

Procedural Content Creation in the Age of Generative AI

Marvin Zammit

Supervised by Professor Antonios Liapis

Co-supervised by Professor Georgios N. Yannakakis

Institute of Digital Games

University of Malta

19th January 2026

*A thesis submitted in partial fulfilment of the requirements for the degree
of Ph.D. in Game Research.*



University of Malta Library – Electronic Thesis & Dissertations (ETD) Repository

The copyright of this thesis/dissertation belongs to the author. The author's rights in respect of this work are as defined by the Copyright Act (Chapter 415) of the Laws of Malta or as modified by any successive legislation.

Users may access this full-text thesis/dissertation and can make use of the information contained in accordance with the Copyright Act provided that the author must be properly acknowledged. Further distribution or reproduction in any format is prohibited without the prior permission of the copyright holder.



Copyright ©2026 University of Malta

WWW.UM.EDU.MT

First edition, 19th January 2026

To my wife Raffaella, our son Tumas, and my late parents Freddie and Mary

You have made this possible with your love, care, patience, and support.

I am eternally grateful for your generosity.

Acknowledgements

Firstly I express my gratitude to my supervisor Prof. Antonios Liapis, who has always been present with his guidance, and prompt to reply to my incessant queries and engage in lengthy discussions. Carrot or stick, his direction was there when, and as needed. I also thank my co-supervisor Prof. Georgios N. Yannakakis, whose vast experience and kind demeanour are second to none. I cannot imagine better tutors for this journey.

I am deeply grateful to the Institute of Digital Games at the University of Malta for their support throughout the course, as well as for facilitating my travel and publications. I appreciate the encouragement of all the academics there; Dr. Daniel Vella, Prof. Stefano Gualeni, and Dr. Ahmed Khalifa. I am especially indebted to Prof. Gordon Calleja and Dr. Costantino Oliva, whose long-standing friendship and ongoing inspiration I greatly value.

My colleagues at the Institute have been both welcoming and supportive. Although I am thankful to many there, I would like to especially acknowledge Dr. Iro Voulgari, Dr. Kostantinos Makantasis, Dr. David Melhart, Dr. Daniele Gravina, Jasper Schellekens, Kostantinos Sfikas, Chintan Trivedi, Matthew Barthet, Kosmas Pinitas, Roberto Gallotta, and Nemanja Rasajski; some of whom were also my fellow students and travel companions, but all of them great friends.

I thank all my colleagues and friends at Mighty Boards, who have been understanding and very supportive. Most of all I am grateful to Christopher Borg and David Chircop for their continued friendship over many years.

I am also sincerely thankful to my family and friends for their constant support during this challenging journey. In particular, I thank Noel Cuschieri, Jeffrey Galea, Dino Mifsud, and Ronald 'Rex' Grech Santucci for being akin to brothers to me, and for their presence through thick and thin.

I hold in loving memory my late parents, Mary and Freddie Zammit, whose sacrifices made so much possible. I am also grateful to my parents-in-law, Mary Rose and the late Gabriel Caruana, for their steadfast presence in our lives. Above all, my deepest appreciation goes to my wife, Raffaella Zammit, and our son, Tumas, who are my daily source of joy and who have upheld me through all my endeavours at every step.

Abstract

Within the domain of Computational Creativity, Procedural Content Generation (PCG) in multimodal domains presents challenges in modal alignment, diversity, and coherence. Although recent generative AI models, including Large Language Models (LLMs), have demonstrated improved cross-modal capabilities, they frequently struggle with balancing autonomy and control to produce creative, high-quality artefacts. This thesis explores how generative AI can be structured to enhance diversity, coherence, and evaluation in multimodal PCG, with a focus on text and image generation. This work addresses the fundamental question: how can AI be guided to generate novel, high-quality, and semantically aligned artefacts across modalities?

To this end, the research introduces four key approaches: Firstly, evolutionary quality diversity algorithms are combined with generative models to improve output diversity in image generation while maintaining semantic relevance.

Secondly, the MAP-Elites algorithm is augmented with Transverse Assessment to explore multimodal search spaces, discover diverse solutions, and improve coherence across text and image generation.

The third approach demonstrates CrawLLM, a zero-shot generative pipeline that uses LLMs to orchestrate the creation of game levels, narrative elements, and visual assets for a video game, ensuring thematic consistency and content structure.

The last approach builds on CrawLLM by incorporating the LLM-driven pipeline into an adaptive evaluation loop to dynamically assess and regenerate underperforming artefacts, improving overall quality and thematic fit.

These contributions advance methodologies for multimodal PCG by integrating generative AI with structured evaluation and optimisation techniques. The findings provide novel insights into search-based creativity, LLM-driven generative orchestration, and AI-assisted evaluation frameworks, with applications in computational creativity, game design, and digital media generation.

Contents

List of Abbreviations	xv
1 Introduction	1
1.1 Challenges in Machine Creativity	1
1.2 Contemporary Generative AI	2
1.3 Towards Generative Controlled Autonomy	3
1.4 Research Questions	4
1.4.1 Key Concepts and Operational Definitions	4
1.4.2 Construct Validity and Scope	6
1.5 Contributions	7
1.5.1 Publications	8
1.6 Structure of Thesis	9
1.7 Summary	10
2 Related Technologies	11
2.1 Evolutionary Algorithms	11
2.1.1 Non-Dominated Sorting Genetic Algorithm II	14
2.1.2 The (1 + 1) Evolutionary Algorithm	14
2.2 Neural Networks and Deep Learning	15
2.2.1 Recurrent Neural Networks	16
2.2.2 The Transformer Model	16
2.2.3 Large Language Models	17
2.2.4 Generative Adversarial Networks	19
2.2.5 Variational Auto-Encoders	19
2.2.6 The Vision Transformer and Vector-Quantised GAN	20
2.3 Multimodal Algorithms	21
2.3.1 Contrastive Language-Image Pretraining	21
2.3.2 Foundational Language And Vision Alignment	22

2.3.3	VQGAN-CLIP Image Generation	22
2.3.4	Diffusion Models	23
2.4	Quality Diversity Algorithms	25
2.4.1	Novelty Search with Local Competition	27
2.4.2	Multidimensional Archive of Phenotypic Elites	28
2.4.3	Monte Carlo Elites	29
2.5	Summary	30
3	Background Work	31
3.1	A Computational Creativity Lens	31
3.2	AI for Text and Image Generation	33
3.2.1	Text generation	34
3.2.2	Image generation	36
3.2.3	Multimodality	41
3.3	Procedural Content Generation in Games	43
3.3.1	Implications of GenAI on PCG	45
3.4	Evaluative Measures	47
3.4.1	Image Metrics	47
3.4.2	Text Metrics	49
3.4.3	Multimodal Metrics	50
3.5	Summary	51
4	Combining Evolution and Backpropagation	52
4.1	Methodology	53
4.2	Use Case: Text-to-Image Generation	54
4.2.1	The Refinement Cycle using Backpropagation	54
4.2.2	The Exploration Cycle using Evolutionary Algorithms	54
4.3	Structure of the Experiment	57
4.3.1	Hypotheses	57
4.3.2	Protocol	57
4.3.3	Performance Metrics	58
4.4	Results and Evaluation	59
4.4.1	Qualitative Observations	62
4.5	Discussion	64
4.6	Limitations and Shortcomings	66
4.7	Summary	67
5	Multimodal Search Space Illumination	68

5.1	Methodology	68
5.2	Use Case: Hypothetical Game Blurbs	70
5.2.1	The Text Modality	70
5.2.2	The Image Modality	73
5.2.3	Applying MELiTA to the Use Case	75
5.3	Structure of the Experiment	76
5.3.1	Hypotheses	77
5.3.2	Protocol	78
5.3.3	Evaluation Metrics	79
5.4	Results and Evaluation	80
5.4.1	Evaluating the Final Quality and Diversity	81
5.4.2	Evaluating the Progress of Quality and Diversity	82
5.4.3	Evaluating Image and Text Diversity in the Final Archive	83
5.5	Discussion	86
5.6	Limitations and Shortcomings	89
5.7	Summary	90
6	A Zero-Shot Multimodal Pipeline	91
6.1	Methodology	91
6.2	Use Case: The CrawLLM Video Game	93
6.2.1	The Game Design	93
6.2.2	The Generative Pipeline	95
6.2.3	Cyclic Dungeon Generation	106
6.2.4	Computational Considerations	108
6.2.5	Generated Games Collection	109
6.3	Structure of the Experiment	109
6.3.1	Hypotheses	109
6.3.2	Experimental Protocol	109
6.4	Results and Evaluation	111
6.5	Discussion	117
6.6	Limitations and Shortcomings	119
6.7	Summary	120
7	An Iterative Generation Pipeline	121
7.1	Methodology	121
7.2	Use Case: The CrawLLM Iterative Pipeline	122
7.2.1	Replacement Candidates	123

7.2.2	Definition of Metrics	123
7.2.3	Leaf Node Selection	125
7.3	Structure of the Experiment	126
7.3.1	Hypotheses	126
7.3.2	Experimental Protocol	126
7.4	Results and Evaluation	127
7.5	Follow-up Study on Branch Selection	131
7.5.1	Branch Selection	133
7.5.2	Experimental Protocol	135
7.5.3	Results	136
7.6	Discussion	140
7.7	Limitations and Shortcomings	143
7.8	Summary	144
8	Discussion	145
8.1	Combining Evolution and Backpropagation	145
8.2	Multimodal Search Space Illumination	147
8.3	A Zero-Shot Multimodal Pipeline	149
8.4	An Iterative Generation Pipeline	152
8.5	Addressing the Research Questions	154
8.6	Observations	156
8.7	Summary	157
9	Conclusion	158
9.1	Future Work	159
9.2	Long-Term Impact	160
	Bibliography	162

List of Figures

1.1	The four studies carried out throughout this work are shown, illustrating their methodologies.	8
2.1	The iterative process of text-to-image generation using VQGAN-CLIP.	23
2.2	The training process of a diffusion-based image model. Noise is gradually added to images in the training dataset. The sequence of images is reversed to train the model to go from pure noise to a complete image.	24
2.3	The principle of operation of the NSLC algorithm. Every individual is compared to its k nearest neighbours, computing its dominance within the neighbourhood, as well as the sparsity of this region as a measure of its diversity. .	28
2.4	An iteration of the MAP-Elites algorithm. An elite E is chosen from the feature map and mutated. Its new placement is found based on its new BCs. If the new location is unoccupied, or if E' has a higher fitness than the elite at that location, it is placed there, replacing any previous occupant.	29
3.1	The complexity of an image may be computed as the proportion of edge pixels detected through an edge detection filter.	48
4.1	Examples of VQGAN models trained on different datasets (<i>ImageNet</i> and <i>WikiArt</i>) generating images for two semantic prompts under CLIP-guidance. .	52
4.2	The basic alternating structure of Refinement and Exploration cycles. A genetic algorithm targetting novelty diversifies the population, whilst a refinement algorithm improves its fitness.	53
4.3	Structure of the experiments alternating between refinement (CLIP-guided VQGAN) and exploratory cycles (NSLC).	58
4.4	Progression of the performance metrics over GAN iterations. The iterations at which evolutionary NSLC cycles were performed are marked in red.	60
4.5	The progression (top to bottom) shows the 5 images with the highest HSV diversity at the end of each cycle in an NSLC-HSV experiment for SP3.	63

5.1	Example of image variation: The text modality of the parent is employed as a prompt for image-to-image SD regeneration, together with “standard” negative prompts.	74
5.2	Some example images showing the combined extremes of complexity and colourfulness adopted as a visual behavioural characteristic.	75
5.3	Simplified feature map of the MELiTA process for this use case, with grey cells representing elites. From a selected elite E , a modified image e'_V generates three candidates from E , R_1 , and R_2 . These are ranked by CLIP score as $\vec{L} = \{f(R'_2), f(E'), f(R'_1)\}$. If $f(R'_2) > f(R_2)$, the candidate R'_2 , combining the image from E' and text from R_2 , replaces R_2 . Otherwise, E' fills the empty cell at row 5, column 1).	77
5.4	The MELiTA experiment structure. 2000 iterations are carried out per title before the final archive is evaluated. 10 independent runs of each experiment are conducted. The entire process is carried out using the MELiTA algorithm as well as for the MAP-Elites baseline.	79
5.5	Box plots compare metrics of the final archives after 2000 iterations of MELiTA to those of the MAP-Elites baseline. Values are averaged over 10 runs for each title.	81
5.6	Box plots compare the area under the graph of QD metrics throughout 2000 iterations of MELiTA to those of the MAP-Elites baseline. Values are averaged over 10 runs for each title.	83
5.7	Box plots of mean and nearest-neighbour distance metrics for visuals and text among final elites. Metrics are aggregated over 10 runs per title for both MELiTA and MAP-Elites algorithms.	84
6.1	Gameplay elements within a CrawLLM game session. Labelled elements are door (1), wall (2), floor (3), and secret door (4) tiles, key (5), PC (6), and enemy (7) sprites, PC (8), location (9), enemy (12), and action card (15) names, and location (10), PC (11), card frame (13), and action card (14) images. The game depicted is “ <i>Luminary’s Legacy: The Renascence Vault</i> ”	96
6.2	The overall structure of the CrawLLM generative pipeline. The static manually-generated templates are used to generate text and visual assets in a format which can be used directly within the game engine. The entire process is driven by the seed thematic details that may be generated by a LLM as shown, or alternatively supplied as natural-language textual input in JSON format.	97

6.3	The sequence of text generation via Mixtral 8×7B. Arrows indicate the reuse of generated text as part of the prompts in the following stages. Steps 4, 5, 6 are repeated for each location in the game (five times in the current paper).	100
6.4	The various SDXL image generation stages are shown with the input semantic elements. Ad-hoc ControlNet guides are shown where applicable. Image dimensions in pixels are shown for reference.	101
6.5	The 3-stage process of tileset generation. Tiles from the game “ <i>Aeternum Ruptura</i> ” are shown.	102
6.6	The spritesheet generation process is executed in 3 stages. Rear-, front-, and side-facing sprites are consolidated into an image containing 12 poses. Figure shows the PC from the game “ <i>Aeternum Ruptura</i> ”.	104
6.7	An example of a procedurally generated dungeon (left) and its underlying graph representation (right). The hierarchical, nested cycle structure is indicated on the graph. The illustrated game instance is “ <i>Spirits of the Ice: Arctic Research Unbound</i> ”.	107
7.1	Processes of the Evo (a), and Regen (b) sets of experiments. Each run consisted of 25 iterations, processing an individual from every group. A run is carried out for each of the 20 generated games from EXP3.	128
7.2	Box plots showing the percentage change in fitness and diversity (a), and the mean number of unique assets chosen for replacement (b). Values are averaged over the 20 runs of each experiment.	129
7.3	The % change in fitness and diversity metrics plotted against iterations for each experiment. Each value represents the mean across the 20 experiment runs for that iteration. The shaded areas denote 95% confidence intervals.	130
7.4	Processes of the MOBranchEvo (a), and MOBranchRegen (b) sets of experiments. Each run processes the most dominated individual across all groups, and iterations stop when the total assets generated reaches 200. A run is carried out for each of the 20 generated games from EXP3.	137
7.5	Box plots showing the percentage change in fitness and diversity (a), and the mean number of unique assets chosen for replacement (b), for the branch selection experiments. Values are averaged over the 20 runs of each experiment.	138
7.6	The % change in fitness and diversity metrics plotted against iterations for the branch selection experiments. Each value represents the mean across the 20 experiment runs for that iteration. The shaded areas denote 95% confidence intervals.	139

List of Tables

2.1	General workflow of a prototypical Evolutionary Algorithm	13
2.2	Approximate parameter counts and training-corpus sizes for selected LLMs. .	19
5.1	Examples of text variation methods. Partial mutation randomly selects a space or punctuation, removes subsequent text, and uses the remaining segment to generate the rest of the description (in red). Full mutation rewrites the entire description, using only the game title (in bold) to generate a new one (in red).	72
5.2	Sample MELiTA output for title T6, “The Shadow Warrior 2: Shadows of the Past” (T6). Each sample was selected as a medoid via k -medoid clustering ($k = 5$) of the final archive. The generated description is shown with its original errors.	85
6.1	Cards in CrawLLM, with their in-game effect; the original card title listed is subsequently adapted by the LLM.	95
6.2	A summary of the user study results. Answer accuracy is shown alongside response entropy per question. An asterisk (*) denotes statistically better outcomes (higher rating, higher accuracy, lower entropy) compared to random selection.	111
6.3	Example dungeon crawler screenshots descriptions and related questions from the user study. The correct options which were used to generate the game are shown as checked.	113
6.4	Example card-based combat screenshots and related questions from the user study. The correct options which were used to generate the game are shown as checked.	114
6.5	Example enemy descriptions and related questions from the user study. The correct options which were used to generate the game are shown as checked.	117
7.1	The leaf node assets grouping by type, used for diversity comparisons. 76 assets (leaf nodes) are considered in total.	124

7.2	Table showing the combined methodologies and the corresponding experiment tags in EXP4.	127
7.3	Mean percentage change in fitness and diversity for each experiment. An * indicates statistically significant change using Wilcoxon signed rank test (corrected for false discovery rate, $p < 0.05$).	129
7.4	The full asset set grouping by type, used for diversity comparisons in branch node selection strategies. 162 assets are considered in total.	134
7.5	Mean percentage change in fitness and diversity for the branch selection experiments. The results of the multi-objective leaf node selection experiments are also listed for reference. An * indicates statistically significant change using Wilcoxon signed rank test (corrected for false discovery rate, $p < 0.05$).	136

List of Abbreviations

2D two-dimensional	44
3D three-dimensional	44
AI Artificial Intelligence	1
AUC Area Under the Curve	80
BC Behavioural Characteristic	26
BERT Bidirectional Encoder Representations from Transformers	17
CAN Creative Adversarial Network	39
CC Computational Creativity	1
CLIP Contrastive Language-Image Pretraining	21
CNN Convolutional Neural Network	20
CPU Central Processing Unit	108
DAG Directed Acyclic Graph	121
EA Evolutionary Algorithm	11
FACE Frame, Aesthetic, Concept, and Expression	3
FLAVA Foundational Language And Vision Alignment	22
GA Genetic Algorithm	12
GAN Generative Adversarial Network	19
GenAI Generative Artificial Intelligence	2
GPT Generative Pre-Trained Transformer	17
GPU Graphics Processing Unit	108
HED Holistically-nested Edge Detection	47
HITL Human-In-The-Loop	3
HP Hit Points	94
HSV Hue, Saturation and Brightness	55
IDEA Iterative Development - Execution - Appreciation cycle	3
IP-Adapter Image Prompt Adapter	41

JSON JavaScript Object Notation	92
LC Local Competition	27
LDA Latent Dirichlet Allocation	49
LLM Large Language Model	4
LMM Large Multimodal Model	43
LPIPS Learned Perceptual Image Patch Similarity	49
LSTM Long Short-Term Memory	16
MAP-Elites Multidimensional Archive of Phenotypic Elites	27
MEliTA MAP-Elites with Transverse Assessment	7
ML Machine Learning	11
MOO Multi-Objective Optimisation	13
MP Mana Points	94
NEAT NeuroEvolution of Augmenting Topologies	40
NLP Natural Language Processing	17
NN Neural Network	1
NSGA-II Non-dominated Sorting Genetic Algorithm II	14
NSLC Novelty Search with Local Competition	7
PC Player Character	93
PCG Procedural Content Generation	1
QD Quality Diversity	4
RNN Recurrent Neural Network	16
RPG Role-Playing Game	46
SBERT Sentence BERT	18
SD Stable Diffusion	24
SDXL Stable Diffusion XL	7
SOO Single-Objective Optimisation	13
SOTA State Of The Art	2
SP Semantic Prompt	57
SSIM Structural Similarity Index Measure	48
TPU Tensor Processing Unit	108
UCB Upper Confidence Bound	29
UI User Interface	99
VAE Variational Auto-Encoder	19
VGDL Video Game Description Language	160

ViT Vision Transformer	20
VQGAN Vector Quantized Generative Adversarial Network	20
VQVAE Vector Quantized Variational Auto-Encoder	20
VRAM Video Random Access Memory	108

Introduction

The interdisciplinary domain of Computational Creativity (CC) has emerged soon after the earliest advances in Artificial Intelligence (AI), in the 1960s and 1970s, with the purpose of exploring how machines can emulate, support, or enhance human creativity. It challenges conventional notions of creativity, expands the boundaries of art and design, and drives forward innovative applications in technology and industry. Research within this domain, in particular Procedural Content Generation (PCG), has tried to address the intricate challenge of creating content across domains like art, music, writing, and design. This involves balancing technical constraints, aesthetic values, and contextual relevance.

1.1 | Challenges in Machine Creativity

Unlike human creators who adapt fluidly to constraints, such as style, thematic setting, and culture, machines struggle to integrate these facets into cohesive and meaningful outputs. This is attributable to the lack of broad understanding of the generative algorithms of the domains they are applied to. The ‘knowledge’ of these models is limited to the information they are provided with, or in the case of Neural Networks (NNs), the relations they surmise within their training datasets or processes.

Compounding the problem is the fact that there exists no absolute measure of quality for the produced artefacts. Dewey [51] argues that art is a personal and interactive experience, varying widely depending on the viewer’s emotions, context, and interpretation. Consequently, there is no single universal aesthetic which is a ‘golden standard’ and evaluation of generated artefacts, whether resulting from PCG or human creation, is highly subjective.

Employing AI to generate content in a directed manner, or to generate content across different modalities highlights this problem. Automated systems often excel in narrow, domain-specific tasks but lack the holistic understanding needed for multi-modality. For example, creating an artwork involves not just generating visual elements but also appreciating context, emotion, and purpose. Colton et al. [34] discuss how PCG

algorithms often fall short in appreciating or imagining beyond predefined parameters, resulting in outputs that can feel mechanical or unnatural.

Multiple assets generated across multiple modalities normally need not only to be relevant and have value, but also to be coherent, increasing the complexity of evaluation for generative algorithms. The domain of games itself is an apt example of the importance of coherence between modalities. Liapis et al. [124] define the primary facets of games as being *visuals*, *audio*, *narrative*, *levels*, *rules*, and *gameplay*. Within most games, all these facets are aligned towards the same objective, carefully designed to be coherent and support each other to generate a holistic experience for players. However, generative algorithms often address a single facet and struggle to interpret or align multiple domains within their operation. Addressing all modalities concurrently is challenging without extensive knowledge of each domain.

So is PCG doomed to fail in creative tasks? Ritchie [173] argues that generated artefacts can be considered creative if they possess novelty, quality, and typicality. Novelty itself is bounded by typicality; an artifact that is overly typical may result in being unoriginal, while one that is too atypical risks being irrelevant. Quality is subject to an evaluation, which determines whether or not the artefact may be deemed valuable, useful, or aesthetically pleasant, depending on its nature. It is therefore useful to frame evaluation within these parameters.

1.2 | Contemporary Generative AI

Recent advances in Generative Artificial Intelligence (GenAI), such as ChatGPT [154] and diffusion models [175], have brought leaps in quality of generated assets, and observable capabilities in cross-modality. They are progressing the State Of The Art (SOTA) towards bridging across multiple forms of input such as text, image, or audio, to generate cohesive cross-modal outputs. One particularly attractive feature for multimodal generation that these models possess is their understanding of natural language input, which is used to constrain their outputs without requiring dedicated parametrisation. Semantic comprehension also endows them with the ability to assess artefacts through the modality of text. This offers an improved evaluation mechanism, and suggests that they can be used in feedback loops to improve generation pipelines. Text in generative models also provides a human-readable intermediary format which is understandable without any requirement for specialised training about anything other than the problem being addressed. It can be used as an input (or output) format at different stages to log progress within a generative pipeline, or, due to its very nature, also as an editable control in a

Human-In-The-Loop (HITL) system. Such systems adopt a co-creative approach to asset generation, involving a human designer as the evaluator of the creation process, with the algorithm merely acting as a muse, proposing options to kindle creativity.

1.3 | Towards Generative Controlled Autonomy

For GenAI to be effectively used for multimodal PCG, there must be a strategic balance between *autonomy*, or the system's ability to independently generate creative output, and *control*, or the capacity of human users or systems to guide and constrain the generative process within the desired bounds of quality, coherence, and relevance. Striking this balance is critical for adhering to domain constraints while ensuring the generative outputs meet creative criteria such as diversity and novelty.

In relation to the Frame, Aesthetic, Concept, and Expression (FACE) and Iterative Development - Execution - Appreciation cycle (IDEA) models of creativity proposed by Colton et al. [34], the author of this thesis argues that recent generative models exhibit characteristics which align, at least partially, to the FACE model as they demonstrate:

Frame: The context or framing is provided through natural language input, which constrains their output and imparts purpose or direction to the generated artefacts.

Aesthetic: Outputs from these models increasingly exhibit qualities that satisfy human aesthetic expectations, particularly for text fluency and image composition.

Concept: Their semantic understanding allows for the generation of conceptually relevant artefacts that correspond to input prompts in meaningful ways.

Expression: The technical execution of these concepts, possible through sophisticated model architectures and training methodologies, is optimised and accessible.

Despite these promising attributes, such systems still struggle to integrate these facets into a cohesive whole when operating in multimodal contexts.

The reciprocity between autonomy and control is imperative for addressing the persistent challenges of achieving coherence, diversity, and semantic relevance across modalities, while also navigating the subjective nature of creativity and evaluation. These challenges reveal a pressing need to explore innovative approaches that combine the strengths of foundation¹ GenAI models with complementary methods to enhance multimodal content generation.

¹Large models trained on extensive corpora, often on specialised hardware and at considerable financial cost.

This thesis takes a step towards addressing these challenges by investigating how advanced generative models can be integrated into functional pipelines to produce cohesive, diverse, and high-quality artefacts across text and image domains.

1.4 | Research Questions

Assessing algorithms in terms of creativity is certainly an ambitious goal in and of itself; there are several thriving communities of researchers whose endeavours target specifically this objective. This thesis, therefore, is more distinctly set to explore the ways in which GenAI models may be used within the domains of text and image generation, since these are the most mature technologies currently available.

More explicitly, the following questions are addressed:

- RQ1 How do evolutionary Quality Diversity (QD) algorithms enhance the semantic alignment and diversity of outputs generated by foundation models in multimodal PCG?
- RQ2 How does an explicit structuring of multimodal search spaces, or the metrics used to traverse them influence the effectiveness of exploration strategies in generating coherent and diverse outputs?
- RQ3 To what extent can Large Language Models (LLMs) direct multimodal generative pipelines to improve coherence, diversity, and overall quality in functional problems such as games?
- RQ4 Which metrics and assessment frameworks are most effective for evaluating and optimising coherence, diversity, and quality in the outputs of multimodal generative systems?

This set of questions is formulated to analyse and assess various aspects of multimodal text and image generation. They aim to explore GenAI's potential to redefine PCG in multimodal creative domains, and to propose structured methodologies for its utilisation. The questions outlined will not be addressed sequentially within this thesis. Rather, they establish a framework of objectives that will be explored through a collection of experiments.

1.4.1 | Key Concepts and Operational Definitions

Throughout the process of interpreting and investigating the research questions, this text makes reference to a set of evaluative concepts, namely *quality*, *diversity*, *semantic*

alignment, and *structuring*. These terms are widely used in CC but their meanings are neither universal nor uncontested. This section therefore introduces explicit operating definitions for these concepts as they are used throughout the thesis, clarifies their scope and limitations, and outlines how they are approximated through computational measures in the subsequent experimental chapters. Importantly, the definitions provided here do not claim to capture these concepts in their entirety. Instead, they attempt to establish a shared interpretative framework within which the experimental results of this thesis should be understood.

1.4.1.1 | Quality

In creative domains, *quality* is an inherently subjective, context-dependent, and normatively defined construct. It has been argued that there exists no universal or objective standard by which creative artefacts can be definitively judged [51, 173]. Within games research, quality may variously refer to aesthetic appeal, functional adequacy, player experience, or thematic appropriateness, depending on the framing of the task. In this thesis, quality is operationalised as the degree to which a generated artefact satisfies task-specific functional, semantic, or thematic constraints, as approximated by automated evaluation metrics and, where applicable, human judgement. Across the experiments presented in the following chapters, quality is assessed using proxies such as semantic similarity scores, fitness values within optimisation processes, LLM-based evaluations, and user-study responses. These measures serve as consistent and reproducible indicators of task-aligned adequacy but do not claim to capture holistic aesthetic or experiential quality. As such, quality in this work should be interpreted as *relative* and *instrumental*, with the purpose of enabling comparative evaluation rather than absolute judgement.

1.4.1.2 | Diversity

Diversity is a central concern in PCG and CC, often associated with notions of novelty, exploration, and avoidance of mode collapse [114, 123, 150]. However, diversity is not a singular construct. It is dependent on the representation space, the chosen features, and the level of abstraction at which variation is considered.

In this thesis, *diversity* refers to the degree of variation among generated artefacts within a population or archive, measured with respect to explicitly defined behavioural, semantic, or perceptual features. Diversity is quantified using feature-space distances (e.g., colour statistics, perceptual similarity, or text embeddings), archive coverage in QD algorithms, and counts of distinct assets selected during iterative regeneration. These measures capture *structural* or *representational* diversity rather than experiential or

interpretive diversity. Consequently, while they are effective for analysing exploration behaviour and population spread, they do not necessarily account for how diverse artefacts may be perceived by human players or designers.

1.4.1.3 | Semantic Alignment

Semantic alignment is a critical concept in multimodal generation, particularly when text and image modalities are expected to cohere around shared meaning or intent [166]. In practical terms, semantic alignment implies that different representations express compatible concepts, themes, or narratives. In computational systems, however, semantic alignment can only be inferred indirectly.

In this thesis, *semantic alignment* denotes the degree of correspondence between the intended meaning expressed in textual prompts and the content expressed in generated artefacts across one or more modalities. Semantic alignment is primarily approximated using shared embedding spaces of multimodal pre-trained models, supplemented by human user studies. It is important to note that such measures reflect statistical alignment within learned representational spaces rather than semantic understanding in a human sense. Accordingly, claims about semantic alignment in this thesis are limited to measured correspondence rather than semantic comprehension or interpretative meaning.

1.4.1.4 | Structuring

The concept of *structuring* is central to the contributions of this thesis and warrants explicit definition. In the context of generative systems, structuring concerns how generative processes are guided, constrained, and decomposed to balance autonomy with control.

In this thesis, *structuring* refers to the explicit imposition of intermediate representations, constraints, evaluation stages, or search-space organisations that guide generative processes toward coherent, diverse, and task-relevant outputs.

Structuring manifests in different forms across the four experimental studies presented in this work [34, 122, 231], which are presented in Section 1.5. In all cases, structuring is not treated as a limitation on creativity, but as a mechanism for enabling controlled exploration and coherence in complex multimodal domains.

1.4.2 | Construct Validity and Scope

The measures used throughout this thesis should be understood as *proxies* for the constructs defined above, consistent with established methodological discussions of con-

struct validity in CC and PCG research [34, 173]. Their validity is therefore necessarily partial: each measure captures certain aspects of a construct while omitting others. Rather than claiming that these measures *are* the constructs themselves, this thesis adopts a pragmatic stance, evaluating whether the measures are sufficiently sensitive, consistent, and aligned with theoretical expectations to support comparative analysis and optimisation.

Evidence for the effectiveness of these measures is provided through convergent results across multiple experiments, observed trade-offs between quality and diversity, comparative baselines, and human validation where applicable. The implications and limitations of these choices are revisited in Chapter 8, where the findings are discussed in relation to the research questions and the broader goals of CC research.

1.5 | Contributions

This work focuses on the generation of content within the text and image domains, using recent GenAI models specifically. With these conditions, the research questions are explored through four different approaches to the use of foundation models for PCG, namely:

- EXP1 **Combining evolution and backpropagation:** The first method explored is the combination of evolutionary algorithms with backpropagation, to diversify the outputs of a text-aligned image generation model. This technique takes advantage of the population-based evolutionary algorithm to assess images vis-a-vis each other in order to diversify a set of images. This is accomplished via a multi-objective Novelty Search with Local Competition (NSLC) algorithm [115] which optimises image fitness to the semantic prompt while simultaneously promoting diversity.
- EXP2 **Multimodal search space illumination:** For this approach, an improvement on the MAP-Elites quality diversity algorithm [150], dubbed MAP-Elites with Transverse Assessment (MEliTA), is proposed for multimodal outputs, in this case text and an accompanying image. The search space is segmented in terms of behavioural characteristics chosen for each modality and a solution space of better fitting individuals is gradually built. This results in a range of solutions spanning a range of properties, which may be equally desirable.
- EXP3 **Zero-shot multimodal pipeline:** The next study utilises a LLM to direct and orchestrate a pipeline that generates various narrative and visual assets for a video game created for this purpose, CrawLLM. The pipeline is made up of LLM instances for the output text as well as to generate semantic prompts for Stable

	EXP1	EXP2	EXP3	EXP4
	Combining Evolution and Backpropagation	Multimodal search space illumination	Zero-shot multimodal pipeline	Iterative multimodal pipeline
Output	Image (1)	Image (1) + Text (1)	Images (74) + Text (24) "Hidden" Text (78)	Images (74) + Text (24) "Hidden" Text (78)
Search	Evolutionary Multi-objective	Evolutionary, MAP-Elites ++	Zero-shot	Generate and test
Objectives	Text to image fitness, Image diversity	Text to image fitness, Text & image diversity	None	Theme fitness, Text & image diversity

Figure 1.1: The four studies carried out throughout this work are shown, illustrating their methodologies.

Diffusion XL (SDXL) image generators [162] to maintain coherence. A user-study was conducted to assess whether the generated assets retained coherence to the seed setting prompts.

EXP4 Iterative multimodal pipeline: The final study advances the CrawlLLM pipeline to dynamically assess diversity and fitness throughout the generative process, to identify the least performing assets and subsequently regenerate them. Depending on which approach is taken, this results in improved overall quality or diversity across the range of artefacts.

Through these methodologies, laid out visually in Figure 1.1, the thesis illustrates a range of use-cases for GenAI models, and the ways that different algorithms may be applied to promote coherence and diversity across modalities. The principal contributions to the domain of CC are therefore the proposed algorithms and methodologies mentioned above, which also highlight several ways in which multimodal generated content may be assessed and applied to practical applications.

1.5.1 | Publications

Throughout the course of this research, the author has contributed to the field through a number of papers which have been published in peer-reviewed journals and conferences. These papers are listed below:

1. Zammit, M., Liapis, A., & Yannakakis, G. N. (2026). CrawlLLM: An LLM-Based Pipeline for Game Asset Generation. *IEEE Transactions on Games*, Special Issue on

Large Language Models and Games. [Accepted for Publication]

2. Zammit, M., Liapis, A., & Yannakakis, G. N. (2024). CrawLLM: Theming Games with Large Language Models. In *Proceedings of the IEEE Conference on Games (IEEE CoG) 2024*.
3. Gallotta, R., Todd, G., Zammit, M., Earle, S., Liapis, A., Togelius, J., & Yannakakis, G. N. (2024). Large Language Models and Games: A Survey and Roadmap. *IEEE Transactions on Games*. [Early Access]
4. Zammit, M., Liapis, A., & Yannakakis, G. N. (2024). MAP-Elites with Transverse Assessment for Multimodal Problems in Creative Domains. In *Proceedings of the 6th International Conference on Computational Intelligence in Music, Sound, Art and Design (EvoMUSART)*.
5. Zammit, M., Liapis, A., & Yannakakis, G. N. (2022). Seeding Diversity into AI Art. In *Proceedings of the International Conference on Computational Creativity*. [Awarded Best Student Paper]

Publications 1 and 2 detail the work carried out in EXP3, whereas publication 4 and 5 relate to experiments EXP2 and EXP1 respectively. Other publications by the author related to other work in the field of games in pedagogy are listed below for reference:

- Zammit, M., Voulgari, I., Liapis, A., & Yannakakis, G. N. (2022). Learn to Machine Learn via Games in the Classroom. *Frontiers in Education*, 7.
- Zammit, M., Voulgari, I., Liapis, A., & Yannakakis, G. N. (2021). The Road to AI Literacy Education: From Pedagogical Needs to Tangible Game Design. In *Proceedings of the European Conference on Games Based Learning*.
- Voulgari, I., Zammit, M., Stouraitis, E., Liapis, A., & Yannakakis, G. N. (2021). Learn to machine learn: Designing a Game Based Approach for Teaching Machine Learning to Primary and Secondary Education Students. In *Proceedings of the ACM Interaction Design and Children (IDC) conference*.

1.6 | Structure of Thesis

This document is structured as follows:

Chapter 2 presents an overview of the algorithms which form the technical basis of the methodologies used throughout the following chapters.

Chapter 3 reviews work in the domains related to this thesis, giving an overview of the SOTA, and describing the most notable contributions therein.

Chapters 4 through 7 respectively detail studies EXP1 to EXP4 listed in Section 1.5 above. Each Chapter describes the experiments carried out, the methodologies used, and the results obtained.

Chapter 8 discusses the collective findings of the conducted experiments and how these may answer the research questions.

Chapter 9 concludes the thesis and proposes avenues for further research in this field.

1.7 | Summary

This chapter outlined the research objectives, which leverage recent GenAI models for multimodal content creation within text and image domains. It introduced four novel approaches: a study combining an evolutionary algorithm with backpropagation, an enhanced MELiTA algorithm for exploring multimodal search spaces, the CrawLLM zero-shot pipeline for coherent asset generation, and its iterative variant for refining diversity and thematic alignment. Four key research questions were posed to address diversity, semantic alignment, multimodal pipeline orchestration, and evaluation methods. These contributions aim to advance CC and set the stage for exploring the related work and SOTA in the next Chapter.

Related Technologies

The term PCG is widely accepted to refer to the algorithmic generation of digital game assets through heuristic, stochastic, or Machine Learning (ML) processes, whereby numerous pieces of content may be produced with minimal direct human authoring [201]. Although, in theory, the domain of PCG is not strictly tied to games, it is generally understood that there is an enduring link to it. GenAI, in contrast, is a broader term referring to computational systems that learn patterns from immensely large corpora of existing data in order to create new artefacts that emulate human-crafted outputs [7]. Although the definition is broad, and technically encapsulates all generative models, its contemporary use is often associated with the SOTA foundational models [177] which achieve near-human level of quality in their output, such as recent chatbots, and text-to-image models.

This Chapter aims to describe those technologies which have contributed to the development of contemporary PCG and GenAI. It outlines the algorithms which are at the core of the methodologies used in the rest of this thesis. This text is by no means comprehensive of the extensive body of work related to these two domains. The objective is merely to describe the most influential algorithms that have paved the way for the ensuing work, and which will be referred to occasionally. The following algorithms are not presented in exact chronological order, although there is a modicum of historical progression in the way they are conferred.

2.1 | Evolutionary Algorithms

Among the numerous classes of generative algorithms, Evolutionary Algorithms (EAs) have constituted a particularly significant category and have seen extensive application across diverse procedural generation tasks. They constitute a family of population-based optimisation methods that emulate the adaptive processes observed in biological evolution. These algorithms, whose origin dates back to the 1960s and 1970s, operate by iteratively improving a set of candidate solutions through mechanisms analogous

to genetic inheritance, mutation, and natural selection [91]. They are fundamentally algorithms which optimise a set of solutions towards one or several objectives.

Fundamentally a number of components comprise an EA. Each candidate solution, often referred to as an *individual* or *phenotype*, is encoded into a concise representation, a *genotype*, which can be subjected to genetic operations. Common encodings include sequences of bits, where each bit (gene) contributes to the phenotype, real-valued vectors, or alphanumerical string representations. However these may become more convoluted as the complexity of the modelled system increases. A *fitness* or *utility* function $f(x)$ needs also to be defined as a measure that evaluates the quality of solution x . Higher fitness indicates superior performance when the goal is maximisation; for minimisation problems, a suitable transformation (e.g. negation) is applied. At every iteration, a *selection mechanism* chooses parent individuals for reproduction, usually on the basis of their fitness. Many strategies may be employed for this process, such as roulette-wheel selection, where selection probabilities are assigned proportionally to fitness; tournament selection, where the fittest individual is chosen from a randomly sampled subset; and rank-based selection, where selection probabilities are assigned by sorted fitness rank [91].

Genetic operators are applied to the selected individuals. *Crossover* (or *recombination*) is used to combine genetic material from two or more parents to produce offspring; in binary-encoded representations, one-point or two-point crossover is commonly applied, while in real-valued encodings, techniques such as simulated binary crossover are frequently utilised [92]. *Mutation* is introduced to select individuals to maintain genetic diversity by randomly altering its genotype. In binary strings, this is typically achieved by flipping bits with a low probability, whereas in real-valued vectors, small Gaussian perturbations are often applied.

Replacement, or *survivor selection*, is then conducted to determine which individuals proceed to the next generation. Generational replacement involves the complete substitution of the parent population by offspring. Steady-state replacement allows only a few offspring to replace less fit individuals, and elitist replacement ensures that the best-performing individuals are preserved, thereby promoting consistent improvement in population fitness over successive generations.

Table 2.1 shows the typical workflow of an EA, indicating the operation and application of the elements just described.

Genetic Algorithms (GAs) are considered a specialised subset of EAs, characterised by their specific use of selection, crossover, and mutation operators inspired by biological evolution. They are distinguished from other evolutionary algorithms by their use of fixed-length, often binary-encoded representations; a strong emphasis on crossover as

Step	Description
1	Initialization: Generate an initial population $\{x_i\}_{i=1}^N$ of size N , typically at random.
2	Evaluation: Compute fitness $f(x_i)$ for each individual.
3	Selection: Select parents based on fitness-biased probabilities.
4	Crossover: Exchange genetic material among selected parents to produce offspring.
5	Mutation: Apply random perturbations to offspring to maintain diversity.
6	Replacement: Form the new population according to the chosen replacement strategy.
7	Termination Check: If a stopping criterion is met (e.g., maximum generations, convergence tolerance), terminate; otherwise, return to Step 2.

Table 2.1: General workflow of a prototypical Evolutionary Algorithm

the primary variation operator; fitness-proportionate or tournament-based selection mechanisms; and a survivor selection process that replaces most or all of the population each cycle. While EAs encompass a broader class of population-based optimisation methods, GAs are defined by these specific representation schemes and reproduction mechanisms [91, 92].

Problems which focus on optimising a single property are referred to as Single-Objective Optimisation (SOO) problems, and are represented by a scalar function $f(x)$ that needs to be minimised (or maximised). The aim is to identify a solution x^* that satisfies

$$x^* = \arg \min_{x \in \Omega} f(x),$$

where Ω denotes the search space. In GA terms, fitness is directly related to $f(x)$, potentially via a transformation if the algorithm expects maximisation. Single-objective GAs excel at rugged, multi-modal landscapes where gradient information is unavailable or unreliable [91, 92].

Many practical problems however, require optimising multiple objectives simultaneously, such as $f_1(x), f_2(x), \dots, f_M(x)$, which often conflict with each other. The Multi-Objective Optimisation (MOO) problem seeks to minimise all objectives at once:

$$\text{Minimise } \mathbf{f}(x) = (f_1(x), f_2(x), \dots, f_M(x)), \quad x \in \Omega.$$

Since no single solution typically optimises all M objectives simultaneously, the concept of *Pareto optimality* is employed: A solution $x^* \in \Omega$ is considered Pareto-optimal when no other feasible solution $x \in \Omega$ exists that performs at least as well in all objectives while performing strictly better in at least one objective. In mathematical terms, no

solution x can satisfy

$$f_i(x) \leq f_i(x^*) \quad \forall i \in \{1, \dots, M\}$$

while also having

$$f_j(x) < f_j(x^*) \quad \text{for some } j \in \{1, \dots, M\}.$$

The complete collection of Pareto-optimal solutions forms the *Pareto set* in the decision space, while their corresponding objective values create the *Pareto front* in the objective space. For problems with two or more objectives, the Pareto front is commonly visualized by plotting $f_1(x)$ against $f_2(x)$, which reveals a curve that represents the fundamental trade-off between objectives; improvements in one objective necessitate deterioration in the other [27, 66].

2.1.1 | Non-Dominated Sorting Genetic Algorithm II

The Non-dominated Sorting Genetic Algorithm II (NSGA-II) algorithm, introduced by Deb et al. [47], is among the most widely adopted multi-objective EAs. Its key strengths lie in combining elitism, fast non-dominated sorting, and a diversity-preservation mechanism (*crowding distance*) without requiring user-specified parameters. These mechanisms ensure that the best non-dominated solutions are always preserved, while crowding distance maintains a well-distributed approximation of the front [27, 47]. Although it has since been outperformed by more recent algorithms, particularly in high-dimensional and computationally intensive problems, it has not been made entirely obsolete. NSGA-II continues to serve as a reliable benchmark and remains of practical value in a variety of optimisation contexts, especially in problems with 2 or 3 objectives.

2.1.2 | The (1 + 1) Evolutionary Algorithm

Certain applications often benefit from simpler variations of EAs, which offer reduced computational overhead at the expense of efficiency.

One such variant is the *(1 + 1) EA* [57], a simple optimisation algorithm in which a single solution is iteratively improved through mutation and selection. The process starts with the generation of an initial candidate solution x . In each iteration, a new offspring x' is created by applying a mutation operation to x . The fitness of the offspring is then evaluated, and selection is applied through elitism, i.e. the offspring replaces the parent if its fitness is at least as high as that of the parent. This procedure is repeated until a predefined termination condition is met, such as a maximum number of iterations or the attainment of a specific fitness value. No recombination or crossover is employed in this

algorithm as there is only ever 1 individual in the population. Only mutation is used to explore the search space.

A similar approach, although not technically an EA except in the broadest of terms, is the *generate-and-test* paradigm [200]. Generate-and-test methods may be applied to an individual solution or to a population. Poorly performing individuals are regenerated randomly across multiple iterations. Each iteration is evaluated, and a candidate is retained if it satisfies predefined usability criteria. Like the (1 + 1) EA, the process continues until the desired fitness level or a different termination criterion is attained.

2.2 | Neural Networks and Deep Learning

The most impressive advancements leading directly to contemporary GenAI resulted from the pursuit of NNs. Although NNs were developed prior to the advent of EAs, the latter were more widely adopted for tasks involving content generation, particularly throughout the periods historically referred to as the ‘AI winters’ [206]. During these intervals, NNs experienced a marked decline in popularity, primarily as a consequence of their inability to meet the inflated expectations that had accompanied their initial emergence. It was only after substantial advancements in hardware capabilities had materialised that NNs began to regain prominence. Subsequently, they were increasingly employed across a diverse range of applications, including their implementation into generative frameworks.

Artificial NNs are computational models inspired by biological neural systems, consisting of interconnected nodes (neurons) organized in layers [142]. Each connection is associated with a weight w_{ij} , and neurons apply activation functions to weighted inputs. The fundamental operation is expressed as:

$$y = f \left(\sum_{i=1}^n w_i x_i + b \right) \quad (2.1)$$

where f represents the activation function and b is the bias term [176].

NNs are trained using labelled datasets $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$ where x_i represents input features and y_i the corresponding target outputs. The training process minimises a loss function $L(\theta)$ that measures the discrepancy between predicted outputs $\hat{y}_i = f(x_i; \theta)$ and true labels y_i , where θ denotes the network parameters [9]. Common loss functions include mean squared error for regression and cross-entropy for classification tasks.

The *backpropagation* algorithm evaluates the weights and biases through gradient-based optimisation [178]. It computes gradients of the loss function with respect to these

network parameters using the chain rule of calculus. For a network with loss function L , the weight update rule is:

$$w_{ij}^{(t+1)} = w_{ij}^{(t)} - \eta \frac{\partial L}{\partial w_{ij}} \quad (2.2)$$

where η is the learning rate. Error signals are propagated backward through the network, enabling efficient computation of gradients for all parameters simultaneously [220].

The topologies of NNs vary greatly, and certain architectures lend themselves to some tasks better than others [111]. A relevant selection of noteworthy ones shall be described in the relevant sections of this chapter. Generally speaking, the capability of the network scales with the complexity and size of its architecture. Models which contain a large number of hidden layers are able to learn more complex hierarchical representations and interrelations of data, and are called *deep learning* networks or deep NNs [111].

2.2.1 | Recurrent Neural Networks

One limitation of generative NNs becomes apparent in domains involving sequential or time-dependent data, such as music or text generation. These networks lack mechanisms for retaining information about previous inputs, making them ill-suited for capturing temporal dependencies. In response to this problem, Recurrent Neural Networks (RNNs) [196] and Long Short-Term Memory (LSTM) networks [90] introduced internal memory states that could carry information across time steps, enabling the modelling of long-range dependencies across time steps. As autoregressive models, they predict future elements in a sequence based on previously observed values, learning dependencies over time through internal state transitions. One notable feature of autoregression is that such models do not require their inputs to have a fixed size, but can easily process varying lengths of input tokens. RNNs capture temporal dynamics via recurrent connections, but suffer from vanishing gradients when modelling long-term dependencies. LSTMs address this limitation by incorporating gating mechanisms that regulate the flow of information and preserve gradients over extended sequences [90, 196].

2.2.2 | The Transformer Model

The transformer architecture [211] replaced the implementation of memory structures with the *attention* mechanisms. It consists of an encoder-decoder architecture, each of these containing a multi-head self-attention module, and a position-wise feed-forward NN. The self-attention mechanism allows each input token in the sequence to attend to all other tokens.

In generation tasks, the transformer decoder produces outputs auto-regressively. At each time step t , the model uses only previously generated tokens:

$$P(y_t | y_1, y_2, \dots, y_{t-1}, X) \quad (2.3)$$

The masked self-attention mechanism ensures that the prediction at time t only depends on tokens preceding it. This autoregressive formulation is essential for generative models and is the basis of most modern LLMs, like the various Generative Pre-Trained Transformer (GPT) models [18, 168]. The transformer architecture demonstrates scalability and flexibility, which has led it to become the dominant architecture in Natural Language Processing (NLP) and has also facilitated its application to other modalities such as vision [55] and audio [212]. The attention mechanism can, in theory, scale up to process any length of sequential input tokens. However since tokens are compared to all other previous tokens in the input, the computational resources used increase rapidly with longer sequences, thereby setting an upper limit to the context length a model can attend to. Given the widespread use of transformers, this limitation has also been addressed by several researchers and several methods have been proposed to mitigate it [20, 46].

2.2.3 | Large Language Models

LLMs are deep learning models built on the transformer architecture, designed to model and generate human-like text. Their efficacy depends on various components, including pre-training objectives, tokenisation, optimisation strategies, scaling laws, and fine-tuning methods.

NN-based models adopt a common approach to pre-processing text, which is tokenisation. This process breaks down words into groups of characters which enables the language models to handle a large vocabulary, including rare words efficiently. Although the process is common, several distinct algorithms are available for this, which split input text into a string of tokens, each representing a key to a group of characters within the tokeniser vocabulary.

Different LLMs exhibit different capabilities, e.g. autoregressive causal language modelling in models such as the GPT family [18, 168], masked language modelling which predicts randomly masked tokens in models like Bidirectional Encoder Representations from Transformers (BERT) [50], or permutation language modelling which allows bidirectional context via permutation of token sequences, as used in XLNet [230].

BERT is considered notable among LLMs due to its significant contributions to natural language understanding and sentence-level representation. It introduced a bidirectional transformer encoder architecture, enabling context to be captured from both

directions and establishing a pre-training and fine-tuning paradigm that achieved state-of-the-art performance across numerous NLPs benchmarks [50]. Sentence BERT (SBERT) extended BERT by employing a Siamese network structure to produce fixed-size sentence embeddings, allowing semantic similarity tasks to be performed efficiently through vector comparisons [172].

LLMs are typically pre-trained using massive unlabelled text data through self-supervised objectives [4]. They are subsequently adapted for downstream tasks through various methodologies, commonly using supervised fine-tuning and reinforcement learning approaches to make their output more useful, safe, and aligned with human preference [26, 155, 194]. Various of these models have been applied to a wide variety of purposes, such as biomedical tasks [6, 112, 158], sentiment analysis on social media [5, 96], synthetic data generation [127], fake news detection (and generation) [236], and story generation [63].

OpenAI's ChatGPT 3.5 and 4 [154] ushered a landmark leap in performance over its predecessors, and allowed the general public to freely chat online with the model about any subject, as long as the subject matter was morally sound. ChatGPT is a "black box" model trained on undisclosed corpora of data drawn from the internet, and the model itself has never been publicly released. Its impressive performance, arising mainly from a larger model architecture, and enormous training corpus size, allowed it to adequately respond to most conversations, without explicitly requiring fine-tuning to a specific domain. Several other competing models followed soon after, such as Facebook's Llama [58, 202, 203] family of models, Mixtral [97], and others. Table 2.2 lists some of the. Unlike ChatGPT, many of these were openly released under various licensing conditions, allowing hobbyists and researchers to run local instances, and carry out their own investigations. Some of these LLMs came in different model sizes, allowing the smaller ones to run on household grade hardware, with the larger and most performant ones requiring specialised machines to infer within a few seconds of the input submission.

LLM performance improves predictably with model size, data, and compute, as shown empirically [102]. As model size increases, research seems to indicate that certain improvements in performance occurs in sudden steps rather than proportionally to model size [18, 218]. However this claim appears to be contended as other research attributes these apparent leaps in performance to variations in testing methodologies [181].

There are several nuances and details pertaining to the creation of a LLM, and the above is but a scratch on the surface of the domain. As interesting as it is, it would be beyond the scope of this work to enter into such details, as there is already ample literature on the topic [147].

Model	Parameters (approx.)	Training Corpora Tokens (approx.)
BERT [50]	110M	3.3B
GPT-2 [168]	1.5B	8B
GPT-3 [18]	175B	300B
GPT-4 [154]	undisclosed	undisclosed
OPT-175B [240]	175B	180B
BLOOM [180]	176B	366B
PaLM [25]	540B	780B
Mixtral 8x7B [97]	47B (13B active)	undisclosed
DeepSeek-V3 [48]	671B	14.8T
LLaMA-65B [202]	65B	1.4T
LLaMA-2 [203]	7B, 13B, 70B	2T
LLaMA-3 [58]	8B, 70B, 405B	>15T
T5-11B [169]	11B	34B

Table 2.2: Approximate parameter counts and training-corpus sizes for selected LLMs.

2.2.4 | Generative Adversarial Networks

Generative Adversarial Networks (GANs) proved to be a notable advance in image generation and were among the first deep learning models to successfully generate realistic images. GANs consist of a pair of NNs trained in parallel, with the ultimate objective being the generation of artefacts in a chosen modality. One network is the generator, which is trained to replicate data from the training corpus. The second network is the discriminator, which is trained to distinguish between original artefacts from the corpus from those created by the generator. The output of the discriminator is also used as part of the fitness function of the generator, i.e. the generator output has a higher fitness if it ‘fools’ the discriminator into classifying it as a ‘real’ artefact, but gets penalised if the discriminator correctly identifies it as generated. After the training terminates, the discriminator is discarded, and the generator is then capable of creating artefacts which are very similar to those in the training data set. This adversarial approach proved to be very successful and was applied to a wide variety of problems [167, 171, 199].

2.2.5 | Variational Auto-Encoders

Another family of deep generative models which predate GANs is that of Variational Auto-Encoders (VAEs). These were proposed by Kingma and Welling [109] and also consist of two NNs. An encoder learns a probabilistic mapping from a dataset to a lower-dimensional latent space $q_\phi(z | x)$, and conversely a decoder learns the inverse

process, i.e. a mapping between the latent space and new data samples similar to the training corpus $p_\theta(x \mid z)$. Not only does the VAE generate artefacts, but since the lower-dimensional latent vector space is continuous, trajectories may be found within to increase or decrease specific features of the output artefacts, e.g. increase the “smile” component in a face generator by moving along the corresponding latent axis [105].

Vector Quantized Variational Auto-Encoders (VQVAEs) [209] replace the continuous latent space with a discrete codebook $\{e_k\}_{k=1}^K$. The codebook (or embedding space) consists of a finite set of learnable embedding vectors, often called codebook entries or codewords, which are optimised during the training process. During encoding, the encoder produces a continuous vector, which is then quantised by finding the nearest codebook vector $k^* = \arg \min_k \|z_e(x) - e_k\|$. The index of this nearest vector is the discrete latent representation. The decoder is trained to reconstruct an artefact typical of the dataset from its corresponding discrete embedding. In practice, a VQVAE trained on an image corpus can discretise key visual patterns, enabling transformers or autoregressive models to sequentially predict these tokens and generate highly detailed images [62].

2.2.6 | The Vision Transformer and Vector-Quantised GAN

The Vision Transformer (ViT) [55] divides an image into fixed-size non-overlapping patches (e.g., 16×16 pixels), flattens each patch into a vector, and linearly projects it into a patch embedding. These embeddings, along with a learnable class token and positional encodings, form a sequence that is fed into a standard transformer encoder. Unlike Convolutional Neural Networks (CNNs), which use a convolution window (or kernel) striding across the processed image instead of processing individual pixels, the ViT captures global dependencies between image patches through self-attention. When trained and fine-tuned on large datasets, ViTs can serve as powerful backbone encoders, extracting high-level representations for downstream tasks such as classification or semantic segmentation [227]. In generative settings, they are often paired with autoregressive decoders to guide the synthesis process. While such models are effective at reconstructing image content, directly applying them to high-resolution images is computationally expensive due to the quadratic complexity of self-attention with respect to input length. To address this, Vector Quantized Generative Adversarial Network (VQGAN) [62] combines vector quantisation with generative adversarial training to learn a compact discrete latent space for images. It consists of an encoder that maps an image to a low-dimensional grid of discrete codebook indices, and a decoder that reconstructs the image from these indices. This discrete representation can then be modelled efficiently using autoregressive transformers, enabling high-quality image

generation while retaining semantic alignment and fine detail.

2.3 | Multimodal Algorithms

The announcement of Dall-E [170] by OpenAI showcased its capabilities for text-guided image generation, and attracted worldwide attention. The authors of this work tried to circumvent the constraints of the training dataset by using a large and varied corpus of 250 million images with corresponding text descriptions, scraped from the internet. The model and its weights were not publicly released by OpenAI but they released those of Contrastive Language-Image Pretraining (CLIP) [166], the classifier at the heart of its text alignment mechanism. CLIP's ability to generate embeddings for hitherto unseen text and images gave *Dall-E* its signature ability to generate novel images corresponding to text prompts describing original objects outside of the training corpus.

2.3.1 | Contrastive Language-Image Pretraining

Fundamentally, CLIP is a text and image classifier, which maps both of these modalities onto a common latent space. CLIP jointly trains image and text encoders, denoted $f_{\text{img}}(x)$ and $f_{\text{text}}(t)$, respectively, using contrastive learning. The model is trained on a large dataset of image-caption pairs, with the goal of semantically aligning related inputs from these modalities within a shared embedding space. Specifically, CLIP maximises the cosine similarity between embeddings of matched image-text pairs while minimising similarity for mismatched pairs. This alignment enables *zero-shot* image classification and cross-modal retrieval, i.e. CLIP can match arbitrary text prompts to images without requiring fine-tuning on a specific task. This leads to the possibility of computing a distance metric between any two artefacts from either text or image modalities to measure alignment using a *CLIP Score*.

Vector embeddings for any pair of assets in either modality, say an image and a text asset, may be obtained from the CLIP model. The CLIP Score of the assets is obtained by computing the cosine similarity of the two vectors and is given by Equation 2.4:

$$\text{similarity}(\mathbf{e}_{\text{image}}, \mathbf{e}_{\text{text}}) = \frac{\mathbf{e}_{\text{image}} \cdot \mathbf{e}_{\text{text}}}{|\mathbf{e}_{\text{image}}| \cdot |\mathbf{e}_{\text{text}}|} \quad (2.4)$$

where $\mathbf{e}_{\text{image}}$ is the CLIP vector embeddings of the image,
 $\mathbf{e}_{\text{text}}$ is the CLIP vector embeddings of the text,
 and $|\mathbf{e}|$ denotes the magnitude of vector $\mathbf{e}$.

Such metrics can be used as a fitness function in various generative methodologies to steer the composition of an image towards some predetermined text input [69, 108].

2.3.2 | Foundational Language And Vision Alignment

It is worth noting that despite being one of the most popular multimodal models, CLIP is not unique, as researchers have replicated and improved on its performance since its release. The Foundational Language And Vision Alignment (FLAVA) model, introduced by Singh et al. [191], is a unified vision and language architecture designed to handle a wide range of tasks involving both modalities. It distinguishes itself from other multimodal models through its integration of a multimodal transformer with modality-specific encoders, allowing it to process visual, textual, and combined inputs effectively. By jointly training on diverse datasets and tasks, FLAVA learns shared representations that support image classification, text understanding, and cross-modal reasoning. This approach enables the model to generalize across unimodal and multimodal scenarios without needing task-specific adjustments.

Unlike models that are designed primarily for cross-modal tasks, such as CLIP, FLAVA incorporates separate encoders for vision and language along with a shared multimodal encoder. This design allows it to learn robust representations not only for joint image-text tasks but also for tasks involving only images or only text, without requiring modality-specific fine-tuning. FLAVA's versatility and breadth of training give it a more general-purpose capability, although this can come at the cost of slightly lower performance on highly specialised tasks compared to models fine-tuned for those specific domains.

2.3.3 | VQGAN-CLIP Image Generation

The ability of multimodal models to align domains led to their incorporation as a semantic prompt alignment mechanism for text-to-image generators. One of the most popular CLIP-based generative techniques, *VQGAN-CLIP*, was developed by Crowson et al. [43]. The methodology was initially released through several online repositories¹ which were made openly available before the work was featured in peer-reviewed publications. This technique uses CLIP to guide the refinement of an image defined as a VQGAN embedding, and follows an iterative sequence starting from a random fractal noise array. The image embedding in the CLIP latent space is computed, as well as that for

¹<https://github.com/lucidrains/big-sleep> and <https://colab.research.google.com/drive/1L8oL-vLJXVcRzCFbPwOoMkPKJ8-aYdPN>, among others.

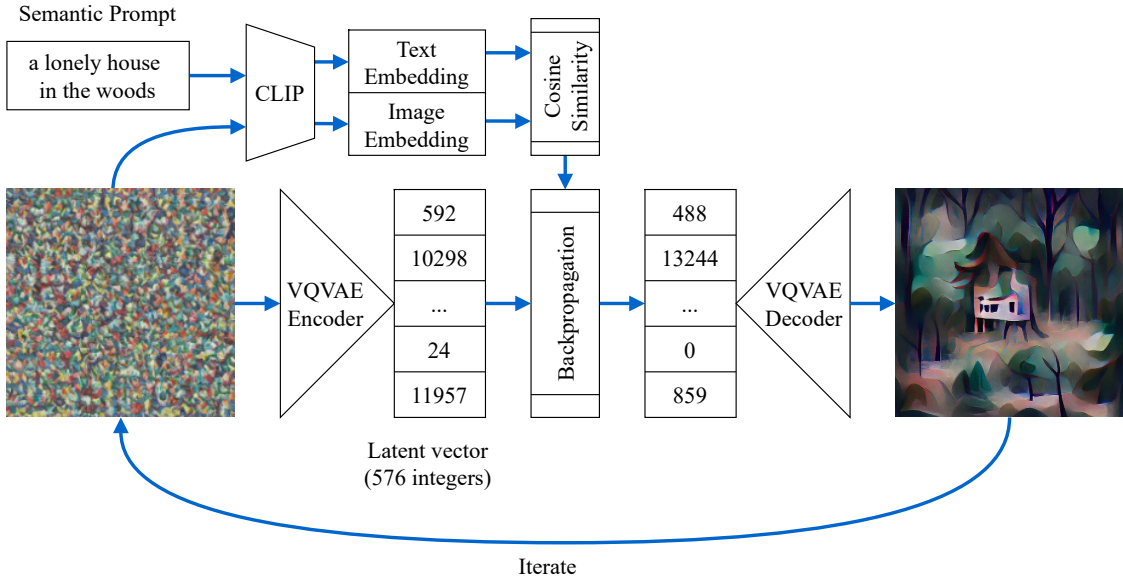


Figure 2.1: The iterative process of text-to-image generation using VQGAN-CLIP.

an input text prompt and their CLIP score, given by Equation 2.4, is computed. The encoder of the VQVAE from the VQGAN developed by [62] is then used to obtain a vector representation of the image. Backpropagation is carried out on the encoded image representation, using the VQVAE decoded image's CLIP score with respect to an input semantic prompt as the fitness function. This process is carried out over a number of iterations, and is illustrated in Figure 2.1. The CLIP-guided VQGAN technique gained significant popularity and was widely adopted by both researchers and enthusiasts.

2.3.4 | Diffusion Models

Diffusion models are a class of generative models that gradually transform data into noise through a fixed forward “noising” process, then learn a parametrised reverse de-noising process to recover samples from pure noise [88, 192] (see Figure 2.2).

In essence, given a data distribution, a Markov chain is defined, having a predefined noise distribution variance schedule that slowly increases from a near-zero value up to nearly 1, resulting in almost pure Gaussian noise after a fixed number of steps. The generative model learns a reverse Markov chain which is trained to invert the forward process, iteratively reducing an amount of noise at each time step. The result is a synthetic sample which approximates the true data distribution. The scheduling of noise

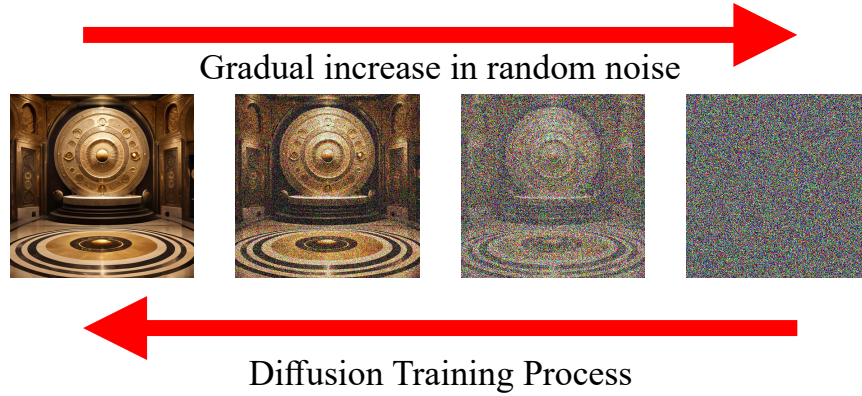


Figure 2.2: The training process of a diffusion-based image model. Noise is gradually added to images in the training dataset. The sequence of images is reversed to train the model to go from pure noise to a complete image.

application and removal affects the overall composition of the output at each step, and different schedulers may yield significantly different final outputs [104].

Text-to-image diffusion models are trained to guide the de-noising process by aligning the image embeddings to text embeddings, typically using CLIP for this comparison. An alternative approach is to train both a conditional and an unconditional diffusion model, and comparing their responses during inference, which strives to balance image quality and coherence to the input prompt. This method removes the dependency on an external classifier, and is thus termed classifier-free [89].

These models achieve a high image quality and fidelity to the given text prompt, but carrying out the diffusion operation directly on pixels is increasingly resource-intensive as images increase. To mitigate this problem, latent diffusion [175] operates the same process on image embeddings instead of pixels. Latent diffusion image generators typically use a VAE or VQVAE to obtain a latent vector representation of the input image, and carry out the noising/de-noising process on it. The VAE encoder and decoder allow a smooth transition between pixels and embedding for the image at any point. This process achieves faster inference times and utilises fewer resources as operations on the latent vector are computationally cheaper, and it has been popularised in the Stable Diffusion (SD) family of models, which have been openly released by Stability AI.

Beyond conditional and unconditional image generation, diffusion models have been effectively adapted to a range of tasks. These include image inpainting, where only the masked regions of an image are synthesized to integrate seamlessly with the unmasked content [175]; super-resolution, which employs a conditional approach to transform low-resolution inputs into high-resolution outputs by progressively refining

details through the reverse diffusion process [118]; and audio synthesis and enhancement, where diffusion is applied in the spectrogram domain to generate or improve speech and music with high fidelity [133, 183].

2.3.4.1 | Controlling Diffusion Models

While diffusion models perform well in generation tasks, practical applications frequently require mechanisms for structured control. Recent advances have introduced architectural modifications that enable diffusion models to condition their outputs on specific forms of external input, thereby increasing both flexibility and precision in generative tasks.

ControlNet [238] introduces an approach in which trainable auxiliary branches are added to a pre-trained and fixed U-Net diffusion backbone. These branches process additional conditioning inputs, such as edge maps, pose annotations, or depth information, and guide the generative process without altering the original model parameters. During training, ControlNet learns to align these external signals with the internal representations produced during the diffusion process. As a result, it can generate outputs that adhere closely to the provided conditioning information. For example, when given a pose skeleton, the model can produce a human figure that conforms precisely to the specified pose. The separation of control-specific parameters from the core diffusion model enables efficient adaptation to various conditioning tasks with minimal impact on the model's generalisation capabilities.

IP-Adapter [232] offers a distinct mechanism for control by introducing high-level visual prompts into text-to-image diffusion pipelines. It employs a lightweight image encoder to extract semantic features from a reference image, such as a sketch or stylistic example. These features are then injected into the cross-attention layers of the diffusion model, alongside tokenized text prompts. This integration allows the model to condition its outputs on both visual and textual sources simultaneously. For example, a user can provide a natural language prompt in combination with a reference image to generate content that reflects the structural or stylistic properties of the image while incorporating the semantic content of the text. The design of IP-Adapter prioritises compatibility with existing diffusion architectures and requires minimal modification to the base model.

2.4 | Quality Diversity Algorithms

Irrespective of which modalities PCG is applied to, the quality of the output artefacts within creative domains is subjective. There exists no painting or book which is liked

by each and every person in the world. There is no absolute metric that can universally determine whether one creative artefact is better than another. Some specific metrics of quality can be attributed during evaluation; for example it is possible to determine analytically which image is the most colourful within a set of images, by counting the total number of different colours, and/or measuring the difference in hues in each image. There are extensive arguments to be made on the subject, and this chapter shall also shallowly dip into this deep ocean of CC.

But it is important to note that GenAI models, being based on deep NNs and trained using backpropagation, require a strictly mathematical fitness metric which they optimise, thereby finding a single, optimal solution. A degree of stochasticity can be applied to most models to yield different outputs for the same inputs. For example, in autoregressive language models, to predict the next token can be changed from simply selecting the next most probable token to a range of strategies which include randomisation [223, 235]. A temperature parameter is often specified, controlling the orientation of the model output between more aligned to the next probable token, or more varied, increasing its ‘creativity’. In diffusion models, the random initial seed noise distribution significantly influences the final composition of the generated image. The model identifies and amplifies latent structural patterns within this noise during the de-noising process, guiding the generation process. Consequently, altering the starting random seed leads to visibly different outputs, even when using identical generation parameters [88]. The problem with using these random controls to generate different outputs is that the process is unguided, and therefore wasteful of resources.

Traditional EAs are population based, and therefore lend themselves better to offering a range of potential solutions. Their disadvantage is that they tend to hone in to a single set of desirable qualities in the proposed solutions, at the expense of others which are potentially equally valid. A subset of these, QD algorithms [79, 164], aim to discover a diverse set of high-performing solutions across a range of behaviours or niches. This focus on both quality and diversity enables the exploration of complex search spaces where multiple viable strategies may exist, each exhibiting its own Behavioural Characteristics (BCs) thereby making it suitable to specific contexts or objectives.

The core idea of QD is founded on the understanding that real-world problems often benefit not from a single optimal answer but from a repertoire of solutions that exhibit functional diversity. This is particularly valuable in domains such as robotics, game design, creative AI, and automated discovery, where adaptability and exploration are as critical as raw performance. QD algorithms formalise this approach through mechanisms that simultaneously promote novelty and optimise performance within behaviour-defined niches.

Notable algorithms in this space include Multidimensional Archive of Phenotypic Elites (MAP-Elites) [150] and NSLC [115], which exemplify different approaches to maintaining behavioural diversity while encouraging high-performing solutions. These methods have demonstrated state-of-the-art results in tasks ranging from locomotion control to PCG, showcasing their ability to discover creative and unexpected solutions that would be overlooked by traditional optimisation.

2.4.1 | Novelty Search with Local Competition

NSLC [115] is an evolutionary algorithms which is inspired by evolution in nature. Its core concept is that flora and fauna exhibit a multitude of variations in the implementation of similar functions. This implies that a novel design may not be universally optimal, but may be dominant within a local neighbourhood. That trait is thus allowed to be retained through evolution, either in its original form or as a channel towards more performant mutations, despite it not being a globally dominant feature. Such a process enables a wider variety of features to survive across several subdomains within the evolutionary space.

The algorithmic form is visualised in Figure 2.3, and is formulated as a dual-objective optimisation problem with the objectives being to maximise both a *novelty score* and a *local competition score*. The novelty score, given by Equation 2.5, is the average distance between an individual and its k nearest neighbours. Thus, it is also a measure of the sparsity of the neighbourhood within the solution space. The Local Competition (LC) score, given by Equation 2.6, represents the proportion of nearest k neighbours that the individual outperforms in fitness.

$$n(i) = \frac{1}{k} \sum_{j=1}^k d(i, \mu_j) \quad (2.5)$$

$$lc(i) = \frac{1}{k} \sum_{j=1}^k o_f(i, \mu_j) \quad (2.6)$$

where the function d represents the behavioural distance between its arguments, μ_j is the j -th nearest neighbour to i , $o_f(x, y)$ is 1 if $f(x) > f(y)$ and 0 otherwise, and f is the fitness function for the problem. This algorithm works on the hypothesis that novelty acts as a stepping stone towards better performing individuals [114]. Searching for novelty has been applied with considerable success in domains such as robotics [114, 115], video game PCG [122, 125, 126] and generative art [116]. Several works of this type apply novelty search to the evolution of the architecture of NN [193], but it can also be applied to that of the population, either directly [126] or indirectly [120, 122].

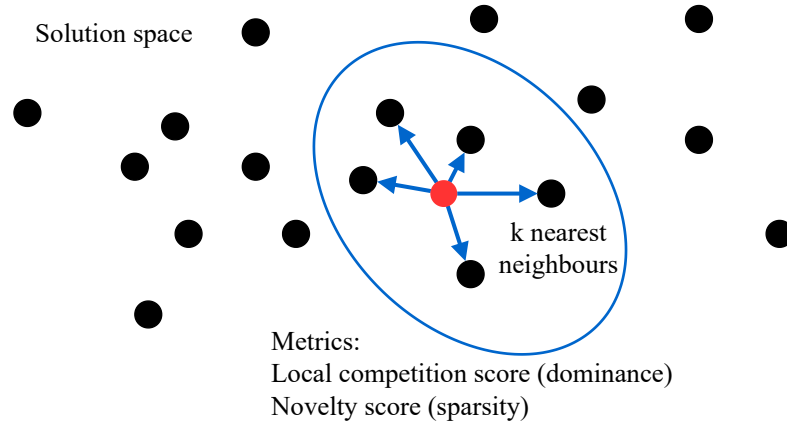


Figure 2.3: The principle of operation of the NSLC algorithm. Every individual is compared to its k nearest neighbours, computing its dominance within the neighbourhood, as well as the sparsity of this region as a measure of its diversity.

2.4.2 | Multidimensional Archive of Phenotypic Elites

Another influential algorithm within the QD paradigm is the MAP-Elites [150]. It is designed to simultaneously search for high-performing and diverse solutions by explicitly partitioning the behaviour space, defined by a set of user-specified BCs, into a discrete, multi-dimensional grid, known as the feature map.

Each axis of the feature map corresponds to a particular behavioural descriptor, or phenotypic trait, of the solutions, such as gait frequency in robot locomotion or level linearity in game design. These continuous behavioural dimensions are discretised into a finite number of intervals (bins), effectively tessellating the search space into a structured grid. For example, if two behavioural dimensions are each divided into n bins, the resulting feature map will contain $n \times n$ cells, where each cell corresponds to a distinct region of the behaviour space.

During the evolutionary search process, a parent is randomly selected from the current set of elites stored in the archive at each iteration. This parent undergoes a variation operator, commonly mutation, to generate a new offspring. This offspring is evaluated according to two criteria: its performance with respect to a global fitness function, and its location in the behaviour space as determined by the chosen BCs. The algorithm maintains a single elite, i.e. the best solution found so far, in each cell of the grid. An offspring is considered for inclusion in a cell if its behavioural descriptors fall within the bounds of that cell. If the cell is unoccupied, or if the new solution has a higher fitness than the existing elite, it replaces the previous occupant. The process

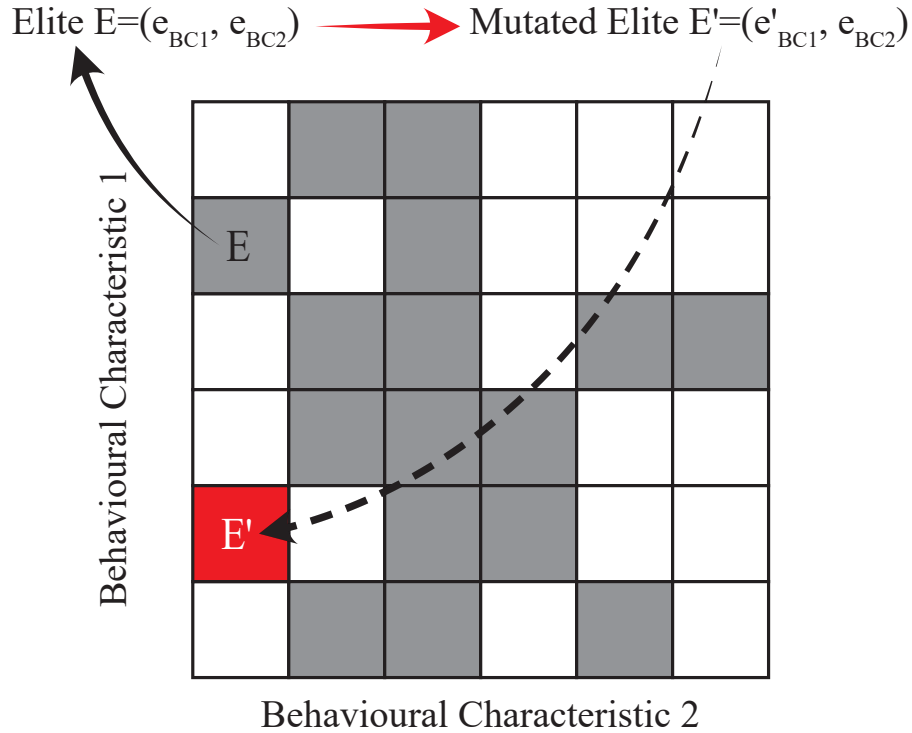


Figure 2.4: An iteration of the MAP-Elites algorithm. An elite E is chosen from the feature map and mutated. Its new placement is found based on its new BCs. If the new location is unoccupied, or if E' has a higher fitness than the elite at that location, it is placed there, replacing any previous occupant.

is illustrated in Figure 2.4. This mechanism promotes a form of localised competition wherein individuals compete only with others that exhibit similar behavioural traits, encouraging the discovery of a diverse set of high-performing solutions throughout the behaviour space. MAP-Elites is said to *illuminate* the search space, as the gradual construction of this diverse archive reveals which kinds of behaviour lead to better performance. This makes it particularly effective in creative or open-ended domains such as PCG and design exploration, where MAP-Elites has demonstrated considerable success [3, 2, 32, 67, 79, 213].

2.4.3 | Monte Carlo Elites

Sfikas et al. [186] propose a variation on the MAP-Elites algorithm called *Monte Carlo Elites*. This algorithm addresses the parent selection in each iteration by using a Upper Confidence Bound (UCB) algorithm [110] and keeping track of how many times each

elite was selected.

The UCB algorithm is employed as a solution to the multi-armed bandit problem, where an agent must balance exploration of unknown options with exploitation of currently promising alternatives. The algorithm is designed to address the fundamental trade-off between gathering information about uncertain actions and maximising immediate rewards based on current knowledge. In this case, the selection of which parent to operate on is considered to be a similar problem, as the MAP-Elites algorithm benefits from both the further optimisation of existing elites which have already been selected, and also from diversifying its feature map by selecting less utilised elites. The UCB for each existing elite is obtained through Equation 2.7:

$$UCB_i(t) = \bar{f}_i + \sqrt{2} \sqrt{\frac{\ln t}{n_i(t)}} \quad (2.7)$$

where $UCB_i(t)$ is the upper confidence bound for elite i at selection t ,

$\bar{f}_i$ is the average fitness of offspring produced by elite i ,

t is the total number of parent selections so far,

$n_i(t)$ is the number of times elite i has been selected.

UCB operates by maintaining confidence intervals around estimated reward values for each available elite individual. An upper confidence bound is calculated for each option based on both the empirical mean reward observed thus far and an uncertainty term that decreases as more observations are collected. The parent with the highest UCB is selected at each time step, ensuring that both high-reward actions and insufficiently explored actions receive appropriate consideration [110].

The addition of this equation to the parent selection mechanism exhibits a consistent improvement in feature map coverage than other methods, with a near-negligible increase in computational overhead [186].

2.5 | Summary

This chapter has provided a comprehensive overview of the foundational technologies deemed essential for a thorough understanding of the research presented in this thesis.

In the following chapter, prior work relevant to the fields of PCG and GenAI will be reviewed, offering a contextual basis for the methodologies and contributions discussed herein.

Background Work

This chapter examines the existing body of work that constitutes the foundational basis of the domains investigated over the course of this thesis.

The scope of the current work is deliberately restricted to the domains of image and text generation, wherein the state-of-the-art in GenAI has demonstrated a notably high level of sophistication with respect to the quality of generated outputs. These two modalities have been the focus of considerable academic and industrial research, resulting in the development and public release of numerous high-performing models and algorithmic frameworks.

Section 3.1 starts by adopting a CC lens to approach these generative processes from an interpretational aspect, to establish their defining characteristics.

Section 3.2 provides a general exposition of GenAI, encompassing a concise historical overview and an outline of its strategies for multimodal content generation. This discussion paves the way for Section 3.3, which details some important work in PCG and explores the influence of GenAI within the domain of games.

Finally, Section 3.4 shall investigate a number of relevant evaluative methodologies.

3.1 | A Computational Creativity Lens

CC is a prolific field that explores various aspects of PCG systems for creative content in domains such as visual arts, music, literature, and game assets. Unlike traditional AI applications focused on optimisation and efficiency, CC seeks to understand, model, simulate, or enhance the human creative process itself [12]. The field encompasses a range of generative techniques, from rule-based systems to deep learning models, some of which have already been described in Section 3.2. The focus on a broad understanding of creativity is important in this field; Colton [31] suggests that researchers should build creative systems without preconceptions or definitions, so that an understanding of what creativity is can be garnered from studying the creations. A clear-cut definition of creativity remains elusive. Jordanous [99] acknowledges that “we assume an intuitive

understanding of the concept of creativity but lack a universally accepted and comprehensive definition of the concept.” Human creativity is often linked to attributes such as originality, aesthetic appeal, and contextual relevance. Such qualities do not lend themselves well to quantification or precise measurement. When applied to machine-generated content, these characteristics become even more difficult to define or assess rigorously and systematically [12, 34, 99]. Yet CC has yielded several interpretations of what defines a creative system, or rather, what features or aspects such systems should have [12, 13, 34, 99, 222]. These categorise different types of creativity and provide alternative approaches to evaluation for creative systems in general.

Colton et al. [34] developed the FACE and IDEA models in CC theory, in order to break down creative processes into components. The FACE model decomposes creative acts into subprocesses and resulting items, namely those pertaining to *Framing* information given in natural language, *Aesthetic* measures, *Concept* generation, and *Expression* of concepts. The IDEA model focuses on the process of developing creative systems, describing an *Iterative Development*, *Execution*, and *Appreciation* cycle. Jordanous [99] also formulates a system for evaluating creative systems. The method she proposed is an overall strategy for evaluation, where one should first define an interpretation of creativity within the active domain that aligns to the generative system’s purpose. This definition is translated into appropriate measurable standards, which are finally used to test the system using both qualitative and quantitative metrics. These frameworks, primarily intend to assess whether the *system* is creative, and do not explicitly provide means of evaluating the generated artefacts themselves.

This thesis does not explicitly seek to explore whether the systems in use can be deemed creative through such formal definitions. Rather, it uses CC as a lens through which to interpret certain properties of generated assets in relation to their peers, similar datasets, or other assets of similar or contrasting modalities. Specifically, it is important to ground certain terms within existing CC frameworks in order to clarify the intentions and assessment of the experimental work.

In the influential work on the evaluation of creative systems, Ritchie [173] proposed that creativity could be analysed in terms of three interrelated but distinct properties: *quality*, *typicality*, and *novelty*. These concepts were introduced to provide an informed basis for assessing generated outputs that is not a subjective or purely aesthetic judgment.

Quality was defined as a measure of the positive value or merit of an artefact, typically assessed in terms of its effectiveness, usefulness, or appeal within a particular domain. It was emphasised that this evaluation must be grounded in domain-specific criteria, such that an artefact’s quality is determined in relation to the standards and expectations of the relevant field of practice.

Typicality was conceptualised as the degree to which an artefact conforms to the norms or patterns associated with a given class of artefacts within the domain. It was acknowledged that while highly typical artefacts may lack surprise or originality, they often possess recognisability and coherence, which can contribute positively to their perceived value. In this formulation, typicality serves as a baseline against which novelty can be assessed.

Novelty was defined as the extent to which an artefact deviates from existing examples within the domain. It was characterised as a necessary but insufficient condition for creativity, as novelty alone does not guarantee quality or appropriateness. Instead, novelty must be balanced with typicality and quality to ensure that creative artefacts are both innovative and meaningful within their context.

In addition to these definitions, it is also useful to distinguish between terms which may bear resemblance to them and their use in CC parlance.

Value refers to the perceived worth or impact of the artefact, often in a social or contextual sense. Both quality and value aim to assess the merit of a creative artefact, often after it has been generated. They both imply a positive assessment, however, the source and scope of the judgment differ [79, 101].

Diversity is the degree of variation within a set of artefacts generated by the same system. Although this term may resemble and reflect novelty, and they are often positively correlated, one does not necessarily imply the other. Jordanous and Keller [101] identify both novelty and diversity (referred to as variety/divergence) as separate components in their 14-factor model of creativity.

Finally, this thesis borrows the term *appreciation* from Colton [31], which defines a tripod framework of skill, imagination, and appreciation as three key pillars of creativity. Appreciation here is understood as the system's capacity to evaluate its own outputs, compare with past work, and understand contextual relevance.

This thesis shall make use of the above definitions to interpret the outputs of the generative models used.

3.2 | AI for Text and Image Generation

GenAI systems have greatly advanced across a wide variety of modalities, including natural language, visuals, audio and more. Their goal goes beyond classification or analysis, instead they produce artefacts that exhibit the structural, semantic, or aesthetic character of their training data. Fundamentally, GenAI models construct an implicit representation of the data distribution and sample from it, so as to generate outputs that

a human observer may find plausible or even indistinguishable from real examples.

3.2.1 | Text generation

The earliest attempts at AI-driven text generation were based on explicitly defined rule-based systems. Whilst capable of generating coherent sentences, these lacked contextual awareness and flexibility. One of the most noteworthy developments in this era was the introduction of Generative Grammar by Chomsky [24], which defines how smaller linguistic units combine into larger structures, using formal rules that reflect the syntax of a language. This linguistic theory laid a structured foundation for early computational approaches to language generation leading to the development of models capable of rudimentary dialogue or narrations. From the early 1960s, Weizenbaum [219] notably introduced *ELIZA*, one of the earliest chatbots, which used pattern-matching templates to generate responses to mimic a Rogerian psychotherapist [174]. While it lacked true understanding of the conversations it was engaged in, it demonstrated how simple rule-based systems could emulate human-like interactions. Similarly *SHRDLU*, developed by Winograd [225], used augmented transition networks [226] to generate text responses in a controlled environment, allowing it to engage in basic conversations about a virtual world of geometric shapes. Despite the fact that these models were unable to adapt to topics other than those predefined for them, they were considered influential milestones for later developments in chatbots.

Statistical methods such as Markov chain text generators proposed by Shannon [189], and n-gram language models [17] next emerged, in which token sequences were sampled according to learned transition probabilities. Such models produce random-walk sequences that can mimic surface word patterns but lack coherence. Despite this, they could be applied to predictive applications such as autocomplete assistance, or predictive text suggestion.

Surface realisation frameworks including FUF/SURGE [98] then allowed semantic representations to be rendered as grammatical English sentences. These models began to approach the composition of coherent sentences, and could be applied to practical use. IBM's Statistical Machine Translation models [16] which emerged during the 1990s demonstrated their practicality. These used vast amounts of bilingual text to generate translations based on statistical correlations.

While statistical models improved fluency and coherence in AI-generated text, they still struggled with long-range dependencies of words or topics, and often resulted in disjointed or repetitive outputs. Although they fared well in topics such as translation, long-term coherence was crucial for automated narrative generation. Narrative planners

were developed to generate goal-driven stories by simulating character motivations, conflicts, and resolutions. One of the earliest AI storytelling systems, *TALE-SPIN* [144], modelled characters as autonomous agents with desires and goals, generating narratives based on their decision-making processes. Later, *MINSTREL* [205] applied case-based reasoning to generate short stories based on characters from the court of King Arthur. It reused past story structures to create new parts of the narrative, thereby maintaining continuity.

In the early 2000s, *MEXICA* [159] was designed to write short stories about the Mexicas (or Aztecs). It introduced emotional tension modelling, using dramatic arcs to make the generated stories more compelling. Around the same time, *Façade* [141] introduced real-time interactive storytelling to video games, creating a procedural narrative system where AI-driven characters reacted dynamically to the player's free textual input. Such works created a library of generative processes, each having a specific focus. Montfort et al. [149] combine several existing subsystems using a blackboard architecture [151] to coordinate the final narrative generation and exchange data between the various components throughout the process.

EAs have also been used to generate poetry [139] and stories [76, 143, 153] through genetic operators acting on various formats of text representations, however the inherent difficulty in interpreting and assessing the sequence of narrative events invariably lead to the generative process focusing more on the plot rather than the text itself.

Neural network architectures rapidly accelerated the development of more capable sequential models. Mikolov et al. [146] proposed RNN language model which pioneered the prediction of successive tokens via learned embeddings and recurrent dynamics. It was one of the earliest iterative models able to generate coherent sentences. Subsequently Seq2Seq [197] used an encoder-decoder model consisting of two stacked LSTMs that maps an input sequence to a fixed vector and then decodes it to an output sequence. This model was capable of conditional text generation, and successfully applied to translation tasks.

With the introduction of the transformer [211], the attention mechanisms supplanted recurrence, yielding state-of-the-art performance in machine translation and broader text generation. BERT's [50] bidirectional pre-training further enhanced masked token prediction. Despite not being intended primarily for sentence generation, it could fill in missing text sequences, or answer questions about a given context.

Autoregressive language models then scaled dramatically. The GPT family [18, 168] employed unsupervised pre-training to generate coherent continuations of text. GPT 2 and 3 expanded this paradigm, demonstrating few-shot generalisation across creative and factual domains. They could be fine-tuned to virtually any domain for text generation

or continuation.

These models led to a rapid advance in conversational systems, such as DialoGPT [242], which featured a GPT-2 model fine-tuned on Reddit dialogues for conversational response generation.

Several models emerged demonstrating excellent performance when fine-tuned for generative tasks such as dialogue, summarisation, and translation. Notable examples include BART [117], T5 [169], and BLOOM [180] among many others. These paved the way for the latest LLMs listed in Section 2.2.3 of Chapter 2.

3.2.2 | Image generation

Procedural image generation has been investigated since the earliest days of computing. Rather than recounting its progress chronologically, however, in this section, the works are investigated by their generative purpose, as this will offer a more meaningful taxonomy than a simple historical sequence. Four different categories of models are considered, namely *Simulation*, *Expression*, *Exploration*, and *Transformation*. This taxonomy is not without overlap, as some algorithms may easily be classified into more than one group.

3.2.2.1 | Simulation

Within this category works aim to emulate or replicate an existing category of artefacts. Early generative approaches relied on explicit rules and procedures to create visual patterns. Generative grammars [56] and fractal algorithms [138] have long been used to model natural forms. Lindenmayer systems [129], or *L-systems* as they are commonly referred to, use rewriting rules to emulate the algorithmic structure of plant growth. Fractal constructs produce complex self-similar imagery by iterating simple functions. These systems aim to emulate existing categories of visual objects (e.g. photographic textures or plant forms) using hand-crafted or statistical rules. Some of these works, such as the development of Perlin noise [160], have become mainstay tools for visual generation to date [137, 188].

Deep learning algorithms have also been extensively used for image generation, primarily due to the fact that they lend themselves well to problems for which large corpora of data exist. Several image datasets are readily available, such as *ImageNet* [49] which contains photographs of real objects classified in various categories, *WikiArt* [179] which is composed of images of artworks in various styles with some basic descriptive qualities, and *CelebA* [131] comprising of photographs of the faces of international celebrities annotated with distinguishing facial features. Various NN architectures have been

trained and tested on such datasets, and they have been successful in reproducing the typicality of the learned corpora [62, 105, 199].

Autoregressive and latent-variable models have played a major role in simulation-style image generation. PixelCNN [208] and PixelRNN [207] models treat an image as a sequence of pixels and predict each pixel in order using masked convolutional or deep recurrent networks respectively. Such models can generate high-quality images by ensuring local coherence. van den Oord, Kalchbrenner, and Kavukcuoglu [207] reports about PixelRNN that “samples generated from the model appear crisp, varied and globally coherent.” These models are trained to capture the full distribution of image pixels, enabling them to sample entirely new images that exhibit the typicality of their training data.

In VAEs [109] an input image is mapped by the encoder to a latent vector distribution, and the decoder maps a sample from that vector back into image space. This allows the model to generate plausible images by sampling the latent space. It excels at capturing global structure but often produces images which are more blurred in comparison to those generated by adversarial networks.

In a GAN, a generator network is trained to produce images that a discriminator network cannot distinguish from real examples. Goodfellow et al. [78] demonstrated that, the adversarial context leads the generator to reproduce the true data distribution if both networks have sufficient capacity. In practice, GANs exhibited an unprecedented ability to generate sharp, realistic images. Subsequent variations of the model improved stability and quality, yielding state-of-art photorealistic samples. *DCGAN*, by Radford et al. [167], uses a stack of strided and transposed convolutions in the generator and discriminator respectively, to map random noise $z \sim \mathcal{N}(0, I)$ into images. A common usage for this model is to interpolate within the latent space to smoothly morph one face into another. Progressive GANs [103] grow both generator and discriminator progressively, starting from low resolution (e.g. 4×4 pixels) and adding layers to reach high resolution (e.g. 1024×1024 pixels). This yields very high-fidelity portraits and landscapes.

StyleGAN [105] introduced a style-based generator that controls images at different spatial scales. It automatically separates high-level attributes, such as object pose and identity, from stochastic detail, like texture and colour patterns. StyleGAN is known for generating highly realistic high-resolution human faces and other objects by blending these separated “styles” at each layer of the network. Modern GANs like StyleGAN are widely used to simulate domains across multiple datasets, producing photorealistic images by learning directly from large image collections [42]. GAN-based models are far too numerous to comprehensively list, and they have been put to several uses, including both guided and unguided generation. Aside from the simulation category, such models

have been applied to transformative applications, such as *pix2pix* [95] which will be described in Section 3.2.2.4.

Recently, diffusion-based models have emerged as a powerful simulation algorithm [88]. These probabilistic models define a forward process that gradually adds noise to data and a learned reverse process that reconstructs images from noise. Diffusion models can match or exceed GANs in sample quality. Ho et al. [88] showed how their model achieved state-of-art scores on standard benchmarks and generated images comparable in quality to GANs. Diffusion methods are particularly versatile. They are likelihood-based and avoid issues like mode collapse, and they can naturally incorporate classifier guidance to align the generated artefacts to a predetermined input, such as text. A later development, Latent Diffusion [175], applies diffusion in the latent space of an autoencoder to reduce computational cost and enable efficient conditioning on text. Latent diffusion models such as Stable Diffusion XL [162] have achieved new state-of-art performance on tasks like text-to-image generation and image in-painting, producing high-resolution images efficiently with relatively low computational demand.

3.2.2.2 | Expression

Works of this type seek an artistic or aesthetic goal beyond mere replication of real-world imagery. Pre-dating neural networks, systems like AARON, proposed by Cohen [29], exhibit this expressive intent. The project started in the 1970s and continued through the early 2000s. AARON was a rule-based painter that used a formal set of extensively detailed drawing rules and heuristics to mimic humans' expressiveness in the visual arts. Cohen defined a vocabulary of shapes and composition rules, and the program autonomously generated novel drawings with an artistic "feel" despite being deterministic. Despite being well received by the CC community and AI researchers, Cohen [29] states that his work did not manage to make the impact he desired within the artistic community, despite his efforts towards that end. However, the works of AARON have since been exhibited in museums [157], so Cohen's work has eventually been acknowledged as Art, posthumously fulfilling his expressed desire. AARON as a system went through several stages, first of which involved the generation of abstract shapes and lines in various colours. Further on, its output transitioned to an impressionistic rendition of rocks and vegetation, before its images became more figurative, depicting human figures, and potted plants. This complexity, and the autonomy of the system are quite remarkable, being the result of symbolic AI.

EAs have been used to generate images in this category without requiring to define an instruction set. Colton and Browne [33] generate images by evolving the placement,

scale, rotation, and colour of geometric shapes. These are then combined with evolved sequences of interactive elements chosen from these shapes to generate simple ‘puzzle’ games where users have to discover the correct sequence. Machado and Cardoso [134] propose the *NEvAR* system, which evolves images through a set of asynchronous GA experiments, with different populations, allowing the interchange of individuals between distinct processes. The genotypical representation of each image is a tree structure, which is subjected through recombination and mutation operators to evolve the image population in each experiment.

The Painting Fool, an extended experiment driven by Colton [31] is another notable system which followed in the footsteps of AARON. It was made of a mixture of rule-based and evolutionary algorithms, and its objective was to create a digital artist, rather than a tool to assist human artists. To that end, it was not intended to be driven by human stimuli, but rather to generate visual compositions independently. In one of its iterations, for example, it was designed to assess the mood from a video of a person, and create a non-photorealistic image that aligns with that emotion, through the chosen colours and painting techniques [35].

These early efforts treated creative style and composition as explicit algorithmic processes, but in the deep learning era, style and aesthetic value became entwined with model design. Google’s *DeepDream* [1] turned neural networks into “dreamers” by amplifying patterns in images, creating psychedelic, abstract renditions that emphasise visual motifs learned by the network. These models are geared toward subjective aesthetic effects rather than accurate simulation. Researchers have also explicitly aimed for creativity and novelty. Elgammal et al. [61] introduced the Creative Adversarial Network (CAN), a variation on the GAN that is trained on artworks but modified to encourage stylistic innovation. In CAN, the generator not only tries to fool a discriminator but also maximises deviation from known art styles, thereby intentionally producing images that are unusual yet still recognizable as art. Human evaluators rated its outputs as creative or comparable to contemporary human-made art [61].

3.2.2.3 | Exploration

This category focuses on works which are intended as creative tools that explore or suggest ideas rather than as creators themselves. These systems generate a range of options or variations to inspire human artists. one of the most representative examples of this taxonomy is *Picbreeder* [185], a collaborative and interactive evolutionary platform. *Picbreeder* allowed users to evolve images by selecting individuals they found appealing, thereby branch off from any published image. The engine behind this interface used

NeuroEvolution of Augmenting Topologies (NEAT) [193] to mutate and combine image parameters. As a result, a rich phylogeny of images emerged. Users could browse and select novel and surprising forms, effectively exploring the space of images without specifying an explicit aesthetic objective. This illustrates exploratory creativity; the system generates artifacts that spark new ideas, but final creative judgment and refinement are left to humans.

Tools such as *GANSpace* [84] provide systematic ways to navigate the latent space of trained generative models. *GANSpace* applies principal component analysis [136] to the latent features of a GAN and finds directions corresponding to interpretable image alterations, such as changing an object's orientation, lighting, or even the inclusion of novel objects. By moving along these directions, a user can quickly generate coherent variations of an image. For example, one principal direction might correspond to "aging" a face or shifting day to night in a scene. These interfaces do not aim to produce a single finished artwork but rather to present many semantically meaningful options. Artists and designers are then able to select and refine from these suggestions. Similarly, platforms like Artbreeder¹, an online tool built on GAN models, let users combine and tweak latent vectors collaboratively to spawn hundreds of image variants. All of these tools showcase the role of muse these models assume. They illuminate and explore regions of the creative space and seed inspiration, rather than seeking a singular output.

3.2.2.4 | Transformation

This section gives a brief survey of algorithms which transform one or more images into another. They are not intended to generate an image completely from the ground up. Typically, these algorithms preserve some structure from the input image, while altering its other aspects.

A number of statistical models applied to texture generation embody this category. Portilla and Simoncelli [163] proposed a statistical texture model using joint wavelet statistics, synthesising images by matching parametrised image constraints. Example-based methods such as image quilting as proposed by Efros and Freeman [60] stitched together patches from source images to generate novel textures or transfer style between images.

A key transformative class of algorithms encompasses style transfer. Neural style transfer [73] is a practical example. It repurposes deep convolutional features to separate content from style in an image, allowing to freely interchange stylistic components between the content of images. Thus, given a photograph and a painting, the algorithm

¹<https://www.artbreeder.com/>

generates a new image that maintains the scene of the photo while adopting the brush strokes or textures of the painting. This method generates images primarily valued for their artistic texture and composition rather than fidelity to a real object. Style transfer was also accomplished through StyleGAN [105], which was described in Section 3.2.2.1.

More generally, image-to-image models convert one representation of an image into another. The pix2pix framework [95] demonstrated that a single conditional GAN architecture can learn many such transformations. Pix2pix was trained on paired datasets and applied to tasks like turning building labels into photorealistic facades, turning sketches or segmentation maps into images, and colourising greyscale photographs. The same architecture learned these mappings without requiring a distinct fitness function for each task. This gave pix2pix a high versatility and it rapidly became a popular tool within the community as it could be repurposed for creative editing without expert tuning.

CycleGAN [243] extended this idea to unpaired settings by enforcing cycle-consistency. In practice, this means that one network learns to map images from domain X to Y, whilst the other learns an inverse mapping, with the constraint that mapping back yields the original image. CycleGAN can therefore perform style swaps between domains without exact pairs, for example converting daytime to nighttime scenes, or horses to zebras.

Diffusion models [175] may also be used as image-to-image models, by adding noise to the input image, and starting the generative diffusion process from that point. ControlNET [238] models can also be expressly trained for a specific image generation constraint, e.g. application of a specific pose for humanoid figures, or following the contours of a given outline. When such a model is added to a diffusion model, the generation is constrained by an input image. Similarly, an Image Prompt Adapter (IP-Adapter) [232] will extract predominant characteristics from an image and constrain a diffusion model's pipeline to include them in the generated images. Thus the resulting image will contain similar features to the reference image supplied to the IP-Adapter. This can be used to transfer artistic style, or to preserve the the facial features of a character across different generative processes.

These transformation techniques are widely used in creative contexts. Style transfer or CycleGANs are used to apply artistic filters to images, colourise old monochromatic photographs, or even complete missing regions (in-painting) in an image.

3.2.3 | Multimodality

Generally speaking, multimodal generation refers to the concurrent creation of artefacts each of which pertains to different domains, such as text and images. However, this

term can arguably be also applied to processes that use input from one modality to generate a corresponding and related asset that pertains to a different modality. Within that definition, text-to-image GenAI models are multimodal models, and incorporate metrics that relate both input and output modalities. This is an important consideration for this work as it relates to the first exploration undertaken in Chapter 4.

Guided image generation using natural language has been a long-standing goal and has been studied before the advent of foundation GenAI models. Reed et al. [171] used GANs to generate images from a semantic description. The study was trained to generate images of either birds inferred from the Caltech-UCSD Birds dataset [215], flowers from the Oxford-102 Flowers dataset [152], or a wider range of objects from the Microsoft COCO: Common Objects in Context dataset [128]. The work demonstrated images that interpreted the textual input fairly well on the birds and flowers domains, but struggled on the generic images of the COCO dataset. Despite achieving successful guidance, the models still suffered from being limited to the domains of their training corpora.

CLIP marked a major milestone for multimodal generation. Its ability to generate embeddings for hitherto unseen text and images gave OpenAI's Dall-E [170] and its subsequent iterations their signature ability to generate novel images corresponding to text prompts describing original objects outside of the training corpus. Unlike prior captioning or conditional models, Dall-E treats the text as part of the same autoregressive stream as the image tokens, allowing it to model more complex relationships between them. It was evident that OpenAI's results could only be achieved through high computational and financial costs. Furthermore, the training datasets for DALL-E and CLIP were never released, compounding the difficulty of training similar models. Despite this, through concerted collaborative efforts, the research and hobbyist community collaborated towards the creation and open-sourcing sufficiently large, hand-crafted, or generated datasets [184] to facilitate future multimodal models.

Contemporary GenAI models are less constrained to narrow corpora due to the vast variety within their training data. However, a wide range of these, such as VQGAN-CLIP [43] and the SD family [162, 175], still rely on CLIP guidance for their generative, training, or evaluative processes.

In broad terms, multimodal generation effectively introduces additional constraints on the creative process. Not only must the generated artefacts have value within their domain, but they also need to be coherent to other artefacts used or produced throughout the process. Despite focusing on image and text domain, it is worth delving into a very brief consideration as to how advancements in these domains can be reflected onto other modalities.

In the aural domain, *AudioCLIP* [83] extends CLIP's capabilities by also mapping

audio to text and images. Other Large Multimodal Models (LMMs) have also been developed to encompass even more modalities; *ImageBind* [77] unifies text, images, audio, depth maps, thermal maps, and Inertial Measurement Unit readings. Thus it is able to pair or compare any set of artefacts across these domains. Google's *Gemini* [74] is also a powerful multimodal model which was trained on text, images, audio and video. Some of these LMMs however, are autoregressive and do not directly map inputs to a fixed-length latent tensor in the same way as CLIP or AudioCLIP do. Thus, despite their ability to generate new artefacts across all modalities, the latent tensors they provide may vary and are difficult to use for comparative metrics.

Closing this brief parenthesis on other modalities, and having established the technical foundations behind GenAI, this text shall now focus on the ways in which these powerful multimodal models have reshaped the domain of PCG.

3.3 | Procedural Content Generation in Games

PCG excels in the rapid creation of numerous adaptive outputs, and is invaluable to industries where scalability and personalisation are crucial. Systems which are able to create novel content reduce the manual labour required to craft each element, and facilitate emergent experiences that can respond to user interactions. Video games are a notable example where procedural generation has been utilised to increase replayability and variety, whilst decreasing development time. Games such as *Elite* (Acornsoft, 1984), *Minecraft* (Mojang Studios, 2011), and *No Man's Sky* (Hello Games, 2016) pioneered PCG algorithms to generate seemingly infinite spaces and worlds for players to explore. Others such as *Diablo* (Blizzard North, 1997) combined PCG with hand-crafted designs to craft new dungeons at each playthrough, for virtually infinite replayability.

Practical applications of PCG extend beyond entertainment. In education and training simulations, procedurally generated scenarios provide learners with diverse challenges tailored to their interactions [93]. Furthermore, in architectural design and urban planning, parametric design (a paradigm of PCG) helps generate conceptual models and quickly iterate through design alternatives, resulting in a more dynamic exploration of creative solutions [70].

Video games remain nonetheless a prime exponent of multimodal content which lends itself well to PCG. Games are composed of several facets; narrative, visuals, audio, levels, rules, and gameplay as defined by Liapis et al. [124]. PCG in any one of these domains is already subject to a number of constraints. Narrative generation needs to take into account pre-existing characters, locations, items, events and relationships. Sprite

sheets used to animate different characters in a two-dimensional (2D) game will often require a specific number of frames depicting the character in a predetermined sequence of poses. Tilesets used to generate the landscape within the game require the individual tiles to appear contiguous when placed next to each other. Generating levels for a game must ensure that every level is playable and is sufficiently entertaining for a human player. Extensive work has been carried out within each of these domains to overcome such constraints, at least in part. Shaker, Liapis, et al. [187] show how constructive algorithms can be used to generate functional levels and layouts, whilst Khalifa et al. [107] use reinforcement learning [198] to do so. Cohen et al. [30] tackle the continuity of tile-based image generation using Wang tiles, although the work does not directly carry over to tileset generation in games. Each generated asset must also be coherent with the rest of the game, and therefore must be consonant to the other assets across all modalities. Cheerful music would sound inappropriate in a game with dark and gloomy visuals. The images of locations and characters have to be consistent with their description within the game narrative. The level design must comply to the rules and mechanics, and often to the pace or rhythm of the accompanying music.

It is therefore useful to consider some important research that has been pursued within the domain of digital games, and how it relates to multimodal content generation. Some of the most explored domains within PCG relate to level generation. Although such works occasionally provide an interesting window on multimodal generation, they are not the focus of this thesis. *Sonancia* [132], for example, is a system that couples a background musical track to a procedurally generated level in a horror-themed game, adapting it to complement the level design.

More comprehensive multimodal examples may be observed in works related to automated game generation. Relatively few successful examples exist in this field, the most notable ones being *Game-o-Matic* [204], the various incarnations of *ANGELINA* [40, 38, 39], and *Puck* [37]. These procedurally create entire games either through a sequential process wherein separate generators each focus on a specific facet, or alternatively, the facets are generated concurrently, then matched to a specific concept. Another approach for automated generation is to keep specific facets fixed, most commonly the game rules, and generate others [80, 121, 234]. Regardless of which method is being employed to generate assets or even which asset is being generated, the output facets have to be consistent and coherent. This is a far from trivial problem. Cook [36] highlights the difficulties encountered when attempting to create a three-dimensional (3D) ‘walking simulator’ game generator. The primary stumbling blocks mainly concerned the coherence between the different elements of the game, e.g. the relative sizing of the generated objects or the semantic identification of props.

3.3.1 | Implications of GenAI on PCG

Since outputs from GenAI models exhibit strong semantic ties to their input prompts, coherence across modalities has become increasingly attainable. Contemporary multimodal architectures such as CLIP are capable of quantifying modal alignment, and have thus become essential in guiding text-to-image generation. Indicatively, diffusion-based sprite generators [165, 195] typically incorporate distance metrics derived from CLIP or similar models to ensure semantic fidelity. Nevertheless, despite these advances, there are still limitations to overcome. For instance, Qiu [165] proposes diffusion models to generate 2D spritesheets, consisting of sets of character images in sequential poses collectively forming an animation cycle. However, the method requires manual intervention and iterative refinement, and it offers limited control over generated poses. This complicates their integration into standard game development pipelines, and forces additional manual work for inclusion of the generated spritesheets into the game engine. Similar limitations are frequent in analogous works.

An attempt to automate this process more fully has been proposed by Sun et al. [195], who describe a pipeline for character generation for 2D video games. However, their approach relies on pose-control models that must be trained individually for each character archetype and artistic style, thereby restricting its applicability across diverse game genres.

Beyond the visual modality, LLMs have increasingly been positioned as creative agents in the orchestration of interactive narratives. Because of these models' understanding of a broad range of topics and domains, Gallotta, Todd, et al. [72] argue that they can be used within games in several roles, namely as player, ancillary non-player characters, a player's assistant, a commentator, a performance analyst, a game master, part of the game mechanics, an automated designer, or a design assistant. Focusing on the roles that generate game content within this taxonomy, a number of studies stand out.

In *1001 Nights* [68], a LLM is cast in the role of a king whom the player must appease through storytelling. The king responds to the player's narrative contributions (effected through natural language input) and attempts to divert the story's direction. The player's objective is to induce the model into mentioning predetermined keywords that represent weapons or items which may aid their escape. The resulting narrative emerges procedurally, with associated visual content rendered dynamically through diffusion models. A comparable, albeit structurally distinct, system is introduced in *PANGeA* [21], wherein the LLM generates narrative sequences for a turn-based game, and also appropriate responses to the player's actions. Similarly, Simon [190] integrates

a narrative planning framework that serves to constrain and guide the generative output of the LLM, to improve coherence across successive narrative events. Further exploring this narrative dimension, Kelly et al. [106] propose a model capable of transforming high-level narrative plans into fully articulated stories and vice versa. Värtinen et al. [210] also focus on narrative generation, and propose a language model fine-tuned on a corpus of Role-Playing Game (RPG) quest descriptions in order to synthesise new playable quests.

In parallel, LLMs have been assigned the role of systems designers, generating the structural and mechanical components of games. Hu et al. [94] utilise an LLM to generate game levels and rule systems using the Video Game Description Language [182]. While this approach demonstrates the expressive potential of language models for content generation, the authors note several limitations, including the frequent emergence of hallucinated content that renders games unplayable or incomplete, often necessitating manual correction.

To redress these limitations, several systems adopt a co-creative, HITL framework, in which the language model functions as a generative collaborator under human supervision. This design philosophy is observed in *LLMaker* [71], an interactive dungeon generator that responds to semantic prompts while allowing human users to guide, curate, and filter the resulting content. Such methods mitigate the well-documented shortcomings of LLMs, including their susceptibility to hallucination and lack of persistent state or memory [147]. The HITL approach thus serves both as a safeguard and as a mode of augmenting human creativity, although the present work shall not substantially engage with this extensive and distinct domain of co-creativity. Rather, the focus of this thesis lies in systems that aspire toward autonomy; those that minimise or eliminate the need for human intervention during content generation.

Full automation remains a challenging goal. However, a number of the aforementioned works, in particular those of Buongiorno et al. [21], Simon [190], and Sun et al. [195], indicate a rapidly growing body of work striving towards generative pipelines in which AI agents assume authorial control across one or more modalities. This is one of the research gaps which this thesis addresses through the development of the *CrawlLLM* multimodal generative pipeline discussed in Chapters 6 and 7.

Taken together, these developments reveal a fundamental shift in the landscape of PCG. Where traditional methods within this domain have been largely algorithmic, rule-driven, and domain-specific, the incorporation of GenAI introduces models that operate in a more generalised, prompt-responsive manner. This transition expands the expressive potential of PCG, enabling systems to generate content that is structurally complex and semantically meaningful. However, this advancement comes with new

design challenges, including a reliance on black-box models, difficulties in ensuring reliability or consistence, and an increased need for verification. As a result, the role of the designer is itself transforming from one of deterministic content authoring to that of curator, collaborator, or critic, navigating a space shaped by the affordances and characteristics of generative systems.

3.4 | Evaluative Measures

Irrespective of which modalities are being generated, evaluating creative content presents significant challenges. The most prominent of these is the inherent subjectivity of the assessment. HITL systems often shift the burden of creativity and evaluation to the human user, and the automated generator serves as a co-creative tool to facilitate and inspire the human creative process [13, 65, 71, 123]. But automated systems require quantifiable metrics for assessment. Creative domains are characterised by the lack of a universal aesthetic that can be used to determine the output quality, be they images, narratives, poetry, music, or other media. Unless a PCG system is trying to create a precisely defined artefact with tight constraints, there is no single ground truth against which to evaluate its output. Instead, the evaluation must be grounded in how well an artefact aligns with the intent behind its creation [14, 15, 99].

3.4.1 | Image Metrics

Conventional uni-modal metrics are based on the properties of the artefacts within their domain. In the context of image analysis, pixel counting and grouping has been frequently employed to quantify visual content properties. This approach involves evaluating the number or distribution of pixels within an image to assess aspects such as complexity or detail.

In one of the approaches detailed by Machado et al. [135], image *complexity* is computed by applying a Sobel [59] or Canny [22] filter. This determines which constituent pixels form an outline of objects within the image. The proportion of these pixels to the image total pixel size thus yields a measure of complexity. Figure 3.1 illustrates this metric computation. This concept may be extended to other type of outline detection algorithms, such as the use of Holistically-nested Edge Detection (HED) [229], which accomplishes this task through a trained NN.

Therefore, in mathematical notation:

$$complexity_{\mathcal{E},\tau}(I) = \frac{1}{|I|} \sum_{(x,y) \in I} \delta(\mathcal{E}(I)(x,y) \geq \tau)$$

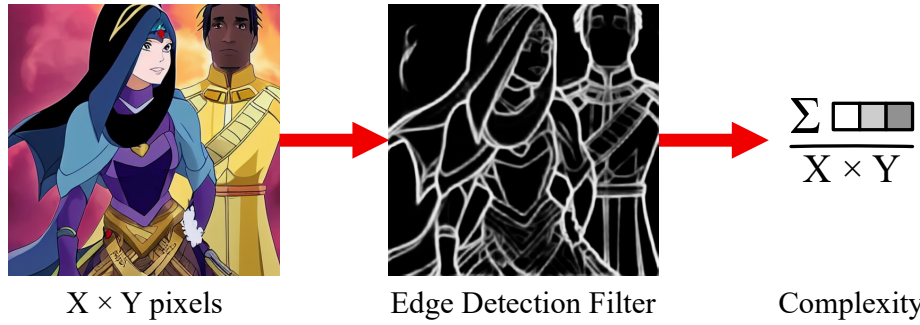


Figure 3.1: The complexity of an image may be computed as the proportion of edge pixels detected through an edge detection filter.

where I is the set of pixels comprising the image,

$|I|$ is the total number of pixels in the image,

$\mathcal{E}(I)(x, y)$ is the edge strength at pixel (x, y) produced by edge detector $\mathcal{E}$,

τ is the threshold for edge classification,

$$\delta(\text{condition}) = \begin{cases} 1, & \text{if the condition is true} \\ 0, & \text{otherwise} \end{cases} \quad \text{is the discrete decision function.}$$

Image metrics may also be derived from colour groupings, where the spatial and statistical properties of colour distributions are analysed to estimate visual characteristics like *colourfulness*. This technique was detailed by Hasler and Suesstrunk [85], who proposed a metric as a perceptually correlated measure for estimating the overall colourfulness of natural images. In this approach, pixel values are first transformed into opposing colour channels, specifically using the red-green difference $\alpha = R - G$ and the yellow-blue difference $\beta = 0.5(R + G) - B$.

For each of these channels, the mean values μ_α and μ_β and the standard deviations σ_α and σ_β are computed over the entire image. These statistical features are then combined to compute a colourfulness index according to the formula

$$\text{colourfulness} = \sqrt{\sigma_\alpha^2 + \sigma_\beta^2} + 0.3\sqrt{\mu_\alpha^2 + \mu_\beta^2}$$

In this expression, the first term reflects the variation in chromatic content, while the second term accounts for the overall deviation from achromaticity. The resulting metric was validated against subjective human colourfulness ratings and was shown to achieve a correlation exceeding 90% with perceptual assessments [85].

The Structural Similarity Index Measure (SSIM) [216] is a method that is used to assess the perceptual *similarity* between two images. It is based on the degradation of

structural information, and it is designed to improve upon traditional statistical which rely on absolute error measures and poorly align with human visual perception. In contrast, SSIM models image quality by considering perceptual phenomena, making it more sensitive to structural distortions and more consistent with human judgments. In this algorithm, image quality is evaluated by comparing local patterns of pixel intensities that have been normalised for luminance and contrast. The metric is typically computed over local windows, and the final similarity score is obtained by averaging over the entire image. SSIM has been widely adopted in image processing tasks due to its strong correlation with perceived image quality.

To compute SSIM, the three components, luminance, contrast, and structure, are calculated between corresponding local windows of the two images. These components are then combined into a single similarity index. The luminance comparison is obtained through mean intensity values, the contrast is assessed via standard deviation, and the structural comparison is based on the correlation (covariance) between the local image patches. The resulting index ranges from -1 to 1 , where a value of 1 indicates perfect structural similarity.

Similarity between images has also been evaluated using deep NNs. Such models are leveraged to extract high-level features from images, which are then compared to assess perceptual similarity. The effectiveness of this technique was explored by Zhang et al. [239], who demonstrated the utility of deep features in capturing human-like perceptions of visual similarity. This resulted in the development of the Learned Perceptual Image Patch Similarity (LPIPS) model, whose benchmark tests showed that it aligned surprisingly well with human preference.

3.4.2 | Text Metrics

For textual data, measurements can be grounded in numerical comparison of words or tokens. Metrics such as the Gunning-Fog index [82] quantify textual *complexity* by analysing sentence structure and word length, providing an estimate of readability based on cognitive demand.

When the classification of textual data is required, the content may be partitioned into thematically coherent segments through a statistical analysis of words or tokens, and the distributions they exhibit within the dataset. A widely adopted methodology for such statistical topic modelling is Latent Dirichlet Allocation (LDA), wherein latent thematic structures are inferred by analysing the patterns of word co-occurrence observed across a collection of documents or sentences. Within this framework, each document is assumed to be a mixture of multiple topics, and each topic is characterised by a

distinct distribution over words. Through iterative estimation procedures, the model parameters are adjusted to refine the groupings that most plausibly explain the observed text distributions. This technique, originally proposed by Blei et al. [11], has since found extensive application within the field of natural language processing, where it has been employed to categorise and facilitate comparative analyses of large textual corpora. LDA remains a pragmatic choice in many applied contexts despite the emergence of more sophisticated topic modeling approaches. While NN-based methods like BERTopic [81] may achieve better coherence scores or capture semantic nuances more effectively, LDA offers several practical advantages. Its computational efficiency enables rapid iteration and experimentation, particularly valuable during exploratory analysis or when working with large corpora under resource constraints. For certain downstream applications, the marginal gains from more complex models may not justify their computational cost, longer training times, or increased difficulty in parameter tuning. Furthermore, its mathematical transparency and well-understood probabilistic foundation make results more interpretable and easier to diagnose.

To assess textual similarity NN-based techniques are commonly adopted. These methods typically utilise pre-trained models to generate latent vector representations of text, which can then be compared using similarity metrics. For example, BERTScore, proposed by Zhang, Kishore, et al. [241], and Sentence-BERT, introduced by Reimers and Gurevych [172], both utilise transformer-based architectures to capture nuanced semantic relationships between sentences or phrases.

3.4.3 | Multimodal Metrics

In multimodal domains that produce distinct sets of components across various modalities, each modality must be evaluated independently through modality-specific assessment techniques to establish its validity. However, such an approach is insufficient in isolation, as it fails to capture the relationships between modalities inherent in many multimodal tasks. Therefore, an additional metric for cross-modal evaluation is required to assess the coherence and alignment between modalities.

Such multimodal evaluations may be relatively straightforward to evaluate in some applications; generating subtitles for a video has very clearly demarcated criteria for success, such as accurate speech detection, matching time frames for corresponding text and audio/video, and other framing metrics for text [156]. However other applications present greater challenges. In many cases, such evaluations are highly context-sensitive and lack standardised or generalisable criteria, making them difficult to transfer across domains.

To address these challenges, large-scale multimodal models such as CLIP, FLAVA, and other foundation models may be leveraged due to their capacity to embed different modalities into a shared representational space, enabling direct comparison via vector similarity metrics, as described by Equation 2.4 in Section 2.3.1 of Chapter 2. In scenarios where such models are unavailable or unsuitable, task-specific measures of cross-modal coherence must be developed, tailored to the unique constraints of the application at hand.

Thus, effective multimodal evaluation necessitates both rigorous intra-modal validation and robust cross-modal coherence assessments, ideally supported by powerful, general-purpose embedding models or by expert ad-hoc evaluation metrics.

3.5 | Summary

This chapter has given a bird's eye view of the evolution of GenAI algorithms, their PCG applications, and a basis for their evaluation. Thus it lays the foundations for the following experimental chapters 4 to 7, which will propose novel ways of combining these principles into novel pipelines for various output complexities. Chapter 4 begins this journey through combining evolution and backpropagation algorithms to seed diversity into a text-to-image generative process.

Combining Evolution and Backpropagation

VAEs and VQGANs are an integral part of text-to-image generative models, as image embeddings within their latent space offer a compact and more manageable representation than pixel space. Images generated by models that use these components thus tend to heavily conform to specific visual patterns, driven by their training corpora. Figure 4.1 illustrates this behaviour via generated images from models trained on the *ImageNet* [49] and *WikiArt* [199] corpora respectively.

In practice, for a specific semantic prompt, the generated images exhibit *typicality* and better *quality* than preceding algorithms, but lack *novelty*, in terms of the requirements for creative systems as described by [173]. This lack of diversity partly stems from the fact that these generative mechanisms operate on a single image at a time. Thus the process lacks knowledge about prior runs and images created hitherto.

To seed diversity into this process, the author proposes a hybrid algorithm which

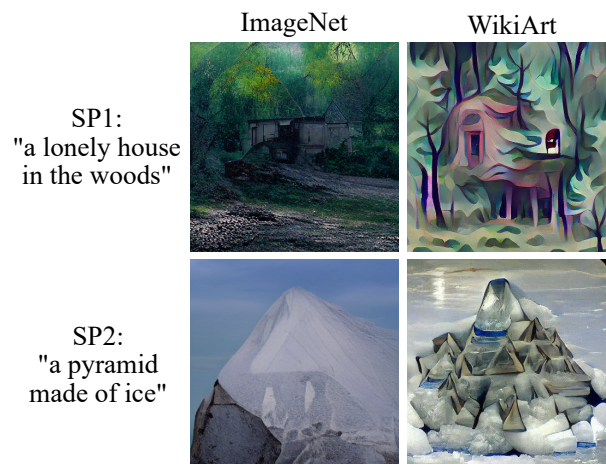


Figure 4.1: Examples of VQGAN models trained on different datasets (*ImageNet* and *WikiArt*) generating images for two semantic prompts under CLIP-guidance.

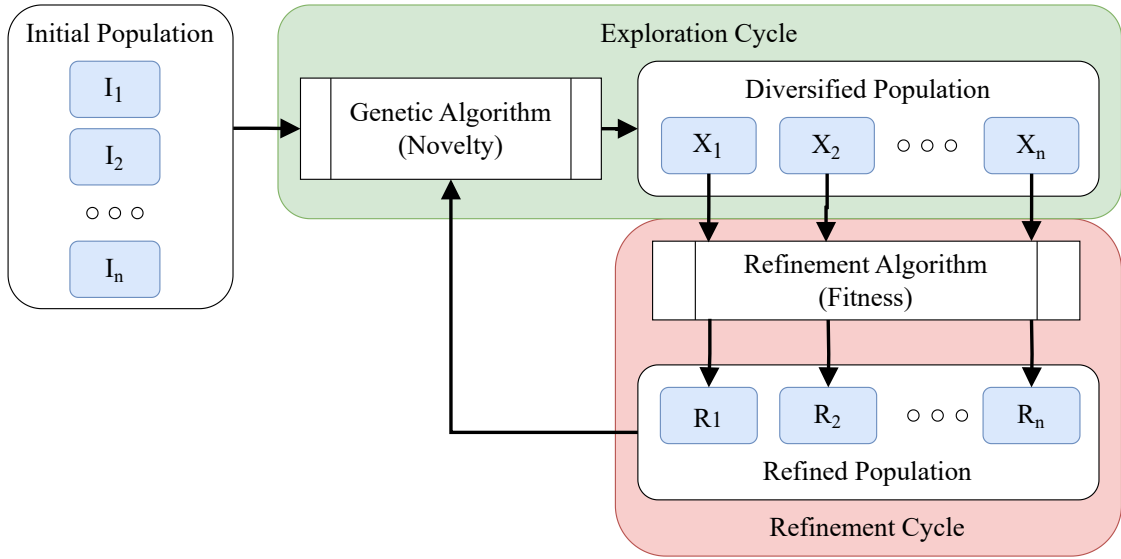


Figure 4.2: The basic alternating structure of Refinement and Exploration cycles. A genetic algorithm targeting novelty diversifies the population, whilst a refinement algorithm improves its fitness.

cycles between a novelty-driven GA and the base generative one. This counteracts the lack of ‘vision’ across individual images through the GA’s awareness of its evolving population’s ensemble of properties. In particular, the use of QD algorithms promotes a wider variety within the final population of generated images. To this end, experiment EXP1 was conducted to demonstrate how the creative visual task of generating images in alignment to given text prompts can be improved in terms of quality and diversity through the use of combined backpropagation and QD algorithms.

4.1 | Methodology

At its core, the proposed process takes an initial population of generated artefacts, and alternates between *refinement cycles*, which operates at an individual level to optimise quality, and *exploration cycles*, which attempt to diversify the population collectively through a novelty-driven GA (see Figure 4.2).

Each of these alternating cycles excels at a specific task. Refinement cycles, commonly based on NNs, are able to rapidly increase the fitness of an individual, but strictly align to the typicality of their training corpora. Exploration cycles, typically based on evolutionary algorithms, operate at a slower rate but affect the global population, and hence are able to diverge the properties of the individual assets to preserve novel

characteristics. This idea of cycling between algorithms to evolve diversity is by no means original [122], but its application to GenAI models is a novel concept [233].

4.2 | Use Case: Text-to-Image Generation

The use case chosen for this process is the VQGAN-CLIP algorithm [43] for text-to-image generation, described in Section 2.3.3 of Chapter 2. The algorithm is alternated with NSLC iterations in order to inject more novel individuals into the generated set of images.

4.2.1 | The Refinement Cycle using Backpropagation

In this work, the VQGAN-CLIP algorithm is implemented via the *Pixray*¹ Python library. Images of 384×384 pixels were generated, using a VQGAN trained on the *WikiArt* corpus [199]. This dataset consists of 81,444 images of paintings and illustrations across various art styles². The model used splits images into patches of 16×16 pixels, i.e. the generated images each comprised of 24×24 patches. The VQVAE representation of an image is thus a flattened vector consisting of 576 integers ranging between 0 and 16,384, each representing the index of a specific patch of the image from a codebook learned by the VQGAN. A backpropagation process is applied to this latent vector, using a CLIP score aligning the image to the text prompt as its fitness function. Since the backpropagation process requires a floating-point vector, a larger internal representation of the patches is used for this stage; elsewhere the integer representation is used for compactness.

4.2.2 | The Exploration Cycle using Evolutionary Algorithms

In order to diversify the generated images, a population-based, QD algorithm is used. Specifically, NSLC [115] operates on the latent vector representation of the image. This algorithm was chosen because of its propensity to seek novel solutions and preserving features which are locally dominant, despite these not belonging to globally optimal solutions.

Since the mapping between each integer within the image embedding to an image patch in the VQGAN codebook is arbitrarily learned during the training process of the model, crossover functions applied to these genes tend to disrupt the image into incoherence. Hence the GA was implemented using only mutation operators. Each

¹<https://github.com/pixray/pixray>

²<https://github.com/cs-chan/ArtGAN/tree/master/WikiArt%20Dataset>

mutation replaces a randomly selected 5% of the individual's genes with random integers chosen from the codebook (i.e. between 0 and 16,384). This optimal mutation rate was preferred to higher ones because during initial trials it was found to create appreciable alterations of the phenotype, without excessively harming the resulting composition.

For each individual, the $k = 15$ nearest neighbours within the current population and within an archive of novel past individuals were found for this experiment. The novelty archive is gradually populated throughout the course of the algorithm with the 3 most novel individuals from every generation. The archive is reset for every exploration cycle as maintaining an increasingly large archive bears a considerable computational cost when calculating neighbourhoods and scores.

The novelty and LC scores were processed via a Non-dominated Sorting Genetic Algorithm (NSGA-II) [47] as a multi-objective optimisation problem, using the *Pymoo* Python library [10]. This library processes minimisation problems, so for this experiment the scores were negated. Individuals for subsequent populations were selected from a Pareto front to ensure both optimality and diversity as follows: Initially, the Pareto front is identified, consisting of individuals that are dominant with respect to the objectives, which are prioritised for selection. Should the Pareto front contain more individuals than required, a subset is selected. Selection is carried out to maximise sparsity based on the Manhattan distance, in order to maintain a well-distributed front across the behavioural space. Conversely, if the Pareto front contains insufficient individuals to populate the next generation, subsequent Pareto fronts are computed, and additional individuals are selected based on their dominance.

The definition of the two metrics, fitness and distance, operating on and between individuals is crucial to the problem, as both the novelty and LC score are derived from these parameters. Neither of these two is straightforward to quantify, as observed by [173]. However, for fitness this work takes the simple approach of measuring the alignment of the image with the input text prompt. The CLIP score as defined in Section 4.2.1, obtained by Equation 2.4, can thus be directly used as it directly reflects this objective.

The distance function should measure the visual diversity between two images. Such a task is relatively simple for humans to accomplish, but it is very challenging to identify a single, adequate metric to automate this assessment or reliably quantify it. This work focuses on two specific metrics for such measurements, namely chromatic diversity, and ViT diversity, subsequently referred to as Hue, Saturation and Brightness (HSV) and ViT diversities, respectively within this text.

4.2.2.1 | Chromatic (HSV) Diversity

Following Machado et al. [135], the distribution of colours in the pixels is surmised to reflect the diversity of the images. The HSV value of each pixel are analysed, and a diversity metric is calculated for any pair of generated images I_1 and I_2 using their means and standard deviations, as described hereunder:

$$m_1 = \Delta \bar{b} = |\bar{b}_1 - \bar{b}_2| \quad (4.1)$$

$$m_2 = \Delta \sigma(b) = |\sigma(b_1) - \sigma(b_2)| \quad (4.2)$$

$$m_3 = \Delta \bar{s} = |\bar{s}_1 - \bar{s}_2| \quad (4.3)$$

$$m_4 = \Delta \sigma(s) = |\sigma(s_1) - \sigma(s_2)| \quad (4.4)$$

$$m_5 = \Delta \bar{h} = |\min[\bar{h}_1 - \bar{h}_2, \bar{h}_1 - (1 - \bar{h}_2)]| \quad (4.5)$$

$$m_6 = \Delta \sigma(h) = |\sigma(h_1) - \sigma(h_2)| \quad (4.6)$$

where h , s and b denote the hue, saturation, and brightness (all normalised to the $[0, 1]$ value range), the means $(\bar{h}, \bar{s}, \bar{b})$ are calculated across all pixels in I_1 and I_2 , and σ denotes the standard deviation of these values.

It is important to note that the hue value is cyclic, hence its mean and standard deviation need to be calculated as follows:

$$\bar{h} = \tan^{-1} \left(\frac{\sum_{i=1}^N \sin(h)}{\sum_{i=1}^N \cos(h)} \right) \quad (4.7)$$

$$\sigma(h) = \sqrt{\frac{\sum_{i=1}^N (\min(h_i - \bar{h}, h_i - (1 - \bar{h})))^2}{N - 1}} \quad (4.8)$$

where N is the number of pixels in the image.

The distance metric d_{HSV} is obtained from the mean square value of the individual metrics $m_1 \dots m_6$.

4.2.2.2 | Vision Transformer (ViT) Diversity

Diversity can also be assessed using the embeddings of pre-trained models. ViT models encode information about various images from their training corpus within their layers, and leverage this information to classify them. For this analysis, a ViT model pre-trained on the *ImageNet* dataset [49] is used, with its final classification layer removed. By excluding this layer, a latent vector of 768 floating-point values is retained for each processed image. This retains the information about the defining features of the image without the constraints of the predetermined classes. The diversity metric (d_{ViT}) is then

calculated as the Euclidean distance between the latent vectors of two images as shown in Equation 4.9:

$$d_{\text{ViT}}(\mathbf{v}_1, \mathbf{v}_2) = \|\mathbf{v}_1 - \mathbf{v}_2\|_2 = \sqrt{\sum_{i=1}^{768} (v_{1i} - v_{2i})^2} \quad (4.9)$$

where $\mathbf{v}_1, \mathbf{v}_2 \in \mathbb{R}^{768}$ are the latent vectors of two images,
and v_{ij} is the j^{th} component of $\mathbf{v}_i$,

4.3 | Structure of the Experiment

4.3.1 | Hypotheses

The aim of EXP1 is to test the following hypotheses:

- H1 The periodic interruption of the refinement process by NSLC evolutionary cycles yields a greater overall diversity in the image population than that produced by an uninterrupted process.
- H2 The introduction of exploration improves or retains, but does not degrade the alignment between generated images and their textual prompts, as measured by CLIP similarity.
- H3 Employing chromatic (HSV-based) novelty metrics during the NSLC cycles will preferentially enhance colour diversity, whereas employing feature-embedding (ViT-based) novelty metrics will preferentially enhance feature-space diversity.

4.3.2 | Protocol

The CLIP-guided VQGAN text-guided image generation in the refinement cycles are carried out using the *Pixray* library using a VQVAE model pre-trained on the *WikiArt* dataset, as described in Section 4.2.1. Five Semantic Prompts (SPs) used by the community³ were used:

SP1: a lonely house in the woods

SP2: a pyramid made of ice

SP3: artificial intelligence

³<https://github.com/lucidrains/big-sleep>

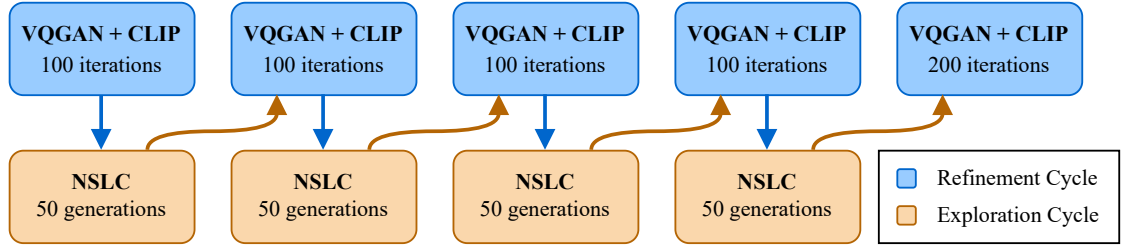


Figure 4.3: Structure of the experiments alternating between refinement (CLIP-guided VQGAN) and exploratory cycles (NSLC).

SP4: cosmic love and attention

SP5: fire in the sky

As a first step, an initial population of 50 latent vectors was generated by using the VQVAE to encode a set of randomly generated fractal noise images. This same initial ‘noise’ population was retained and used as the set of seed images across all prompts and in all variations of the algorithms described in the experiment.

A baseline was established by running *Pixray* for a total of 600 uninterrupted iterations on each of these 50 seed images, for each of the SPs. This is referred to as the GAN baseline (GAN-BSL) for brevity. Initial trials have shown that increasing the iterations beyond 600 will refine the generated image quality, but its general composition is already stable at around 500 iterations, beyond which no appreciable structural change is noticed.

For the experiment protocol, the same number of refinement iterations as the baseline are carried out, but the process is interrupted after 100, 200, 300 and 400 iterations. The latent vectors of the population are extracted at each of those points and used as the starting population of a NSLC exploration cycle as detailed in Section 4.2.2 above. This cycle is carried out for 50 generations with the aim to diversify the population, then the resulting latent vectors are returned to the next refinement cycle, where the GAN iterations continue operating on them until the next scheduled exploration cycle interruption is due. The experiment was carried out in two variants based on each of the chosen diversity metrics, i.e. guiding the NSLC process by either ViT diversity (NSLC-ViT experiment) or by HSV diversity (NSLC-HSV experiment). This protocol is outlined in Figure 4.3.

4.3.3 | Performance Metrics

For consistency, the same metrics as those of the exploration cycles are employed to obtain the quality and diversity of the final images, thus comparing the different algorithms.

They are applied to the populations within each experiment by calculating the following:

- **mean fitness** - the average CLIP score across all 50 images in the population.
- **mean ViT diversity** - the average ViT distance from the nearest 15 neighbours per individual, averaged across all 50 images in the population.
- **mean HSV diversity** - identical to mean ViT novelty but uses the HSV metric to measure distance and finding nearest neighbours.

Note that since the two mean diversity metrics are calculated on a population separately from the NSLC processes, no archive is considered. Neighbours are obtained only from the same population being evaluated.

4.4 | Results and Evaluation

In Computational Creativity, assessing the *process* followed by the algorithms is as relevant as evaluating the resulting *product* at the end of the 600 iterations [100]. Therefore, the significant metrics described in Section 4.3.3 are calculated at regular iteration intervals throughout each experiment. Figure 4.4 plots these metrics for all experiments conducted, including the baseline reference, and across all the prompts used. It is interesting to note how pronounced the effect of the exploration cycles is, when compared to the GAN baseline. At a glance the plots indicate how the NSLC cycles disrupt the fitness of the population but there is a drastic improvement in the mean diversities, irrespective of which method is used to compute it.

Figure 4.4(a) shows the mean fitness, or CLIP score, which rapidly increases over the initial 20 to 40 iterations, then stabilises to a very gradual increase thereon. The exploration cycles decrease the mean fitness across the evolved population by an average of 12% for the NSLC-HSV runs, and 21% for the NSLC-ViT ones. This drop is consistent throughout all NSLC cycles, including the last one, despite the composition of the image having been solidly affirmed at that point. Surprisingly, the refinement cycles are quick to restore the mean fitness over the first few iterations following the NSLC interruption to practically the same levels of the uninterrupted baseline. This indicates the remarkable efficacy of the refinement cycle to operate on the image, independently of whether it is primarily random noise, or an already composed image. Despite the interruptions, both NSLC experiments reached a marginally higher mean fitness than the GAN baseline at the end of the experiments, with the exception of the NSLC-ViT run of SP4, whose final value was 2.9% lower than the GAN-BSL.

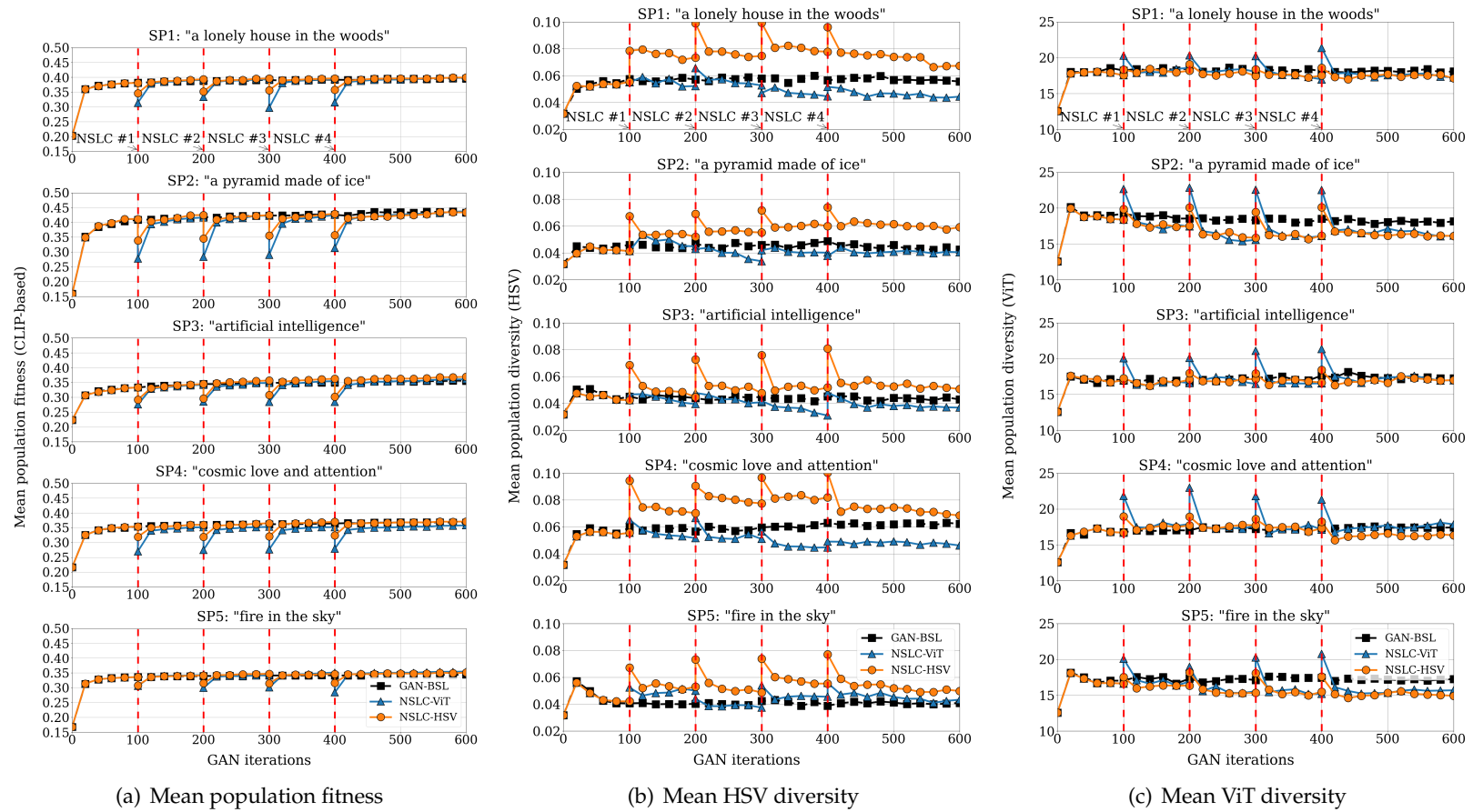


Figure 4.4: Progression of the performance metrics over GAN iterations. The iterations at which evolutionary NSLC cycles were performed are marked in red.

The two NSLC experiments differed slightly in behaviour. NSLC-HSV mean fitness surpassed that of the GAN baseline by 1.5% on average, and exhibited smaller variations between prompts. For SP3 this algorithm showed the largest improvement of 3.9% over the baseline. In comparison, NSLC-ViT had only an average increase of 0.7% in mean fitness over the baseline, and was subject to greater disparities across the SPs.

Figures 4.4(b) and 4.4(c) plot the diversity of the populations from all experiments using the HSV and ViT metrics respectively, so that the populations evolved via each metric driving the NSLC cycles are evaluated both vis-a-vis their evolving metric as well as an orthogonal one. In both sets of graphs a sharp increase in mean diversity can be observed over the first 20 iterations. This is due to the fact that the initial seed images, composed of random fractal noise, have similar average distributions of colour and brightness, as well as a lack of form and structure. Consequently, both HSV and ViT metrics yield small distances between them. Conversely, the VQGAN model manifold is expected to push the formation of the images towards the typicality of the training corpora, especially given the sharp increase in fitness shown in Figure 4.4(a) within these same iterations. This is observed in subsequent iterations, during which the mean diversity stabilises or gradually decreases. This trend is prevalent for the GAN-BSL experiments throughout, except for SP4 in the HSV diversity evaluations. The trend to decrease is most noticeable in SP5 for both HSV and ViT, which is attributed to the prompt constraining the images to have primarily red and blue colours, and to be similar in terms of image composition and content.

The NSLC experiments show an evident spike after every exploration cycle when the same metrics as those of the NSLC cycles is used. It is interesting to note that NSLC-HSV exhibits the increase with both diversity metrics, whereas NSLC-ViT does not always do so for a HSV evaluation. The average increase in ViT diversity for each exploration cycle is 25% for NSLC-ViT and 10% for NSLC-HSV. In comparison, NSLC-HSV achieves an average increase of 43% HSV diversity in every NSLC cycle, and NSLC-ViT obtains 14%.

However the refinement cycles rapidly attract the population back towards the style of the manifold, decreasing the diversity over the iterations to a stable level as the mean population fitness increases. Generally, this leads to a lower average ViT diversity than the GAN baseline. This was unexpected, since the ViT diversity increases during the NSLC cycles for both variants of the experiments. Interestingly, the GAN cycles do not exhibit a similar increase in diversity as in the initial 20 iterations when the NSLC cycles inject ‘noise’ in the images (due to the mutations) at iterations 100, 200, 300, and 400. By 600 iterations, the final images produced by NSLC-HSV show an average 6.3% increase in HSV diversity compared to the GAN baseline, but an average 11.5% decrease in ViT diversity per prompt. On the other hand, the final images generated by NSLC-ViT exhibit

lower diversity for both ViT and HSV compared to the GAN baseline, with decrements of 5.8% and 13.7%, respectively.

In view of the experimental hypotheses, the above analysis shows that the alternating evolutionary cycles produced a measurable increase in visual diversity when HSV distance guided the exploration. Thus, hypothesis H1 holds for the HSV-driven variant but is not supported when ViT-based novelty is used.

Although each NSLC cycle caused an immediate drop in mean CLIP fitness, the refinement cycle consistently recovered semantic correspondence by its termination. By iteration 600, both NSLC variants attained mean CLIP scores at or slightly above the baseline, with the exception of one prompt (SP4) in NSLC-ViT. These findings support the validity of H2.

During each evolutionary cycle, NSLC-HSV substantially increased both HSV and ViT, whereas NSLC-ViT increased ViT diversity but showed negligible gains in HSV diversity. These cycle-level trends translated into final outputs in which NSLC-HSV preferentially enhanced chromatic diversity and NSLC-ViT failed to preserve either form of diversity. Hence, hypothesis H3 was only partially confirmed, i.e. the choice of novelty metric directs diversity gains toward its corresponding feature space, which is particularly seen with HSV but not with ViT.

4.4.1 | Qualitative Observations

To appreciate the proposed process and visualise its various stages, the 5 most diverse images at the final iteration of each cycle are presented in Figure 4.5. In this diagram, the most diverse individuals are selected by computing the HSV distance to the nearest-neighbours. The NSLC-HSV experiment was chosen because this algorithm has been demonstrated to produce more diverse individuals while maintaining quality comparable to the baseline. In particular, SP3 is shown due to its final images achieving the highest increase in CLIP score (3.9% above the GAN baseline) and a notable improvement in HSV diversity (7% above the GAN baseline).

After the first 100 GAN iterations, the image structure is recognisable, but still somewhat noisy, and with rough details. At this stage, diversity among images is appreciable. Following the first exploration cycle, new patterns and elements emerge, such as the introduction of human nose features. Despite this, some images become more noisy and less distinguishable. These rough details are progressively refined in the subsequent GAN cycle, resulting in crisper yet visually similar images. Each successive round of exploration and refinement adds further details. Later NSLC cycles yield images that are more recognisable and less noisy. However, after 400 iterations, increased similarity

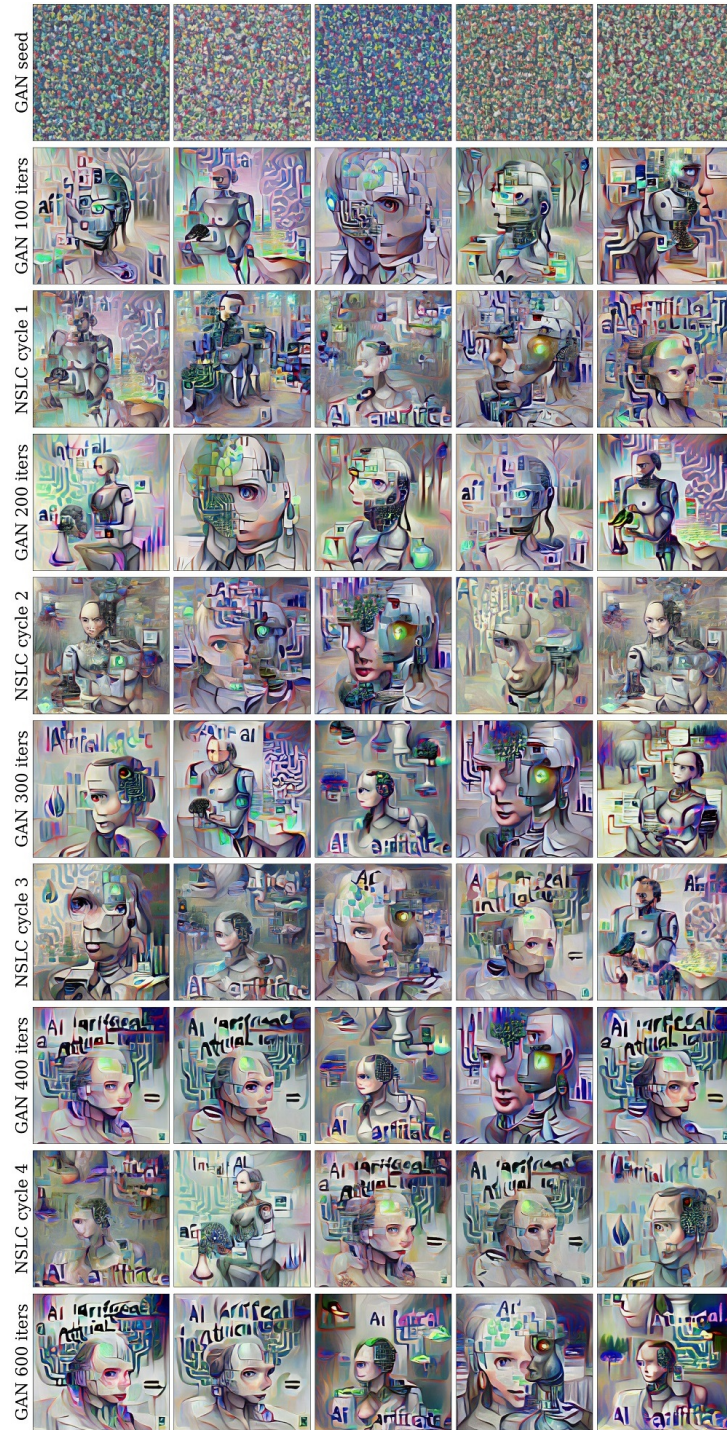


Figure 4.5: The progression (top to bottom) shows the 5 images with the highest HSV diversity at the end of each cycle in an NSLC-HSV experiment for SP3.

among images becomes apparent, with recurring patterns such as the appearance of the text “artificial” across most outputs. At this stage, the final NSLC cycle struggles to disrupt these patterns, and the final products at 600 iterations exhibit more similarities than earlier interim images, such as those at 300 iterations.

The analysis suggests that NSLC plays a more significant role during earlier stages of the process. Over time, GANs tend to enforce dominant patterns, even on initially novel images. It is inferred that stopping the process earlier or incorporating NSLC interventions during earlier stages, e.g. at 20 or 50 iterations instead of 400, may better mitigate this convergence toward dominant patterns.

4.5 | Discussion

EXP1 has demonstrated how a QD algorithm, namely NSLC, can be combined into generative processes to diversify the output artefacts. Specifically, the population-based evolutionary search was useful to break away from the constraints of the text guidance and the model’s training corpora, which otherwise lacks *appreciation* of its individual creations. The results have shown how alternating between NSLC and GAN cycles shows a temporary increase in visual diversity but the effect decreases over time the more the GAN process is allowed to operate.

In view of the hypotheses set out in Section 4.3.1, these have been directly addressed in Section 4.4, and were mostly supported; in part for H1 and H3, in full for H2.

This confirms that evolutionary QD can be leveraged to broaden the stylistic variety of generated images without permanently degrading their adherence to text prompts, thereby answering this thesis’ RQ1. Specifically, EXP1 shows that these algorithms effectively increase the fitness and diversity of PCG models operating in multiple modalities, with the caveat that the chosen metrics have to be carefully selected to reflect specific qualities of interest within the resulting artefacts.

RQ2, which addresses the structuring of the search space, is answered by EXP1’s distinct subdivision of its novelty search space into two distinct feature spaces, namely chromatic (HSV) and learned-embedding (ViT). The experiment showed that the choice of representation directs diversity gains toward its respective domain. Despite its simplicity, HSV novelty produced significant chromatic variation with modest embedding changes, whereas ViT novelty failed to sustain gains when measured with both diversity metrics, illustrating the critical role of search-space structuring in driving desired exploratory outcomes.

The poor performance of the ViT metric could be attributed to the fact that the model

was originally trained as an image classifier, and might be unable to properly differentiate between two images if the composition contains elements of the same class, despite their having different visual properties. For example, the model is trained to classify a building as a house if it has specific structural properties, but it might not discern between a red-painted, two-storey house and a white, single-story one, irrespective of how different these may visually appear to a human.

Other visual distance metrics based on deep learning such as LPIPS [239] are more targetted towards finding similarities in images, but their training is more attuned towards detecting the same image which has undergone some transformation, such as rotation, scaling, translation, or others. LPIPS was also evaluated during initial trials, however its performance as a distance metric for the NSLC cycles fared much worse than the GAN baseline and thus it was abandoned.

This experiment is also relevant to RQ4 as it employed CLIP similarity as a proxy for semantic coherence. In response to the question, it shows that multimodal latent spaces, CLIP in particular for the investigated modalities, are effective metrics for asset alignment and comparisons. In addition, EXP1 proposes HSV and ViT-embedding distances as both quantitative measures of visual diversity, as well as drivers for evolutionary novelty search. The cycle-level and endpoint analyses using these metrics provided clear, interpretable signals of transient versus cumulative effects, thereby validating this combination of assessment tools for multimodal generative systems.

The generalisation strength of these conclusions lies primarily at the methodological level rather than at the level of specific models or prompts. While the experiments make use of VQGAN-CLIP and particular image datasets, the underlying principle of alternating between population-level exploration and local gradient-based refinement is not specific to these choices. Similar dynamics can reasonably be expected to emerge when pairing other differentiable generative models with evolutionary novelty or diversity objectives.

However, the conclusions do not fully generalise to all multimodal or creative domains. The metrics used and the reliance on CLIP embeddings bias the exploration toward visually measurable properties and semantic similarity as defined by a specific pre-trained model. Consequently, the results generalise best to domains where a shared embedding space is available, and where diversity can be meaningfully captured by perceptual features of low complexity. Claims about broader creative diversity or human-perceived novelty should therefore be interpreted with caution.

It warrants brief consideration that the computational cost necessitated by the exploratory NSLC cycles surpasses that demanded by the refinement cycles by several orders of magnitude. Given the generally higher computational demands of population-

based evolutionary methods relative to gradient-based optimisation, this observation is entirely anticipated. Indeed, it was observed that the refinement cycles themselves impose a substantially greater computational burden than the baseline backpropagation process, a consideration that becomes particularly salient when the relatively modest improvements in diversity are taken into account. Nevertheless, this computational overhead was deemed to be acceptable within the context of the present proof-of-concept investigation, to illustrate the hybrid approach.

Overall, notwithstanding mixed results, this study showcases the approach of combining algorithms into more effective hybrids, and shows great potential when applied to GenAI models. It is hypothesised that adaptive scheduling strategies, in which the influence of novelty is increased as refinement stagnates, may mitigate late-stage homogenisation. Additionally, embedding spaces explicitly trained to reflect perceptual similarity rather than classification performance may yield more stable and interpretable diversity gradients.

4.6 | Limitations and Shortcomings

This study relies on a tightly coupled optimisation and evaluation scheme, wherein CLIP-based similarity is used both to guide evolutionary selection and to assess final outcomes. While the implications of CLIP’s semantic bias are discussed earlier in this chapter, a broader limitation is that this coupling introduces an element of evaluative circularity. Improvements observed should therefore be interpreted as relative to the model-defined objective, rather than as evidence of improvement over any universal absolute, or human-centric visual quality or creativity.

Furthermore, evaluation in this chapter is fully automated and does not involve human judgement. Although this choice enables controlled experimentation and repeatability, it limits the extent to which claims can be made about aesthetic value or semantic appropriateness beyond the embedding space of the model. This limitation is particularly relevant given that evolutionary pressure may amplify artefacts that are favoured by the metric while remaining ambiguous or unconvincing to human observers.

Diversity is addressed through novelty-based competition and low-level visual descriptors, which effectively demonstrate that evolutionary processes can counteract convergence in generative image models. However, this form of diversity remains primarily perceptual and does not account for higher-level conceptual variation. As a result, visually distinct artefacts may still occupy a narrow semantic space. The difference in performance of the different metrics used also tempers the generality of the findings,

indicating that evolutionary enhancement of GenAI outputs is contingent on the metrics used to define diversity. At the same time, the progressive homogenisation of the population during later refinement cycles reveals an important shortcoming of the approach. Despite repeated injections of novelty, backpropagation increasingly dominates the search dynamics, pulling solutions toward a limited subset of high-fitness regions. As a result, diversity gains diminish over time, suggesting the presence of a practical ceiling beyond which exploration yields diminishing returns.

These limitations frame EXP1 as a proof-of-concept rather than a comprehensive creative system. In particular, they motivate the need for a more explicit structuring of the search space and a broader treatment of multimodal diversity, which is addressed in Chapter 5.

4.7 | Summary

This chapter has shown the combination of an QD evolutionary algorithm with a VQGAN-based generative model. Two distance metrics, HSV and ViT, were used within a NSLC algorithm to diversify the generated images. Despite having a unimodal output, the process was multimodal as the image generation was guided by text, and the implemented algorithms targeted both the increase in modal coherence through a CLIP score, as well as diversity through distance metrics. In the next chapter, a multimodal GenAI problem dealing with models producing both text and images will be examined.

Multimodal Search Space Illumination

This chapter focuses on multimodal generation for both text and images using GenAI models. An illustrative example will be used in which text descriptions and cover (or ‘hero’) images are generated for hypothetical video games. Through this work the author proposes an improvement on the MAP-Elites algorithm [150]. This novel method, dubbed MELiTA, implements an inter-modal evaluation strategy that facilitates the exchange of partial artefacts (such as images or texts) among phenotypically related elite candidates, to promote the emergence of more coherent pairings. This novel mechanism enriches the creative dynamics of co-evolution, leading to the generation of solutions that are both more diverse and better adapted.

This approach is useful in its capacity to illuminate creative multimodal search spaces, particularly in scenarios where it is desirable to obtain solutions that comprehensively span a distinct set of BCs associated with each modality.

5.1 | Methodology

MELiTA is an extension of the MAP-Elites framework, purposefully designed to evolve multimodal artefacts and reduce their inter-modal divergence. The methodology assumes several structural assumptions. In brief, MELiTA differs from MAP-Elites during the placement of new candidate solutions. In the baseline MAP-Elites algorithm, when a new candidate is found, its BCs are computed to determine in which position it may be placed in the archive. If the space is vacant, the candidate is placed there, becoming the new elite. Otherwise, the candidate’s fitness is compared to that of the existing elite at that location. If higher, it replaces the elite, otherwise it is discarded.

MELiTA operates in multimodal domains wherein each individual has components in multiple modalities. The search space is subdivided into BC axes each pertaining to one specific modality. During placement, when the BCs of the new candidate place it at a specific location, its fitness is evaluated against not only the existing elite at that position, but also against new potential candidates obtained by pairing the altered modalities of

the new candidate with the other modalities of the existing elites which are placed along the same dimension of the archive. The resulting hybrid candidate with the highest fitness is then placed in its respective location, replacing the existing elite.

More formally, the operation of MELiTA is explained as follows. While illustrated here in the context of two modalities, the framework generalises to any N -modality scenario:

- Each individual A comprises N discrete artefacts, each associated with a distinct modality M_i , such that $A = \{a_{M_1}, a_{M_2}, \dots, a_{M_N}\}$.
- For each modality M_i , a variation operator V_{M_i} exists and can be applied to the corresponding artefact a_{M_i} , potentially conditioned on other modalities, without altering them. For instance, $a'_{M_1} = V_{M_1}(a_{M_1}, a_{M_2}, \dots, a_{M_N})$.
- A coherence function $f : A \mapsto \mathbb{R}$ returns a scalar value q representing the degree of inter-modal consistency: $q(A) = f(a_{M_1}, a_{M_2}, \dots, a_{M_N})$.
- Each modality M_i is associated with a behaviour characterisation function $g_{M_i} : a_{M_i} \mapsto \beta_{M_i}$, where β_{M_i} denotes the behaviour descriptor of artefact a_{M_i} .

Under this formulation, MELiTA constructs an N -dimensional archive in which each dimension corresponds to a behaviour characterisation β_{M_i} . Every generation, an elite individual $E = \{e_{M_1}, e_{M_2}, \dots, e_{M_N}\}$ is sampled from the archive, via uniform or adaptive selection strategies [44].

A single artefact, say e_{M_k} , is selected uniformly at random, and transformed using its corresponding variation operator, yielding $e'_{M_k} = V_{M_k}(e_{M_1}, \dots, e_{M_N})$. The associated behaviour descriptor is computed as $g_{M_k}(e'_{M_k})$.

A new candidate individual $E' = \{e'_{M_k}, e_{M_j \neq k}, \dots\}$ is then formed by substituting the modified artefact while preserving all others. In the standard MAP-Elites algorithm, E' would be evaluated against an incumbent elite E'_{old} located at the cell defined by $\{g_{M_1}(e'_{M_1}), \dots, g_{M_N}(e'_{M_N})\}$, and inserted if $q(E') > q(E'_{\text{old}})$, or if the cell is unoccupied.

MELiTA augments this process by evaluating the mutated artefact e'_{M_k} in combination with artefacts from elites $R = \{r_{M_1}, \dots, r_{M_N}\}$ occupying archive cells with matching behaviour descriptor $\beta_{M_k} = g_{M_k}(e'_{M_k})$. For each such elite, a hybrid individual $R' = \{e'_{M_k}, r_{M_j \neq k}, \dots\}$ is generated, and its coherence score $q(R')$ computed.

The resulting set of candidate individuals $\vec{R}'$, together with E' , is sorted in descending order of q to form a ranked list $\vec{L}$. Each candidate $\phi \in \vec{L}$ is sequentially considered for insertion into the archive. If the archive cell defined by the behaviour descriptors of ϕ is empty, ϕ is inserted. Otherwise, ϕ replaces the current occupant ϕ_{old} if and only if

$q(\phi) > q(\phi_{\text{old}})$. If neither condition is met, the process continues to the next candidate in the list.

At most one individual is inserted into the archive per generation. This mechanism facilitates progressive refinement of the archive while allowing unoccupied cells to be filled, potentially including the unpaired candidate E' .

The application of MELiTA will be further illustrated through the use case presented in Section 5.2.

5.2 | Use Case: Hypothetical Game Blurbs

As already explained in previous chapters, the modalities of image and text were selected due to their compatibility with robust, publicly available AI generation systems. Within this circumscription, MELiTA is applied to the automated generation of fictional video games, each defined by a *game title*, a *short textual synopsis*, and a *cover image*. The objective is to produce a varied set of artefacts that are both internally coherent and stylistically aligned with established conventions of game media, with the potential to inform or inspire players and developers.

For each modality, two primary aspects are addressed: the mechanism by which artefacts are generated or altered (typically via mutation), and the manner in which they are characterised for placement within the MAP-Elites feature map [150]. Subsequent sections describe the generation and characterisation processes for each modality in detail, followed by a formal outline of the MELiTA procedure as implemented in this study.

5.2.1 | The Text Modality

The GPT-2 model [168] serves as the backbone for generating both game titles and descriptions within this study. Although more advanced language models have since emerged, offering improved fluency and reasoning capabilities [18, 154, 202, 203], GPT-2 remains relevant for tasks that benefit from its significantly lower inference latency. Despite its limited scale and correspondingly reduced performance, the model's speed and adaptability made it a suitable choice for integration into the iterative, population-based search characteristic of this evolutionary framework. Additionally, GPT-2's relatively lightweight architecture compared to LLMs with billions of parameters, simplifies the fine-tuning process, allowing for efficient specialisation toward domain-specific tasks.

5.2.1.1 | Generating Titles and Descriptions

To create plausible and contextually appropriate titles and descriptions for fictional games, a pre-trained GPT-2 language model was fine-tuned using a dataset derived from the Steam platform’s extensive catalogue of real-world game metadata¹. This dataset was carefully curated to ensure relevance and quality: entries lacking English text, non-game software (e.g., utilities, videos), and titles flagged for adult content or nudity were excluded. The final corpus comprised roughly 72,000 paired entries of titles and their corresponding descriptions.

For title generation, the model was fine-tuned to the curated set of video game titles. Due to the relatively small size of the dataset, the smallest GPT-2 variant was used, which consists of approximately 124 million parameters. Training data was formatted with clearly defined boundary tokens, using the syntax:

```
<|begin|>game title<|end|>
```

Initial fine-tuning efforts resulted in overfitting, as the model tended to reproduce known titles. To address this, the weights and biases of the final output layer were re-initialised prior to training, which encouraged greater novelty in the generated content and reduced memorisation of the original data.

A second GPT-2 model was fine-tuned to handle both the title and its corresponding description, again building on the pre-trained weights from [168]. The input format for this training included both components structured as:

```
<|begin|>game title<|body|>description<|end|>
```

Table 5.1 shows an example of a generated title and description pair.

This dual-model setup enables two-step generation. First, a new game title is produced by prompting the title-only model with

```
<|begin|>
```

Second, the generated title is passed to the title-description model using the structure

```
<|begin|>game title<|body|>
```

to yield a matching description.

¹<https://store.steampowered.com>

Table 5.1: Examples of text variation methods. Partial mutation randomly selects a space or punctuation, removes subsequent text, and uses the remaining segment to generate the rest of the description (in red). Full mutation rewrites the entire description, using only the game title (in bold) to generate a new one (in red).

Original	Hooey! You Got a Monster!? , You’ve been selected for an experiment at the University of Chicago’s Animal Research Center. A girl, who has lost her memory in this creepy and magical place will use your memories to discover why you got kidnapped by Professor Teller on his way there...
Partial mutation	Hooey! You Got a Monster!? , You’ve been selected for an experiment at the University of Chicago’s Animal Research Center. A professor with no memories about his past finds you. An unforgettable story about two animals, their lives and what happened to them in this dream-like place.
Full mutation	Hooey! You Got a Monster!? , The best game of all time is back, and better than ever. This year we’ve reimagined the classic arcade platformer with over 100 levels to conquer. Gameplay has been revamped for even more fun, it’s more difficult then ever but much harder in this new version!

5.2.1.2 | Mutating the Description Text

Two distinct variation operators are employed during the mutation of textual descriptions. Partial mutation preserves continuity with the preceding description, whereas full mutation replaces it entirely to mitigate the risk of premature convergence. In the case of partial mutation, the initial segment of the description is retained, determined by splitting the text at a selected space or punctuation mark near its midpoint. To increase the likelihood that this split occurs near the centre, three such positions are randomly chosen within the text, and the middle one is selected. The text from that point onwards is discarded. This retained fragment, along with the corresponding game title, is provided as input to the second GPT-2 model to generate the continuation.

Conversely, full mutation involves discarding the entire original description; only the game title is supplied to the second model, which then produces a novel description anew. Full mutation is applied with a probability of 20%, otherwise a partial mutation is used. Examples of the resulting textual variations are presented in Table 5.1 to illustrate the process.

5.2.1.3 | Behavioural Characteristics of Text

In order to segment the feature map of MELiTA metrics for the BCs of each modality have to be defined. A range of techniques for text characterisation were investigated

in the context of this application, including traditional readability metrics such as the Gunning-Fog index [82]. This index estimates the years of formal education required to comprehend a piece of writing, based primarily on sentence length and word complexity. Ultimately, topic modelling was selected as the most suitable approach, using topics derived from the Steam dataset. The objective is to align with prevalent thematic patterns observed in game descriptions, which represent the typicality of the domain of this use case [173].

For this purpose the LDA algorithm was used to categorise the generated text into topics [11]. The algorithm was trained across a range of topic counts, varying from 4 to 30. Based on evaluations using both perplexity and complexity metrics, 16 topics were identified as the optimal number.

The refined LDA model is applied in order to characterise each generated description, generating a distribution of probabilities which reflects the extent to which the description corresponds to each of the learned topics. When no single topic achieves a probability that exceeds others by a margin of at least 40%, the description is deemed unclassified. In such cases, the description is excluded from the feature map and is thereby omitted from subsequent stages of the evolutionary process [145]. This mechanism may be interpreted as a rudimentary instance of constrained optimisation, as discussed in prior literature [28].

5.2.2 | The Image Modality

To generate cover images for the hypothetical games a SD pre-trained model [175] with openly released code and weights is used. This model was chosen because it was able to produce images of good quality with modest computational resources. At the time this experiment was conducted, the image quality it delivered was considered to represent the state of the art.

For the generative process, the model was supplied with two prompts. The positive prompt, to which the model attempts to align the generated image, contains the title and the description of the target game in the following format:

“title, description”

Initial trials showed that the image aesthetics were considerably improved if a negative prompt is also supplied to the model. When provided, the model will use both convergence towards the positive prompt as well as divergence from the negative prompt as its generative goals. Guided by community suggestions and following experimentation by the author, the negative prompt used was constant for all processes and set as follows:

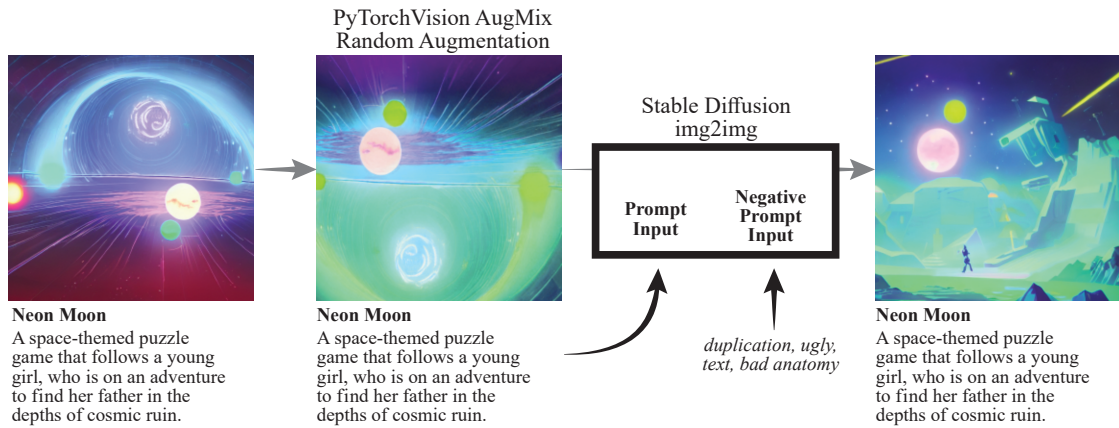


Figure 5.1: Example of image variation: The text modality of the parent is employed as a prompt for image-to-image SD regeneration, together with “standard” negative prompts.

“duplication, ugly, text, bad anatomy”

5.2.2.1 | Mutating the Cover Images

Analogously to the approach taken with text, image mutation is performed at the phenotypic level. For this modality, this operation is conducted through a structured, two-stage process. The first stage involves applying the AugMix augmentation function [87], which is part of the TorchVision software library [140], in order to introduce controlled distortions to the original image. In the second stage, this distorted image is combined with the original textual inputs, including the game title, description, and negative prompt. The combined inputs are then passed into a Stable Diffusion (SD) model configured for image-to-image transformation, as illustrated in Fig. 5.1.

All image generation and mutation tasks are carried out using a fixed schedule of 40 diffusion steps. This choice reflects a compromise between achieving high image quality and maintaining computational efficiency. Due to the inherent variability in the distortions introduced by the augmentation function, the similarity between the final output and the original image can vary considerably. Unlike textual mutation, which allows for more deterministic control over the extent of change, image-based mutation involves greater uncertainty in the resulting phenotype. However, this variability is viewed as beneficial, as it helps prevent premature convergence and encourages a more dynamic and exploratory evolutionary process.



Figure 5.2: Some example images showing the combined extremes of complexity and colourfulness adopted as a visual behavioural characteristic.

5.2.2.2 | Behavioural Characteristics of Images

Several quantitative metrics were evaluated for the purpose of categorising the image modalities. Ultimately, the selection was guided by the pragmatic balance between computational simplicity and interpretability, resulting in the adoption of two complementary measures: *complexity* and *colourfulness*.

Image complexity is defined as the proportion of edge pixels detected using the HED model [229], relative to the total number of pixels in the image. Although traditional approaches in evolutionary art have employed Sobel or Canny filters for estimating complexity [135], the HED model confers a significant improvement, offering greater precision while effectively suppressing noise at the detected edges.

Image colourfulness, is quantified through a metric that captures the perceptual intensity of chromatic content, effectively measuring saturation and richness [86].

The benefit of using these metrics is that they represent properties of the image that can be readily interpreted by a human observer. To fuse these two metrics into a unified scale, each of the two values, complexity and colourfulness, is discretised into four ordinal categories: very low, low, high, and very high. The pairwise combination of these categories yields a total of sixteen distinct bins, enabling a classification scheme that mirrors the dimensionality of the corresponding textual BC. Some example images illustrate typical categorisations combining the extremes of colourfulness and complexity may be seen in Figure 5.2

5.2.3 | Applying MELiTA to the Use Case

Having defined all the necessary metrics and algorithms, MELiTA may be applied to the chosen bimodal use case. Parent selection is carried out in accordance with the

Monte Carlo Elites algorithm [186], which has been shown to enhance the diversity and coverage of final elites in the solution archive. At every iteration of the evolution, the UCB algorithm [110] is evaluated to select the next parent. This balances exploration and exploitation by considering the frequency with which each parent has been selected.

The underlying feature map is structured as a grid of 16×16 cells, with each individual assigned a position based on two BCs: one derived from text topics and the other from image complexity and colourfulness, as described in Sections 5.2.1.3 and 5.2.2.2.

Fitness is evaluated using the CLIP score from Equation 2.4, which measures semantic alignment between an image and its corresponding description, inclusive of the title as outlined in Table 5.1. This score determines whether a given individual is retained in the archive. In the baseline MAP-Elites approach, a selected parent undergoes variation to generate either a new image or a new textual description. The resulting artefact is then paired with the unmodified component of the parent (i.e., the original description or image, respectively) to form a novel offspring.

In contrast, the Transverse Assessment procedure introduces a more elaborate evaluation mechanism. When MELiTA selects a parent, it randomly chooses to apply variation to either the image or the description using operators specified in Sections 5.2.1.2 or 5.2.2.1 respectively. For illustration, Figure 5.3 depicts a scenario in which the image component e_V is transformed into e'_V . This newly generated image is then combined with the parent's original text component e_T to form a candidate solution E' . The CLIP score, text BC, and visual BC of E' are computed accordingly; in the example shown, E' maps to the previously unoccupied cell at row 5, column 1.

The process continues by pairing the new image e'_V with the title and description of each elite individual whose visual BC matches that of E' , i.e. all elites residing in the same row as E' in the feature map (as illustrated with R_1, R_2 in Figure 5.3). Each resulting pairing forms a new temporary candidate whose CLIP score is computed. As these combinations involve existing BC values, these need not be recalculated. These candidate solutions are ranked based on fitness and starting from the highest, each is evaluated for potential archive inclusion. If a candidate exceeds the fitness of the elite currently occupying its corresponding cell, or if the cell is vacant, it is added to the archive, replacing the incumbent if necessary. Once a successful insertion occurs, the assessment concludes. If none of the candidates meet the inclusion criteria the archive remains unmodified and a new parent is selected for the next iteration.

5.3 | Structure of the Experiment

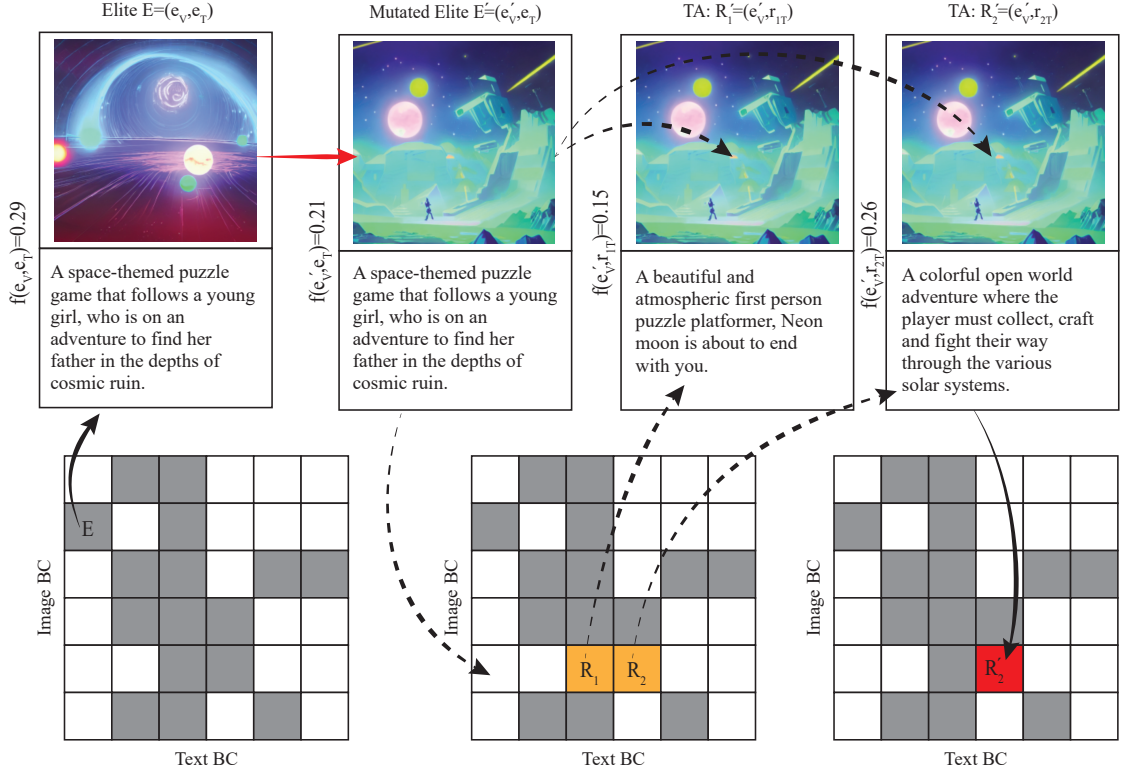


Figure 5.3: Simplified feature map of the MELiTA process for this use case, with grey cells representing elites. From a selected elite E , a modified image e'_v generates three candidates from E , R_1 , and R_2 . These are ranked by CLIP score as $\vec{L} = \{f(R'_2), f(E'), f(R'_1)\}$. If $f(R'_2) > f(R_2)$, the candidate R'_2 , combining the image from E' and text from R_2 , replaces R_2 . Otherwise, E' fills the empty cell at row 5, column 1).

5.3.1 | Hypotheses

The central hypothesis proposed in this experiment postulates that MAP-Elites with Transverse Assessment yields multimodal assets exhibiting higher fitness and greater diversity relative to the baseline MAP-Elites algorithm implemented *without* Transverse Assessment.

More specifically, this overarching hypothesis, is further subdivided into the following sub-hypotheses:

- H1 The final archive of solutions produced by MELiTA exhibits higher quality and greater diversity than that produced by the baseline MAP-Elites algorithm.
- H2 MELiTA identifies solutions of higher quality and diversity more rapidly than MAP-Elites.

H3 The final archive of elites discovered by MELiTA is perceptually and semantically more diverse in comparison to MAP-Elites without transverse assessment.

Hypothesis H1 concerns the outcome of the evolutionary process, focusing on the end-state archive. Hypothesis H2 evaluates the efficiency and progression of the evolutionary dynamics throughout the search. Both H1 and H2 quantify diversity based on the BCs explicitly encoded in the feature map. In contrast, H3 relies on alternative diversity metrics, selected to better align with human judgments of diversity within each modality under consideration. Experiment EXP2 tests these hypotheses and is detailed in this section.

5.3.2 | Protocol

Initially, 100 game titles were generated as described in Section 5.2.1, from which 7 titles listed here were hand-picked based on criteria of thematic diversity and a varied word count.

T1 “Neon Moon”

T2 “Lion King”

T3 “Hexgrave”

T4 “Fantasy Fables: The Legend of the Flying Sword”

T5 “The Princess of Thieves”

T6 “The Shadow Warrior 2: Shadows of the Past”

T7 “Hooey! You Got a Monster!?”

Ten independent evolutionary runs are conducted for each of the above title, using both algorithms, namely MELiTA and MAP-Elites. In order to ensure a standardised and fair initialisation of the population, the GPT-2 language model is employed to generate one hundred textual descriptions per title. Subsequently, for each of these descriptions, four corresponding images are generated using the SD model. Within each evolutionary run, the initial population of 100 hundred individuals is formed by pairing each textual description with a randomly selected image from its four associated candidates. These individuals are then placed into the appropriate cells within the feature map as dictated by the algorithm in use. Each experiment consisted of 2000 evolutionary iterations, i.e. 2000 parent selections, mutations, and placements of novel individuals. This structure is visualised in Figure 5.4.

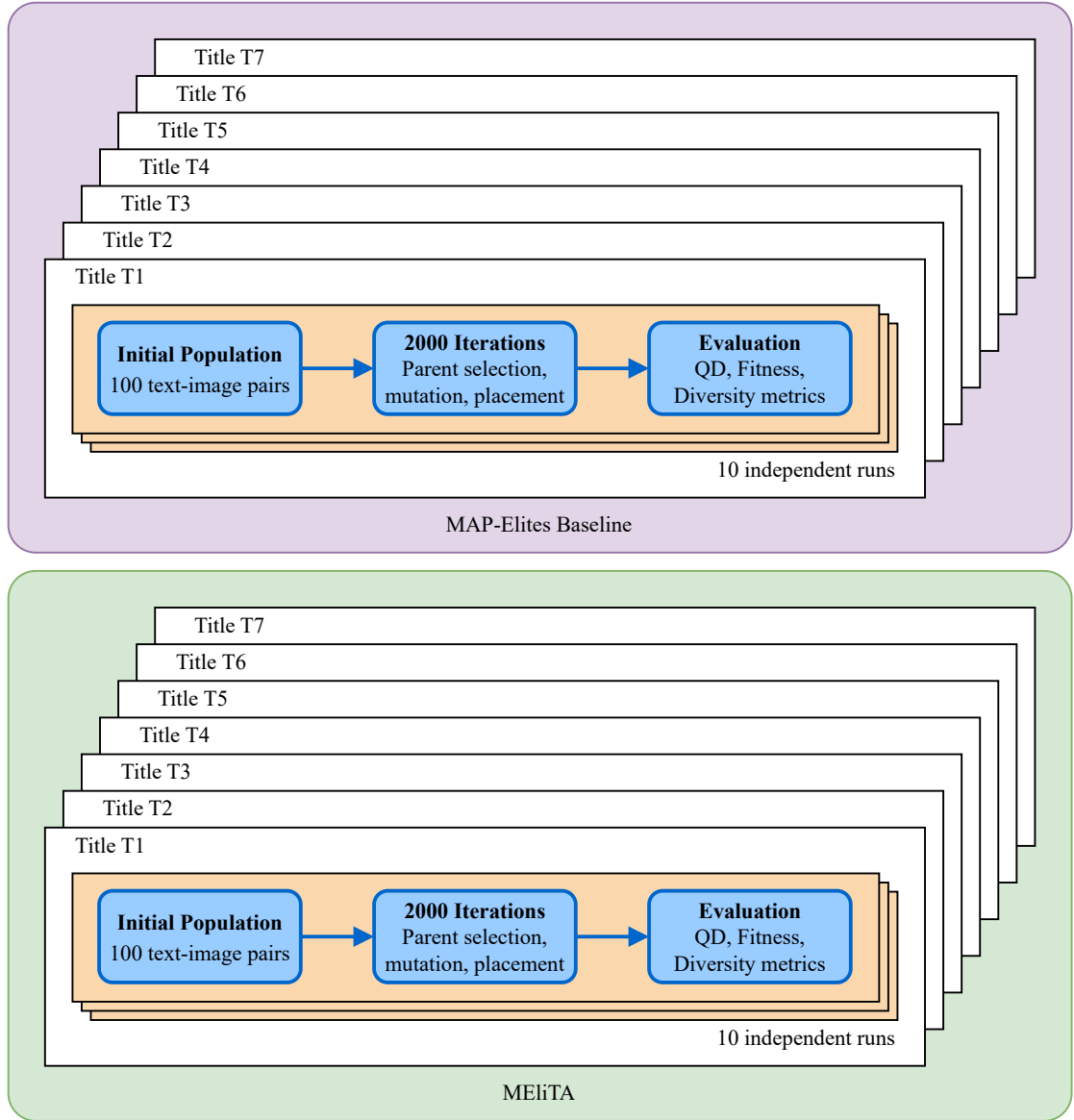


Figure 5.4: The MELiTA experiment structure. 2000 iterations are carried out per title before the final archive is evaluated. 10 independent runs of each experiment are conducted. The entire process is carried out using the MELiTA algorithm as well as for the MAP-Elites baseline.

5.3.3 | Evaluation Metrics

In order to evaluate hypotheses H1 and H2, standard performance metrics established in the context of QD evolutionary search are employed [150].

With regard to H1, the evaluation is conducted using the final archive, from which

four key metrics are computed: the mean fitness of occupying elites, the maximum fitness observed among these elites, the coverage (defined as the proportion of occupied cells relative to the total number of cells in the grid), and the QD score (calculated as the sum of all elite fitness values). The first two of these metrics are indicative of solution quality, whereas coverage serves as a measure of diversity with respect to the specified BCs, and the QD score reflects an integrated assessment of both quality and diversity. In contrast to the procedure outlined in [150], the values are not normalised to the maximum obtained across runs or methods.

In the case of H2, the same metrics are monitored throughout the evolutionary process, specifically after each instance of parent selection, and subsequently evaluated via the Area Under the Curve (AUC). A high AUC value may suggest either that strong performance metrics were attained by the end of the process, or that such performance was achieved at earlier stages.

To address H3, the final elites are analysed using diversity measures that are orthogonal to those employed in populating the feature map. For visual diversity, two complementary metrics are used: the LPIPS, which provides a distance measure from a NN trained on human perceptual judgments of visual difference [239], and the SSIM [216], which is a metric frequently applied to quantify image quality degradation due to compression or transmission artefacts. In assessing textual diversity, the semantic embeddings produced by the SBERT encoder [172] are used, as these representations encapsulate the meaning of each description. The cosine similarity (see Equation 2.4) between the latent vectors of elite descriptions is computed in order to derive the SBERT distance [45]. For each distance metric, both the mean distance of an elite to all other elites and the nearest-neighbour distance to the most similar elite are calculated, the latter serving as a more conservative indicator of visual or textual similarity.

5.4 | Results and Evaluation

The comparison between MELiTA and MAP-Elites is carried out by evaluating the final feature maps after 2000 evolutionary iterations, for each of the 10 runs performed. The non-parametric two-tail Wilcoxon Rank-Sum test [224] is used to establish statistical significance in the performance differences observed between the algorithms under test. A significance level of $p < 0.05$ was used for these tests.

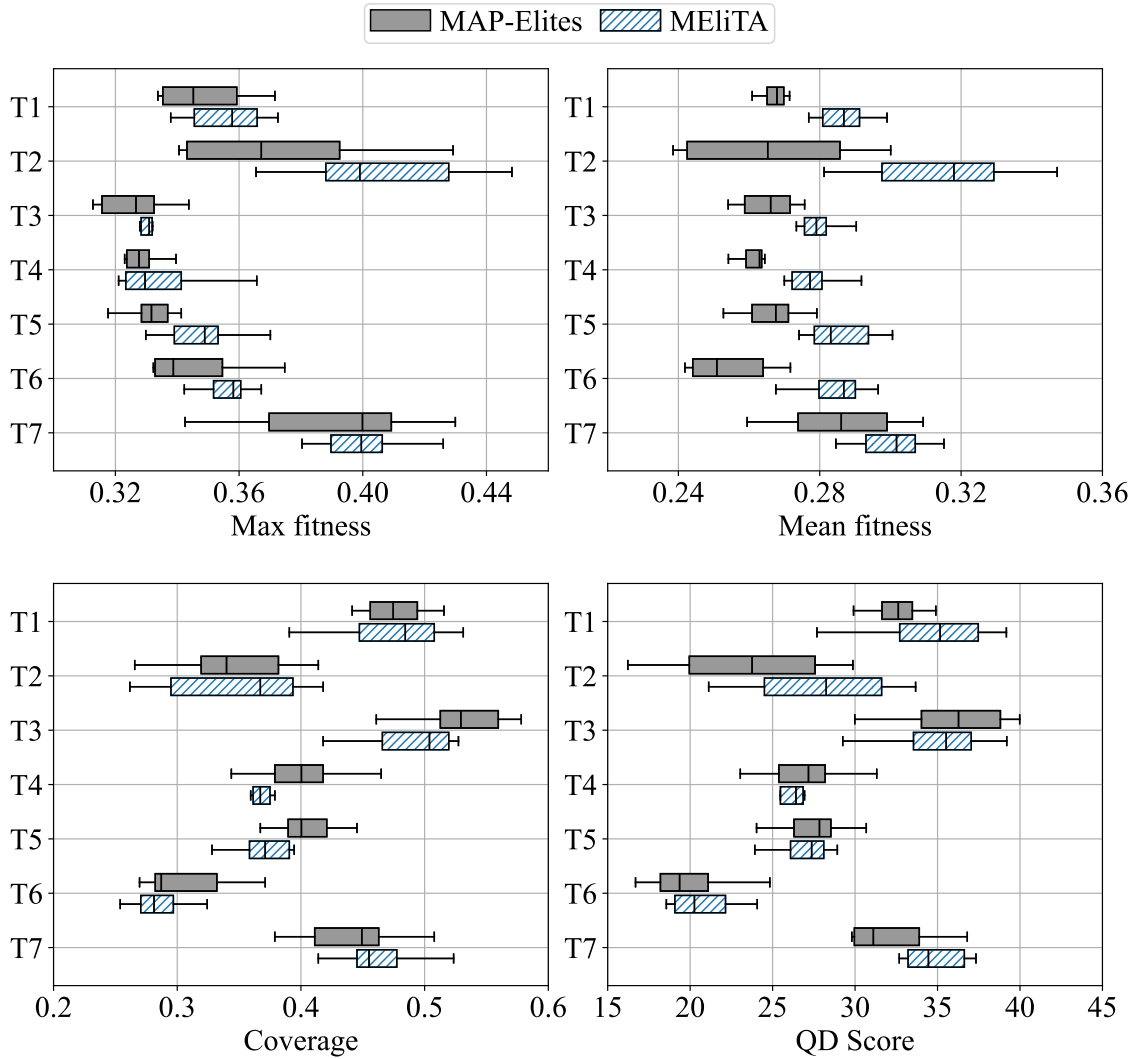


Figure 5.5: Box plots compare metrics of the final archives after 2000 iterations of MELiTA to those of the MAP-Elites baseline. Values are averaged over 10 runs for each title.

5.4.1 | Evaluating the Final Quality and Diversity

Figure 5.5 presents the four evaluated performance metrics of the QD algorithm as applied to the final archive. It can be observed that MELiTA generally yields individuals of higher fitness than those produced by MAP-Elites. This is evidenced by statistically significant improvements in mean fitness across six out of seven game titles, with the exception of T7. Additionally, a statistically higher maximum fitness is achieved in three of the seven titles, specifically T2, T5, and T6.

These outcomes suggest that the solutions evolved through MELiTA tend to exhibit

a greater degree of coherence between textual and visual elements. Such coherence is likely attributable to the mechanism by which text or image outputs are selectively paired with counterparts from the archive that provide a more semantically aligned match, as opposed to relying solely on the original image-text pairings produced by MAP-Elites. However, this strategic preference for combining artefacts across distinct elites appears to come at the expense of reduced feature space coverage. Although this reduction in coverage is minor for the majority of game titles, MAP-Elites achieves significantly greater coverage in two instances: T3 and T5.

Given that MELiTA produces a smaller set of elites, albeit with higher fitness, the resulting QD scores between the two methods are, in general, comparable. A notable exception arises in the case of T7, for which MELiTA attains a significantly higher QD score than MAP-Elites.

From this analysis, it can be concluded that MELiTA offers enhanced performance in terms of quality, though this is occasionally offset by a diminished exploration of the feature space. Consequently, hypothesis H1 receives partial validation: support is found with regard to quality, but not with respect to diversity.

5.4.2 | Evaluating the Progress of Quality and Diversity

Figure 5.6 presents the AUC values computed across 2000 parent selections for various QD performance metrics. These AUC results are consistent with the trends previously identified in Section 5.4.1. In particular, both the mean and the maximum fitness values were observed to increase at a faster rate under MELiTA when compared to MAP-Elites. Out of the seven titles under study, this trend reached statistical significance in three of them for the mean fitness, and in four for the maximum fitness. In contrast, coverage tended to improve more rapidly when using MAP-Elites, with statistically significant differences observed in two of the seven titles. With respect to the QD score, the AUC values were largely comparable between the two methods, although MELiTA exhibited a significantly higher AUC exclusively in title T1.

From these observations, it can be inferred that MELiTA tends to identify higher-fitness individuals more rapidly than MAP-Elites. Consequently, hypothesis H2 is supported in terms of quality improvements but not fully validated with respect to diversity.

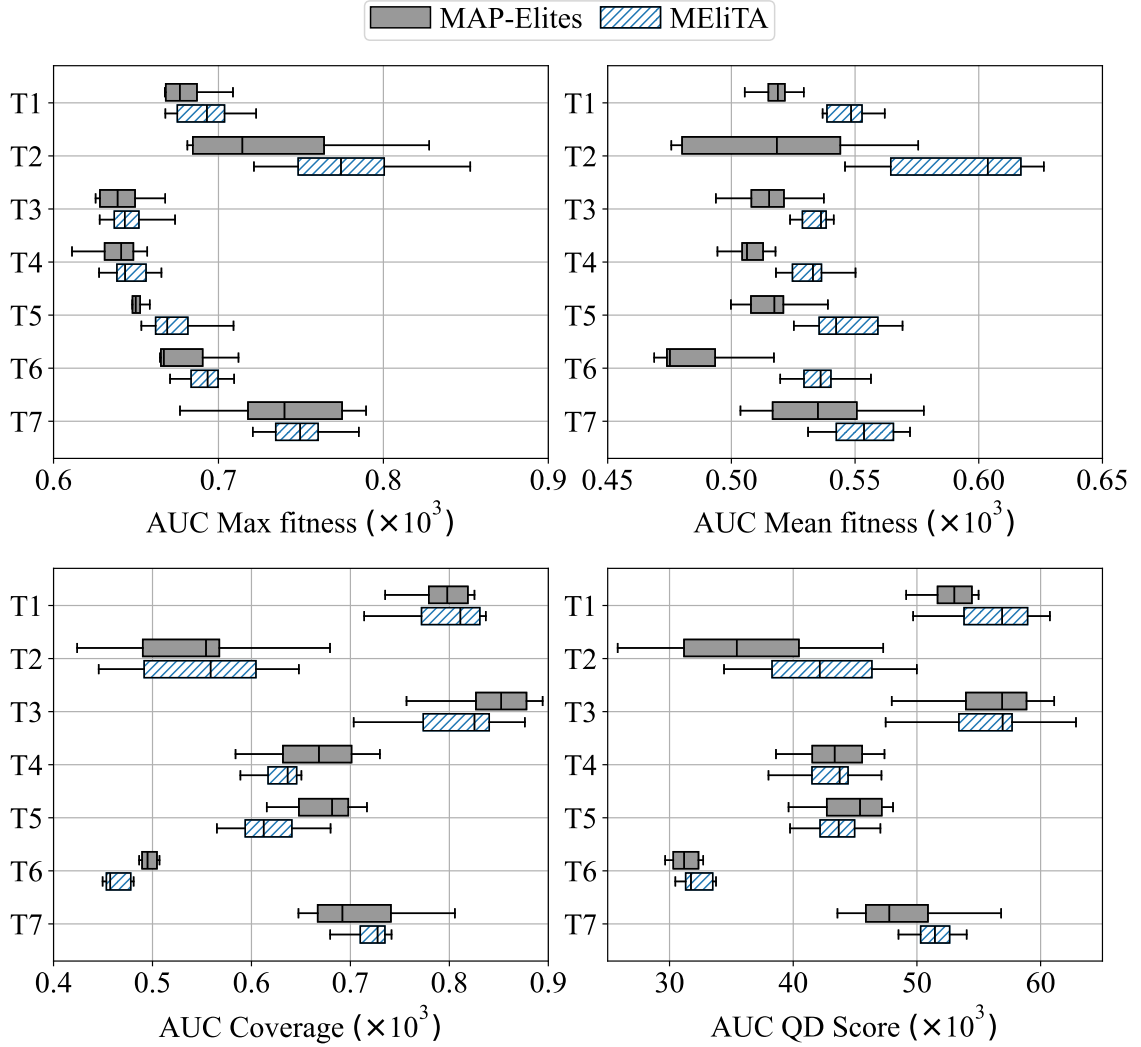


Figure 5.6: Box plots compare the area under the graph of QD metrics throughout 2000 iterations of MELiTA to those of the MAP-Elites baseline. Values are averaged over 10 runs for each title.

5.4.3 | Evaluating Image and Text Diversity in the Final Archive

In order to evaluate the perceptible differences between the archives generated by the two methods, visual and textual distances were employed. Specifically, perceptual metrics LPIPS and SSIM were used to quantify visual dissimilarities, while SBERT embeddings were utilised to assess textual divergence. These comparisons were made between the final elites obtained after 2000 parent selections in both MELiTA and MAP-Elites. The results, as presented in Figure 5.7, include both mean and nearest-neighbour distances for each metric. Notably, these measures are largely independent of the BCs used by

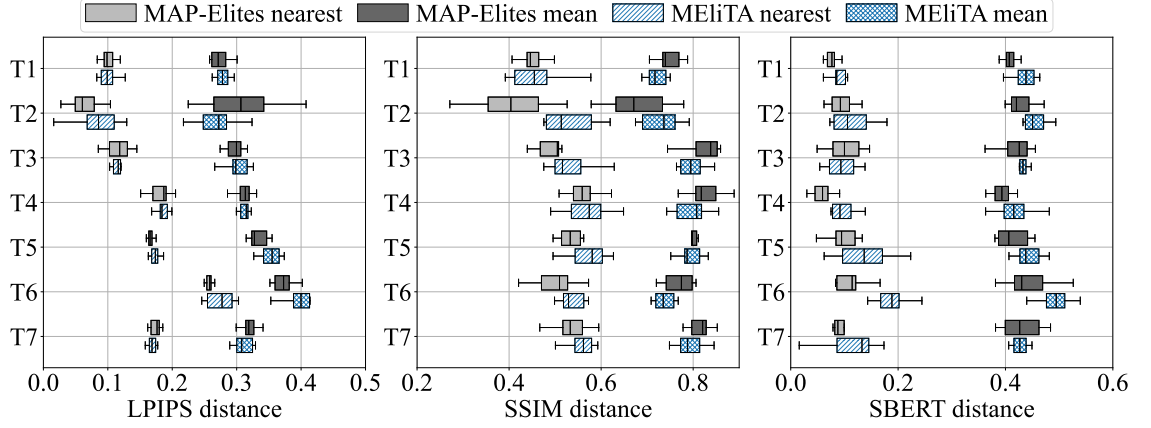


Figure 5.7: Box plots of mean and nearest-neighbour distance metrics for visuals and text among final elites. Metrics are aggregated over 10 runs per title for both MELiTA and MAP-Elites algorithms.

MAP-Elites.

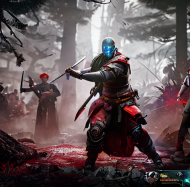
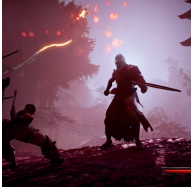
Figure 5.7 reveals that, in aggregate, the final outputs produced by MAP-Elites do not exhibit a consistently higher or lower degree of diversity in comparison to those generated by MELiTA. When visual diversity is considered, MELiTA demonstrates significantly greater mean LPIPS distances in the outcomes associated with two game titles (T5 and T6), and significantly higher nearest-neighbour LPIPS distance in one title (T5). Furthermore, a significantly higher nearest-neighbour SSIM distance is observed for MELiTA in two additional game titles (T2 and T3). Although these results do not establish a globally consistent trend, they suggest a tendency for MELiTA to yield more visually diverse images than MAP-Elites.

More pronounced trends are observed with respect to textual diversity. MELiTA produces significantly higher mean SBERT distances in the final archives for three game titles (T1, T2, and T6), and significantly higher nearest-neighbour SBERT distances for four titles (T1, T4, T5, and T6). Despite the absence of any single game title for which MELiTA significantly outperforms MAP-Elites across all diversity metrics, these findings collectively support the validation of hypothesis H3.

It can thus be concluded that, although the archive sizes may be smaller, MELiTA consistently produces archives that are both more visually and textually diverse than those yielded by the baseline MAP-Elites.

It is also insightful to examine variations among elite samples through direct analysis. The title T6, “The Shadow Warrior 2: Shadows of the Past”, was selected due to its consistently elevated values in both LPIPS and SBERT distances, as well as the presence of a representative run exhibiting high scores for both metrics. To this end, k -medoid

Table 5.2: Sample MELiTA output for title T6, “The Shadow Warrior 2: Shadows of the Past” (T6). Each sample was selected as a medoid via k -medoid clustering ($k = 5$) of the final archive. The generated description is shown with its original errors.

Game 1	Game 2	Game 3	Game 4	Game 5
				
In this game, players have to use their sword and weapons wisely because there are many enemies who will try very hard for you. After many defeats in a long time, each one offers new challenges.	After a global catastrophe, you are left alone to face your past. The greatest evil - an alien race called Korda - is planning on tearing Earth apart as they do every few centuries in this new adventure from Arkane Studios and Infamous Games.	Play as a shadow warrior, taking on different threats in this third person shooter that's more brutal than ever before.	In this game, players have to use their swords and bows in order not only survive but also fight with others.	After years' long struggle, Dark Lord Arthur's dark legacy is coming to light, and a new evil seems lurking at every corner. TheShadow Warrior2 : Shadows OfthePast.

clustering was conducted using a Euclidean distance metric that integrates both SBERT and LPIPS components. The resulting medoids for $k = 5$ are presented in Table 5.2.

The textual descriptions corresponding to these samples demonstrate considerable variation. In certain cases (specifically Games 1, 3, and 4), the descriptions primarily summarise gameplay mechanics or actions. In contrast, other descriptions reflect narrative elements, with Game 2 incorporating motifs from science fiction and Game 5 adopting a medieval Camelot-inspired theme. Several of the textual descriptions also include extraneous or seemingly irrelevant content, as demonstrated by the conclusion of Game 5's description, which may be attributed to generative noise. Furthermore, overlap among descriptions can be observed, particularly in the case of Games 1 and 4, which may result from the application of partial mutation operations within the generation process.

With respect to the associated imagery, most samples portray dynamic or action-oriented scenes that give the impression of being extraneous from the bounds of the hypothetical gameplay. An exception can be found in Game 5, which features a first-person shooter perspective that includes a visible firearm, situated within an otherwise fantastical environment. The characters across the medoid images are generally clad in armour and tend to have obscured facial features, a pattern that may be influenced either by genre-specific visual tropes (e.g., ninja iconography) or by negative prompt constraints that strive to suppress undesirable anatomical inconsistencies. Swords are

frequently depicted as weapons, although modern firearms are also present, most notably in Games 3 and 5. The backgrounds exhibit a degree of uniformity, with most images featuring a fog-laden forest landscape. However, Game 2 presents a notable deviation, showcasing a warmer colour palette and flame-like wing structures emanating from the central character.

In summary, while the visual medoid samples do not exhibit an especially high degree of diversity, particularly in terms of thematic representation of “ninja” figures or variation in colour schemes, a consistent relationship can nonetheless be observed between the textual descriptions and their corresponding images, despite the sometimes generic nature of the descriptions. Of particular concern is the inclusion of real-world game studio names in the description for Game 2. This phenomenon is likely a consequence of GPT-2’s exposure to the Steam dataset during the fine-tuning process, and raises potential issues regarding intellectual property, in scenarios which would involve public or commercial dissemination of the generated content.

5.5 | Discussion

Through Experiment EXP2, which focused on a use case involving the generation of fictional game descriptions and corresponding cover images, it was determined that MELiTA is capable of producing a smaller, yet more suitable set of elite solutions. It was further acknowledged that this improvement in quality is accompanied by a reduction in feature map coverage. This reduction arises from the algorithmic process wherein a new elite is added to the archive only if no superior alternative is identified when evaluating the offspring’s modified modality in conjunction with the modalities of existing elites.

Despite this decline in coverage, no corresponding reduction was observed in the visual and textual diversity of the generated elites, provided that the assessment metrics were decoupled from the BCs. When measured using both the mean pairwise distance and the nearest-neighbour distance, it was observed that the elites generated by MELiTA exhibit greater visual and semantic diversity. Furthermore, these elites demonstrated higher coherence, as evidenced by elevated fitness scores, in comparison to those produced without the use of transverse assessment.

It should also be noted that throughout its process, MELiTA yields a substantially larger number of candidate solutions due to its use of transverse assessment, in contrast to the single offspring typically generated by MAP-Elites. As a result, the number of fitness evaluations required by MELiTA can be significantly higher, although the number of BC evaluations remains constant, since only a single individual bearing the unaltered

modality of the parent is subject to such an evaluation. For instance, in the examined use case featuring a 16×16 feature map, MELiTA may conduct up to 16 fitness evaluations per parent selection, whereas MAP-Elites would require only one. The computational cost associated with calculating the CLIP score is minimal, rendering the increased number of fitness evaluations largely inconsequential in this context. However, it must be acknowledged that this may not hold true in scenarios involving simulation-based fitness functions [150], or if the algorithm is applied to more than two modalities.

Observations of the outputs illustrated in Table 5.2 further revealed certain limitations inherent in the employed variation operators. When variation was performed directly on the phenotype (either image or text), the resulting artefacts were found to be more controlled and less noisy compared to those produced via latent variable evolution [67, 233]. However, textual variation specifically was prone to generating individuals with descriptions which overlap in part. This issue stems from the partial nature of mutation (refer to Section 5.2.1), and other mutation strategies, such as the employment of a reverse text generation model [221], or genotypic-level operators that modify the underlying text embeddings [113] could potentially mitigate this. It is also important to acknowledge that the GPT-2 model used in the experiment is relatively outdated within the evolving ecosystem of LLMs. Although more recent, larger generators offer improved performance, their use presents practical limitations, particularly with respect to computational resources, longer generation times, and traceability of generated content.

In terms of computational resources, MELiTA processes additional fitnesses during placement than the baseline MAP-Elites. In this experiment, since the archive comprised of 16×16 cells, the maximal additional computation would have entailed 15 individuals. In several cases, especially during the initial iterations, the actual calculations were decisively fewer. However, the additional computational cost was negligible, compared to that of the text and image mutations, which dominated the computational budget of the overall process.

While the current use case focused on a bimodal problem that was well-suited to visualisation on a two-dimensional feature map, the MELiTA framework could be extended to accommodate a broader range of artefact modalities and higher-dimensional BC spaces. The correspondence between artefact modalities and BC axes is not strictly necessary. In fact, each modality could comprise multiple BCs. For example, in the present use case, the two image-based metrics complexity and colourfulness could have been treated as distinct BCs, thereby forming a three-dimensional feature space. The potential of MELiTA also extends to more complex multimodal creative domains. An illustrative example would be the co-evolution of text, images, and audio within an interactive storytelling environment. In such an application, a LLM could be employed

to generate the narrative and associated locations, while SD could generate background visuals for each scene. A text-to-music system such as *MusicGen* [41] could be utilised to produce an appropriate atmosphere soundtrack. The fitness of these multimodal outputs could then be evaluated using a cross-modal embedding model such as *ImageBind* [77], by computing cosine similarities across different modalities.

In reference to the Research Questions posed in Section 1.4 of Chapter 1, particular attention has been paid to the manner in which EXP2 engages with and responds to the central investigative aims of this study.

Specifically, RQ1 is directly answered through the deliberate incorporation of GenAI methodologies with QD algorithms. In particular, it has been demonstrated that the application of MELiTA to the use case incorporating GenAI models, leads to an enhancement in both the semantic alignment among the final elite solutions as well as their perceived diversity, in comparison to the baseline algorithm.

RQ2, which pertains to the structural organisation of the search space, is likewise addressed through the construction of the feature map and the selection of suitable BCs, which together facilitate the effective operation of MELiTAs. This structural strategy, whereby each modality relevant to the use case is allocated a distinct dimension within the search space, has been instrumental in enabling the observed improvements produced by the proposed algorithm.

Furthermore, while EXP2 is not explicitly oriented toward the full resolution of RQ4, partial progress in that direction is presented. A central contribution of this experiment lies in the application of a set of carefully chosen evaluation metrics, which serve to characterise the structure of the underlying search space in a multimodal generative setting. These metrics have been selected both for their light computational requirements, as well as for their effectiveness in quantifying the various characteristics of the generated artefacts. These metrics align with the objectives outlined in RQ4. By employing such measures, it becomes possible to obtain a more nuanced and quantifiable understanding of the generative landscape, especially with regard to the interplay between diversity and quality. This is particularly the case through the adoption of established similarity and fitness evaluation frameworks. The CLIP score has been employed as a means of assessing fitness by measuring semantic alignment between generated asset pairs. Diversity has been analysed through the use of perceptual similarity metrics such as LPIPS and SSIM, alongside semantic textual diversity measures like SBERT.

The success of these metrics within the context of the present experiment, not only partially answers RQ4, but also suggests that their application may extend beyond this specific use case. Here, the generalisability of the conclusions is stronger than in EXP1, as the contribution is largely algorithmic and representation-independent. MELiTA's core

idea does not depend on the specific choice of text descriptions, image generation models, or game-themed content. The observed benefits arise from how multimodal candidates are compared and promoted, rather than from domain-specific heuristics. That said, the instances used still constrain the scope of generalisation. The behavioural characteristics selected reflect a particular interpretation of diversity that may not transfer cleanly to other multimodal pairings.

Within similar domains, it is plausible to envisage that employing more expressive language models and mutation operators capable of global semantic restructuring would further improve diversity and yield more meaningful variation. Additionally, adaptive or hierarchical behavioural characterisations may preserve coverage while maintaining transverse coherence across modalities.

5.6 | Limitations and Shortcomings

This chapter extends QD optimisation to multimodal artefacts, but remains constrained by the use of predefined behavioural characteristics and internally defined fitness measures.

A key limitation is that while MELiTA avoids populating regions of the search space with weak elites, the resulting sparsity limits exploration of niche or marginal regions that may still hold creative value. As such, the algorithm prioritises quality over representational completeness, a trade-off that may not be appropriate for all creative applications.

As in the previous chapter, evaluation is performed without direct human grounding. The automated metrics employed enable systematic comparison between MAP-Elites and MELiTA, but they do not capture subjective judgements of coherence, originality, or usefulness. This limits the extent to which conclusions can be generalised to creative practice or player-facing applications.

Diversity is more explicitly addressed than in EXP1, yet it remains discretised through selected behavioural dimensions. While this discretisation is necessary for illumination-based search, it inevitably privileges certain forms of variation and excludes others that may be equally meaningful but harder to quantify. The resulting archives should therefore be interpreted as structured design spaces rather than complete representations of multimodal diversity. Qualitative inspection of the elite artefacts reveals that increased metric diversity does not always correspond to strong thematic differentiation. Many textual descriptions remain generic, and several visual outputs share common compositional tropes. This suggests that MELiTA enhances diversity largely within the expressive limits of the underlying generative models, rather than expanding those limits outright.

These shortcomings highlight the trade-off between structured exploration and expressive freedom. They also motivate a shift in focus from abstract multimodal artefacts to functional content generation, which is next undertaken in Chapter 6.

5.7 | Summary

This chapter has detailed Experiment EXP2 which combines GenAI models for visual and text assets with MAP-Elites, a QD algorithm. A variation on the latter, MELiTA, was proposed to improve fitness and diversity of the resulting feature map. In the next chapter, an automated LLM-driven pipeline for a wider range of game visual and narrative asset generation will be explored.

A Zero-Shot Multimodal Pipeline

This chapter continues to build on multimodal generation, addressing a more pragmatic use case within digital game development. Taking advantage of the flexibility of pre-trained LLMs, Experiment EXP3, proposes a fully automated, generative zero-shot pipeline that regenerates the visual and text assets required to replace the theme of the game with any given setting provided as a semantic input. The pipeline is orchestrated by a LLM and iteratively constructs the required narrative elements, and the text prompts that subsequently drive a text-to-image model to also generate the game visuals. This work emphasises two aspects:

1. All the models used in the pipeline are pre-trained, and have not been fine-tuned for this specific task, hence the pipeline being labelled zero-shot.
2. The generated assets are ready to use directly in the game engine. No manual processes are required to select, adjust or format them for use, making the pipeline automated.

The objective of the experiment is to investigate the coherence between the generated assets and the top-level theme. To assess this, a user study is also presented, in which human participants are asked to assess the coherence of the generated content to the top-level theme descriptors.

6.1 | Methodology

The out-of-the-box adaptability of pre-trained LLMs foundation models elicits their use in a wide variety of roles within the domain of games [72]. Drawing inspiration from the top-down orchestration described by [124], this work proposes a pipeline for game asset generation orchestrated by a LLM. Driven by a top-level theme, the LLM generates the narrative elements which form the backbone of the game, such as the game name, protagonist, locations, antagonists, objectives, and others. The orchestrator generates

elements whose functionality in the game is predefined, and its prompts to generate these are hand-crafted specifically for the type of game, or more specifically, the ruleset of the game being processed. It produces not only semantic descriptors for player-facing narrative, but also for other elements of the game whose visuals need to be generated. Diffusion models are then employed, driven by these LLM-generated descriptions to generate the graphical elements. Since the game engine has specific constraints on the format of the assets, algorithms such as ControlNet [238] and [232] can be leveraged in order to generate the assets in the correct configuration directly.

Since the pipeline was intended to be automated, all prompts submitted to the LLM request its output to be given in JavaScript Object Notation (JSON) format, giving an example of the required schema, to enable automated parsing of the LLM output in ensuing stages.

Although prior research has made attempts to address the challenge of game asset creation through the application of GenAI models [228], and even full game generation encompassing the game engine itself [19], such efforts have typically relied on either models that were specifically trained for the task, or on manual processes to ensure the correct integration of the generated assets into the target game engine.

The present chapter introduces several key distinctions. Notably, the models employed herein are pre-trained and utilised without any task-specific fine-tuning, i.e. the proposed pipeline operates in a zero-shot fashion. Furthermore, the entire asset generation process has been fully automated, thereby enabling the output to be directly incorporated into the game engine without the need for human intervention.

A further, albeit subtle, point of differentiation lies in the choice of model architecture. Whereas a substantial portion of existing literature has favoured the use of large, proprietary models such as GPT-4 [154] owing to their impressive capabilities, the approach presented here instead leverages smaller-scale, open-source models that are capable of running on local hardware.

From the author's perspective, this methodological choice carries significant implications. Proprietary models hosted on third-party infrastructure are subject to updates or alterations without prior notification, often resulting in unpredictable shifts in output. In contrast, locally run models offer greater consistency and reproducibility. Moreover, the relatively modest computational requirements of the selected models enhance the accessibility of the pipeline, thereby broadening its applicability and potential utility within the wider research and development community.

6.2 | Use Case: The CrawLLM Video Game

CrawLLM is a procedurally generated dungeon crawler game which has been designed and developed specifically as a use case for the proposed generative pipeline. It was developed in the Unity game engine¹ and comprises a short, but fully specified game, from start to finish. It is used as a template that defines both the player's experience and the assets required, in both form and format, which are further outlined in Section 6.2.1.

A hierarchical generative pipeline was also developed to produce the required assets in a top-down, sequential manner [124]. The pipeline is driven by LLMs, which are tasked with generating high-level descriptions of the entire game. These descriptions subsequently serve to guide and constrain the downstream generative processes responsible for producing the textual and visual assets required by the game. The pipeline is triggered by inputting a textual description of the required theme (see Section 6.2.2). Since these descriptions are purely textual, they may also be generated by the LLM itself, as was done for Experiment EXP3.

6.2.1 | The Game Design

CrawLLM was designed as a top-down 2D dungeon crawler with card-based combat mechanics. The term “dungeon” is employed here in a broad sense, drawing on genre conventions; while it typically refers to labyrinthine and constrained spaces, the actual structure in each instance of CrawLLM need not visually resemble a castle or medieval prison, but is rather dictated by the given theme descriptors.

The dungeon itself is composed of interconnected rooms and corridors, which are organized into discrete locations. A maximum of five such locations may exist within any given dungeon. Although the dungeon's layout is generated anew for each play-through (as described in Section 6.2.1), the textual and visual assets used are pre-generated (see Section 6.2.2). At the outset of gameplay, a brief introductory message is presented, establishing the identity of the Player Character (PC), their narrative context, and their objective. Thereafter, the player navigates the PC across the 2D map in cardinal directions, attempting to locate the dungeon's exit, thereby complete the game. This goal necessitates traversal through various locations (illustrated in Figure 6.1(a)), each of which contains multiple rooms and corridors.

Rooms within the dungeon may be separated by doors, which can either be locked, or function as one-way portals, closing behind the PC and rendering reverse passage impossible. Locked doors require one or two keys to open. Keys are scattered throughout

¹<https://unity.com/>

the dungeon and may be collected through direct collision with them; no other collectible items are present. To augment navigational complexity, secret doors may also be present. When uncovered, these are converted into regular corridors, revealing otherwise hidden passageways. Each location contains a unique enemy type and up to two minion types associated with it. The primary enemy appears in the final room of the location, while minions are distributed throughout the other rooms. Both enemies and minions exhibit movement within rooms, following randomised, simplistic patterns. Upon colliding with a mobile adversary, the player is transitioned into a distinct interface for card-based combat.

Within the card combat view (shown in Figure 6.1(b)), the PC and their adversaries are represented as cards annotated with hit points and status indicators, while the player’s hand comprises action cards. Although minions may participate in combat, they remain invisible in the dungeon crawling view. The number of accompanying minions is determined by the room’s danger level, a parameter controlled by the dungeon generation system (refer to Section 6.2.1).

Combat proceeds in a turn-based manner. At the start of each turn, the player draws cards into their hand until a total of five is reached. Each action card incurs a mana cost and is discarded upon use. Alternatively, cards may be discarded without effect by choosing to end the turn. If the player’s deck is exhausted, the discard pile is shuffled to form a new deck. A standard allocation of three Mana Points (MP) per turn is granted, though certain cards may modify this amount. During the enemy’s turn, adversaries act randomly by either attacking for variable damage, healing for a random amount of Hit Points (HP), or deploying a shield to absorb incoming damage. Additionally, some minions are capable of inflicting poison damage. Should the PC’s HP reach zero, the game is lost. Conversely, adversaries reduced to zero HP are removed from combat and cease to act. Victory in combat is achieved when all opponents have been defeated, upon which the player is presented with a choice to acquire one new card from a set of three options.

The initial composition of the player’s deck remains constant across all games. However, reward cards are selected at random from the complete set of predefined cards in the system. Each card is framed in bronze, silver, or gold, with the frame’s colour providing a visual cue of the card’s relative power. While the mechanical effects and mana costs of cards are predefined and listed in Table 6.1, their titles and visual designs, including card frames, are generated in accordance with the game’s overarching theme. For instance, the “Basic Attack” card referenced in Table 6.1 may appear as “Renaissance Riposte” in-game (as illustrated in element 15 of Figure 6.1(b)).

The quantity of assets required for a complete game is directly influenced by the

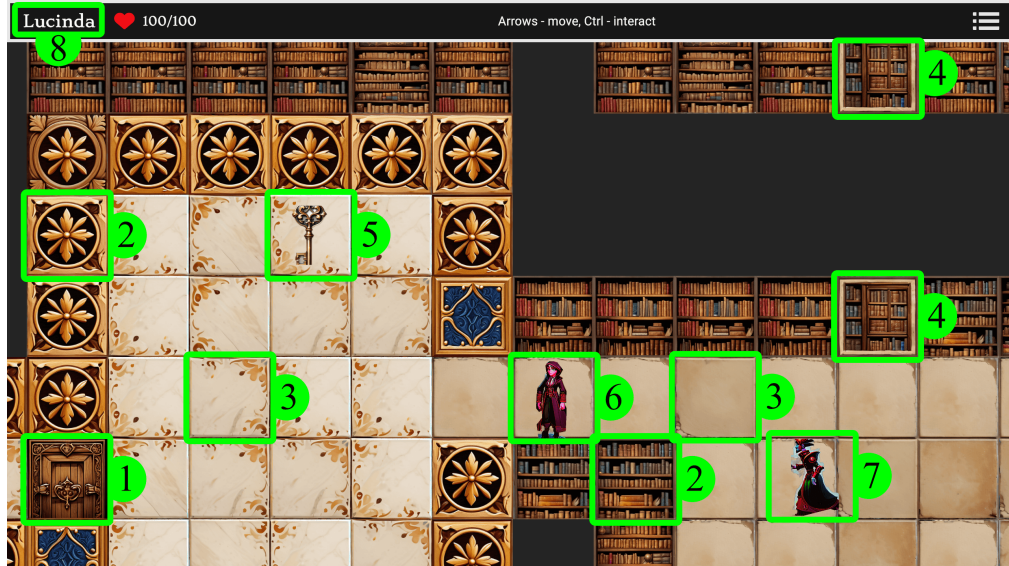
Table 6.1: Cards in CrawLLM, with their in-game effect; the original card title listed is subsequently adapted by the LLM.

Original title	Effect	Cost	Frame
Basic Attack	Deal 6 damage	1	Bronze
Fast Attack	Deal 3 damage	0	Bronze
Basic Block	Gain 5 block	1	Bronze
Basic Draw	Gain 2 cards	0	Silver
Basic Heal	Heal 3 HP	1	Bronze
Increase Max Health	Increase max HP by 5	2	Gold
Increase Strength	Attacks do 4 more damage	2	Gold
Gain Mana	Gain 2 mana	0	Silver
Life Steal	Deal 5 damage, heal 5 HP	2	Gold

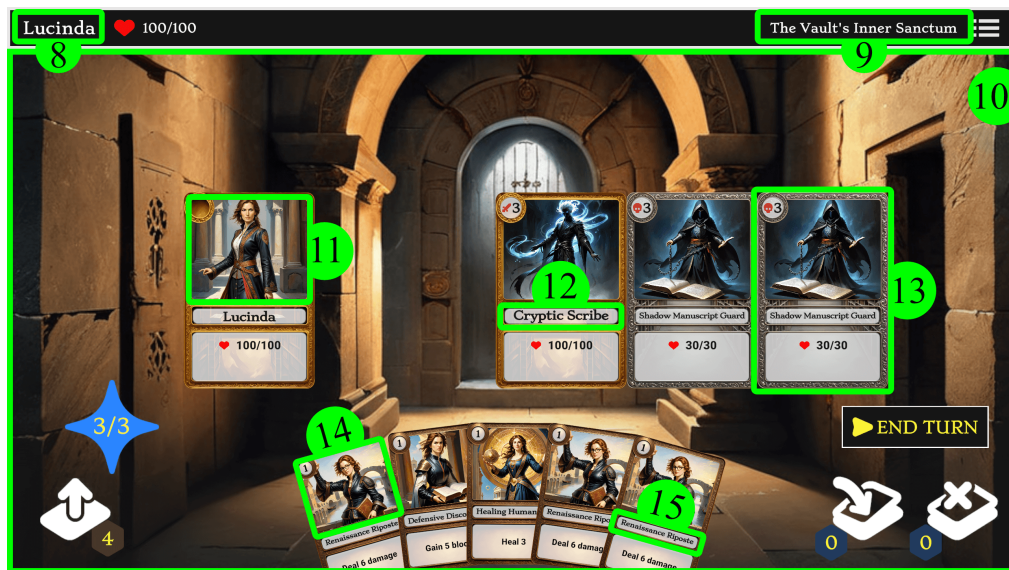
number of locations present in the dungeon. Given that up to five locations may be included, the textual assets to be generated comprise names for one PC, five enemies, ten minions, and nine action cards, as well as one introductory and one completion message. Visual assets include three card frames (one per rarity tier), sixteen sprite sheets and sixteen combat portraits (covering the PC, five enemies, and ten minions), nine action card illustrations, five combat backgrounds, twenty key sprite images (four per location), and five tileset images. Each tileset pertains to one location within the game, and contains a 4×4 grid of tiles comprising nine wall tiles, four floor tiles, two door tiles, and one secret door tiles. These assets are required to exhibit coherence not only with the thematic setting of the game, but also internally within their respective contexts of use, e.g. ensuring that wall textures are visually compatible with corresponding floor tiles in a location. A detailed explanation of the processes used to generate these assets is provided in Section 6.2.2.

6.2.2 | The Generative Pipeline

The aim of this chapter is to implement a pipeline based on LLMs, designed to orchestrate [124] the automatic generation of all requisite textual and visual assets for the CrawLLM game. This orchestration facilitates the re-theming of the system across a wide spectrum of distinct narrative settings. The pipeline is initiated through the generation of high-level semantic constructs (see Section 6.2.2.1), which, although not presented directly to the end user, are utilised as auxiliary prompts for the generation of both player-



(a) Dungeon Crawling view



(b) Card Combat view

Figure 6.1: Gameplay elements within a CrawlLLM game session. Labelled elements are door (1), wall (2), floor (3), and secret door (4) tiles, key (5), PC (6), and enemy (7) sprites, PC (8), location (9), enemy (12), and action card (15) names, and location (10), PC (11), card frame (13), and action card (14) images. The game depicted is “*Luminary’s Legacy: The Renaissance Vault*”

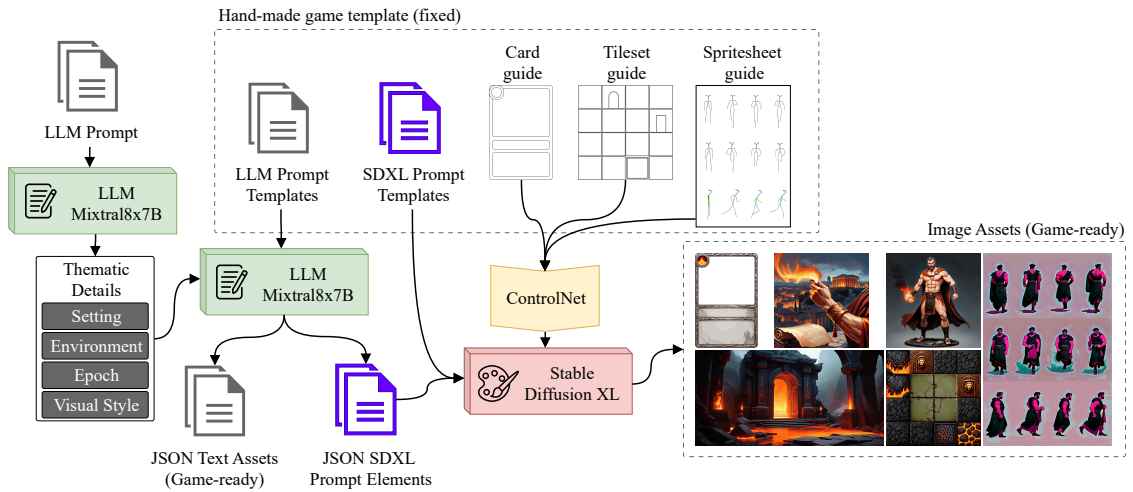


Figure 6.2: The overall structure of the CrawlLLM generative pipeline. The static manually-generated templates are used to generate text and visual assets in a format which can be used directly within the game engine. The entire process is driven by the seed thematic details that may be generated by a LLM as shown, or alternatively supplied as natural-language textual input in JSON format.

facing textual elements (see Section 6.2.2.1) and visual components (see Section 6.2.2.2). Figure 6.2 illustrates the overall structure and flow of the pipeline.

Initially, a series of preliminary tests were conducted to evaluate the performance of select LLMs for generating video game content. The smaller models such as the 2-billion parameter Gemma 2 [75] managed to generate content, but often did not respect specific request, such as generating an exact number of location when requested, and formatting its output in correctly formatted JSON, which was the selected data transfer format across the generative pipeline. Larger models such as the 8-billion parameter versions of LLAMA2 [203] and LLAMA3 [58] offered a more consistent output, and were observed to generate more consistent JSON. However, their output was considered by the author to be repetitive for the chosen domain in comparison to the larger models. The next models to be tested, comprising 13-billion active parameters were Mixtral 8×7B [97] and Orca 2 [148]. Both of them performed reasonably well both in terms of instruction following, domain knowledge, as well as JSON formatting. Both models were set an appraisal task to generate a set of fifty video game themes. The request was repeated 10 times per model. A cursory examination of the results revealed a marginally more diverse output being given by the Mixtral 8×7B model, as the Orca 2 output contained more repetitions of similar themes. Thus Mixtral 8×7B was selected to perform all LLM-driven text generation tasks within the CrawlLLM pipeline. This model is executed

locally through the Ollama platform².

The full set of prompts employed in the generation of both textual and visual content is made openly accessible via an online repository³ for reference.

6.2.2.1 | Text Assets Generation

The semantic information forming the game's thematic setting is represented in textual form and constitutes the initial output of the generative process. Wei et al. [217] highlights the importance of time and space within the narrative of video games. Therefore, the primary thematic descriptors were chosen to be the overall *setting* to establish the overall genre of the narrative (e.g., "A cursed carnival during a sandstorm"), the *environment* which establishes the narrative space (e.g., "Tilted rides, ghost tents, dunes and ruins"), and the *epoch* to locate the narrative within a time frame (e.g., "Late 20th century"). In addition, an artistic *style* (e.g., "Watercolor, whimsical") is specified to steer the generation of visual assets in subsequent stages. Since each of these four thematic elements are specified in natural language, it is possible to initiate the pipeline process with human-generated input for each of these. However, for this experiment, this task was also delegated to the LLM by formulating a single prompt to produce all these components concurrently, yielding a response in JSON format.

All thematic elements, with the exception of the visual style, inform the generation of a range of textual artefacts that are at least partially revealed to the player in the game. This part of the pipeline focusing on text generation is detailed in Figure 6.3.

Through carefully constructed prompt templates, populated with the setting, environment, and epoch, the LLM is first prompted to generate an initial set of game components. These include the game title, the PC profile, an introductory message shown at the beginning of the game, and a completion message presented upon the successful conclusion of the game at the dungeon's exit. Furthermore, five discrete in-game locations are generated, each described by name and accompanying narrative content, to guide subsequent generative steps. While the game title and the two narrative messages are presented in full to the player, the PC's detailed profile and the narrative description of the locations do not form part of the game. All generated characters' profiles include their name, species, gender, and description. Despite the species and gender seeming superfluous, they are essential to ensure that these initial characteristics are preserved through the various stages of image assets generation, as this information may not always be explicitly stated in the character description, yet it has a fundamental

²<https://ollama.com/>

³<https://github.com/m-a-r-v-i-n/CrawLLMPrompts>

impact on the visual representation. Only the PC name and the names of the locations are disclosed in the User Interface (UI) of the game. The undisclosed elements serve primarily as conditioning inputs for downstream content generation.

The PC profile, together with the core thematic attributes (excluding visual style), is employed in a series of prompts aimed at generating the game's action cards. Each such prompt is augmented with the structured descriptors of the action cards as outlined in Table 6.1. The LLM is then requested to generate a name for each action card, to repurpose the original title in Table 6.1, along with an accompanying action description. While this description is not visible to the player, it is utilised in the subsequent generation of the card's visual representation, as detailed in Section 6.2.2.2.

Each of the previously generated locations is associated with one primary enemy character and two subordinate minions. The creation of these assets is handled through dedicated LLM invocations. The primary enemy is generated using a prompt that includes the relevant location description, the PC details, and all core thematic components except for visual style. The enemy's name, species, gender, and description are returned. Subsequently, a single prompt is used to generate two minions per enemy. This prompt includes the primary enemy profile, the relevant location's details, and the core thematic components (again excluding visual style). As with other background elements, the descriptive details of these adversaries are not disclosed directly to the player. Only their names are presented during combat sequences (see Fig. 6.1(b)).

Finally, the semantic specification of each location is extended to include textual descriptions for its walls, floors, doors, and keys. These elements are generated based on the core thematic elements, excluding the visual style, and the semantic attributes of each specific location. While these descriptions remain hidden from the player, they serve as inputs to the generation of the tileset used in the dungeon-crawling interface (see Section 6.2.2.2).

6.2.2.2 | Generation of Visual Assets

In contrast to the semantic assets produced during the preceding stage (see Section 6.2.2.1), all generated visual assets are intended for display to the player. Figure 6.4 indicates all the generated images, showing which semantic elements are used within their input prompts.

These assets are used across the two distinct gameplay contexts, namely the dungeon crawling interface (e.g., tilesets, player character and enemy animations), and the card combat interface (e.g., action cards and location backgrounds). Due to the intricate nature of the generation processes associated with the dungeon crawling view, these processes

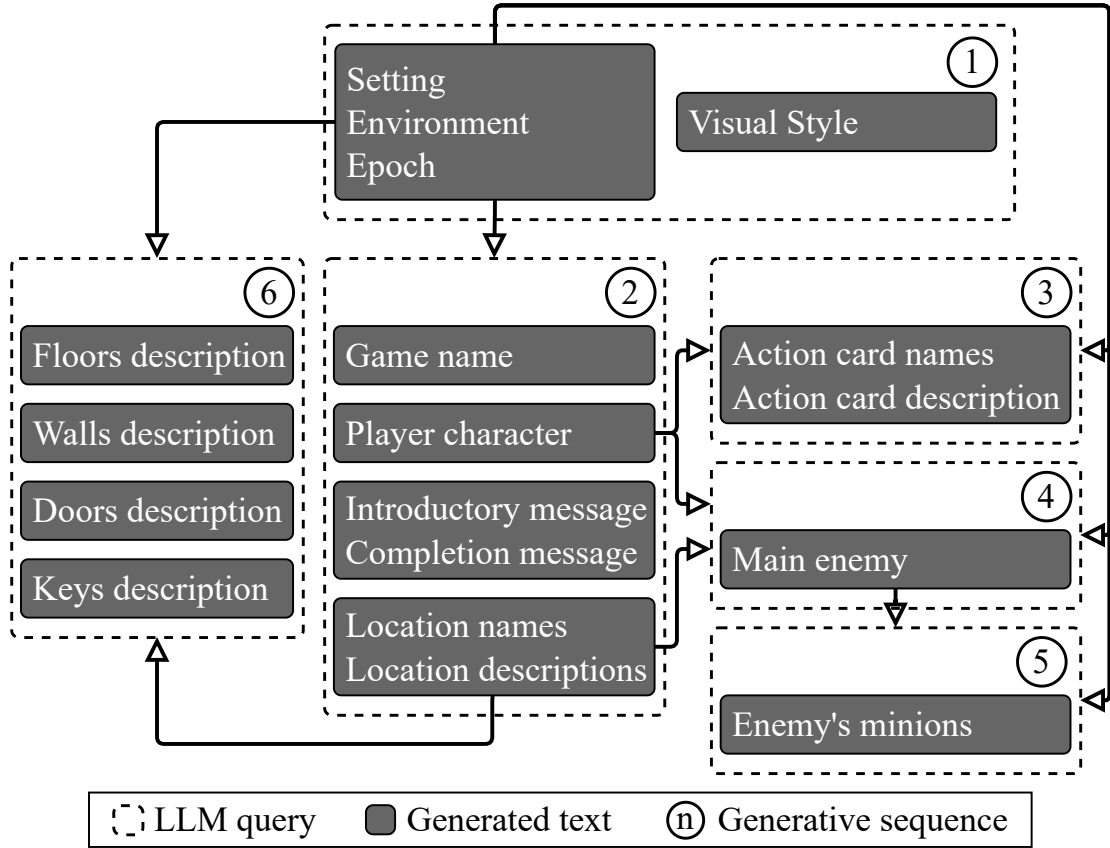


Figure 6.3: The sequence of text generation via Mixtral 8×7B. Arrows indicate the reuse of generated text as part of the prompts in the following stages. Steps 4, 5, 6 are repeated for each location in the game (five times in the current paper).

are addressed in detail in dedicated subsections below. The visual asset generation related to the card combat interface is subject to fewer constraints in comparison.

All visual assets have been generated using *Artium V2.0*⁴, a community fine-tuned version of the SDXL model [162], selected for its versatility across a broad range of artistic styles. Furthermore, the SDXL ecosystem includes several ControlNet models [238], which allow the generated imagery to conform to either a predetermined line drawing or a specified character pose. To facilitate visual coherence across character representations, the IP-Adapter [232] has also been employed. This tool enables character consistency by conditioning the generation process on a provided reference image.

It should be noted that SDXL typically produces high-resolution outputs, with optimal results obtained at resolutions of 1024×1024 pixels. Although this resolution

⁴<https://civitai.com/models/216439/artium>

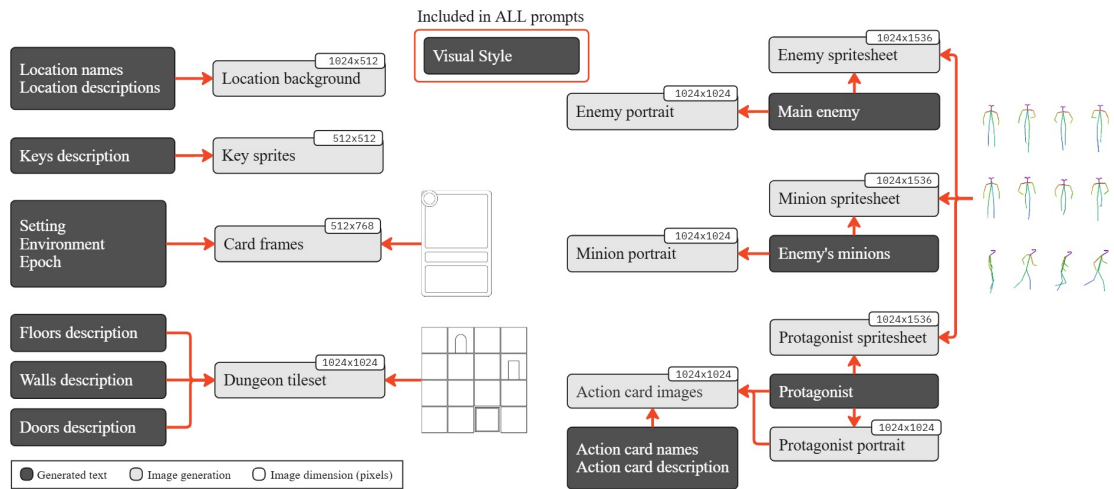


Figure 6.4: The various SDXL image generation stages are shown with the input semantic elements. Ad-hoc ControlNet guides are shown where applicable. Image dimensions in pixels are shown for reference.

exceeds the requirements for simple tile-based level design, high-resolution assets have been retained throughout the CrawlLLM system in order to preserve visual fidelity. All prompts used for visual asset generation incorporate stylistic guidance derived from LLM outputs (see Section 6.2.2.1); this aspect will not be reiterated in the descriptions that follow. All line drawings and masks referenced herein were created ad hoc by the author and are included as components of the hand-generated game template.

6.2.2.3 | Tilesset Generation

Each location is composed of floor, wall, and door tiles, which are required to be arranged within a 2D grid, serving as a navigational backdrop for the player. The visual generation of these elements must clearly delineate traversable surfaces (floors), impassable boundaries (walls), and interactive elements (doors and secret doors). Ensuring visual clarity and proper demarcation between these elements has proven non-trivial and thus warrants further elaboration.

In preliminary experiments, floor, wall, and door tiles were generated independently as separate image assets. While these components exhibited acceptable quality when assessed in isolation, their integration often resulted in visual inconsistencies. Specifically, floor tiles were frequently rendered with excessive detail, or wall textures failed to provide sufficient contrast, thereby undermining the perceptual distinction between navigable and obstructed areas.

Subsequent experiments adopted a revised methodology in which the entire tileset

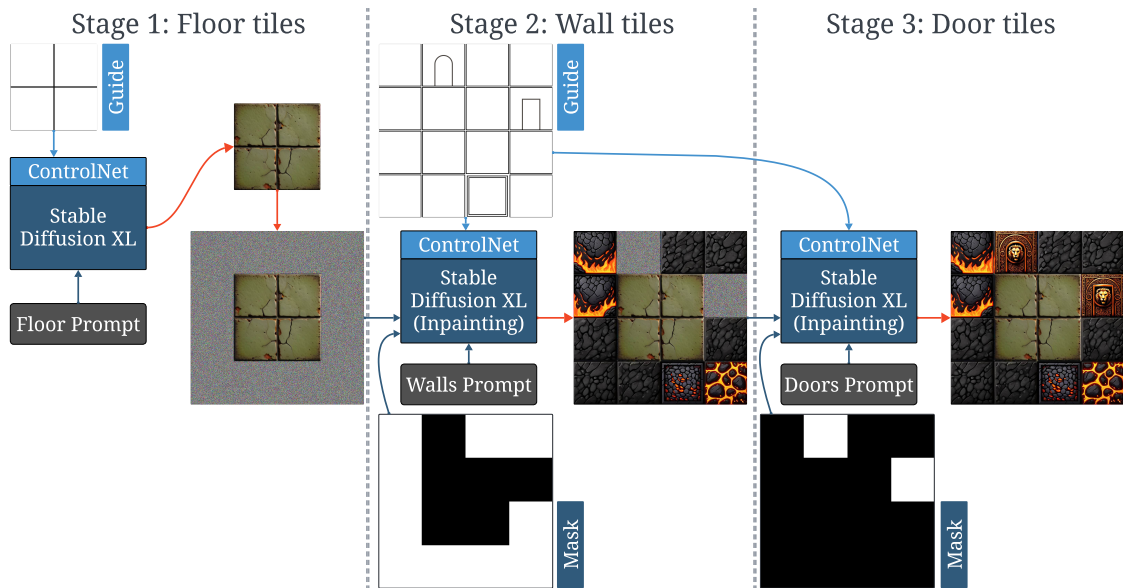


Figure 6.5: The 3-stage process of tileset generation. Tiles from the game “Aeternum Ruptura” are shown.

for each location was produced as a unified image. Distinct tile types were encoded within specific regions of this image, guided by ControlNet using a custom-designed outline. Despite this refinement, the SDXL model demonstrated limited success in reliably distinguishing wall regions from floor regions, suggesting the need for a more structured generation process.

In the current implementation of CrawLLM, tilesets for each location are generated as a composite image across three sequential stages, as illustrated in Figure 6.5. The process begins with a simple line drawing that specifies the configuration of the central four floor tiles. This outline, alongside a semantic prompt describing the floor tile characteristics (refer to Section 6.2.2.1), is input to SDXL via ControlNet. The resulting image, containing the initial floor tiles, is then expanded to the full dimensions of the tileset, with stochastic noise filling the remaining unpainted regions.

During the second stage, this intermediate image is supplied to the model together with semantic prompts describing the wall textures, an updated line drawing to guide ControlNet, and a binary mask that designates the wall tile regions for inpainting. This stage produces an image in which nearly all tiles are rendered, including the secret door tile, which is intentionally generated at this stage to be visually similar to other wall tiles. Only two placeholder tiles remain unpainted at this stage, and are once again overpainted with random noise.

In the final stage, these two noisy regions are used as inpainting targets for the insertion of two door variants. The image is reintroduced into the model along with semantic prompts specific to doors, a corresponding ControlNet line drawing, and a mask indicating the door tile regions. This yields the completed tileset, shown at the far right in Figure 6.5.

It should be noted that when assembling the final dungeon visual (see Figure 6.7), floor, wall, and door tiles are sampled randomly from their respective subsets within the generated tileset. Specifically, each wall tile is selected at random from the set of nine wall textures, each floor tile from the four floor textures, and each door from the two door variants. While this approach introduces greater visual variety within a given location, it may also result in stylistic dissonance when tiles of the same type differ significantly in appearance.

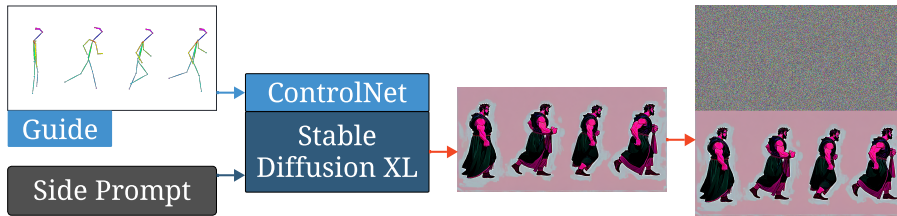
6.2.2.4 | Generating Character Animations

Within the dungeon-crawling environment, the PC, enemies, and minions have to be depicted moving along the four cardinal directions. Therefore movement animations are implemented through the use of *spritesheets*, each of which contains twelve frames of motion subdivided into three rows each corresponding to a primary orientation: front-facing, rear-facing, and right-facing. Leftward movement is rendered as a mirrored version of the rightward animation. Every row comprises four animation frames for movement in that specific orientation, with the first frame corresponding to a stationary pose.

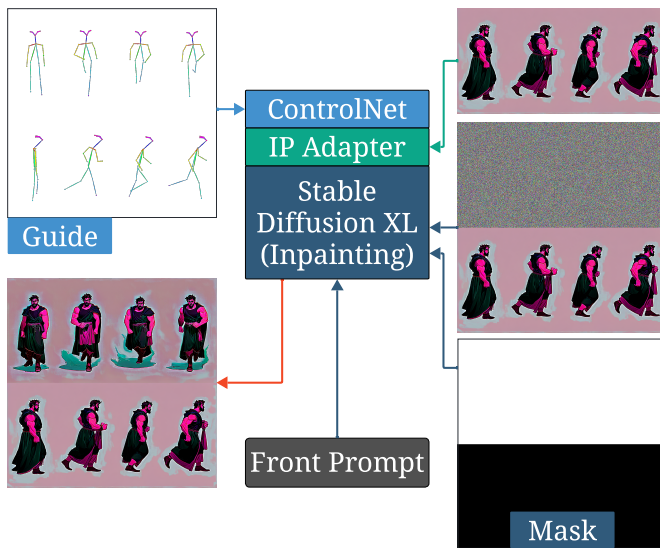
A high degree of visual consistency is required across the frames of a spritesheet, mandating precise control over both pose and positioning for each frame. In order to generate the spritesheets, guide images incorporating predefined OpenPose poses [23] were created within the Unity engine. A three-dimensional humanoid model was captured at sequential key moments during idle and jogging animations. At each selected keyframe, the joints and bones defined by the OpenPose standard were rendered as points and connecting lines. These annotated frames were subsequently compiled into a single composite image to serve as a reference for generation (see Figure 6.6).

Initial attempts to generate complete spritesheets in a single step using ControlNet proved unreliable, as the model frequently confused rear-facing poses with those facing forward. To mitigate this issue, a set of depth maps depicting a humanoid figure aligned to the same set of poses was provided as an additional ControlNet input. Despite improving the pose consistency, this additional depth map input greatly constrained the overall shape of the generated images, which limited the diversity and was deemed too

Stage 1: Side-facing sprites



Stage 2: Front-facing sprites



Stage 3: Back-facing sprites

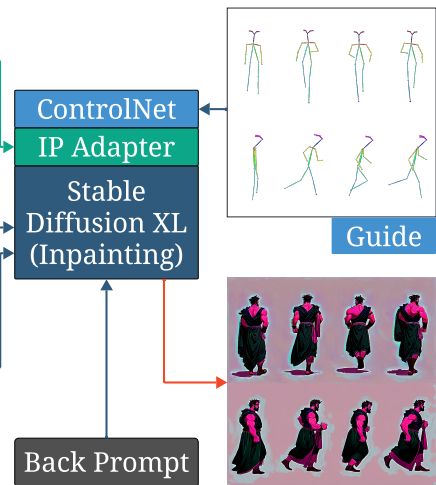


Figure 6.6: The spritesheet generation process is executed in 3 stages. Rear-, front-, and side-facing sprites are consolidated into an image containing 12 poses. Figure shows the PC from the game “*Aeternum Ruptura*”.

restrictive, and the depth map input was removed. Instead, the generation process was subdivided into three discrete stages, each corresponding to a different pose orientation (refer to Figure 6.6). Every stage incorporates text prompts containing relevant character attributes as defined in Section 6.2.2.1.

During the first stage, the side-facing guides were input as a ControlNet reference. Upon generation of the side-facing sprites, the image was extended vertically with randomised noise. This expanded image was then used as a starting point for the next generative stage, with the front- and side-facing pose guides applied to the ControlNet. Simultaneously, an IP-Adapter referencing the previously generated side-facing sprites was incorporated into the SDXL model. A corresponding inpainting mask was supplied to constrain generation to the newly extended region, and prompts were configured to instruct the model to render front-facing sprites of the same character. This process was

repeated to yield the rear-facing poses, using the side-facing set as input, in conjunction with appropriate prompts and guide images. Ultimately, the three resulting sprite groups were collated into a single cohesive spritesheet.

As a final step, since the pre-trained SDXL model does not generate an alpha channel for transparency, background removal was performed to isolate the character sprites. To assist in this process, throughout the generative pipeline, prompts consistently instructed the model to produce imagery against a blank background. Transparency was introduced using *Rembg*⁵, an open-source Python library employing a pre-trained neural network for background identification and removal. On rare occasions, minor elements near the image periphery were incorrectly identified as background and consequently removed.

6.2.2.5 | Generation of Additional Dungeon-Crawling Assets

The visual asset category which remains unaddressed is that of the dungeon keys. These sprites are produced through the use of location-specific key descriptions, in conjunction with the corresponding visual style descriptor, which are combined into the semantic prompts for the SDXL model. Following the initial generation process, a background removal stage is applied using the *Rembg* tool.

6.2.2.6 | Generation of Card Combat Assets

The visual assets for the card combat system are generated through straightforward text-to-image prompting. These prompts require the specification of the visual style as well as asset-specific descriptors: background imagery is derived from location information; enemy character artwork is informed by enemy descriptions; and PC card artwork is generated based on PC attributes.

The generation of the action cards utilise both the name and description of the corresponding action. However, it was observed that the textual descriptions of the actions frequently involved the PC engaged in some action. Therefore, an IP-Adapter guided by the generated PC portrait is coupled to the SDXL model to promote visual consistency of the character within the resulting image.

The generation of card frames was also a non-trivial element within this asset pipeline. A total of three distinct frames are produced per instance of the CrawLLM game, each one colour coded in gold, silver, or bronze, to reflect the rarity of the card it depicts. In contrast to the prompts used for other assets, the prompts for these frames incorporate the full set of thematic descriptors, including setting, epoch, environment, and visual style.

⁵<https://github.com/danielgatis/rembg>

Each card frame is designed with clearly demarcated zones reserved for specific content. A primary region is allocated for card artwork, auxiliary regions for descriptive or functional text, and a compact section is dedicated to iconography that conveys the anticipated actions of the opponent's cards. This layout is defined for all card frames, irrespective of colour, through a predefined line drawing template created by the author. This design is imposed on the SDXL model by supplying the template to a ControlNet model coupled to it. This ensures precise spatial alignment of visual components across the three frame variants. The provided text prompt directs the model to apply graphical variations, not simply colour, to each card class.

6.2.3 | Cyclic Dungeon Generation

The dungeon layout is procedurally generated at the start of every play-through. This layout adheres to a cyclic structure, as described by [54], drawing influence from the algorithm implemented in the video game *Unexplored* (Ludomotion, 2017). The resulting structure ensures the formation of solvable key-and-lock puzzles. Each dungeon *cycle* is composed of multiple interconnected rooms, collectively forming a self-contained location (refer to Section 6.2.1) with its own sequence of doors and keys. The conceptual graph of these locations is structured in a circular fashion, containing a start node and a goal node, which are commonly connected by two distinct edges. Within this abstraction, each node corresponds to an individual room, whereas each edge represents a connecting corridor.

The inclusion of multiple such cycles introduces a layer of narrative coherence and thematic depth. For instance, some cycles may offer alternate paths of varying length towards the goal, while others may necessitate the acquisition of multiple keys before progression is permitted. In order to heighten complexity without compromising the layout's solvability, individual rooms within a cycle may be substituted with entire sub-cycles. This recursive mechanism introduces additional layers of puzzle-solving and strategic decision-making, while maintaining the overall structural integrity and progression logic of the dungeon.

The system employed by CrawLLM relies on a curated set of hand-designed cycles, adapted from those described in [214]. Each node within these cycles encodes specific narrative roles, such as designating a goal room (e.g., one containing an enemy encounter or locked door), indicating the presence of a key, or determining the danger level, which influences the number of enemy minions encountered during card-based combat. Dungeon generation begins with the random selection of a base cycle from this curated set. Subsequently, expansion is achieved by substituting one of the nodes in the selected

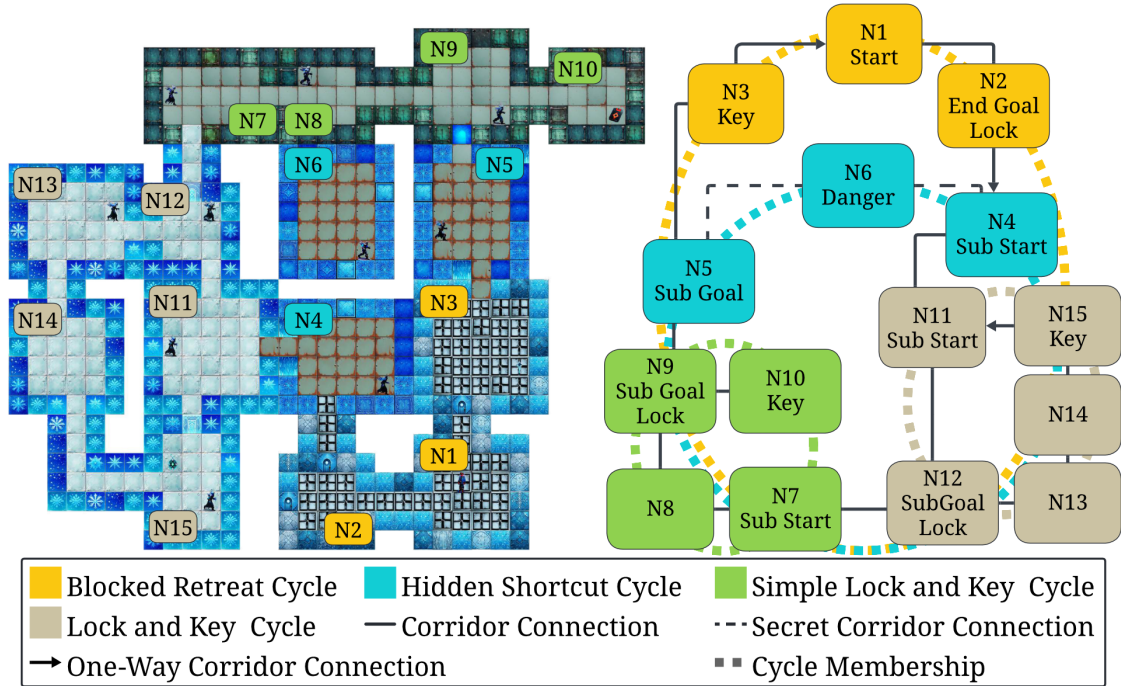


Figure 6.7: An example of a procedurally generated dungeon (left) and its underlying graph representation (right). The hierarchical, nested cycle structure is indicated on the graph. The illustrated game instance is “*Spirits of the Ice: Arctic Research Unbound*”.

cycle with another complete cycle. This substitution process may be recursively applied, as nodes within the newly inserted cycle can themselves be further replaced in similar fashion. The generative procedure continues until a defined set of termination criteria is met, which ensures that the resultant graph can still be feasibly embedded onto a 2D grid. Once completed, cycles are grouped to form discrete locations, as illustrated in Figure 6.7. Although the maximum number of such cycles per dungeon is arbitrarily capped at five, fewer cycles may be produced if the graph generation process terminates prematurely, as is the case of the layout depicted in Figure 6.7. This requires a corresponding set of five unique locations and associated assets to be generated, as discussed in Section 6.2.2

Despite the relative simplicity of generating a cyclic dungeon graph, the task of embedding this structure onto a 2D grid is considerably more challenging. In CrawlLLM, nodes are instantiated as rectangular rooms, and edges are converted to either straight or right-angled corridors between them. The placement process proceeds iteratively, beginning from the innermost cycle. One node is selected and placed at the origin of the grid, after which its connected nodes are positioned in adjacent grid locations. This recursive process is then applied to all remaining nodes and cycles. Since CrawlLLM

enforces a purely 2D layout without any overlapping rooms or corridors, certain edge placements may occasionally be infeasible due to spatial constraints. However, because the graph-routing algorithm operates rapidly, such failures can be mitigated by restarting the placement process until a viable configuration is obtained. The final dungeon, as shown in Figure 6.7, is rendered with appropriate tiles and visual elements, faithfully representing the underlying graph structure in a coherent and navigable spatial format.

6.2.4 | Computational Considerations

It is noteworthy to consider the computational demands of the CrawlLLM generative pipeline. As has been previously discussed, local models were employed for both textual and visual generation tasks, thereby necessitating hardware infrastructure capable of executing such models with adequate performance. Of particular note, LLMs are generally characterised by substantially larger architectures, frequently consisting of several billion parameters in excess of those found in diffusion-based text-to-image generative models. The pipeline architecture required that multiple instances of these models be instantiated and utilised for the purpose of generating the diverse assets comprising the game environment. While Graphics Processing Units (GPUs) or Tensor Processing Units (TPUs) demonstrate significantly superior efficiency compared to Central Processing Units (CPUs) when executing such models, a critical limitation is that the former two devices possess must sufficient Video Random Access Memory (VRAM) capacity to accommodate the models in their complete form. The hardware configuration employed for executing the CrawlLLM pipeline incorporated a GPU equipped with 48 Gigabytes of VRAM, which were adequate to support the experiments conducted. Nevertheless, it was necessary for the LLM and SDXL model instances to be alternately loaded into and unloaded from memory when transitioning between text and image generation operations.

With respect to computational time requirements, it was observed that an individual instance of text generation consumed, on average, an order of magnitude greater duration than that required for the generation of a single image, though this relationship was dependent upon the quantity of text and the pixel dimensions of the images being produced. Notwithstanding this disparity, given that multiple textual assets were generated through identical text prompts while images were produced on an individual basis, and further considering that a greater quantity of image assets was required relative to textual assets, it was found that the generation of images consumed approximately twice to four times the duration required for text generation when considering the entire pipeline.

6.2.5 | Generated Games Collection

To evaluate the pipeline, it was used to produce a collection of 20 distinct themes, each supplemented with a complete set of game assets. The outputs were validated through playability testing of the generated games, where it was confirmed that all assets conformed to the formats required by the game engine.

6.3 | Structure of the Experiment

This section outlines the study undertaken to assess the coherence of the generated assets within the game context, and to validate their relevance to the parent theme. A user survey was thus conducted to verify these aspects in relation to both narrative and visual elements of the games.

6.3.1 | Hypotheses

The objective of the user study was to confirm that the core thematic descriptors that guided the generation of both text and visuals are evident in the final result. It was also desirable to ascertain that the final results are appealing to human players. More formally, these can be written as the following hypotheses:

H1 Participants in the study are able to identify the core thematic details intended in the generated text and visuals at a rate significantly above chance.

H2 Human participants will rate the visuals and text as appealing.

6.3.2 | Experimental Protocol

The collection of 20 generated games from Section 6.2.5 was played by the author, during which illustrative screenshots were captured from key gameplay phases, including dungeon exploration and card-based combat. One stimulus was generated per game for user presentation. The first 5 games were utilised to create dungeon-crawling screenshots (denoted D1-D5), while the subsequent 5 games served as sources for card-combat screenshots (C1-C5). From the following 5 games, textual descriptions of a single enemy character were extracted (E1-E5). The remaining 5 games were not directly presented to users, however, thematic elements from these games were incorporated into the survey's multiple choice questions in order to increase the potential for user confusion, as outlined in a later section.

The selection of the 15 games from which the stimuli were extracted, was carried out without any subjective filtering or manual curation. The decision was purely based on sequence, and driven purely by considerations of participant burden. Including assets from all 20 games would have resulted in an overly prolonged and potentially fatiguing survey experience.

For each stimulus, pertinent thematic attributes of the corresponding game were collected. Specifically, dungeon-crawling screenshots were annotated with the game's setting, environment, historical epoch, and visual style. The card-combat screenshots included setting, epoch, and visual style, but not the environment, since it was not explicitly visible in this context. The textual enemy descriptions were associated with setting and epoch alone. For every thematic attribute, the three most semantically similar counterparts among the remaining 19 games were identified. This similarity was quantified through the FLAVA model [191], which computes distances in a shared latent space by embedding both textual and visual descriptions. This model enables the direct comparison of thematic vectors irrespective of modality. Notably, this embedding-based distance metric is orthogonal to the similarity function employed in CLIP [166], which underlies image generation used by the pipeline.

As a result, each stimulus was paired with four candidate options per thematic attribute (e.g., setting, epoch), one of which corresponded to the correct attribute of the game used to generate the assets contained in the stimulus. The remaining three options were distractors, selected based on their proximity in the semantic latent space.

The survey was deployed online via Google Forms. Participation links were disseminated through the author's social media platforms. An informed consent section was presented at the beginning of the form, followed by questions assessing participants' gaming experience, but demographic information was not collected. Each stimulus was displayed on a dedicated page within the form. After viewing the stimulus, participants were asked to select the most appropriate setting, epoch, environment, or visual style (depending on the stimulus type) via a multiple choice format consisting of the four predetermined options. The order of these options was randomised in advance and remained consistent across all participants. Additionally, a concluding question for each stimulus solicited a user rating on a five-point Likert scale, phrased as either "The Screenshot is visually appealing" or "The Description is entertaining to read."

The experimental protocol was registered with and processed by the University of Malta's Research Ethics Committee.

It is important to emphasise that participants did not interact with the generated games themselves. Rather, they were exposed exclusively to static screenshots or textual excerpts. This design decision was based on the assumption that coherence between

Table 6.2: A summary of the user study results. Answer accuracy is shown alongside response entropy per question. An asterisk (*) denotes statistically better outcomes (higher rating, higher accuracy, lower entropy) compared to random selection.

#	Accuracy (Entropy)				Mean Rating
	Setting	Environment	Epoch	Style	
D1	79%* (0.37*)	68%* (0.56*)	44%* (0.76*)	50%* (0.89*)	2.5
D2	82%* (0.34*)	94%* (0.16*)	15% (0.3*)	47%* (0.89*)	3.4*
D3	32% (0.79*)	15% (0.61*)	0% (0.67*)	47%* (0.5*)	2.4
D4	53%* (0.8*)	74%* (0.52*)	62%* (0.73*)	27% (0.99)	3.2
D5	41%* (0.94)	29% (0.53*)	9% (0.73*)	68%* (0.61*)	3.2
C1	65%* (0.65*)	—	18% (0.81*)	71%* (0.58*)	4.0*
C2	15% (0.51*)	—	15% (0.66*)	35% (0.94)	3.9*
C3	88%* (0.32*)	—	0% (0.76*)	9% (0.82*)	3.6*
C4	97%* (0.10*)	—	94%* (0.19*)	100%* (0*)	3.6*
C5	94%* (0.19*)	—	100%* (0*)	15% (0.78*)	3.8*
E1	77%* (0.56*)	—	0% (0.37*)	—	3.1
E2	79%* (0.52*)	—	91%* (0.26*)	—	3.4*
E3	0% (0.59*)	—	77%* (0.46*)	—	3.4*
E4	79%* (0.46*)	—	68%* (0.67*)	—	3.4*
E5	82%* (0.39*)	—	85%* (0.37*)	—	3.5*

the theme and content (visual or textual) can be effectively assessed through static representations. This approach also reduces cognitive load relative to gameplay-based evaluations, and provides a more controlled experimental context, as the author retains complete knowledge of the stimuli observed by each participant.

6.4 | Results and Evaluation

A total of 34 responses were obtained through a process of convenience sampling. The majority of participants self-identified as gamers: only 6% reported never engaging in video game play, whereas 47% indicated that they play regularly. A preference for dungeon crawlers was expressed by 61% of respondents, while 21% stated that they had never played such games. Similarly, 62% reported enjoyment of card-based video games, with 12% noting they had never experienced this genre.

The collected responses were analysed in relation to the predefined hypotheses, with quantitative metrics computed independently for each stimulus. In evaluating the dimension of appeal, the *mean rating* was calculated based on participants' responses to a 5-point Likert item pertaining to appeal (as described in Section 6.3.2). Coherence was

assessed through two principal measures: *accuracy*, defined as the proportion of participant responses that corresponded correctly to the thematic details of each stimulus, and Shannon’s *entropy* [189], normalised by the number of available options. A low entropy value was interpreted as an indication of consensus among participants, irrespective of correctness.

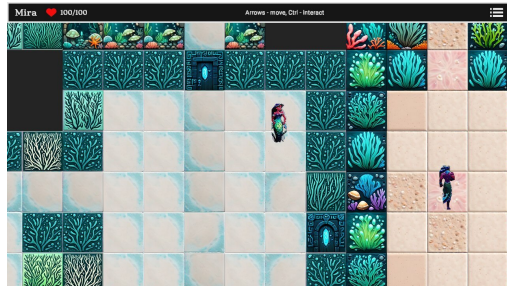
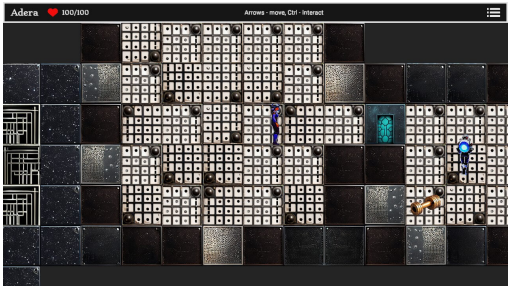
Statistical significance was evaluated at the $p < 0.05$ level under the assumption of random selection. For Likert-based ratings, significance was determined by comparing the 95% confidence interval of the observed mean rating to the neutral baseline score of 3. Accuracy results were tested using a binomial test predicated on a 1-in-4 probability of correct choice [52], whereas entropy values were assessed using a χ^2 goodness-of-fit test [53], comparing the deviation of participants’ response distributions to a uniform distribution. All the tests were subjected to a False Discovery Rate correction using the Benjamini-Hochberg process [8].

Table 6.2 summarises the resulting metrics for each stimulus, presented in the order shown to participants. A majority of stimuli were rated as appealing, with statistically significant results above the neutral baseline recorded in 10 out of 15 cases. Card combat screenshots were consistently rated as highly appealing in comparison to other stimuli. Ratings for dungeon crawler screenshots exhibited greater variability, with 2 out of 5 instances (specifically, D1 and D3) receiving significantly lower ratings than the baseline.

With regard to coherence, a high degree of variation in accuracy was observed across different stimuli. Although 30 out of 45 questions yielded accuracy levels significantly above the random baseline of 25%, performance was highly contingent upon the nature of the stimulus. Questions related to setting achieved the highest average accuracy (64%), despite 3 out of 15 such questions failing to achieve statistical significance, including one instance (E3) in which all 34 participants selected the incorrect option. Environment questions, whilst being the fewest in number, counted a mean accuracy of 56%, but these results were ultimately significant only for 3 of the 5 questions in this category. Epoch-related questions demonstrated the lowest average accuracy (43%), with 6 out of 15 cases not significantly above chance, followed closely by visual style questions, which averaged 45% accuracy and included 4 out of 10 cases lacking statistical significance.

Response dispersion was also evident from the entropy analysis. Out of the 45 questions, 28 exhibited a normalised entropy score exceeding 0.50, and 7 surpassed 0.80. This dispersion occurred even in scenarios where accuracy was low (e.g., the visual style of C3, with only 9% accuracy and an entropy score of 0.82), and also in cases where accuracy was relatively high (e.g., the visual style of D1, with 50% accuracy and an entropy score of 0.89). While entropy varied by stimulus, questions related to visual style consistently produced higher entropy values, with an average of 0.70, compared

Table 6.3: Example dungeon crawler screenshots descriptions and related questions from the user study. The correct options which were used to generate the game are shown as checked.

D2	D3
	
Which setting best describes the above Screenshot?	
☐ A subterranean realm of mushroom forests and crystal caves ☐ A sunken pirate city in a stormy sea ☒ A mystical underwater city ☐ A lost colony on an alien world	☐ A lost colony on an alien world ☒ A drifting asteroid mining facility ☐ A subterranean realm of mushroom forests and crystal caves ☐ An abandoned space station overrun by mutants
What best describes the environment in the above Screenshot?	
☐ Flooded streets, shipwrecks, treasure troves ☐ Trap-filled corridors, massive machinery, hidden chambers ☒ Coral reefs, sunken temples, deep sea trenches ☐ Trap-filled corridors, mummies and scarabs, hidden treasures	☐ Trap-filled corridors, massive machinery, hidden chambers ☐ Zero gravity corridors, malfunctioning machinery, airlocks and vents ☒ Gravity-less corridors, cryogenic chambers, dark caves ☐ Trap-filled corridors, mummies and scarabs, hidden treasures
Which epoch is depicted in the above Screenshot?	
☐ Fantasy ☒ Ancient ☐ Medieval ☐ Renaissance	☐ Lost civilization ☐ Space age ☐ Science-fiction ☒ Near future
What best describes the visual style in the above Screenshot?	
☐ Semi-abstract, vivid alien landscapes ☐ Digital matte painting, cold color palette ☒ Stylized realism, bioluminescent color palette ☐ Mystical realism, moody	☐ Science-fiction, gritty ☐ Retro-futuristic, vibrant and nostalgic ☒ Science-fiction, minimalist ☐ Retro-futuristic, neon-lit

to other question types whose entropy scores tended to cluster around 0.50. These findings suggest that visual style identification may have been particularly challenging for participants.

In order to better interpret the distinctions observed in user response patterns across different question types, representative examples are presented herein. Dungeon crawling screenshots (D2, D3) are provided in Table 6.3, while card combat screenshots (C3, C4) are shown in Table 6.4.

As can be seen in Table 6.2, screenshot D2 yielded a high level of accuracy with regard to both setting and environment, whereas D3 exhibited low accuracy scores for both attributes. In the case of D2, all six participants who did not correctly identify the setting (“A mystical underwater city”) instead selected “A subterranean realm of mushroom

Table 6.4: Example card-based combat screenshots and related questions from the user study. The correct options which were used to generate the game are shown as checked.

C3	C4
	
Which setting best describes the above Screenshot?	
☐ An enchanted forest on a floating island ☒ A lost colony on an alien world ☐ A subterranean realm of mushroom forests and crystal caves ☐ A mystical underwater city	☐ An ancient Roman city swallowed by a volcano ☒ An ancient labyrinth beneath a desert pyramid ☐ A forgotten underground library ☐ A lost colony on an alien world
Which epoch is depicted in the above Screenshot?	
☐ Near future ☐ Science-fiction ☒ Space age ☐ Lost civilization	☒ Ancient Egypt ☐ Medieval ☐ Fantasy ☐ Science-fiction
What best describes the visual style in the above Screenshot?	
☒ Semi-abstract, vivid alien landscapes ☐ Science-fiction, minimalist ☐ Retro-futuristic, vibrant and nostalgic ☐ Dreamlike, surreal	☐ Grimy steampunk, surrealistic ☐ Semi-abstract, vivid alien landscapes ☒ Egyptian relief, earthy tones ☐ Science-fiction, gritty

forests and crystal caves”. This outcome is notable, particularly given that only two participants failed to correctly identify the environment (“Coral reefs, sunken temples, deep sea trenches”). It may therefore be inferred that while most users successfully recognised the marine environment, possibly due to the depiction of aquatic fauna on the dungeon walls, several of those users still interpreted the overarching setting as subterranean rather than marine (e.g. “A sunken pirate city in a stormy sea” and “A mystical underwater city”).

Regarding epoch identification, both D2 and D3 demonstrated very low accuracy, with no participant correctly identifying the epoch for D3. It must be acknowledged that the multiple-choice options provided for the epoch-related questions were, in these cases, suboptimal. For D2, a significant majority (85%) selected “fantasy”, while for D3, the responses were divided predominantly between “science fiction” (50%) and “space age” (41%). It is plausible that participants gravitated toward familiar genre labels often encountered in literature and media. Depictions of abstract temporal concepts are inherently challenging to disentangle from similar terms through visual cues alone. Visually, “Ancient”, the correct option for D2, may easily be misconstrued as “Fantasy”

or “Medieval”. “Near future”, the correct epoch for D3, is difficult to extricate from “Science-fiction” and “Space age”. A more fundamental concern, however, lies in the apparent dissonance between setting and epoch: mystical underwater cities are not commonly associated with ancient times (except through tenuous symbolic associations), and asteroid mining facilities are generally imagined within the context of a distant, technologically advanced future. The issue at hand does not concern feasibility but rather the alignment of conventional tropes. Indeed, mystical underwater cities are a staple of fantasy narratives, while asteroid mining facilities are frequently depicted in science fiction cinema.

For the style attribute, both D2 and D3 achieved identical accuracy levels (47%), which, although not particularly high, surpasses random chance by a meaningful margin. However, substantial differences in entropy scores were observed between the two stimuli. In D2, incorrect style selections were distributed among several alternative options, with the most commonly chosen being “Semi-abstract, vivid alien landscapes” (26%). By contrast, in D3, all incorrect responses were concentrated on a single alternative: “Science-fiction, gritty” (53%). This again reflects a recurrent inclination toward labels featuring the term “science fiction”, as previously noted in the context of the epoch responses. This observation brings to light a broader issue with the experimental design: by selecting the semantically nearest alternatives from a pool of 19 additional games, many of the presented options were rendered nearly indistinguishable from one another. Although efforts were made to eliminate duplicate labels from the questionnaire, numerous options remained only marginally different. An example of this ambiguity is evident in the epoch labels presented for D5, where both “Gothic Victorian” and “Victorian era” were offered. There is substantial evidence, drawn both from the quantitative data of Table 6.2 and the qualitative examples in Tables 6.3 and 6.4, suggesting that the attributes of epoch and style were particularly difficult to render comprehensibly through in-game screenshots. These limitations of the experimental protocol are further discussed in Section 6.5.

To conclude the comparative analysis between D2 and D3, it is noteworthy that D2 received the highest average rating (3.4) among all dungeon crawler screenshots, whereas D3 received the lowest (2.4). It is conjectured that the presence of a visually noisy background in D3, particularly in the floor tile textures, negatively impacted participants’ evaluations. In contrast, D2 employed flat colours for the floor tiles alongside strong contrasting hues for the walls, which may have contributed to a more aesthetically pleasing impression. The colourful, organic shapes observed in the wall structures may also have been perceived favourably in contrast to the dark, blank tiles in D3. It should be observed that while the visual style of D2, defined as “bioluminescent colour palette”, was reasonably preserved, the noisy texture of the floor tiles in D3 undermines the

“Science-fiction, minimalist” aesthetic intended by its style descriptor. This discrepancy is revisited in Section 6.5.

Turning now to the card combat scenarios, both C3 and C4 provided settings that were consistently identified correctly by participants. However, in the case of C3, neither the epoch nor the style was accurately recognised. The epoch choices for C3 mirrored those of D3, albeit with a different order of presentation. No participants selected the correct epoch of “space age”; instead, responses were nearly evenly divided between “near future” (41%) and “science fiction” (38%). This finding is intriguing, especially in light of the responses to D3, where the correct “near future” epoch was never chosen, yet “space age”, which was correct for C3, was selected by 41% of users. Assuming that ordering effects are negligible, this divergence in behaviour remains difficult to rationalise. It may be hypothesised that the warehouse interior depicted in C3’s background failed to evoke associations with the “space age” epoch, likely due to the absence of visual cues such as celestial bodies or spacecraft, which, although not strictly necessary, are often used in popular culture as visual shorthand for that period.

With respect to visual style, the options provided may help explain the low correctness rate (9%) for C3. The intended style, “Semi-abstract, vivid alien landscapes”, was poorly reflected in the background imagery, and as a result, the majority of users instead selected “Science-fiction, minimalist” (50%). As with previous cases, this underscores the difficulty of communicating visual style effectively, particularly when the semantic descriptors lack specificity or when the image-generation system (in this case SDXL) fails to render distinguishing features clearly. A notable counterexample is found in C4, where the style label “Egyptian relief, earthy tones” is both semantically precise and visually unmistakable. The clarity and specificity of the visual and textual elements resulted in a correct consensus amongst participants. This was the only instance across all stimuli where perfect accuracy was observed for visual style identification.

The final observations are related to the textual descriptions of the enemies. These assets not displayed directly to players in the CrawLLM game, but are rather used in the prompts to generate the character images and to generate the semantic descriptions of the minions. These descriptions generally received above-average evaluations and high accuracy scores in thematic identification tasks, as can be surmised from Table 6.2. However, two results stand out for E1 and E3, whose full text and multiple choice options are presented in Table 6.5.

Specifically, none of the participants correctly identified the setting of E3, and the epoch of E1. For the latter, “Fantasy” was the most selected option, with the correct one being “Medieval”. This is consistent with the conclusions drawn about epoch from the screenshots part of the survey. The case of the setting of E3 is interesting in that

Table 6.5: Example enemy descriptions and related questions from the user study. The correct options which were used to generate the game are shown as checked.

E1	E3
Astral Harpy The Astral Harpy is a mesmerizing creature, her form a harmonious blend of ethereal energy and humanoid structure. She possesses the upper body of a woman with skin that shimmers like twilight stars, while her lower half transforms into powerful, feathered wings of deep cosmic blue. Her eyes burn with a fierce, otherworldly light, and she is adorned in ceremonial robes intricately woven from threads of pure starlight. The Astral Harpy wields a divine harp that, when strummed, can summon forth bolts of radiant energy to strike down any who dare to tread within the hallowed chambers of The Celestial Choir Chapel.	Vortexian Sentinel A towering figure of swirling cosmic energy, the Vortexian Sentinel stands as a guardian of the Nebula Nexus. Its body is composed of intertwining streams of vibrant blues, purples, and silvers, constantly in motion. The entity has no distinguishable features beyond its energy form, but it does possess four arms that extend from its core, each ending in five long, slender fingers. These hands are capable of manipulating the energy around them to create weapons or barriers as needed. The Sentinel wears no clothing, but a faint outline of a circular badge can be seen near its center, glowing with a white-hot intensity - possibly indicative of its allegiance or purpose.
Which setting best describes the above Description?	
☒ A forgotten cathedral in the clouds ☐ An ancient labyrinth beneath a desert pyramid ☐ A lost colony on an alien world ☐ A mystical underwater city	☐ A drifting asteroid mining facility ☐ A lost colony on an alien world ☒ A haunted forest on a foggy moon ☐ A forgotten cathedral in the clouds
Which epoch is depicted in the above Description?	
☐ Ancient ☐ Renaissance ☐ Fantasy ☒ Medieval	☐ Fantasy ☐ Near future ☒ Science-fiction ☐ Ancient

despite the verbose description, there were insufficient indications for the participants to distinguish between the correct choice (“A haunted forest on a foggy moon”), and the most commonly selected one (“A lost colony on an alien world”). There is a degree of similarity between the two settings, and the description of the enemy in this case is markedly alien, which could equally fit both of these. Despite the similarity, the direct “alien” term in the decoy setting deceived the entire population of participants.

6.5 | Discussion

From the survey results it is observed that thematic coherence is not consistent across both modalities and across all theme descriptors. Whilst it can be seen that the narrative assets are for the majority recognisable, certain thematic elements, specifically those concerning visual style and epoch, proved difficult to communicate effectively through in-game visuals alone. In conjunction with the qualitative analysis of both the visual stimuli and user responses presented in Section 6.4, these results partially support hypothesis H1, i.e. that the core thematic details seeding the assets are recognisable. This is true for the setting and environment descriptors, successfully conveyed the thematic information through the given stimuli. In contrast, epoch was only identified correctly from the

narrative stimuli most of the time, and style recognition varied greatly between dungeon and card interfaces. Notably, it was found that thematic clarity was typically achieved only when the underlying details were highly specific, such as references to particular places rather than abstract genre conventions.

Furthermore, it was observed that the textual descriptions and the screenshots of the card combat interface were generally well perceived by the participants. However the dungeon crawling images were not received as favourably. Despite this, in the majority of cases the preference of the participants was superior to a random selection, which reinforces hypothesis H2.

The study also brought to light several limitations of CrawLLM that had previously been identified during internal playtesting sessions. One such limitation concerned the generation of a coherent and stylistically consistent tileset, a process that necessitated numerous iterations and refinements. Although the three-stage methodology illustrated in Fig. 6.5 addresses various shortcomings of earlier attempts, it continues to yield noisy background tiles in certain instances, as exemplified by game C3 in Table 6.4. Prospective improvements could involve the application of quantitative image metrics, such as visual complexity [135] and colourfulness [85], to systematically select the least noisy tile from the generated alternatives shown in Fig. 6.5, rather than relying on random selection.

Additional shortcomings were attributed to the experimental design itself. Specifically, the inclusion of near-duplicate response options appeared to have introduced confusion among participants, particularly in the evaluation of stylistic attributes as seen in Tables 6.4 and 6.3. A measure of misdirection was desirable so as not to overly facilitate the correct choice, and thus making the survey futile. Nonetheless, the high similarity between options proved to be far too misleading, and the choices offered could have been filtered to at least remove options which contain identical words, instead of omitting only full duplication. Future iterations of user studies would benefit from eliminating such near-duplicates and presenting only distinct alternatives that exceed a predetermined minimum distance threshold in the FLAVA latent space.

A more consequential limitation pertains to the interaction between thematic elements and the processes governing text and image generation. Empirical evidence suggests that, in numerous cases, the specification of visual style had little to no discernible influence on the resulting visual assets, whether tilesets or card combat backdrops. This phenomenon is hypothesised to arise from constraints inherent in the prompting methodology: visual style descriptors are often positioned at the end of prompts directed at the SDXL model, and may thereby fall outside the effective context window of the CLIP architecture [166], or be semantically diluted by preceding elements of the prompt, such as scene descriptions. Alternatively, limitations may stem from the LLM responsible for

generating the visual style descriptions themselves. While some styles (e.g., “Egyptian relief”) lend themselves more readily to downstream generation processes, others (e.g., “semi-abstract”) may be less practical. The LLM, lacking awareness of its output’s intended role in guiding subsequent image generation, is unable to self-correct or anticipate such issues.

In view of the research questions of this thesis, this experiment EXP3 primarily addresses RQ3. Specifically, it describes and successfully implements a fully automated pipeline orchestrated by a LLM, the output of which human evaluators have found to have good quality, deeming the assets to be overall pleasant. Furthermore the experiment shows that the assets have good coherence to the thematic intent, which is also queried in RQ3.

In future work, introducing constrained human oversight at high-impact stages, such as theme definition or tileset selection, could substantially improve interpretability while preserving the pipeline’s zero-shot character. Alternatively, models explicitly optimised for stylistic control may better support the visual grounding of abstract descriptors.

Generalisation of the pipeline is intentionally limited. *CrawlLLM*, as implemented, will not necessarily generalise directly to all game genres or creative pipelines. The findings only generalise at the conceptual level of orchestration, i.e. that LLMs can act as effective high-level controllers for decomposed generative processes without task-specific training.

The reliance on a specific game genre, fixed templates, and a particular set of generative models constrains broader claims. The user study further reflects recognition-based coherence within the *CrawlLLM* design space rather than absolute measures of creative quality or player enjoyment.

Therefore, the conclusions generalise best to structured creative pipelines where content can be decomposed into semantically linked stages, and where natural language can function as a stable intermediate representation. They generalise less strongly to open-ended creative systems lacking clear hierarchical structure.

6.6 | Limitations and Shortcomings

The *CrawlLLM* pipeline demonstrates that LLMs can coherently orchestrate multimodal asset generation in a zero-shot setting; however, it does not incorporate explicit optimisation or selection mechanisms. As discussed earlier in the chapter, asset quality and consistency are largely a consequence of prompt design and model capabilities, rather than the result of systematic improvement or search. This limits the pipeline’s ability to

recover from weak generations or to explore alternative design possibilities.

Although this chapter introduces human grounding through a user study, the scope of that grounding is deliberately narrow. The study focuses on recognisability and thematic coherence, rather than on broader notions of creative quality, engagement, or player preference. As such, positive results indicate that the generated assets are internally consistent, but do not imply that they represent optimal or particularly compelling designs.

The results also point at uneven recognisability of the thematic descriptors. This highlights a limitation in the pipeline's ability to ground abstract or stylistic descriptors visually. While narrative text retained thematic clarity, corresponding visual assets frequently failed to encode these attributes in a manner that was interpretable to users. This suggests a disconnect between internal prompt alignment and external human perception.

Diversity is not explicitly measured or controlled within the pipeline. Variation emerges implicitly from stochastic generation, but there is no mechanism to detect redundancy or ensure coverage across asset types. This reflects a design choice prioritising coherence and structural completeness over exploratory breadth, but it also limits the system's ability to reason about diversity as a first-class objective.

These limitations motivate the iterative approach introduced in Chapter 7, which augments the CrawlLLM pipeline with explicit evaluation and regeneration steps in order to address underperforming assets and introduce adaptive control over quality and diversity.

6.7 | Summary

This chapter has detailed the design of CrawlLLM, a zero-shot, fully automated, GenAI pipeline for coherent game asset generation. An online user study engaged a total of 34 participants who were tasked with evaluating game assets through screenshots and textual descriptions, and inferring the thematic details that had informed their generation. The results showed overall good coherence with the thematic elements, but poor for specific ones. However the output was deemed mostly pleasant by the majority of users. In the next chapter, the CrawlLLM pipeline will be extended to include an iterative workflow in order to automatically assess and refine the generated assets.

An Iterative Generation Pipeline

The CrawLLM multimodal generative pipeline described in Chapter 6 was developed as a zero-shot system in which all assets are generated in a forward pass, without subsequent assessment or refinement. This implementation lacks mechanisms for evaluating and improving the coherence of the asset set or ensuring sufficient variation across individual assets. This uniformity can limit the expressive capacity and the creative scope of the system. It is however crucial that the generated assets not only differ from one another but also function cohesively within a shared narrative and style. The equilibrium between coherence and diversity is therefore critical to the perceived value of the generated content.

This chapter thus advances the CrawLLM pipeline by introducing a set of simple iterative asset regeneration processes. These are supported by automated asset evaluation methodologies aimed at improving both the fitness and diversity of the generated content.

7.1 | Methodology

Consider a generic generative multimodal pipeline which generates or transforms assets sequentially, in a similar fashion to CrawLLM. The pipeline structure may be considered a hierarchical directed graph, as it is generated in consecutive stages, and it is also acyclic since each stage does not affect earlier stages. Such a hierarchical Directed Acyclic Graph (DAG) is therefore similar to a tree structure, with the exception that each node may have more than one parent. The output set of assets from such a pipeline may also be considered as forming a similar DAG, preserving the dependencies between assets as they occurred during the generative process.

Within creative domains, coherence between the assets in a generated set along the DAG directions is generally desirable, as this denotes parent-child relationships. It is equally advantageous for assets of the same type to be diverse enough from each other. Appropriate fitness (or coherence) and diversity metrics to assess each node within the

DAG of the asset set require definition to match the attributes of the system. Once defined, these metrics enable the pipeline to systematically evaluate its output and identify the least effective individual assets or asset groups. This information may then be leveraged for targeted regeneration of assets, in a conceptually similar fashion to the operation of EAs.

A population-based GA may be employed by generating and evolving multiple sets of assets. However, in systems of similar or greater complexity than CrawLLM, each asset set comprises a considerable number of elements as established in Chapter 6. Consequently, the computational demands associated with maintaining, regenerating, and evaluating a large population of such sets may render this approach impractical. Alternative strategies which iterate over single solutions, such as the generate-and-test paradigm [200] or a basic $(1 + 1)$ EA [57] (refer to Section 2.1.2) are more adequate in such a circumstance. In either case, after defining the target metric for optimisation, a series of iterations are carried out through which the least performing asset, or subset of assets, within the set are selected for regeneration and potential replacement.

A distinction must here be drawn between assets representing *inner* nodes which are parents to other nodes, and *leaf* nodes (which have no children). Challenges arise during evaluation, when inner nodes are selected for replacement, as all other downstream assets depending on them will in turn also require regeneration. If the pipeline supports partial regeneration from arbitrary points, this issue is mitigated. Conversely, if the pipeline's complexity renders such regeneration infeasible, it may be prudent to restrict candidate selection for replacement to the leaf nodes. These considerations are inherently context-dependent, influenced by the structural and computational characteristics of the specific pipeline in question.

The iterative search process itself may be formulated either as a SOO problem, in which a single metric (e.g., fitness or diversity maximisation) is targeted, or as a MOO problem, where individuals dominated across multiple metrics are selected for replacement. The subsequent sections offer practical illustrations of how these algorithms may be implemented.

7.2 | Use Case: The CrawLLM Iterative Pipeline

In the context of EXP4, the CrawLLM pipeline has been extended to incorporate iterative refinements, with a particular focus on enhancing both the internal coherence of generated assets and the diversity among asset groups of the same type within a given set.

A comparative evaluation is conducted between two iteration strategies: the generate-and-test approach and a simplified $(1 + 1)$ EA, hereafter referred to as *Regen* and *Evo*, respectively. Within each strategy, assets demonstrating the weakest performance are identified at every iteration, and for each such asset, a newly generated alternative is produced. The key distinction between the two methods lies in their replacement criteria: the Regen strategy indiscriminately substitutes the older asset with the newly generated one, whereas the Evo strategy enforces a performance-based criterion, allowing replacement only if the new asset surpasses its predecessor in the target metrics.

7.2.1 | Replacement Candidates

In view of the complex structure of the CrawLLM pipeline, EXP4 restricts replacement eligibility to *leaf node* assets, totalling 76 candidates (see Table 7.1). As detailed in Section 7.1, these leaf nodes correspond to player-facing assets which possess no downstream dependencies within the CrawLLM DAG. In other words, they represent final outputs that are not reused as inputs for the generation of further assets. This selective approach offers a high efficiency, as it forgoes the need for cascading regenerations.

7.2.2 | Definition of Metrics

The principal enhancement introduced to the CrawLLM pipeline involves the computation of metrics for both the fitness and diversity of the generated asset set. Given that the assets span across both visual and textual modalities, the use of multimodal models is proposed to carry out this evaluation.

It should be noted, however, that CLIP is already integral to the prompt alignment mechanism employed within SDXL, the image generation model utilised by CrawLLM [162]. To provide an orthogonal evaluation while preserving functionality, a distinct model with analogous capabilities is employed. For this purpose, FLAVA [191] was utilised in Experiment EXP4 detailed in this chapter.

A corresponding *FLAVA score* is computed for any two assets in a manner similar to the CLIP score, as defined in Equation 2.4. The distinction lies solely in the substitution of FLAVA-derived embeddings in place of those produced by CLIP. However, it is important to note that whilst in previous experiments the CLIP score was being computed between the input prompt for a model and its generated output, in EXP4 the FLAVA score is used to capture the conceptual proximity between any two assets under comparison.

Within this use case, asset coherence is interpreted as the fitness of a given asset with respect to its immediate parents within the hierarchical pipeline. This is based on

Table 7.1: The leaf node assets grouping by type, used for diversity comparisons. 76 assets (leaf nodes) are considered in total.

Group	Asset Count	Type
1	3	Game name, Introduction, Conclusion
2	16	Character Spritesheets
3	3	Card Frames
4	5	Location Tilesets
5	5	Location Backgrounds
6	15	Character Portraits
7	20	Key Images
8	9	Action Card Images

the assumption that, since all assets are hierarchically derived following the pipeline’s DAG, the coherence of the complete asset set can be inferred if each individual asset is evaluated and found to be coherent in relation to its direct parent. In other words, global coherence is obtained through the transitive evaluation of local parent-child relationships across the hierarchy.

An asset’s fitness is defined as the mean similarity between its FLAVA embeddings and those of its parents. The similarity is computed as per Equation 2.4 within the FLAVA latent space. Thus the equation for the fitness of each asset is given by:

$$fitness(i) = \frac{1}{N} \sum_{p=1}^N similarity(\mathbf{e}_i, \mathbf{e}_p)$$

where $\mathbf{e}_i$ is the FLAVA embedding of the asset being evaluated,

$\mathbf{e}_p$ is the FLAVA embedding of parent p ,

N is the total number of direct parents of asset i .

Assessing diversity within the CrawlLLM pipeline is challenging due to the considerable range of assets it produces. Since there is a distinct disparity in the *format* of the generated assets, a direct comparison between, for example, a character spritesheet and a location tileset inherently yield low FLAVA similarity scores. Such an observation, however, does not yield any useful insight other than that pertaining to structure. It is far more informative to recognise when there are, say, two (or more) robot characters, whether or not they have been depicted in a visually similar manner, and whether there is appreciable overlap between their names and descriptions.

Therefore the assets were grouped by type (e.g. location, character, etc.) and format (e.g. background image, text description, spritesheet, etc.), as shown in Table 7.1. The

diversity of an asset was defined to be its mean distance to its peers within that group, as shown in Equation 7.2.2.

$$diversity(i) = \frac{1}{n-1} \sum_{\substack{j=1 \\ j \neq i}}^n (1 - similarity(\mathbf{e}_i, \mathbf{e}_j))$$

where n is the total number of assets in the group,

$\mathbf{e}_i, \mathbf{e}_j$ are the FLAVA embeddings of assets i and j ,

$\cos(\mathbf{e}_i, \mathbf{e}_j) = \mathbf{e}_i^\top \mathbf{e}_j$ is the cosine similarity between the embeddings.

During each iteration, the fitness and diversity for each asset is recorded. Every replaced asset was also noted, in order to calculate the total count of unique assets replaced during every run. For every experiment conducted, the final values of fitness and diversity associated with the asset sets are averaged, and the resulting means are used to compute percentage changes relative to their respective initial values.

7.2.3 | Leaf Node Selection

When selecting replacement candidates from leaf node assets, multiple individuals may be chosen during the same iteration, as there are no downstream repercussions. Therefore, the asset with the lowest objective metric from each group shown in Table 7.1 was chosen for replacement. Therefore, in the experiments which optimise fitness, the 8 assets with the lowest fitness score as calculated from Equation 7.2.2 were selected at every iteration. Analogously, those runs optimising diversity select the 8 assets exhibiting lowest diversity, computed as shown in Equation 7.2.2.

When applying MOO to target both fitness and diversity, a *domination* count is established for each group. This domination metric is obtained for each asset by counting the number of other assets within the group which have higher values of both fitness and diversity. The asset with the maximum domination count is considered the worst performing one and will be selected for regeneration. In case there is a tie between two or more assets, the sum of each individual's domination counts in fitness and diversity computed separately is used as a secondary score to arbitrate the selection. In mathematical notation, if $G = \{x_1, x_2, \dots, x_n\}$ is a group of assets, where each $x_i \in G$ has an associated fitness $f(x_i)$ and diversity $d(x_i)$, then define the following for each $x_i \in G$:

$$P(x_i) = |\{x_j \in G \mid j \neq i, f(x_j) \geq f(x_i), d(x_j) \geq d(x_i), (f(x_j) > f(x_i) \vee d(x_j) > d(x_i))\}|$$

$$T(x_i) = |\{x_j \in G \mid j \neq i, f(x_j) > f(x_i)\}| + |\{x_j \in G \mid j \neq i, d(x_j) > d(x_i)\}|.$$

Then the selected (worst performing) individual is:

$$x^* = \arg \max_{x_i \in G}^{\text{lex}} (P(x_i), T(x_i))$$

where $\arg \max^{\text{lex}}$ selects the individual with the highest domination count $P(x_i)$, and breaks ties using the total domination score $T(x_i)$.

7.3 | Structure of the Experiment

EXP4 has been structured to explore multiple methodologies. Firstly, it comparatively explores the performance of Regen and Evo iterative algorithms, and secondly it observes the properties of the asset sets when the target metric is either fitness, diversity, or both evaluated through a MOO approach.

7.3.1 | Hypotheses

The aim of EXP4 is to establish whether iterative regeneration of assets within the CrawLLM pipeline will increase the assets' fitness and diversity. It also aims to establish the extent through which simple algorithms such as Regen and Evo as outlined in Section 7.2 affect these metrics. The hypotheses for this experiment are thus given below:

- H1 When using SOO the targeted metric will be improved by both Regen and Evo algorithms.
- H2 When targeting both fitness and diversity concurrently through MOO, both metrics will improve using both algorithms.
- H3 Evo algorithms are expected to perform better than Regen variants, achieving better overall improvement in the targeted metric.

7.3.2 | Experimental Protocol

The formulated hypotheses will be investigated through a series of experiments designed to address distinct objectives, and categorised by their objective metrics. These are listed in Table 7.2, outlining the combined methodologies used, and noting the tags used to refer to each, for brevity.

Each experiment commences with the 20 asset sets generated by the CrawLLM pipeline in Experiment EXP3, as detailed in Chapter 6. 25 iterations are then carried out

Table 7.2: Table showing the combined methodologies and the corresponding experiment tags in EXP4.

Algorithm	Target Metric		
	Fitness	Diversity	Multi-Objective
(1+1) Evolutionary	FitEvo	DivEvo	MOEvo
Regenerate and Test	FitRegen	DivRegen	MORegen

according to the experiment parameters for each of these sets, thereby constituting 20 runs for each experiment.

During each of these iterations, the least performant individual within each group vis-a-vis the target metrics is identified and selected for regeneration, in accordance with the strategies described in Section 7.2.3. Given that this selection process operates across 8 distinct groups at each iteration, a cumulative total of 200 individual assets is identified for potential replacement in each run. Figure 7.1 illustrates the overall process utilised for the experiments.

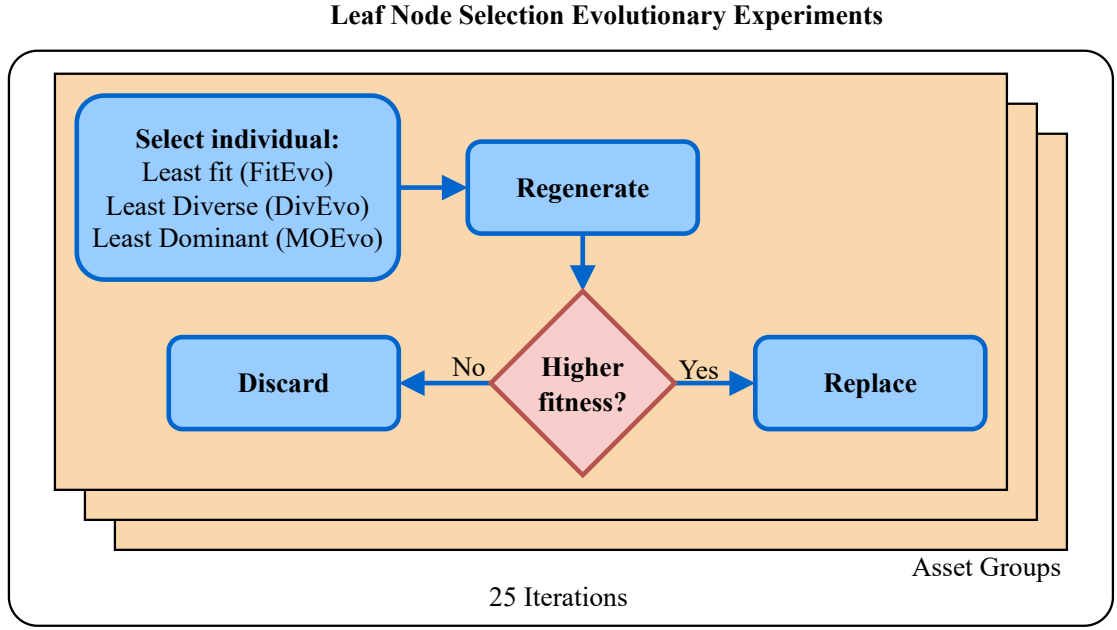
7.4 | Results and Evaluation

The central aim of the experiments, as outlined by the hypotheses presented in Section 7.3.1, is to assess the extent to which the fitness and diversity of the asset sets undergo improvement over iterative regeneration within the parameters of each experiment.

The Wilcoxon signed-rank test [224] was conducted upon the computed values to assess statistical significance of the observed changes. One-tailed variants of the test were applied in both directions in order to evaluate whether the observed changes exhibit meaningful increases or decreases. Statistical significance was presumed at a threshold of $p < 0.05$.

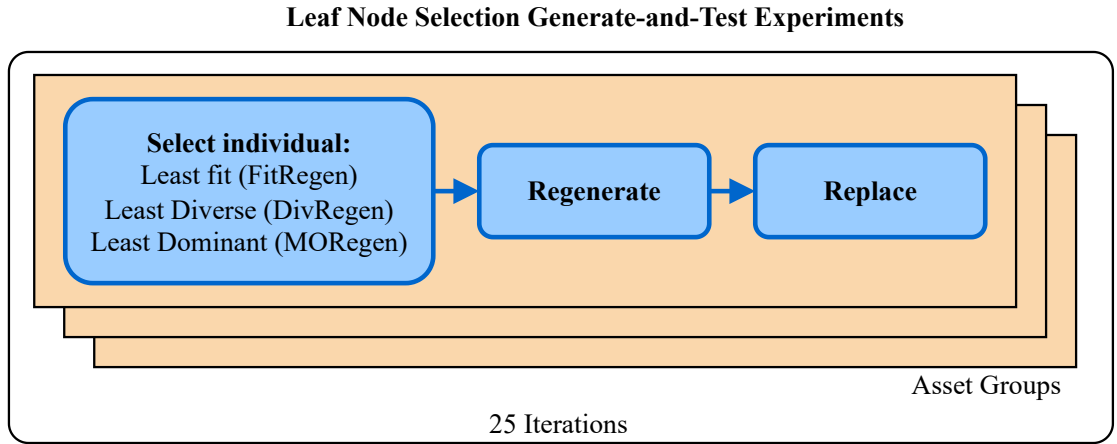
Furthermore, to adjust for the false discovery rate, the Benjamini-Hochberg correction [8] was applied.

The outcomes of these analyses are summarised in Table 7.3, which highlights how those experiments targeting an individual metric successfully improved it at the expense of the other. Thus DivEvo and DivRegen improved diversity significantly (by 18.72% and 17.68% respectively), but consistently worsened the fitness by around 1% from its initial value. Similarly, FitEvo and FitRegen improved the fitness by 4.07% and 3.08% respectively, and had an insignificant effect on diversity. It is disappointing to see that the maximum % change in fitness was only around 4%, despite achieving statistical significance. It can be noted that the Evo experiments generally outperformed the Regen



20 Independent Runs / Asset Sets

(a) (1+1) Evolutionary Experiments



20 Independent Runs / Asset Sets

(b) Regenerate and Test Experiments

Figure 7.1: Processes of the Evo (a), and Regen (b) sets of experiments. Each run consisted of 25 iterations, processing an individual from every group. A run is carried out for each of the 20 generated games from EXP3.

experiments in SOO, however for the MOO variants, the final diversities are seen to be

Table 7.3: Mean percentage change in fitness and diversity for each experiment. An * indicates statistically significant change using Wilcoxon signed rank test (corrected for false discovery rate, $p < 0.05$).

Experiment	Mean % Fitness Change	Mean % diversity Change
DivEvo	-1.00*	18.72*
DivRegen	-1.07*	17.69*
FitEvo	4.07*	-0.67
FitRegen	3.08*	0.53
MOEvo	2.03*	9.01*
MOREgen	1.46*	11.87*

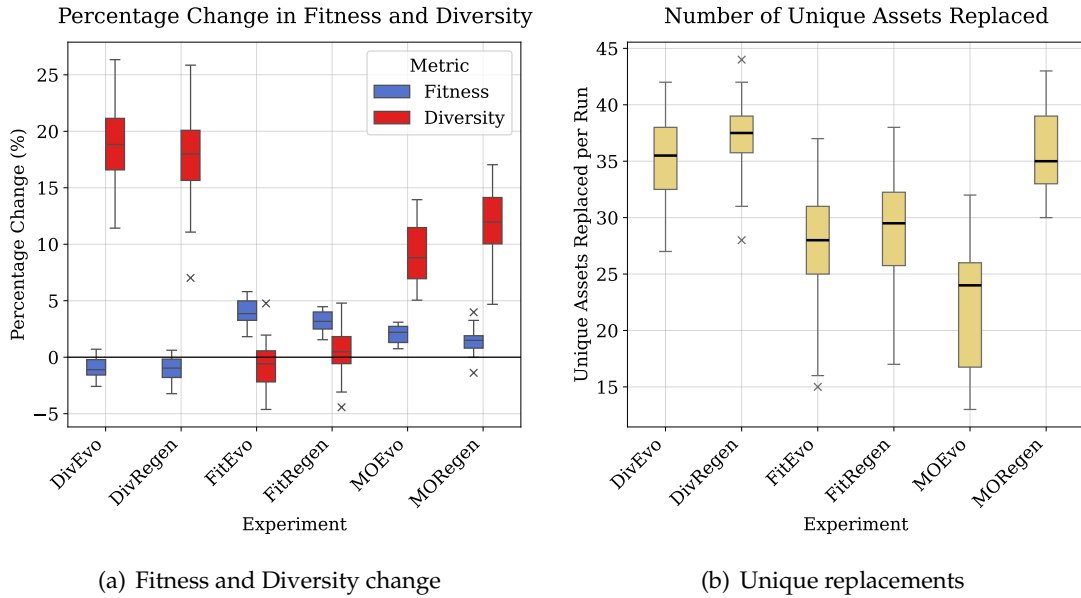


Figure 7.2: Box plots showing the percentage change in fitness and diversity (a), and the mean number of unique assets chosen for replacement (b). Values are averaged over the 20 runs of each experiment.

higher for the Regen experiments.

The total count of unique assets replaced during every run, along with the percentage variations in fitness and diversity, are visualised through box plots in Figure 7.2.

The box plots corresponding to the target metrics (Figure 7.2(a)) complement the results of Table 7.3. They assist in visualising the spread of the values across the experiments, which immediately convey the significant disparity between diversity and fitness, with the former achieving significantly higher changes when directly targeted.

Additional insights into the replacement dynamics are conveyed by the box plots shown in Figure 7.2(b). It is evident that targetting diversity resulted in a high number

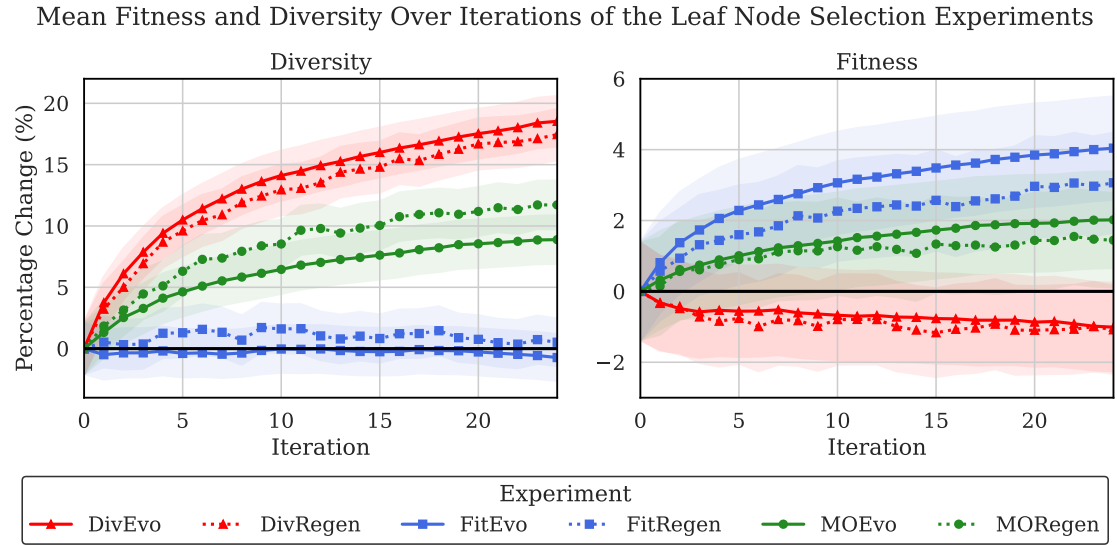


Figure 7.3: The % change in fitness and diversity metrics plotted against iterations for each experiment. Each value represents the mean across the 20 experiment runs for that iteration. The shaded areas denote 95% confidence intervals.

of unique replacements, i.e. the process had a wider-reaching effect on different assets from the set. Notably, the Regen experiments yielded an increased number of unique replacements, a result that aligns with expectations. In the Evo experiments which regenerate the least performant individual, and the replacement fails to surpass the existing asset in quality, the unchanged asset remains the poorest in subsequent iterations. This recurrence reduces the frequency with which new candidate assets are introduced for replacement.

This pattern is particularly pronounced in the MOEvo experiment. As the regenerated asset must satisfy multiple selection criteria, specifically, simultaneous improvements in both fitness and diversity, fewer replacements are ultimately achieved. Consequently, repeated regeneration attempts are concentrated on the same individual, thus constraining the overall variety of the regeneration process across the assets.

Finally, the progression of the fitness and diversity change over the iterations, averaged over the 20 runs are plotted in Figure 7.3, which also shows the 95% confidence intervals. This last set of graphs highlight the processes of each experiment in further detail.

The graphs (Figure 7.3) confirm that both Evo and Regen are effective, and gradually improve their target metric consistently, with a degree of randomness expected in the Regen experiments due to the unrestricted replacement of assets at every iteration.

7.5 | Follow-up Study on Branch Selection

An explorative study into a secondary methodology considered the entire set of assets as potential candidates for replacement. This includes both leaf nodes and *inner* assets that contribute to the generation of downstream content yet remain hidden from the end user. Under this approach, termed *branch selection*, the replacement of a single asset imposes the regeneration of all its descendant elements within the CrawlLLM pipeline, thus reflecting a broader scope of iterative influence. Using this strategy, a total of 162 assets are eligible for selection at every iteration, in contrast to the 72 for the leaf node selection strategy.

This illustrates the motivation for this experiment, arising from the observed constraints inherent in the leaf node selection mechanism. Since inner nodes are excluded from the replacement process, the fundamental thematic components remained unaltered across successive iterations. This homogeneity may have impeded the degree to which asset diversity could be meaningfully expanded. To illustrate further, all the descriptive text of the assets are inner nodes of the DAG since they are employed in the generation of images. Consequently, these textual elements remained unchanged within the asset set. Thus, although the visual representations of characters and locations were subject to regeneration, their underlying semantic identities were effectively immutable.

The branch selection approach, despite its complexity, is technically feasible in CrawlLLM due to the pipeline's capacity to regenerate any sub-branch of its generative process. This flexibility is illustrated in Figures 6.3 and 6.4. The stages for generation of enemies which produces a single enemy at every call, the same prompt and location description are provided for a new enemy to be generated. A minor adaptation is required in stages where multiple text assets are concurrently generated, as observed in steps 2 and 6 of Figure 6.3. When an inner asset from such a stage is selected for regeneration, a targeted prompt is provided to the LLM to regenerate only the specific asset in question. The remaining assets from the same stage are included unchanged within the prompt to preserve consistency. For example, the original prompt which generates the location details for a given location name and description is as follows:

```
We are working on a top-down dungeon crawler game of the following type:
Setting: ${setting}
Environment: ${environment}
Epoch: ${epoch}
Within this game, the main character will have to navigate though a dungeon-
like environment, collect keys or items which enable them to unlock doors,
and engage in combat with hostile enemies until they reach the final exit
. The view-point is a top-down orthographic view of the rooms. The levels
```

```

    consist of tiled sprites, which contain floor, wall/boundary, and door
    tiles. One of the wall tiles should have a visible indicator that there is
    a secret passage behind it, but it should not be too obvious.
The location ${location} has the following description: ${
    location_description}
Create a brief description of what the floor, walls (or boundaries), and
doors would look like. Limit the descriptions to what the floor, walls or
doors are made of and their appearance, and do not include anything about
their surroundings. The idea is to create a brief for an artist to draw a
tileable section of each. Next describe an object which can be used as a
key to unlock doors within this place.
All descriptions shall be strictly visual and shall be coherent within the
game setting and location.
Your output should be given in the following JSON format only:
{
    "floor": "...",
    "wall": "...",
    "door": "...",
    "key":
    {
        "name": "...",
        "description": "..."
    }
}

```

In order to replace a specific item from the location details, the replacement prompt becomes:

```

We are making a dungeon crawler game of the following type:
Setting: ${setting}
Environment: ${environment}
Epoch: ${epoch}
Within this game, the main character will have to navigate though a dungeon-
like environment, collect keys or items which enable them to unlock doors,
and engage in combat with hostile enemies until they reach the final exit
. The view-point is a top-down orthographic view of the rooms. The levels
consist of tiled sprites, which contain floor, wall/boundary, and door
tiles.
The location we are working on has the following properties:
${location}
Generate a different description for the ${detail_type} within this location.
    Limit the descriptions to what the ${detail_type} are made of and their
    appearance, and do not include anything about their surroundings. The
    description will be used as a brief for an artist to draw a tileable
    section of ${detail_type}.
Ensure that:

```

```

1) Everything remains coherent and make sense within the game setting,
   environment, and epoch.
2) Do not change any of the other elements in the location.
Your output should be given in the following JSON format only:
{
  "${detail_type}": "...
}

```

As an example of the branch regeneration in CrawlLM, if description of the main enemy in a specific location results to be the most dominated individual, then Figure 6.3 reveals that its minions' descriptions will also be selected for replacement. Figure 6.4 then shows that the enemy's spritesheet and portrait, as well as those of the selected minions are also flagged for replacement.

7.5.1 | Branch Selection

When considering branch replacement, multiple asset selections are more complex to manage as there may be common descendants in the different branches. In this case, it is by far more convenient to select only one asset for replacement from the full set, and regenerate its branch, one generative step of the pipeline at a time. These experiment variants were configured for MOO, so the selection of the least performing asset had to take into consideration a compound measure of fitness and diversity of the assets. A similar concept of domination used in the leaf node selection methodology is used. However, since in this case a unique asset across all groups is required, a direct comparison of fitness and diversity cannot be performed. This is due to the fact that the means of these metrics for each group vary significantly, and therefore those groups with higher average metrics would exert dominance and thus exclude themselves from the selection process.

The full set of assets is thus partitioned into K groups, as detailed in Table 7.4.

$$\mathcal{G} = \{G_1, G_2, \dots, G_K\}.$$

Each group G_k contains n_k individuals, indexed

$$G_k = \{x_{k,1}, x_{k,2}, \dots, x_{k,n_k}\}.$$

Each individual $x_{k,i}$ has two measured objectives:

$$\text{fitness: } f_{k,i}, \quad \text{diversity: } d_{k,i}.$$

In order to enable comparison across groups with potentially differing ranges of objective values, each individual's fitness and diversity values are normalised within

Table 7.4: The full asset set grouping by type, used for diversity comparisons in branch node selection strategies. 162 assets are considered in total.

Group	Asset Count	Type
1	3	Game name, Introduction, Conclusion
2	16	Character Names
3	16	Character Descriptions
4	16	Character Portraits
5	16	Character Spritesheets
6	3	Card Frames
7	5	Location Names
8	5	Location Descriptions
9	5	Location Floor Descriptions
10	5	Location Wall Descriptions
11	5	Location Door Descriptions
12	5	Key Names
13	5	Key Descriptions
14	5	Location Tilesets
15	5	Location Backgrounds
16	20	Key Images
17	9	Action Card Names
18	9	Action Card Descriptions
19	9	Action Card Images

their respective group using percentile ranks. For each group G_k , the fitness and diversity values are ranked, and each individual $x_{k,i}$ is assigned normalized scores:

$$\hat{f}_{k,i} = \frac{\text{rank}_f(f_{k,i}) - 1}{n_k - 1}, \quad \hat{d}_{k,i} = \frac{\text{rank}_d(d_{k,i}) - 1}{n_k - 1},$$

where $\text{rank}_f(f_{k,i})$ denotes the ascending rank of $f_{k,i}$ among all fitness values in G_k , and similarly for $\text{rank}_d(d_{k,i})$. This places all individuals in a common normalised space $[0, 1]^2$, with higher values indicating better fitness and diversity.

The full population is then considered as the union of all individuals across groups, with each individual represented by its normalised objective vector:

$$\mathcal{P} = \bigcup_{k=1}^K \left\{ \left(\hat{f}_{k,i}, \hat{d}_{k,i} \right) \mid i = 1, \dots, n_k \right\}.$$

Domination is defined in the standard Pareto sense for maximisation: an individual $p \in \mathcal{P}$ is said to *dominate* another individual $q \in \mathcal{P}$, denoted $p \succ q$, if

$$\hat{f}_p \geq \hat{f}_q, \quad \hat{d}_p \geq \hat{d}_q, \quad \text{and} \quad (\hat{f}_p > \hat{f}_q \text{ or } \hat{d}_p > \hat{d}_q).$$

For each individual $x \in \mathcal{P}$, its domination count is defined as the number of other individuals in $\mathcal{P}$ that dominate it:

$$\delta(x) = \sum_{\substack{y \in \mathcal{P} \\ y \neq x}} \begin{cases} 1, & \text{if } y \succ x, \\ 0, & \text{otherwise.} \end{cases}$$

Individuals with the highest domination counts are considered the most dominated and are candidates for replacement or regeneration.

In the event of a tie (i.e., multiple individuals sharing the highest domination count), a secondary criterion is applied. Let $\mathcal{C}(x)$ denote the set of all descendants of individual x , as determined by known parent-child relationships. The number of *dominated descendants* of x is defined as:

$$\Delta(x) = \sum_{y \in \mathcal{C}(x)} \begin{cases} 1, & \text{if } \delta(y) > 0, \\ 0, & \text{otherwise.} \end{cases}$$

Among tied individuals, preference is given to those with the largest $\Delta(x)$, as these are viewed as propagators of poor-quality traits within the population.

The final selected individuals for regeneration are those with maximal domination count $\delta(x)$, and among them, maximal dominated descendant count $\Delta(x)$.

7.5.2 | Experimental Protocol

The protocol for the branch selection experiments was kept as similar as possible to that of the leaf node selection MOO experiments. Two experiments were carried out, both concurrently targeting fitness and diversity; MOBranchEvo uses a (1 + 1) GA whereas MOBranchRegen employs the regenerate-and-test strategy. These are meant to mirror MOEvo and MOREgen experiments respectively, and are thus analogously carried out for 20 runs each starting from the same 20 CrawlLLM asset sets from Chapter 6.

In these experiments, however, a single individual is selected from the entirety of the groups at each iteration. If the regenerated asset satisfies the predefined criteria for replacement (unconditionally in the Regen variants, and conditionally in the Evo variants, where both fitness and diversity must be improved), the entire asset branch derived from that individual is replaced. The exact number of assets replaced in this manner varies as a function of the specific asset selected.

To ensure an equitable comparison with the leaf node experiments, the total number of regenerated assets is tallied. Iterations continue until this count reaches or marginally exceeds 200, thereby approximating computational equivalence. Figure 7.4 illustrates the overall process utilised for the experiments.

Table 7.5: Mean percentage change in fitness and diversity for the branch selection experiments. The results of the multi-objective leaf node selection experiments are also listed for reference. An * indicates statistically significant change using Wilcoxon signed rank test (corrected for false discovery rate, $p < 0.05$).

Experiment	Mean % Fitness Change	Mean % diversity Change
MOBranchEvo	0.26	2.67*
MOBranchRegen	-0.29	6.06*
MOEvo	2.03*	9.01*
MOREgen	1.46*	11.87*

In practice, due to the greater number of groups and the substantially higher volume of assets evaluated during each iteration, the computational time associated with the branch selection experiments is significantly higher. Although this computational difference varies by how many assets are changed in the branch replacement algorithm, the increase in duration was measured to fall between 30% to %60.

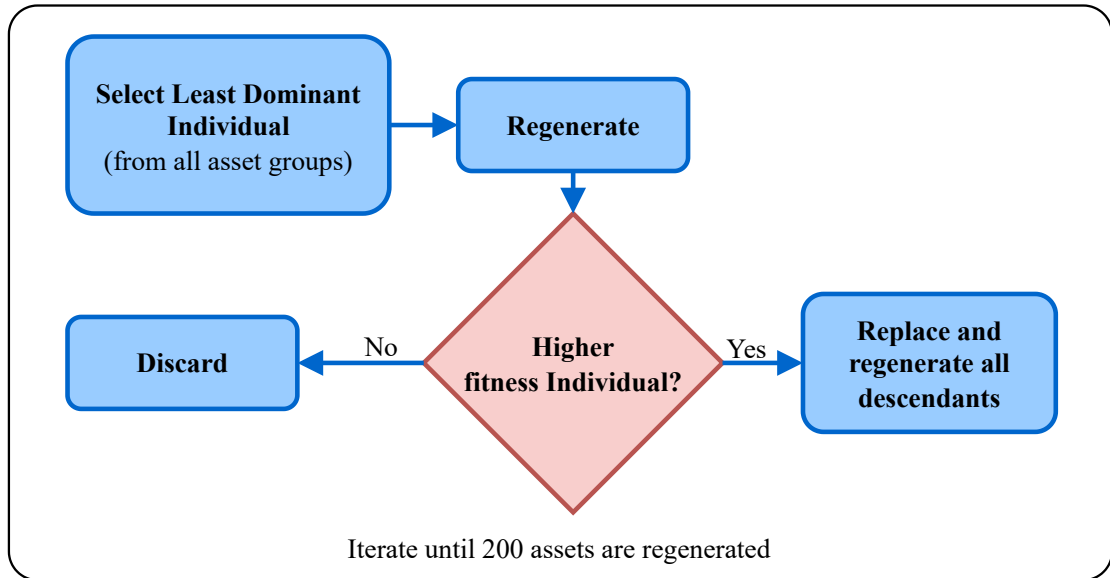
7.5.3 | Results

The percentage changes observed in fitness and diversity resulting from the branch selection experiments are presented in Table 7.5. To facilitate comparison, the corresponding values obtained from the MOEvo and MOREgen leaf node selection experiments are likewise included. In order to assess the statistical significance of these changes, Wilcoxon tests were performed in an analogous manner, with similar adjustments applied for false discovery rate control.

Comparing the leaf node and branch selection experiments, MOEvo and MOREgen performed better than MOBranchEvo and MOBranchRegen, respectively. The % change in fitness was negligible for the branch experiments, and the increase in diversity was 2.67% for MOBranchEvo and 6.06% for MOBranchRegen. This contrasts with the leaf node selection experiments, which achieved a statistically significant increase in both metrics, albeit a marginal one for fitness. In order to further investigate these findings, the data is plotted using a visualisation method similar to that which was applied in the leaf node selection experiments.

Figure 7.5 shows the box plots for the % change in fitness and diversity (Figure 7.5(a)), and the mean number of unique replaced assets over each experiment's 20 runs (Figure 7.5(b)). The graphs show an almost even distribution of the fitness values around 0% change, confirming that there is no significant change in this metric for branch selection. The diversity increase in these experiments is confirmed, however the plot whiskers show

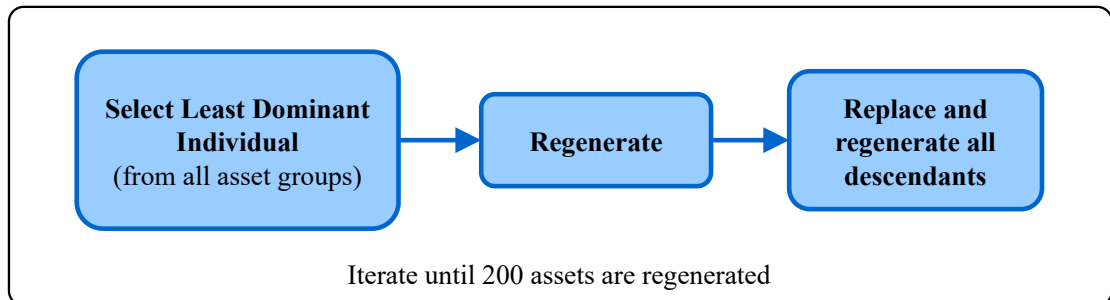
Branch Selection Evolutionary Experiments (MOBranchEvo)



20 Independent Runs / Asset Sets

(a) (1+1) Evolutionary Experiments

Branch Selection Generate-and-Test Experiments (MOBranchRegen)



20 Independent Runs / Asset Sets

(b) Regenerate and Test Experiments

Figure 7.4: Processes of the MOBranchEvo (a), and MOBranchRegen (b) sets of experiments. Each run processes the most dominated individual across all groups, and iterations stop when the total assets generated reaches 200. A run is carried out for each of the 20 generated games from EXP3.

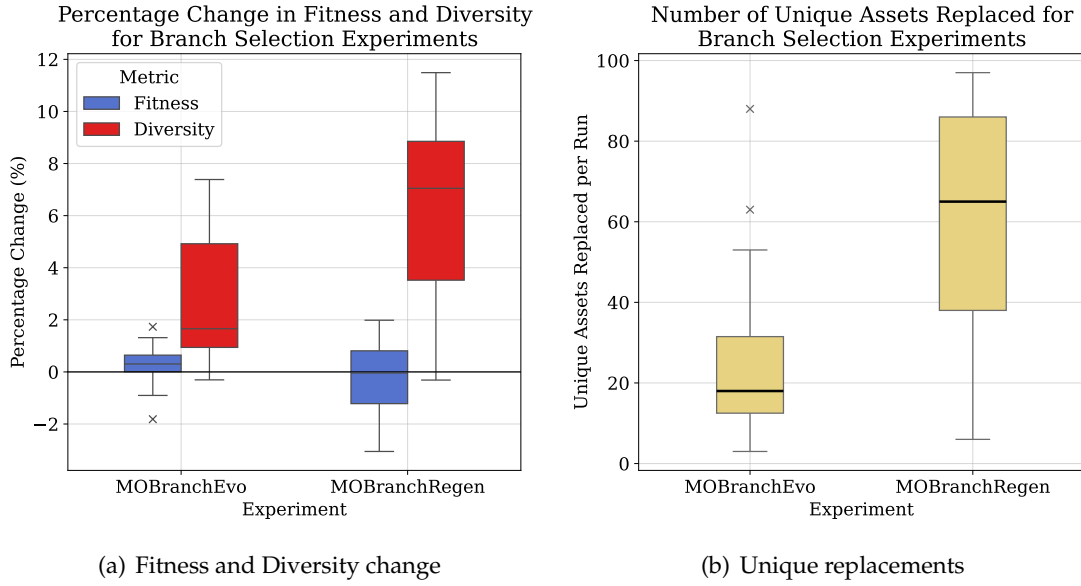


Figure 7.5: Box plots showing the percentage change in fitness and diversity (a), and the mean number of unique assets chosen for replacement (b), for the branch selection experiments. Values are averaged over the 20 runs of each experiment.

that there are instances during which the change in the metric was very low, even with marginally negative values. This behaviour does not mirror that of the corresponding leaf node experiments, where the diversity spread occurred in the region above 5% of positive change, even at minimal increase.

The box plots showing unique replacements cannot be compared directly, due to the considerable disparity between the number of selection candidates in the two selection methods. However, it can be noted that the Evo algorithms consistently replaced far fewer unique assets than the Regen variants. This was an expected result, as analogously to the leaf node experiments, Evo require strict conditions for actual asset replacements unlike the Regen method. Furthermore, failure to replace the assets because of suboptimal resulting regenerated outputs, would reiterate the asset as the least performing one in the following iteration, thus reducing the total possible replacement counts altogether. It is worth noting that the mean number of unique replacements for MOEvo, MOREgen, MOBranchEvo, and MOBranchRegen were 22, 36, 26, and 28 respectively, while the minimum unique replacements were 13, 30, 3, and 6. The difference in unique replacements between the two methodologies (leaf node and branch) is particularly striking in the lower bound of the count in MOBranchRegen. It may be envisaged that over-

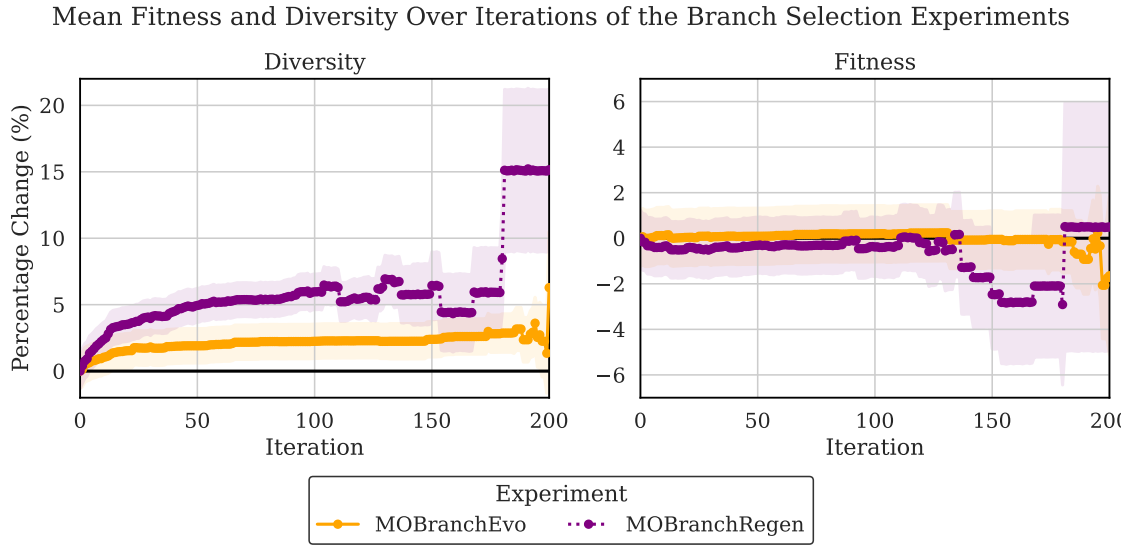


Figure 7.6: The % change in fitness and diversity metrics plotted against iterations for the branch selection experiments. Each value represents the mean across the 20 experiment runs for that iteration. The shaded areas denote 95% confidence intervals.

laps between replaced assets across different iterations will reduce the count of unique replacements. However, the minimum value of 6 is considerably smaller than the 30 minimal count in MORegen, despite the latter having a much smaller selection pool (76 assets, compared to the 162 in MOBranchRegen). Given the larger pool, MOBranchRegen would normally be expected to have a higher or at least comparable minimum number of unique replacements. The observed minimum suggests that the selection mechanism in MOBranchRegen may be converging on a small subset of assets, likely due to regenerated assets failing to improve the rankings of the computed metrics.

The progression of the % change in the targeted metrics over the iterations are shown in Figure 7.6, together with their 95% confidence intervals. The most prominent observation is the highly stochastic nature of the changes in the computed metrics for the branch experiments (Figure 7.6). In particular an evident sharp increase is observed at around the 180th iteration. The confidence intervals show that this is a relatively unique occurrence, within the overall noise progression. In contrast, the progress of all the leaf node experiments are much more continuous, even monotonic for the Evo variants. This difference is due to the nature of the branch replacement strategy, wherein multiple assets may be replaced within the same iteration, leading to drastically different fitnesses and diversity computations.

7.6 | Discussion

The preceding section has analysed the experimental results, illustrating that iterative asset replacement leads to notable enhancements in the quality of the asset set, through improvements in both fitness and diversity metrics.

A particularly unexpected observation was the relatively modest increase in fitness when compared to the more pronounced improvement in diversity. This phenomenon may be attributed to the inherent capabilities of the SDXL image generator, which already achieves a high degree of alignment between generated images and the corresponding input text prompts. These prompts serve as the parent elements within the CrawlLLM pipeline and, consequently, constitute the baseline values against which fitness is evaluated. A similar explanation may be applied to the Mixtral 8×7B LLM employed for text generation. This model consistently produces coherent outputs in response to its prompts, thereby resulting in an already elevated baseline for fitness. Given this strong inherent alignment of the utilised models, there is an overarching high level of coherence also due to the fact that some of the assets are parents of multiple children within the pipeline. For example, the visual style is used in the prompts pertaining to the inputs of all image generation stages.

It had been anticipated that the Evo experimental conditions would outperform the Regen variants overall, and this expectation was largely met with the notable exception of the diversity metric. This divergence is understandable. In Regen experiments, asset replacement was applied at every iteration, thereby introducing a consistent source of variation. In contrast, the Evo methodology only permitted replacement when superior individuals were identified, leading to fewer opportunities for diversity to be introduced. The only case in which the Evo variant surpassed Regen in diversity was within the DivEvo experiment, which was explicitly targeting improvements in that metric.

The branch replacement strategy was observed to be particularly disruptive in the magnitude of change it effected on the metrics of the asset set, as illustrated in Figure 7.6. These disruptions resulted in significant shifts in both fitness and diversity values at points where replacements occurred. The extent of these shifts was observed to correlate with the hierarchical position of the selected replacement candidate within the CrawlLLM pipeline. Specifically, higher placements in the hierarchy led to a greater number of assets being replaced, thus contributing to larger metric fluctuations. Given the inherently stochastic effects of this selection strategy, the terminal condition used in the experiments, namely the replacement of 200 assets, may have been insufficient. It is plausible to hypothesise that extending the number of iterations would attenuate the observed noisiness in the metrics. Nevertheless, such an extension would entail

increased computational overhead, particularly given the demands of evaluating metrics over the enlarged asset group sizes (refer to Tables 7.1 and 7.4), thereby rendering the approach potentially impractical.

An evident advantage of the branch selection algorithm lies in its potential to modify core structural elements of the asset set, which remain inaccessible when relying solely on leaf node selection. This broad scope makes the methodology potentially suitable for integration within a HITL interface, where human designers may be prompted to evaluate and approve more substantial changes that impact a wider spectrum of game content.

However branch selection has demonstrated instances where the count of unique assets over an experiment run was excessively low, in comparison to the relatively large selection pool. This failure to replace unique assets is difficult to assess holistically, without seeking explanations within individual occurrences. However, it could indicate a flaw within the strategy adopted to select a single, most dominated individual across the entire asset pool, rather than within a group. (refer to Section 7.5.1). Metric comparison between assets in different type groups were accomplished by using percentile ranks, to mitigate the disparity in the mean values within the groups. The strategy may still not sufficiently compensate for the group differences, or the variety of group sizes could be interfering with the percentile rankings, i.e. smaller groups would have larger stepped values in percentile ranking than numerous groups. This would lead to some assets appearing more frequently than others as the worst global performer, despite small improvements in their value.

The computational resources required for EXP4 were identical to those delineated in Section 6.2.4 of Chapter 6 with respect to hardware requirements. In the leaf node selection experiments, each iteration regenerated an asset from every group, thereby ensuring that computational time remained relatively consistent across iterations. It should be noted, however, that the diversity calculations were rendered marginally more computationally complex due to the fact that most assets possessed a greater number of peers within their respective groups than parent nodes. Nevertheless, the resulting differences were observed to be practically negligible. Furthermore, since both fitness and diversity metrics were being computed at every iteration across all experimental configurations, the particular metric defining the objective did not have any appreciable impact upon the overall computational time. The MOO variants of the experiments incurred an additional, albeit modest, computational overhead attributable to the calculation of dominance relations. However, given that all fitness and diversity metrics were already being computed as part of the standard procedure, the dominance computation did not realistically influence the total computational time in any meaningful way.

The branch selection experiments, by contrast, exhibited considerably greater variance with respect to computational time. Although these experiments were designed to terminate upon reaching the same total number of generated assets as their leaf node selection counterparts, the *type* of assets being generated was observed to vary substantially across different experimental runs, depending on which individual was chosen for regeneration. Owing to the substantial disparity between text and image assets in terms of generation duration, a randomly variable computational cost was observed across the overall experimental procedure.

Referring back to the experimental hypotheses, they are found to be largely supported. Specifically, the FitEvo and FitRegen experiments yielded improvements in the fitness metric, while DivEvo and DivRegen succeeded in enhancing diversity, thereby satisfying hypothesis H1. Hypothesis H2 was affirmed through the results of the MOEvo and MORegen experiments, both of which demonstrated significant improvements in fitness and diversity, as shown in Table 7.3. Hypothesis H3 was also mostly validated. FitEvo outperformed FitRegen in terms of fitness, and DivEvo produced greater improvements in diversity than DivRegen. For the MOO experiments, this hypothesis did not hold in the case of diversity, as previously discussed.

Although increases in diversity were recorded for both leaf node and branch selection experiment sets, fitness exhibited no significant change under branch replacement. Furthermore, these effects may not be reproducible with consistency, owing to the observed randomness of the changes in the evaluated metrics following the concurrent replacement of multiple assets.

Considering the research questions of the thesis, Experiment EXP4 is relevant to all four. It answers question RQ1 because it uses the $(1 + 1)$ GA and generate-and-test mechanisms to steer the outputs of the pipeline towards more coherent and diverse outputs. Despite not strictly belonging to the QD family, these two methodologies were implemented here with similar objectives, and hence they also align to the spirit of the question. The answer to RQ2 lies in the chosen selection strategies. The leaf-node and branch strategies each restructure the asset set into distinct groups, which in turn dictate the candidates that may be selected. Although this may not be directly construed as a partitioning or shaping of the search space, it bears an impact on how the exploration is effected. Furthermore it has been noted that the different selection strategies have distinct results in terms of improvement of the target metrics, fitness and diversity.

In terms of EXP4, RQ3 can be answered through the iterative approach itself. In practice, the experiment has shown how applying an iterative approach to the generative process, both for the LLM and diffusion models, enables the pipeline to improve the coherence and diversity of the generated assets. Both the Evo and Regen approach have

exhibited improvements in both of these metrics, showcasing a practical application of this iterative technique.

In response to RQ4, EXP4 has employed a different multimodal model, FLAVA, to determine a meaningful distance between any two or more assets within the asset set. It has been shown as an effective orthogonal metric to the multimodal models used in the generative processes themselves (CLIP). FLAVA was the foundation for the defined framework of metrics and comparison metrics employed in the experiment, namely the computation of fitness for each asset, the diversity within groups of assets, and the domination metrics employed to determine the least fit individuals. It has been shown to be consistent and efficient in this role, establishing the use of such models as a solid foundation in similar evaluative roles.

One possible direction for further research would be the incorporation of heterogeneous evaluation signals, such as surrogate human preference models or task-specific constraints, which could reduce evaluative circularity which may be present when comparing assets to a multimodal latent space. Adaptive selection strategies balancing local refinement with occasional structural intervention may further stabilise convergence behaviour of the evolutionary processes proposed herein.

Because EXP4 operates within the same domain and asset pool as EXP3, its empirical generalisation is narrow. However, the procedural insight offers a broader scope for generalisation. Iterative evaluation–regeneration loops, guided by approximate metrics and selective replacement, can incrementally steer multimodal systems toward desired trade-offs. This principle is applicable beyond games, provided that artefacts can be grouped meaningfully, and evaluation proxies are available at intermediate stages.

7.7 | Limitations and Shortcomings

The iterative pipeline improves upon earlier approaches by introducing adaptive regeneration based on fitness and diversity metrics; however, it remains constrained by the reliance on automated evaluation. As discussed in previous sections, the same metrics used to identify weak assets are also used to assess improvement, which introduces a degree of evaluative circularity. Observed gains should therefore be interpreted as alignment with the chosen objectives rather than as definitive improvements in player-facing quality.

The domination-based selection strategy appears sensitive to group heterogeneity and size disparities, potentially biasing replacement toward certain asset types. This raises concerns regarding the robustness of the selection mechanism across heterogeneous

multimodal asset pools.

One other limitation is the fact that overall coherence is being assessed through alignment between assets and their parents within the CrawLLM pipeline. There is an implicit assumption that if all descendants are evaluated as being coherent to their respective parents, then the player-facing leaf node assets are also coherent to each other. Although this assumption is convenient for an automated evaluation of the asset set, it is not explicitly validated, and may not necessarily always be the case.

Human judgement is not incorporated directly into the iterative loop. While the selected metrics are informed by prior experiments and user-facing validation, they cannot capture subjective or experiential dimensions such as narrative impact or gameplay enjoyment. Integrating human feedback at this stage would likely improve grounding but would also substantially increase complexity and cost.

Although diversity is explicitly targeted, it is treated primarily in aggregate at the level of asset groups. The pipeline does not model structural relationships between assets beyond their metric scores, and may therefore increase numerical diversity while still producing thematically redundant or functionally similar content.

Finally, the computational demands of repeated multimodal generation impose practical limits on scalability. The cost of large language model inference and image generation constrains the number of iterations that can be performed, which in turn limits analysis of long-term convergence behaviour.

These limitations situate EXP4 as an important step towards adaptive multimodal PCG, while also underlining the need for richer evaluation frameworks and tighter integration of human judgement, which are discussed further in Chapter 8 and inform the future work outlined in Section 9.1 of Chapter 9.

7.8 | Summary

This chapter has demonstrated a practical application of applying iterative regeneration techniques to multimodal pipelines to improve overall coherence and diversity within the generated assets. This set of experiments concludes the work proposed within this thesis. The following chapter collates the contributions and discusses their merits and opportunities for future work in this domain.

Discussion

The primary objective of this thesis was to explore the integration of contemporary GenAI models into pipelines designed for multimodal content generation, with particular attention given to their adaptation for enhancing both the coherence and diversity of the resulting assets. The experimental investigations documented across Chapters 4 to 7 focused exclusively on the generation of images and text. Despite this constraint, multiple strategies were revealed through which multimodal models can be employed to generate, retrieve, and evaluate assets in a manner that preserves their coherence holistically. In addition, the algorithms introduced were shown to foster diversity at various stages of the generative process.

This chapter is dedicated to revisiting the experimental findings in a concise format, offering a critical discussion of the strengths and limitations inherent to each methodology. Moreover, the research questions initially formulated in Section 1.4 of Chapter 1 are systematically answered, with the aim of contextualising the contributions derived from the experiments performed.

8.1 | Combining Evolution and Backpropagation

Chapter 4 explored a novel hybrid approach for improving PCG in GenAI systems, particularly in text-to-image generation using VQGAN-CLIP. The study recognises the limitations of the generative model, which produces high quality but semantically narrow and stylistically homogeneous outputs due to the biases of their training corpora. It thus proposes an alternating structure combining backpropagation-based refinement with novelty-driven QD exploration. The method revolves around the alternating cycles where backpropagation optimises fitness using CLIP scores to measure text alignment, and an EA, more specifically NSLC, injects diversity at a population level. The process is guided by either chromatic (HSV) or NN activation space (ViT) novelty metrics.

This dual-cycle methodology is empirically tested through experiment EXP1, which involves generating image populations for five textual prompts and analysing per-

formance across three metrics: CLIP fitness, HSV diversity, and ViT diversity. These respectively represent the semantic alignment of the images to the input text prompt, their colour variance, and their feature variance. The results show that exploration cycles disrupt semantic fitness temporarily but lead to notable diversity gains, particularly in the HSV-driven variant. However, ViT-guided exploration fails to sustain diversity improvement, likely due to the classifier nature of the ViT embedding space, which collapses perceptually distinct images into similar representations if they share high-level class features. While the refinement cycle reliably restores semantic fitness after each exploration phase, it also gradually homogenises the population, especially in later iterations, indicating diminishing returns on late-stage NSLC interventions.

The experiment validates several hypotheses: that NSLC increases diversity and preserves semantic alignment, though its effectiveness is contingent on the choice of diversity metric. HSV-driven diversity correlates more strongly with perceptual novelty and yields better results overall. These findings underscore the importance of structuring the latent search space effectively and selecting metrics that align with human perceptions of variation.

The proposed hybrid algorithm offers several advantages, including enhanced image diversity without sacrificing semantic fidelity, adaptability to multiple feature domains via metric choice, and the ability to explore and refine in tandem. The population-based perspective introduced by EAs provides a global view of the creative outputs, countering the limited perspective of gradient-based refinement which only focus on a single artefact at a time. Moreover, the design of the experiment is modular in nature, facilitating substitution of metrics or models.

The primary disadvantage, on the other hand, arises from the computational overhead of alternating cycles, especially due to the reliance on nearest-neighbour calculations and multi-objective optimisation of the NSLC algorithm. Another limitation of the experiment lies in the genetic operations chosen. The EA relied exclusively on mutation operations, foregoing crossover or recombination due to the restrictions imposed by the latent space structure of the VQGAN model used. This may have the effect of limiting evolutionary expressiveness throughout the extent of the process. Furthermore, the poor performance of the ViT metric suggests challenges in generalising feature-space novelty measures, particularly when relying on embeddings derived from NNs trained for specific tasks, such as classification.

The conceptual framework presented, consisting of cycling between refinement and exploration, is presumably applicable to multimodal generative systems in a broad scope. However, the effectiveness of this paradigm depends heavily on the choice of diversity metrics, the nature of the generative model's latent space, and the training corpus.

Applying this method to other modalities such as audio or text, or to multimodal applications requires rethinking the representation and metric structures to retain effectiveness. Therefore, although generalisable in principle, this method needs careful adaptation for the constraints of the particular use case it is applied to. The VQGAN-CLIP algorithm is no longer considered SOTA in text-to-image generation, having been supplanted by more performant models, particularly diffusion-based ones. Nevertheless, the principle of alternating cycles may still be relevant to other models operating through iterative refinement, even if these originate from other unrelated domains, provided the proposed algorithms are adapted accordingly with appropriate metrics.

8.2 | Multimodal Search Space Illumination

Chapter 5 presented an alternative approach to evolving multimodal creative artefacts combining GenAI and QD algorithms. Specifically, it describes MELiTA, a novel extension of MAP-Elites tailored to the evolutionary exploration of a bimodal search space, which employs a transverse assessment mechanism that enhances modal coherence by pairing mutated artefacts with complementary elites already in the archive. MELiTA is designed to generate artefacts that are both semantically coherent and behaviourally diverse, addressing divergence between the two modalities by evaluating and recombining components from phenotypically related elites during search. The method was illustrated through the generation of text and image, in the form of fictional video game descriptions and associated cover images using GPT-2 for text and SD for images. Variation operators target both modalities independently, incorporating partial and full mutations to avoid premature convergence and enhance the search space illumination.

Text artefacts are behaviourally characterised through topic modelling using LDA, while image artefacts are classified using discretised measures of colourfulness and edge-based complexity. These metrics define the axes of a 2D feature map, enabling structured illumination of the multimodal search space. Fitness is assessed using CLIP score, which quantifies semantic alignment between paired text and images.

Experimental comparisons between MELiTA and baseline MAP-Elites, across multiple runs and distinct game title seeds, resulted in MELiTA consistently yielding elites with higher mean and maximum fitness scores faster than MAP-Elites in most cases. This indicates that the algorithm is efficient in discovering solutions toward higher performing regions of the search space.

However, diversity in terms of behavioural coverage often accumulates more slowly under MELiTA. The elite update strategy inserts new solutions only if a new individual

outperforms all other viable combinations, restricts the rate of exploration. Therefore, MELiTA demonstrates accelerated convergence to higher fitness solutions, but not accelerated exploration across the totality of the search space. MELiTA generates artefacts that are more visually and semantically diverse, when diversity is assessed using metrics decoupled from the search structure. The trade-off is reduced coverage in the feature space and increased computational overhead due to the evaluation of multiple hybrid offspring per generation.

While MELiTA archives are more sparse due to stricter replacement criteria, it generates solutions with increased visual and semantic diversity. This is assessed using metrics decoupled from the search structure. Specifically, MELiTA achieved significantly higher distances using LPIPS for perceptual visual distance, SSIM for image structure variation, and SBERT embeddings for semantic textual diversity across different titles in the archive. These metrics enable an assessment that aligns with human perception of diversity. Importantly, even with fewer individuals in the archive, MELiTA preserved or improved perceptual and semantic spread, indicating that transverse assessment can inject diversity indirectly, even when archive coverage is reduced.

Despite its constraints, MELiTA demonstrates potential for generalisation. Its feature map may be expanded to accommodate more complex or higher-dimensional multimodal applications, such as those involving text, image, and audio modalities. In fact, MELiTA's architecture can theoretically support any number of modalities and behavioural axes. However, it is worth noting that despite its extensibility, increasing the feature map's dimensionality will also increase the number of elite comparisons required at each iteration, thereby limiting its effective scalability.

The advantages of MELiTA include improved fitness between modalities, and greater observed visual and semantic diversities. Its design allows it to address multimodal coherence more effectively than the basic MAP-Elites algorithm. The relatively small models adopted by the use case in EXP2, make the process faster, but may be easily replaced with more performant models with minimal architectural changes to the process. Since the algorithm operates at the phenotypical level, all components of the pipeline, such as the mutation genetic operators, the fitness and diversity metrics, and the BCs across modalities may be interchanged freely.

The limitations observed from the experiment include the reduced feature map coverage and increasingly higher computational cost due to transverse fitness evaluations with increasing modal dimensions. Within the implemented use case, potentially simplistic BCs were selected. More practical applications may require a different reorganisation of the search space through a different set of BCs. The chosen text mutation strategy, especially partial mutations, may yield overlapping or repetitive outputs. This could be

mitigated through the use of bidirectional generative models, such as BERT to regenerate a randomly selected segment of the text. Lastly, the use case relied on an older generative model, GPT-2 for text generation, which is limiting in terms of overall narrative coherence compared to more recent models. Since this experiment was designed and carried out, several more capable language models have emerged, some even designed to be lightweight LLMs [119] intended to operate on mobile devices. The substitution of such models into the algorithm would enhance text quality and coherence. Other models, such as the Mixtral $8\times 7B$ used in EXP3 and EXP4, would also improve the text quality at the expense of computation time.

8.3 | A Zero-Shot Multimodal Pipeline

Chapter 6 presented CrawLLM, an automated, zero-shot generative AI pipeline that orchestrates the creation of textual and visual assets for video games. It uses pre-trained LLMs and diffusion-based SDXL image generation models to re-theme a complete dungeon crawler game with card-based combat, from high-level textual prompts describing a desired narrative setting and visual style. All models used in the pipeline are open-source, pre-trained, and executed locally, promoting reproducibility, accessibility, and independence from proprietary platforms.

The pipeline first generates thematic descriptors, setting, environment, epoch, and visual style, that may be either manually provided or generated by the LLM. These descriptors are cascaded into the hierarchical generation of narrative elements such as the game's title, PC, locations, enemies, and cards, all constrained to a JSON format for further automation. The visual assets, intended for direct use in the game engine, include tilesets, spritesheet animations, combat backgrounds, card illustrations, and card frames. Image generation is conditioned by ControlNet models, with additional tools like IP-Adapter to ensure consistency in content and format. Game visuals with specialised functionality, such as location tilesets and character animation spritesheets are generated in staged processes to maintain coherence and structure. For example, the tileset generation is carried out through a three-stage inpainting method that ensures distinction between floor, wall, and door tiles.

The significant strengths of the pipeline may be identified as its zero-shot design and its completely autonomous structure. No model is fine-tuned for the specific task, yet the entire output is functional and coherent enough to be deployed directly within the Unity game engine with no human intervention required. To evaluate the quality of generated content and the coherence to the seed setting, a user study was conducted

using screenshots and textual descriptions from the generated games. 34 participants were tasked with identifying thematic elements and rating appeal.

The results of the user study show partial but meaningful support for the hypotheses posed at the beginning of the experiment. Participants were able to identify the thematic details of the generated assets significantly above chance, particularly with respect to the setting and environment attributes. Across the 15 stimuli presented (5 screenshots from the dungeon crawling environment, 5 from the card-based combat, and 5 narrative descriptions), 30 out of 45 multiple-choice questions yielded accuracies significantly above the 25% chance baseline, indicating that the thematic coherence of the assets is generally perceivable within the game. Recognition of the setting was the most successful, with average accuracy at 64%, indicating that participants could often discern the intended theme. Environment identification also showed strong performance in dungeon-crawling scenarios, with a 56% accuracy. However, epoch and visual style were much less reliably identified, with accuracy averages of 43% and 45% respectively, and some individual stimuli failing to significantly surpass chance levels. This indicates a clear limitation in the experiment. The cause may be twofold; the issue may stem from the pipeline's inability to generate clear temporal and stylistic intent visually, but it may also result from the way in which those thematic elements were defined. Since for the experiment the seed thematic elements were generated by the LLM itself, this possibility in turn may be due either to the LLM's failure to generate meaningful descriptions of the intended epoch and visual style, or the source of the problem may lie within the definition of the input prompt from which the descriptions are generated. This is an interesting point of exploration for future work.

The entropy scores measure the consistency of participant responses, and were used to better interpret these findings. High entropy in many of the lower-accuracy questions reveals significant participant disagreement, particularly in style and epoch classifications, confirming the ambiguity between the stimuli and these thematic descriptors. Some visual styles, like "Egyptian relief," were correctly identified with 100% accuracy, indicating that specific descriptors, with high semantic relevance can guide successful image generation and recognition. Others, such as "semi-abstract" or "minimalist," were far less effective, likely due to their vagueness or the generative model's difficulty in visualising them consistently.

The metrics robustly indicated that participants found the assets visually and narratively appealing. 10 out of 15 stimuli received ratings significantly above the neutral baseline of 3 on a five-point Likert scale, especially in the case of card combat visuals and narrative descriptions, which persistently scored higher than dungeon-crawling images. The most appealing visuals were those featuring strong thematic cohesion and high

visual clarity, such as those with flat floor tiles, vivid colour contrast, or well-composed card art. Conversely, dungeon scenes which incorporated convoluted textures, or offered low contrast between elements which are functionally distinct, obtained low ratings, indicating a sensitivity to visual noise and asset integration quality.

Overall, the CrawlLLM pipeline offers several notable advantages. Foremost among them is its full automation and general applicability across diverse themes without requiring model fine-tuning. This enables rapid re-theming of entire games using only high-level semantic prompts. Another key strength lies in its use of open-source models executed locally, which not only promotes reproducibility and control over model behaviour but also reduces dependency on third party or commercial infrastructure. Furthermore, the modular architecture of the system, including hand-crafted prompt templates and structured output formatting, makes it potentially extensible, or adaptable to other game genres with reasonable effort.

However, the system also presents some limitations. One significant drawback is the inconsistent control over fine-grained visual attributes, particularly visual style, which often fails to manifest clearly in the generated assets despite being explicitly included in the prompts. This may be due to the placement of style descriptors within the prompts or inherent limitations of the generative models in interpreting abstract stylistic cues.

There is also semantic ambiguity in theme descriptors for the epoch, which may lead to poor alignment of the generated assets to the thematic intent. Another issue lies in the visual coherence of certain asset categories, particularly the dungeon tilesets, where randomly sampled tiles occasionally produce stylistic dissonance or excessive visual noise. The reliance on random sampling from generated tile variants further compounds this problem, occasionally undermining the aesthetic consistency within a single game environment. Another limitation in the visuals arises from the fact that pre-trained diffusion models are unable to generate images with incorporated transparency. Assets such as the character spritesheet thus require an additional background removal process which may also be imprecise and detrimental to the resulting image. Moreover, while the pipeline automates complex content generation, it requires significant expertise in prompt engineering and content structuring to adapt it effectively to other domains, which may limit its implementation by less technical users. These challenges suggest a possible need for improvements in prompt design and visual conditioning techniques. Although the pipeline performs well in generating content that is cohesive to the high-level thematic settings and is generally appreciated by players, its ability to convey more nuanced meaning may require improvement.

8.4 | An Iterative Generation Pipeline

The CrawlLLM generative pipeline is improved in Chapter 7 by the addition of iterative refinement methods. This alteration recognises the shortcoming of the original pipeline in its absence of automated output evaluation. This systemic lack of awareness could produce visually or thematically redundant assets, or those which may exhibit poor semantic alignment to the theme elements. Two simple iteration strategies are thus introduced to enhance both the internal coherence (fitness) and diversity within asset group types, namely the regenerate-and-test method and a (1+1) evolutionary algorithm. These are designed to selectively regenerate underperforming assets within the DAG structure of the generative system, primarily focusing on leaf nodes to avoid computationally demanding cascaded regenerations.

EXP4 tests six variations of these strategies targeting either fitness or diversity as a SOO problem, or both in a MOO configuration. The results show statistically significant improvements in the targeted metric in all four SOO variants. FitEvo and FitRegen achieved fitness improvements of around 4% and 3% respectively, while DivEvo and DivRegen boosted diversity by around 19% and 18% respectively. These results are corroborated by the Wilcoxon signed-rank tests adjusted for false discovery rate, confirming the effectiveness of both algorithms in optimising their respective objectives. Targeting both fitness and diversity using MOO led to improvement in both metrics. MOEvo improves fitness by around 2% and diversity by 9%, while MOREgen achieves just above 1% fitness gain and almost 12% rise in diversity. The overall improvements in fitness may be attributed to the already high alignment of outputs from the advanced models utilised, namely SDXL and Mixtral 8×7B. In comparison, diversity exhibits much greater improvement, particularly in Regen experiments due to their unrestricted replacement policy. Evo consistently outperforms Regen in the fitness experiments, FitEvo and FitRegen, and shows greater diversity increase in DivEvo compared to DivRegen. However, as already stated, MOREgen outperforms MOEvo in diversity improvement, albeit with a slightly lower gain in fitness.

A follow-up study explored the branch selection methodology, expanding candidate assets to include internal nodes whose regeneration triggers cascading replacement through the pipeline. This strategy potentially increases novelty within the asset set and enables alteration of core structural elements, such as descriptive text which is otherwise immutable with leaf node selection. The methodology is applied to MOO using both Evo and Regen algorithms. The experiments show no overall change in fitness, but an increase of almost 3% and 6% in MOBranchEvo and MOBranchRegen respectively. Despite being more computationally demanding, it introduces considerable stochasticity

within the progressive values of the metrics over the iterations, and results in lower average improvements. This may be occurring due to the percentile-based normalisation used to compare asset groups to identify globally poor performers. This may not fully mitigate group size and mean metric disparities between asset type groups, possibly skewing the replacement candidate selection.

The advantages of the proposed iterative algorithms lie in their capacity to systematically enhance the asset set using metrics based on multimodal embeddings for both evaluation and optimisation. Evo strategies enforce quality control by allowing replacement only when improvements are verified, making them ideal for controlled refinement, while Regen offers more exploratory variation suitable for increasing creative breadth. The use of FLAVA as an orthogonal evaluator to CLIP increases the evaluative robustness, and promotes model modularity.

There is, however, evidence of disadvantages of the system. The improvement in fitness is constrained by the high baseline quality of SOTA models, counteracting the observable benefit of optimisation. The Regen strategy may replace assets indiscriminately, risking regression in quality. Branch selection strategies, though theoretically powerful, suffer from low efficiency, poor generalisability across asset types, and significant randomness due to cascading effects. Additionally, the selection of a single asset within the multiple asset groups is challenging, and the chosen strategy may introduce bias or instability.

An interesting topic of discussion which emerges from this experiment involves the delicate balance between coherence and novelty in content generation. The experiments demonstrate that optimising one metric often impairs the other, suggesting an inherent tension between producing internally consistent content and introducing diverse and novel content. This balance is especially prominent in games, or applications in general, which are strongly rooted in narrative. Excessive diversity may disrupt the coherence of the narrative, while strict cohesion to the theme may feel repetitive or uninspiring.

In terms of generalisability, the proposed techniques are flexible and are not dependent on specific models. Their effectiveness, however, depends heavily on the structure of the generative pipeline and the complexity of asset interdependencies. The methodologies are suitable for pipelines with clearly defined hierarchical structures and asset type groupings, but may require modifications for more complex pipelines or more modalities.

The question of scalability is also prominent. The iterative methods used are conceptually simple, however their practical implementation across large asset sets and complex generative DAGs requires non-negligible computational costs. The branch selection strategy, although theoretically promising for novelty search, becomes impractical due to

the explosion of downstream dependencies. As the pipelines scale to larger and more complex structures, different evaluation strategies may be required, such as surrogate models for evaluating fitness and diversities across asset groups.

8.5 | Addressing the Research Questions

The aim of the experiments above has been to provide empirical evidence in support of the four research questions outlined in Section 1.4 of Chapter 1. Each investigation contributes towards an understanding of how GenAI and QD algorithms can be orchestrated to enhance multimodal PCG. Although each chapter has discussed its contribution towards answering these question, it is convenient to revisit the questions and seek their resolution of each, holistically across all experiments.

RQ1 How do evolutionary QD algorithms enhance the semantic alignment and diversity of outputs generated by foundation models in multimodal PCG?

Experiments EXP1, EXP2, and EXP4 each demonstrated in different ways how QD algorithms may improve the semantic alignment and the diversity of outputs across multiple GenAI models. NSLC improved these metrics for VQGAN-CLIP generation in EXP1, for text-guided image generation. It demonstrated that the qualities of interest within the image must be targeted by specific metrics to improve them. MELiTA did the same for GPT-2 and SD in EXP2 for bimodal text and image generation. Both the alignment between the modalities of each output pair, as well as the perceived overall diversity of the final elites within the archive were improved. In EXP4, the (1 + 1) GA and generate-and-test algorithm improved the fitness and diversity in the CrawLLM pipeline which uses Mixtral 8×7B and SDXL for multimodal asset sets. It is acknowledged that these methodologies are not part of the QD family, nonetheless they are implemented here with similar objectives, especially in the MOO variant. This thesis therefore offers a selection of methodologies which may be utilised over a range of scenarios, proving that hybrid approaches to PCG are advantageous when targeting the improvement of fitness and diversity in GenAI models.

RQ2 How does an explicit structuring of multimodal search spaces, or the metrics used to traverse them influence the effectiveness of exploration strategies in generating coherent and diverse outputs?

EXP1 demonstrates such a subdivision of the search space through the distinct metrics guiding the search, namely the HSV and ViT diversity measures. The experiment high-

lighted how the metrics guiding the search offer their distinctive exploratory advantages, leading to solutions whose diversity is aligned to specific properties. EXP2 further exemplifies this concept, and shapes the search space through the BCs chosen to delineate the feature map. Once again, this ad hoc structure serves to direct the exploration towards solutions which feature the desired characteristics. EXP4 does not directly segment the search space, but the different selection strategies, leaf node and branch-based, create distinct groupings of the assets within the solution set. The chosen diversity metrics, as well as the domination count, were defined through this segmentation, which in turn directs the search process. Collectively, these respond to RQ2 by showing that the structuring of the domain, or that of the metrics used to explore it, is effective in steering an increase in fitness and diversity, promoting within the assets the distinct characteristics selected through the segmentation.

RQ3 To what extent can LLMs direct multimodal generative pipelines to improve coherence, diversity, and overall quality in functional problems such as games?

The CrawlLLM pipeline described in EXP3 and its iterative version presented in EXP4 answer this question directly through their results. The basic pipeline, without refinement already shows that the LLM is an effective orchestrator, directing the pipeline towards generating asset sets which humans perceived as being appealing and coherent to the overall thematic setting. The iterative refinement added in EXP4 further improves the coherence and diversity of the asset set, whilst retaining the functionality of the assets within the game engine. The fact that this iterative approach only achieved minor increases in fitness is itself attributable to the already coherent and well-aligned content generated under direction of the LLM, underlining its effectiveness in this role.

RQ4 Which metrics and assessment frameworks are most effective for evaluating and optimising coherence, diversity, and quality in the outputs of multimodal generative systems?

Throughout the ensemble of experiments in this thesis, several metrics have been identified to fulfil this evaluative role productively. These may be grouped into 2 distinct categories. The first comprises those metrics which are computed through empirical, logical, or statistical formulae, such as HSV diversity, colourfulness BC, and SSIM similarity for images, and LDA for text. The second category includes all those which employ a NN-based model to compute the desired property. These may be further subdivided into those models which are multimodal in nature, and those which operate on a single modality. The former is the one which was most commonly used across all experiments,

and includes multimodal models such as CLIP and FLAVA. These are crucial for the work described in this thesis, as they form the core of all fitness calculations through their ability to generate comparable embeddings for both text and images. The diversities in EXP4 was also measured using FLAVA, demonstrating the flexibility and utility of these models. The second subcategory comprises metrics based on unimodal NNs, such as ViT diversity, complexity characteristic (based on HED), and LPIPS similarity for images, and SBERT similarity for text. These altogether form a body of metrics for characterisation, optimisation, comparison, or final evaluation for images and text, which may be used within generative systems or pipelines as showcased throughout these experiments.

The documented findings collectively encompass a range of algorithms for enhancing the outputs of systems comprising GenAI models, incorporating hybrid algorithmic approaches that enhance the coherence and diversity of generated assets. Such processes advance the field and contribute towards controllable multimodal generators, opening new avenues for exploration in the domain of GenAI. The experiments offer novel multimodal optimisation strategies, leading to more autonomous content creation systems. In addition to their theoretical contributions, the proposed algorithms offer practical and relevant functions within the domains of PCG and game development. Their application is intended to facilitate rapid prototyping processes and to support co-creative development, increasing efficiency in content design.

8.6 | Observations

A prominent observation from these studies concerns the interpretability and reliability of similarity metrics, specifically those based on embeddings. While models such as FLAVA or CLIP offer a robust multimodal embedding space, the resulting scores are inherently abstract and lack direct semantic meaning. This raises concerns about the alignment between computational similarity and human perception of coherence or diversity. For example, two images may appear visually distinct to a human designer, but may yield a low diversity score due to shared conceptual features in the embedding space. This misalignment underscores the need for complementary human evaluation or more interpretable metrics that may bridge the gap between latent similarity that perceived by humans.

Another observation concerns the rapid advance of technology within this domain. Throughout the duration of this thesis, the landscape of GenAI has shifted and changed drastically. This accelerated development has not slowed down, and the flow of new models and methodologies continues undeterred. Although this rapid development

is both interesting and valuable, it also results in rapid obsolescence of models and algorithms. The work presented in this thesis has emphasised the *modularity* of the models used. This is particularly useful as newer and more powerful models can easily be replaced in to the proposed systems, thereby taking advantage of the progressing SOTA, and offering longevity to the work.

8.7 | Summary

This chapter has summarised the findings from the experiments carried out in this thesis, showing the advantages and disadvantages of the presented systems. The research questions have been holistically answered through the results obtained. The next chapter concludes this manuscript with a final synopsis and thoughts about future work.

Conclusion

This thesis has explored various methodologies which advance the utility of GenAI algorithms, developing strategies that serve to better align output modalities in terms of coherence, and stimulate the increase of novelty within the process. While a detailed synthesis has been provided in Chapter 8, this concluding chapter offers a broader reflection on the contributions and overarching themes of the work.

It has been shown that hybrid generative frameworks that combine exploratory QD methods with gradient-based refinement can effectively improve semantic alignment and perceptual novelty. Such an amalgamation enables pre-trained models to be guided beyond their inherent likelihood distributions, thereby promoting the emergence of outputs that are both diverse and consistent with conditioning prompts.

Structured approaches to behavioural characterisation have been demonstrated to enhance the illumination of generative search spaces. Improvements in both search efficiency and diversity have been achieved by considering the relationships between modalities throughout induced variations.

Large pre-trained models have been repurposed from static generators to adaptable components within the broader scope of orchestration pipelines. Through zero-shot prompting, iterative refinement, and algorithmic feedback, these have been shown to exhibit a high degree of flexibility without requiring bespoke models, additional training, or domain-specific adjustments.

Throughout the various experiments multiple metrics have been adopted for different tasks including search space characterisation, search direction, and assessment. Whether targeting a single modality or more, the studies show that a single metric cannot capture the full breadth of properties pertaining to a problem, and a manifold approach is more effective for a holistic assessment.

Collectively these contributions underline the importance of appropriate search methodologies and metrics in the design of generative systems. They suggest a path toward more controllable, efficient, and meaningful applications of GenAI across multiple domains.

9.1 | Future Work

Several limitations were encountered throughout the experiments conducted, each of which presents opportunities for further investigation. A number of potential extensions have been identified, offering promising directions in which the work presented in this thesis may be expanded. A selection of those deemed most consequential by the author is presented hereunder.

A valuable continuation of the experiment outlined in EXP1 would involve the integration of exploratory cycles within contemporary GenAI frameworks. Diffusion-based generative models, such as SD and SDXL, are characterised by an iterative refinement process [175]. In these systems, the initial seed selection often bears a considerable impact on the output image composition, thereby resulting in a high baseline diversity. Furthermore, diffusion models are known to exhibit strong alignment with the guiding text prompt, which implies that any improvement in fitness from any addition to the baseline process may be marginal, as was observed in EXP4. Nonetheless, with the incorporation of appropriate fitness metrics within exploratory cycles, it would be feasible to introduce exploratory cycles with the aim to produce outputs with specific desired attributes such as chromatic divergence or increased structural complexity, without altering the original text prompt.

Regarding EXP2, its scope may be broadened to encompass multiple modalities, employ more advanced generative models, integrate different behavioural characterisations, and incorporate alternative mutation operators, thereby confirming its validity and generalisability.

The CrawlLLM pipeline from EXP3 may be improved in several ways. One evident improvement is the addition of other modalities, for example audio components. Text-to-audio models, such as the Audio Latent Diffusion Model [130], could be incorporated to generate sound effects and music within the pipeline, under the direction of the orchestrating LLM.

A staged interactive user interface for the CrawlLLM generative pipeline can readily be envisioned, wherein designers are afforded the ability to modify the text produced by the pipeline, triggering downstream regeneration, or requesting image regeneration or refinement throughout the process, perhaps in a similar fashion to EXP4.

Furthermore, the current generation of animation spritesheets is restricted to bipedal creatures due to the limitations imposed by the OpenPose ControlNet applied. This process could be revised to utilise pre-trained image and text to video generative models [64]. This would contribute to the simplification of the multi-stage procedure, and enable the inclusion of a broader range of creature types. Another shortcoming of the existing

pipeline lies in the reliance on background removal tools for producing transparent images, where these are required for sprites. This frequently resulted in imprecise cropping of salient features of the image. A more robust solution has recently been introduced, wherein a model trained on transparent images is applied directly to the latent space of diffusion models, altering their output to generate images incorporating a transparency layer [237]. The integration of this model into the pipeline is expected to improve the quality of transparent sprites.

Additional mechanics to the CrawLLM gameplay could be introduced by incorporating further interactive elements, such as collectible items or environmental hazards. However, a pivotal research direction involves transitioning the pipeline from the current CrawLLM game to an entirely different game genre. This adaptation would serve as a critical test of the generalisability of the orchestration capabilities demonstrated by the LLM.

Building upon EXP4, a redesign of the fitness evaluation strategy could be implemented, in order to determine whether alternative formulations influence the extent of improvement throughout the algorithm. Other refinements to the branch selection methodology may also be considered, such as the investigation of alternative selection mechanisms for the least fit individuals, as well as the exploration of novel criteria governing branch replacements. Such changes may result in more stability over iterations for the target metrics.

Beyond the scope of the experiments described, this line of research may promote a wide array of future advancements. Within the domain of game orchestration, the CrawLLM system may be generalised and further extended to encompass diverse facets of game development, including level and rules generation. It may be extended to generate its own templates for the graphic design and layouts. Integration with a formal description language such as the Video Game Description Language (VGDL) proposed by Schaul [182], could enable the orchestration of fully-fledged games through a unified representational framework. Each facet within the orchestration process may be characterised and explored through illumination of a feature map analogous to that employed in MELiTA, thus enabling the generation of a diverse spectrum of game variants across distinct behavioural axes.

9.2 | Long-Term Impact

The CrawLLM and MELiTA systems suggest the potential for generative ecosystems, where multiple GenAI models evolve together, each within its domain (e.g., audio,

narrative, level design), yet coordinated under high-level orchestration. Future work could explore agentic systems [161] in an evolutionary approach, where AI agents compete and cooperate across shared fitness landscapes, generating novel design spaces that human designers may later interpret, or draw inspiration from. Such systems could generate novel genres or hybrid modalities, such as dynamic rulesets co-evolved with audiovisual themes.

Beyond immediate extensions of the present work, the systems and methodologies proposed in this thesis suggest a foundation for the emergence of autonomously creative agents which generate content within predetermined constraints and also actively engage in self-directed exploration of a characterised search space. Future research could investigate system creativity, where GenAI systems develop internal representations of novelty, value, and style preferences, adapting their behaviour over time through interactions with users or other agents. This could transform generative pipelines into self-evolving creative collaborators, reaching beyond the domain of games into interactive storytelling, virtual world-building, or educational applications.

With further advancements in GenAI technologies, HITL systems can progress from simple prompt-response systems to dialogic, multimodal co-creation frameworks, in which the system actively proposes, critiques, and refines content as a collaborator of human designers. The multimodal structuring and behavioural characterisation strategies developed in this thesis offer a roadmap for such systems to understand and navigate semantic spaces coherently, even across disparate modalities like visuals, narrative, gameplay, and audio. Future research could build on these principles to develop cognitive interfaces that blend human intent and machine assistance.

Beyond metrics of fitness and diversity, future systems could consider intentionality, interpretability, and possibly alignment with cultural and ethical norms. Research inspired by this thesis might explore goal-aware generation, where systems evaluate their performance and also their objectives. This would enable them to adapt their strategies based on inferred user goals, aesthetic values, or ethical constraints.

Finally, the structured generative search strategies developed in this thesis could be adapted for domains beyond creative media, such as hypothesis generation in scientific research, or generative design in engineering. The principles of multimodal illumination, fitness-diversity balancing, and iterative refinement could be used to explore unbounded design spaces in fields like pharmaceutical discovery or materials science, where solutions are concurrently evaluated across semantic and functional behavioural axes.

Bibliography

- [1] Al-E'mari, S., Sanjalawe, Y., & Fataftah, F. (2025). A review on DeepDream neural networks. *Proceedings of the 1st International Conference on Computational Intelligence Approaches and Applications (ICCIAA)*. <https://doi.org/10.1109/ICCIAA65327.2025.11013153> (cit. on p. 39).
- [2] Alvarez, A., Dahlskog, S., Font, J., & Togelius, J. (2019). Empowering quality diversity in dungeon design with interactive constrained map-elites. *Proceedings of the IEEE Conference on Games*. <https://doi.org/10.1109/CIG.2019.8848022> (cit. on p. 29).
- [3] Alvarez, A., & Font, J. (2022). TropeTwist: Trope-based narrative structure generation. *Proceedings of the Foundations of Digital Games conference*. <https://doi.org/10.1145/3555858.3563271> (cit. on p. 29).
- [4] Balestrieri, R., Ibrahim, M., Sobal, V., Morcos, A., Shekhar, S., Goldstein, T., Bordes, F., Bardes, A., Mialon, G., Tian, Y., Schwarzschild, A., Wilson, A. G., Geiping, J., Garrido, Q., Fernandez, P., Bar, A., Pirsiavash, H., LeCun, Y., & Goldblum, M. (2023). A cookbook of self-supervised learning. *arXiv preprint arXiv: 2304.12210*. <https://doi.org/10.48550/arXiv.2304.12210> (cit. on p. 18).
- [5] Bello, A., Ng, S.-C., & Leung, M.-F. (2023). A BERT framework to sentiment analysis of tweets. *Sensors*, 23(1). <https://doi.org/10.3390/s23010506> (cit. on p. 18).
- [6] Beltagy, I., Lo, K., & Cohan, A. (2019). SciBERT: A Pretrained Language Model for Scientific Text. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, 3615–3620. <https://doi.org/10.18653/v1/D19-1371> (cit. on p. 18).
- [7] Bengesi, S., El-Sayed, H., Sarker, M. K., Houkpati, Y., Irungu, J., & Oladunni, T. (2024). Advancements in generative AI: A comprehensive review of GANs, GPT, autoencoders, diffusion model, and transformers. *IEEE Access*, 12, 69812–69837. <https://doi.org/10.1109/ACCESS.2024.3397775> (cit. on p. 11).

- [8] Benjamini, Y., & Hochberg, Y. (2018). Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B (Methodological)*, 57(1), 289–300. <https://doi.org/10.1111/j.2517-6161.1995.tb02031.x> (cit. on pp. 112, 127).
- [9] Bishop, C. (2006). *Pattern recognition and machine learning*. Springer. (Cit. on p. 15).
- [10] Blank, J., & Deb, K. (2020). Pymoo: Multi-Objective Optimization in Python. *IEEE Access*, 8, 89497–89509. <https://doi.org/10.1109/ACCESS.2020.2990567> (cit. on p. 55).
- [11] Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of machine Learning research*, 3(Jan), 993–1022 (cit. on pp. 50, 73).
- [12] Boden, M. A. (1998). Creativity and artificial intelligence. *Artificial Intelligence*, 103(1-2), 347–356. [https://doi.org/10.1016/s0004-3702\(98\)00055-1](https://doi.org/10.1016/s0004-3702(98)00055-1) (cit. on pp. 31, 32).
- [13] Boden, M. A. (2004). *The creative mind: Myths and mechanisms*. Routledge. <https://doi.org/10.4324/9780203508527> (cit. on pp. 32, 47).
- [14] Bodily, P. M., & Ventura, D. (2024). Operationalizing essential characteristics of creativity in a computational system for music composition. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(1), 447–455. <https://doi.org/10.1609/aaai.v38i1.27799> (cit. on p. 47).
- [15] Boggia, M., Ivanova, S., Linkola, S., Kantosalo, A., & Toivonen, H. (2022). One line at a time — Generation and internal evaluation of interactive poetry. *Proceedings of the 13th International Conference on Computational Creativity* (cit. on p. 47).
- [16] Brown, P. F., Cocke, J., Pietra, S. A. D., Pietra, V. J. D., Jelinek, F., Lafferty, J. D., Mercer, R. L., & Roossin, P. S. (1990). A statistical approach to machine translation. *Computational Linguistics*, 16(2), 79–85 (cit. on p. 34).
- [17] Brown, P. F., Della Pietra, V. J., deSouza, P. V., Lai, J. C., & Mercer, R. L. (1992). Class-based n-gram models of natural language. *Computational Linguistics*, 18(4), 467–480 (cit. on p. 34).
- [18] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Nee-lakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., ... Amodei, D. (2020). Language models are few-shot learners. *Proceedings of the Neural Information Processing Systems Conference* (cit. on pp. 17–19, 35, 70).

- [19] Bruce, J., Dennis, M., Edwards, A., Parker-Holder, J., Shi, Y. (, Hughes, E., Lai, M., Mavalankar, A., Steigerwald, R., Apps, C., Aytar, Y., Bechtle, S., Behbahani, F., Chan, S., Heess, N., Gonzalez, L., Osindero, S., Ozair, S., Reed, S., ... Rocktäschel, T. (2024). Genie: Generative interactive environments. *Proceedings of the 41st International Conference on Machine Learning*, 4603–4623 (cit. on p. 92).
- [20] Bulatov, A., Kuratov, Y., Kapushev, Y., & Burtsev, M. (2024). Beyond attention: Breaking the limits of transformer context length with recurrent memory. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(16), 17700–17708. <https://doi.org/10.1609/aaai.v38i16.29722> (cit. on p. 17).
- [21] Buongiorno, S., Klinkert, L., Zhuang, Z., Chawla, T., & Clark, C. (2024). PANGeA: Procedural artificial narrative using generative AI for turn-based, role-playing video games. *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*. <https://doi.org/10.1609/aiide.v20i1.31876> (cit. on pp. 45, 46).
- [22] Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6), 679–698. <https://doi.org/10.1109/TPAMI.1986.4767851> (cit. on p. 47).
- [23] Cao, Z., Hidalgo, G., Simon, T., Wei, S.-E., & Sheikh, Y. (2021). OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 43(01), 172–186. <https://doi.org/10.1109/TPAMI.2019.2929257> (cit. on p. 103).
- [24] Chomsky, N. (1957). *Syntactic structures*. De Gruyter Mouton. <https://doi.org/doi:10.1515/9783112316009> (cit. on p. 34).
- [25] Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., Schuh, P., Shi, K., Tsvyashchenko, S., Maynez, J., Rao, A., Barnes, P., Tay, Y., Shazeer, N., Prabhakaran, V., ... Fiedel, N. (2023). PaLM: Scaling language modeling with pathways. *The Journal of Machine Learning Research*, 24(1), 11324–11436 (cit. on p. 19).
- [26] Christiano, P. F., Leike, J., Brown, T. B., Martic, M., Legg, S., & Amodei, D. (2017). Deep reinforcement learning from human preferences. *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 4302–4310 (cit. on p. 18).
- [27] Coello Coello, C. A. (2006). Evolutionary multi-objective optimization: A historical view of the field. *IEEE Computational Intelligence Magazine*, 1(1), 28–36. <https://doi.org/10.1109/MCI.2006.1597059> (cit. on p. 14).

- [28] Coello Coello, C. A. (2010). Constraint-handling techniques used with evolutionary algorithms. *Proceedings of the Genetic and Evolutionary Computation Conference*, 1310–1333. <https://doi.org/10.1145/3520304.3533640> (cit. on p. 73).
- [29] Cohen, H. (1995). The further exploits of AARON, painter. *Stanford Humanities Review*, 4(2), 141–158 (cit. on p. 38).
- [30] Cohen, M. F., Shade, J., Hiller, S., & Deussen, O. (2003). Wang tiles for image and texture generation. *ACM Transactions on Graphics*, 22(3), 287–294. <https://doi.org/10.1145/882262.882265> (cit. on p. 44).
- [31] Colton, S. (2012). The painting fool: Stories from building an automated painter. In *Computers and creativity* (pp. 3–38). Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-31727-9_1 (cit. on pp. 31, 33, 39).
- [32] Colton, S. (2021). Evolving neural style transfer blends. *Proceedings of the International Conference on Computational Intelligence in Music, Sound, Art and Design* (cit. on p. 29).
- [33] Colton, S., & Browne, C. (2009). Evolving Simple Art-Based Games. *Applications of Evolutionary Computing*, 283–292. https://doi.org/10.1007/978-3-642-01129-0_32 (cit. on p. 38).
- [34] Colton, S., Charnley, J., & Pease, A. (2012). Computational creativity theory: The FACE and IDEA descriptive models. *Proceedings of the 2nd International Conference on Computational Creativity, ICC3 2011* (cit. on pp. 1, 3, 6, 7, 32).
- [35] Colton, S., Valstar, M. F., & Pantic, M. (2008). Emotionally aware automated portrait painting. *Proceedings of the 3rd International Conference on Digital Interactive Media in Entertainment and Arts*, 304–311. <https://doi.org/10.1145/1413634.1413690> (cit. on p. 39).
- [36] Cook, M. (2021). The road less travelled: Trying and failing to generate walking simulators. *arXiv preprint arXiv:2104.10789*. <https://doi.org/10.48550/arXiv.2104.10789> (cit. on p. 44).
- [37] Cook, M. (2022). Puck: A slow and personal automated game designer. *Proceedings of the Eighteenth AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*. <https://doi.org/10.1609/aiide.v18i1.21968> (cit. on p. 44).
- [38] Cook, M., Colton, S., & Gow, J. (2017a). The ANGELINA videogame design system - Part I. *IEEE Transactions on Computational Intelligence and AI in Games*, 9(2), 192–203. <https://doi.org/10.1109/TCIAIG.2016.2520256> (cit. on p. 44).
- [39] Cook, M., Colton, S., & Gow, J. (2017b). The ANGELINA videogame design system - Part II. *IEEE Transactions on Computational Intelligence and AI in Games*, 9(3), 254–266. <https://doi.org/10.1109/TCIAIG.2016.2520305> (cit. on p. 44).

- [40] Cook, M., Colton, S., & Pease, A. (2012). Aesthetic considerations for automated platformer design. *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*. <https://doi.org/10.1609/aiide.v8i1.12524> (cit. on p. 44).
- [41] Copet, J., Kreuk, F., Gat, I., Remez, T., Kant, D., Synnaeve, G., Adi, Y., & Défossez, A. (2023). Simple and controllable music generation. *Proceedings of the 37th International Conference on Neural Information Processing Systems* (cit. on p. 88).
- [42] Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., & Bharath, A. A. (2018). Generative adversarial networks: An overview. *IEEE Signal Processing Magazine*, 35(1), 53–65. <https://doi.org/10.1109/MSP.2017.2765202> (cit. on p. 37).
- [43] Crowson, K., Biderman, S., Kornis, D., Stander, D., Hallahan, E., Castricato, L., & Raff, E. (2022). VQGAN-CLIP: Open domain image generation and editing with natural language guidance. *Proceedings of the European Conference on Computer Vision 2022*, 88–105. https://doi.org/10.1007/978-3-031-19836-6_6 (cit. on pp. 22, 42, 54).
- [44] Cully, A., & Demiris, Y. (2017). Quality and diversity optimization: A unifying modular framework. *IEEE Transactions on Evolutionary Computation*, 22(2), 245–259. <https://doi.org/10.1109/TEVC.2017.2704781> (cit. on p. 69).
- [45] Dangeti, P. (2017). *Statistics for machine learning*. Packt Publishing. (Cit. on p. 80).
- [46] Dao, T., Fu, D. Y., Ermon, S., Rudra, A., & Ré, C. (2022). FLASHATTENTION: Fast and memory-efficient exact attention with IO-awareness. *Proceedings of the 36th International Conference on Neural Information Processing Systems* (cit. on p. 17).
- [47] Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2), 182–197. <https://doi.org/10.1109/4235.996017> (cit. on pp. 14, 55).
- [48] DeepSeek-AI, Liu, A., Feng, B., Xue, B., Wang, B., Wu, B., Lu, C., Zhao, C., Deng, C., Zhang, C., Ruan, C., Dai, D., Guo, D., Yang, D., Chen, D., Ji, D., Li, E., Lin, F., Dai, F., ... Pan, Z. (2025). DeepSeek-V3 technical report. *arXiv preprint arXiv:2412.19437*. <https://doi.org/10.48550/arXiv.2412.19437> (cit. on p. 19).
- [49] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). ImageNet: a large-scale hierarchical image database. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 248–255. <https://doi.org/10.1109/CVPR.2009.5206848> (cit. on pp. 36, 52, 56).

- [50] Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 4171–4186. <https://doi.org/10.18653/V1/N19-1423> (cit. on pp. 17–19, 35).
- [51] Dewey, J. (1934). *Art as experience*. Minton, Balch & company. (Cit. on pp. 1, 5).
- [52] Dodge, Y. (2008a). Binomial test. In *The concise encyclopedia of statistics* (pp. 47–49). Springer New York. https://doi.org/10.1007/978-0-387-32833-1_55 (cit. on p. 112).
- [53] Dodge, Y. (2008b). Chi-square goodness of fit tes. In *The concise encyclopedia of statistics* (pp. 72–76). Springer New York. https://doi.org/10.1007/978-0-387-32833-1_55 (cit. on p. 112).
- [54] Dormans, J. (2017). Cyclic Generation. In T. X. Short & T. Adams (Eds.), *Procedural Generation in Game Design* (pp. 83–96). Taylor & Francis, CRC Press. (Cit. on p. 106).
- [55] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2021). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *Proceedings of the International Conference on Learning Representations* (cit. on pp. 17, 20).
- [56] Drewes, F. (2006). *Grammatical picture generation*. Springer Berlin. <https://doi.org/10.1007-3-540-32507-7> (cit. on p. 36).
- [57] Droste, S., Jansen, T., & Wegener, I. (2002). On the analysis of the (1+ 1) evolutionary algorithm. *Theoretical Computer Science*, 276(1-2), 51–81. [https://doi.org/10.1016/S0304-3975\(01\)00182-7](https://doi.org/10.1016/S0304-3975(01)00182-7) (cit. on pp. 14, 122).
- [58] Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Yang, A., Fan, A., Goyal, A., Hartshorn, A., Yang, A., Mitra, A., Sravankumar, A., Korenev, A., Hinsvark, A., Rao, A., Zhang, A., ... Zhao, Z. (2024). The Llama 3 Herd of Models. *arXiv preprint arXiv:2407.21783* (cit. on pp. 18, 19, 97).
- [59] Duda, R. O., & Hart, P. E. (1973). *Pattern classification and scene analysis*. Wiley; Sons. (Cit. on p. 47).
- [60] Efros, A. A., & Freeman, W. T. (2001). Image quilting for texture synthesis and transfer. *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, 341–346. <https://doi.org/10.1145/383259.383296> (cit. on p. 40).

- [61] Elgammal, A., Liu, B., Elhoseiny, M., & Mazzone, M. (2017). CAN: Creative adversarial networks generating “art” by learning about styles and deviating from style norms. *Proceedings of the International Conference on Computational Creativity* (cit. on p. 39).
- [62] Esser, P., Rombach, R., & Ommer, B. (2021). Taming Transformers for High-Resolution Image Synthesis. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12868–12878. <https://doi.org/10.1109/CVPR46437.2021.01268> (cit. on pp. 20, 23, 37).
- [63] Fagroud, F. Z., Rachdi, M., & Ben Lahmar, E. H. (2022). Automatic Story Generation: A Case Study of English Children’s Story Generation Using GPT-2. *Proceedings of the International Conference on Digital Technologies and Applications*, 54–62. https://doi.org/10.1007/978-3-031-02447-4_6 (cit. on p. 18).
- [64] Fan, X., Dan, Z., Qiaomu, S., & Bo, T. (2025). A comprehensive survey on text-to-video generation. *Chinese Journal of Electronics*, 34(3), 1–27. <https://doi.org/10.23919/cje.2024.00.151> (cit. on p. 159).
- [65] Feng, B., Su, M., Zeng, K., & Li, X. (2024). Player-oriented procedural generation: Producing desired game content by natural language. *Proceedings of the International Conference on Human-Computer Interaction*, 260–274. https://doi.org/10.1007/978-3-031-60692-2_18 (cit. on p. 47).
- [66] Fonseca, C. M., & Fleming, P. J. (1995). An overview of evolutionary algorithms in multiobjective optimization. *Evolutionary Computation*, 3(1), 1–16. <https://doi.org/10.1162/evco.1995.3.1.1> (cit. on p. 14).
- [67] Fontaine, M. C., & Nikolaidis, S. (2021). Differentiable quality diversity. *Proceedings of the Neural Information Processing Systems Conference* (cit. on pp. 29, 87).
- [68] Fu, J., Wang, X., Vi, K., Li, Z., Xu, C., & Sun, Y. (2025). “I Like Your Story!”: A co-creative story-crafting game with a persona-driven character based on generative AI. *Proceedings of the CHI Conference on Human Factors in Computing Systems*. <https://doi.org/10.1145/3706599.3721163> (cit. on p. 45).
- [69] Gal, R., Patashnik, O., Maron, H., Bermano, A. H., Chechik, G., & Cohen-Or, D. (2022). StyleGAN-NADA: CLIP-Guided Domain Adaptation of Image Generators. *ACM Transactions on Graphics*, 41(4). <https://doi.org/10.1145/3528223.3530164> (cit. on p. 22).
- [70] Galanos, T., Liapis, A., Yannakakis, G. N., & Koenig, R. (2021). ARCH-Elites: Quality-Diversity for Urban Design. *Proceedings of the Genetic and Evolutionary Computation Conference*. <https://doi.org/10.1145/3449726.3459490> (cit. on p. 43).

- [71] Gallotta, R., Liapis, A., & Yannakakis, G. (2024). Consistent Game Content Creation via Function Calling for Large Language Models. *Proceedings of the 2024 IEEE Conference on Games (CoG)*. <https://doi.org/10.1109/CoG60054.2024.10645599> (cit. on pp. 46, 47).
- [72] Gallotta, R., Todd, G., Zammit, M., Earle, S., Liapis, A., Togelius, J., & Yannakakis, G. N. (2024). Large Language Models and Games: A Survey and Roadmap. *IEEE Transactions on Games, Early Access*. <https://doi.org/10.1109/TG.2024.3461510> (cit. on pp. 45, 91).
- [73] Gatys, L. A., Ecker, A. S., & Bethge, M. (2016). Image style transfer using convolutional neural networks. *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition*, 2414–2423. <https://doi.org/10.1109/CVPR.2016.265> (cit. on p. 40).
- [74] Gemini Team, Anil, R., Borgeaud, S., Alayrac, J.-B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A. M., Hauth, A., Millican, K., Silver, D., Johnson, M., Antonoglou, I., Schrittwieser, J., Glaese, A., Chen, J., Pitler, E., Lillicrap, T., Lazaridou, A., ... Vinyals, O. (2024). Gemini: A family of highly capable multimodal models. *arXiv preprint arXiv: 2312.11805*. <https://doi.org/10.48550/arXiv.2312.11805> (cit. on p. 43).
- [75] Gemma Team, Riviere, M., Pathak, S., Sessa, P. G., Hardin, C., Bhupatiraju, S., Hussenot, L., Mesnard, T., Shahriari, B., Ramé, A., Ferret, J., Liu, P., Tafti, P., Friesen, A., Casbon, M., Ramos, S., Kumar, R., Lan, C. L., Jerome, S., ... Andreev, A. (2024). Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv: 2408.00118*. <https://doi.org/10.48550/arXiv.2408.00118> (cit. on p. 97).
- [76] Gervás, P. (2023). Improving Efficiency and Coherence in Evolutionary Story Generation. *Proceedings of the International Conference on Computational Creativity* (cit. on p. 35).
- [77] Girdhar, R., El-Nouby, A., Liu, Z., Singh, M., Alwala, K. V., Joulin, A., & Misra, I. (2023). ImageBind: One embedding space to bind them all. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15180–15190. <https://doi.org/10.1109/CVPR52729.2023.01457> (cit. on pp. 43, 88).
- [78] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2020). Generative adversarial networks. *Communications of the ACM*, 63(11), 139–144. <https://doi.org/10.1145/3422622> (cit. on p. 37).

- [79] Gravina, D., Khalifa, A., Liapis, A., Togelius, J., & Yannakakis, G. N. (2019). Procedural content generation through Quality-Diversity. *Proceedings of the IEEE Conference on Games*. <https://doi.org/10.1109/CIG.2019.8848053> (cit. on pp. 26, 29, 33).
- [80] Green, M. C., Barros, G. A. B., Liapis, A., & Togelius, J. (2018). DATA agent. *Proceedings of the 13th Conference on the Foundations of Digital Games*. <https://doi.org/10.1145/3235765.3235792> (cit. on p. 44).
- [81] Grootendorst, M. (2022). BERTopic: Neural topic modeling with a class-based TF-IDF procedure. *arXiv preprint arXiv:2203.05794*. <https://doi.org/10.48550/arXiv.2203.05794> (cit. on p. 50).
- [82] Gunning, R. (1973). The technique of clear writing. McGraw-Hill Book Co. (Cit. on pp. 49, 73).
- [83] Guzhov, A., Raue, F., Hees, J., & Dengel, A. (2022). AudioCLIP: Extending CLIP to image, text and audio. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 976–980. <https://doi.org/10.1109/ICASSP43922.2022.9747631> (cit. on p. 42).
- [84] Härkönen, E., Hertzmann, A., Lehtinen, J., & Paris, S. (2020). GANSpace: Discovering interpretable GAN controls. *Proceedings of the 34th International Conference on Neural Information Processing Systems* (cit. on p. 40).
- [85] Hasler, D., & Suesstrunk, S. (2003a). Measuring colourfulness in natural images. *Proceedings of SPIE - The International Society for Optical Engineering*, 87–95. <https://doi.org/10.1117/12.477378> (cit. on pp. 48, 118).
- [86] Hasler, D., & Suesstrunk, S. (2003b). Measuring colourfulness in natural images. *Proceedings of IS&T/SPIE Electronic Imaging*. <https://doi.org/10.1117/12.477378> (cit. on p. 75).
- [87] Hendrycks, D., Mu, N., Cubuk, E. D., Zoph, B., Gilmer, J., & Lakshminarayanan, B. (2020). AugMix: A simple data processing method to improve robustness and uncertainty. *Proceedings of the International Conference on Learning Representations* (cit. on p. 74).
- [88] Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. *Proceedings of the 34th International Conference on Neural Information Processing Systems* (cit. on pp. 23, 26, 38).
- [89] Ho, J., & Salimans, T. (2021). Classifier-free diffusion guidance. *Proceedings of the NeurIPS Workshop on Deep Generative Models and Downstream Applications* (cit. on p. 24).

- [90] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735> (cit. on p. 16).
- [91] Holland, J. H. (1992a). *Adaptation in natural and artificial systems*. The MIT Press. <https://doi.org/https://doi.org/10.7551/mitpress/1090.001.0001> (cit. on pp. 12, 13).
- [92] Holland, J. H. (1992b). Genetic algorithms. *Scientific American*, 267(1), 66–73 (cit. on pp. 12, 13).
- [93] Hooshyar, D., Yousefi, M., Wang, M., & Lim, H. (2018). A data-driven procedural-content-generation approach for educational games. *Journal of Computer Assisted Learning*, 34(6), 731–739. <https://doi.org/10.1111/jcal.12280> (cit. on p. 43).
- [94] Hu, C., Zhao, Y., & Liu, J. (2024). Game generation via Large Language Models. *2024 IEEE Conference on Games (CoG)*. <https://doi.org/10.1109/CoG60054.2024.10645597> (cit. on p. 46).
- [95] Isola, P., Zhu, J.-Y., Zhou, T., & Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition*, 5967–5976. <https://doi.org/10.1109/CVPR.2017.632> (cit. on pp. 38, 41).
- [96] Jain, P. K., Quamer, W., Saravanan, V., & Pamula, R. (2023). Employing BERT-DCNN with Sentic Knowledge Base for Social Media Sentiment Analysis. *Journal of Ambient Intelligence and Humanized Computing*, 14(8), 10417–10429. <https://doi.org/10.1007/s12652-022-03698-z> (cit. on p. 18).
- [97] Jiang, A. Q., Sablayrolles, A., Roux, A., Mensch, A., Savary, B., Bamford, C., Chaplot, D. S., de las Casas, D., Hanna, E. B., Bressand, F., Lengyel, G., Bour, G., Lample, G., Lavaud, L. R., Saulnier, L., Lachaux, M.-A., Stock, P., Subramanian, S., Yang, S., ... Sayed, W. E. (2024). Mixtral of experts. *arXiv preprint arXiv:2401.04088*. <https://doi.org/10.48550/arXiv.2401.04088> (cit. on pp. 18, 19, 97).
- [98] Jing, H., Netzer, Y. D., Elhadad, M., & McKeown, K. R. (2000). Integrating a large-scale, reusable lexicon with a natural language generator. *Proceedings of the First International Conference on Natural Language Generation*, 209–216. <https://doi.org/10.3115/1118253.1118282> (cit. on p. 34).
- [99] Jordanous, A. (2012). A standardised procedure for evaluating creative systems: Computational creativity evaluation based on what it is to be creative. *Cognitive Computation*, 4(3), 246–279. <https://doi.org/10.1007/s12559-012-9156-1> (cit. on pp. 31, 32, 47).

- [100] Jordanous, A. (2016). Four PPPerspectives on computational creativity in theory and in practice. *Connection Science*, 28(2), 194–216. <https://doi.org/10.1080/09540091.2016.1151860> (cit. on p. 59).
- [101] Jordanous, A., & Keller, B. (2016). Modelling creativity: Identifying key components through a corpus-based approach. *PLOS ONE*, 11(10). <https://doi.org/10.1371/journal.pone.0162959> (cit. on p. 33).
- [102] Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., & Amodei, D. (2020). Scaling laws for neural language models. *arXiv preprint arXiv: 2001.08361*. <https://doi.org/10.48550/arXiv.2001.08361> (cit. on p. 18).
- [103] Karras, T., Aila, T., Laine, S., & Lehtinen, J. (2018). Progressive growing of GANs for improved quality, stability, and variation. *Proceedings of the 6th International Conference on Learning Representations 2018* (cit. on p. 37).
- [104] Karras, T., Aittala, M., Laine, S., & Aila, T. (2022). Elucidating the design space of diffusion-based generative models. *Proceedings of the 36th International Conference on Neural Information Processing Systems* (cit. on p. 24).
- [105] Karras, T., Laine, S., & Aila, T. (2021). A style-based generator architecture for generative adversarial networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(12), 4217–4228. <https://doi.org/10.1109/TPAMI.2020.2970919> (cit. on pp. 20, 37, 41).
- [106] Kelly, J., Calderwood, A., Wardrip-Fruin, N., & Mateas, M. (2023). There and back again: Extracting formal domains for controllable neurosymbolic story authoring. *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, 64–74. <https://doi.org/10.1609/aiide.v19i1.27502> (cit. on p. 46).
- [107] Khalifa, A., Bontrager, P., Earle, S., & Togelius, J. (2020). PCGRL: Procedural Content Generation via Reinforcement Learning. *Proceedings of the Sixteenth AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, 95–101. <https://doi.org/10.1609/aiide.v16i1.7416> (cit. on p. 44).
- [108] Kim, G., & Ye, J. C. (2021). DiffusionCLIP: text-guided image manipulation using diffusion models. *arXiv preprint arXiv:2110.02711*. <https://doi.org/10.48550/arXiv.2110.02711> (cit. on p. 22).
- [109] Kingma, D., & Welling, M. (2014). Auto-encoding variational bayes. *Proceedings of the International Conference on Learning Representations 2014* (cit. on pp. 19, 37).
- [110] Kocsis, L., & Szepesvári, C. (2006). Bandit based Monte-Carlo planning. *Proceedings of the European Conference on Machine Learning*. https://doi.org/10.1007/11871842_29 (cit. on pp. 29, 30, 76).

- [111] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444. <https://doi.org/10.1038/nature14539> (cit. on p. 16).
- [112] Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., & Kang, J. (2020). BioBERT: A pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4), 1234–1240. <https://doi.org/10.1093/bioinformatics/btz682> (cit. on p. 18).
- [113] Lehman, J., Gordon, J., Jain, S., Ndousse, K., Yeh, C., & Stanley, K. O. (2023). Evolution through large models. In *Genetic and evolutionary computation* (pp. 331–366). Springer Nature Singapore. https://doi.org/10.1007/978-981-99-3814-8_11 (cit. on p. 87).
- [114] Lehman, J., & Stanley, K. O. (2008). Exploiting open-endedness to solve problems through the search for novelty. *Proceedings of the Conference on Artificial Life* (cit. on pp. 5, 27).
- [115] Lehman, J., & Stanley, K. O. (2011). Evolving a diversity of virtual creatures through novelty search and local competition. *Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation*, 211–218. <https://doi.org/10.1145/2001576.2001606> (cit. on pp. 7, 27, 54).
- [116] Lehman, J., & Stanley, K. O. (2012). Beyond open-endedness: Quantifying impressiveness. *Proceedings of the Thirteenth International Conference on Artificial Life*. <https://doi.org/10.1162/978-0-262-31050-5-ch011> (cit. on p. 27).
- [117] Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., & Zettlemoyer, L. (2020). BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7871–7880. <https://doi.org/10.18653/v1/2020.acl-main.703> (cit. on p. 36).
- [118] Li, H., Yang, Y., Chang, M., Chen, S., Feng, H., Xu, Z., Li, Q., & Chen, Y. (2022). SRDiff: Single image super-resolution with diffusion probabilistic models. *Neurocomputing*, 479(C), 47–59. <https://doi.org/10.1016/j.neucom.2022.01.029> (cit. on p. 25).
- [119] Li, X., Lu, Z., Cai, D., Ma, X., & Xu, M. (2024). Large language models on mobile devices: Measurements, analysis, and insights. *Proceedings of the Workshop on Edge and Mobile Foundation Models*. <https://doi.org/10.1145/3662006.3662059> (cit. on p. 149).
- [120] Liapis, A. (2016). Exploring the visual styles of arcade game assets. *Proceedings of Evolutionary and Biologically Inspired Music, Sound, Art and Design*. https://doi.org/10.1007/978-3-319-31008-4_7 (cit. on p. 27).

- [121] Liapis, A. (2020). 10 years of the PCG workshop: Past and future trends. *Proceedings of the FDG Workshop on Procedural Content Generation*. <https://doi.org/10.1145/3402942.3409598> (cit. on p. 44).
- [122] Liapis, A., Martínez, H. P., Togelius, J., & Yannakakis, G. N. (2013). Transforming exploratory creativity with DeLeNoX. *Proceedings of the Fourth International Conference on Computational Creativity*, 56–63 (cit. on pp. 6, 27, 54).
- [123] Liapis, A., Yannakakis, G. N., Alexopoulos, C., & Lopes, P. (2016). Can computers foster human users' creativity? Theory and praxis of mixed-initiative co-creativity. *Digital Culture & Education (DCE)*, 8(2), 136–152 (cit. on pp. 5, 47).
- [124] Liapis, A., Yannakakis, G. N., Nelson, M. J., Preuss, M., & Bidarra, R. (2019). Orchestrating Game Generation. *IEEE Transactions on Games*, 11(1), 48–68. <https://doi.org/10.1109/TG.2018.2870876> (cit. on pp. 2, 43, 91, 93, 95).
- [125] Liapis, A., Yannakakis, G. N., & Togelius, J. (2013). Sentient world: Human-based procedural cartography. *Proceedings of Evolutionary and Biologically Inspired Music, Sound, Art and Design*, 180–191. https://doi.org/10.1007/978-3-642-36955-1_16 (cit. on p. 27).
- [126] Liapis, A., Yannakakis, G. N., & Togelius, J. (2015). Constrained novelty search: A study on game content generation. *Evolutionary Computation*, 23(1), 101–129. https://doi.org/10.1162/EVCO_a_00123 (cit. on p. 27).
- [127] Libbi, C. A., Trienes, J., Trieschnigg, D., & Seifert, C. (2021). Generating synthetic training data for supervised de-identification of electronic health records. *Future Internet*, 13(5), 136. <https://doi.org/10.3390/fi13050136> (cit. on p. 18).
- [128] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. *Proceedings of the ECCV 2014 European Conference on Computer Vision*, 740–755. https://doi.org/10.1007/978-3-319-10602-1_48 (cit. on p. 42).
- [129] Lindenmayer, A. (1968). Mathematical models for cellular interactions in development I. Filaments with one-sided inputs. *Journal of Theoretical Biology*, 18(3), 280–299. [https://doi.org/10.1016/0022-5193\(68\)90079-9](https://doi.org/10.1016/0022-5193(68)90079-9) (cit. on p. 36).
- [130] Liu, H., Chen, Z., Yuan, Y., Mei, X., Liu, X., Mandic, D., Wang, W., & Plumbley, M. D. (2023). AudioLDM: Text-to-audio generation with latent diffusion models. *Proceedings of the 40th International Conference on Machine Learning*, 21450–21474 (cit. on p. 159).
- [131] Liu, Z., Luo, P., Wang, X., & Tang, X. (2015). Deep learning face attributes in the wild. *Proceedings of the 2015 IEEE International Conference on Computer Vision*, 3730–3738. <https://doi.org/10.1109/ICCV.2015.425> (cit. on p. 36).

- [132] Lopes, P., Liapis, A., & Yannakakis, G. N. (2015). Sonancia: Sonification of procedurally generated game levels. *Proceedings of the ICCG workshop on Computational Creativity & Games* (cit. on p. 44).
- [133] Lu, Y.-J., Wang, Z.-Q., Watanabe, S., Richard, A., Yu, C., & Tsao, Y. (2022). Conditional diffusion probabilistic model for speech enhancement. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 7402–7406. <https://doi.org/10.1109/ICASSP43922.2022.9746901> (cit. on p. 25).
- [134] Machado, P., & Cardoso, A. (2002). All the truth about NEvAr. *Applied Intelligence*, 16(2), 101–118. <https://doi.org/10.1023/A:1013662402341> (cit. on p. 39).
- [135] Machado, P., Romero, J., Nadal, M., Santos-del-Riego, A., Correia, J., & Carballal, A. (2015). Computerized measures of visual complexity. *Acta Psychologica*, 160, 43–57. <https://doi.org/10.1016/j.actpsy.2015.06.005> (cit. on pp. 47, 56, 75, 118).
- [136] Maćkiewicz, A., & Ratajczak, W. (1993). Principal components analysis (PCA). *Computers & Geosciences*, 19(3), 303–342. [https://doi.org/10.1016/0098-3004\(93\)90090-R](https://doi.org/10.1016/0098-3004(93)90090-R) (cit. on p. 40).
- [137] Maesumi, A., Hu, D., Saripalli, K., Kim, V., Fisher, M., Pirk, S., & Ritchie, D. (2024). One Noise to Rule Them All: Learning a Unified Model of Spatially-Varying Noise Patterns. *ACM Transactions on Graphics*, 43(4). <https://doi.org/10.1145/3658195> (cit. on p. 36).
- [138] Mandelbrot, B. B. (1982). *The fractal geometry of nature*. W. H. Freeman and Company. (Cit. on p. 36).
- [139] Manurung, R., Ritchie, G., & Thompson, H. (2011). Using Genetic Algorithms to Create Meaningful Poetic Text. *Journal of Experimental & Theoretical Artificial Intelligence*, 24(1), 43–64. <https://doi.org/10.1080/0952813X.2010.539029> (cit. on p. 35).
- [140] Marcel, S., & Rodriguez, Y. (2010). Torchvision the machine-vision package of Torch. *Proceedings of the ACM International Conference on Multimedia*. <https://doi.org/10.1145/1873951.1874254> (cit. on p. 74).
- [141] Mateas, M., & Stern, A. (2021). Structuring Content in the Façade Interactive Drama Architecture. *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*. <https://doi.org/10.1609/aiide.v1i1.18722> (cit. on p. 35).
- [142] McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4), 115–133. <https://doi.org/10.1007/BF02478259> (cit. on p. 15).

- [143] McIntyre, N., & Lapata, M. (2010). Plot induction and evolutionary search for story generation. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, 1562–1572 (cit. on p. 35).
- [144] Meehan, J. R. (1977). TALE-SPIN, an Interactive Program that Writes Stories. *Proceedings of the 5th International Joint Conference on Artificial Intelligence*, 91–98 (cit. on p. 35).
- [145] Michalewicz, Z. (1995). Do not kill unfeasible individuals. *Proceedings of the 4th Intelligent Information Systems Workshop* (cit. on p. 73).
- [146] Mikolov, T., Karafiát, M., Burget, L., Černocký, J., & Khudanpur, S. (2010). Recurrent neural network based language model. *Proceedings of the Interspeech Conference*, 1045–1048. <https://doi.org/10.21437/Interspeech.2010-343> (cit. on p. 35).
- [147] Minaee, S., Mikolov, T., Nikzad, N., Chenaghlu, M., Socher, R., Amatriain, X., & Gao, J. (2024). Large language models: A survey. *arXiv preprint arXiv: 2402.06196*. <https://doi.org/10.48550/arXiv.2402.06196> (cit. on pp. 18, 46).
- [148] Mitra, A., Corro, L. D., Mahajan, S., Coda, A., Simoes, C., Agarwal, S., Chen, X., Razdaibiedina, A., Jones, E., Aggarwal, K., Palangi, H., Zheng, G., Rosset, C., Khanpour, H., & Awadallah, A. (2023). Orca 2: Teaching small language models how to reason. *arXiv preprint arXiv: 2311.11045*. <https://doi.org/10.48550/arXiv.2311.11045> (cit. on p. 97).
- [149] Montfort, N., Pérez y Pérez, R., Harrell, D. F., & Campana, A. (2013). Slant: A blackboard system to generate plot, figuration, and narrative discourse aspects of stories. *Proceedings of the Fourth International Conference on Computational Creativity*, 168–175 (cit. on p. 35).
- [150] Mouret, J.-B., & Clune, J. (n.d.). Illuminating search spaces by mapping elites. *arXiv preprint arXiv: 1504.04909* (cit. on pp. 5, 7, 27, 28, 68, 70, 79, 80, 87).
- [151] Nii, H. P. (1986). The Blackboard model of problem solving and the evolution of Blackboard architectures. *AI Magazine*, 7(2), 38–53. <https://doi.org/10.1609/aimag.v7i2.537> (cit. on p. 35).
- [152] Nilsback, M.-E., & Zisserman, A. (2008). Automated flower classification over a large number of classes. *Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing* (cit. on p. 42).
- [153] Ong, T., & Leggett, J. J. (2004). A genetic algorithm approach to interactive narrative generation. *Proceedings of the Fifteenth ACM Conference on Hypertext and Hypermedia*, 181–182. <https://doi.org/10.1145/1012807.1012856> (cit. on p. 35).

- [154] OpenAI. (2023). GPT-4 technical report. *arXiv preprint arXiv:2303.08774*. <https://doi.org/10.48550/arXiv.2303.08774> (cit. on pp. 2, 18, 19, 70, 92).
- [155] Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C. L., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., Schulman, J., Hilton, J., Kelton, F., Miller, L., Simens, M., Askell, A., Welinder, P., Christiano, P., Leike, J., & Lowe, R. (2022). Training language models to follow instructions with human feedback. *Proceedings of the 36th International Conference on Neural Information Processing Systems* (cit. on p. 18).
- [156] Papi, S., Gaido, M., Karakanta, A., Cettolo, M., Negri, M., & Turchi, M. (2023). Direct Speech Translation for Automatic Subtitling. *Transactions of the Association for Computational Linguistics*, 11, 1355–1376. https://doi.org/10.1162/tacl_a_00607 (cit. on p. 50).
- [157] Paul, C., & Lisbon, D. (2024). Harold cohen: Aaron [February 3–May 19, 2024]. <https://whitney.org/exhibitions/harold-cohen-aaron> (cit. on p. 38).
- [158] Peng, Y., Yan, S., & Lu, Z. (2019). Transfer learning in biomedical natural language processing: An evaluation of BERT and ELMo on ten benchmarking datasets. *Proceedings of the 18th BioNLP Workshop and Shared Task*, 58–65. <https://doi.org/10.18653/v1/W19-5006> (cit. on p. 18).
- [159] Pérez y Pérez, R., & Sharples, M. (2001). MEXICA: A computer model of a cognitive account of creative writing. *Journal of Experimental & Theoretical Artificial Intelligence*, 13(2), 119–139. <https://doi.org/10.1080/09528130010029820> (cit. on p. 35).
- [160] Perlin, K. (1985). An image synthesizer. *Proceedings of the 12th Annual Conference on Computer Graphics and Interactive Techniques*, 287–296. <https://doi.org/10.1145/325334.325247> (cit. on p. 36).
- [161] Plaat, A., van Duijn, M., van Stein, N., Preuss, M., van der Putten, P., & Batenburg, K. J. (2025). Agentic large language models, a survey. *arXiv preprint arXiv:2503.23037*. <https://doi.org/10.48550/arXiv.2503.23037> (cit. on p. 161).
- [162] Podell, D., English, Z., Lacey, K., Blattmann, A., Dockhorn, T., Müller, J., Penna, J., & Rombach, R. (2023). SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis. *arXiv preprint arXiv:2307.01952* (cit. on pp. 8, 38, 42, 100, 123).
- [163] Portilla, J., & Simoncelli, E. P. (2000). A parametric texture model based on joint statistics of complex wavelet coefficients. *International Journal of Computer Vision*, 40(1), 49–70. <https://doi.org/10.1023/A:1026553619983> (cit. on p. 40).

- [164] Pugh, J. K., Soros, L. B., & Stanley, K. O. (2016). An extended study of quality diversity algorithms. *Proceedings of the 2016 on Genetic and Evolutionary Computation Conference Companion*, 19–20. <https://doi.org/10.1145/2908961.2909000> (cit. on p. 26).
- [165] Qiu, S. (2023). Generative AI processes for 2d platformer game character design and animation. *Lecture Notes in Education Psychology and Public Media*, 29(1), 146–160. <https://doi.org/10.54254/2753-7048/29/20231440> (cit. on p. 45).
- [166] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., & Sutskever, I. (2021). Learning transferable visual models from natural language supervision. *Proceedings of the 38th International Conference on Machine Learning* (cit. on pp. 6, 21, 110, 118).
- [167] Radford, A., Metz, L., & Chintala, S. (2016). Unsupervised representation learning with deep convolutional generative adversarial networks. *Proceedings of the 4th International Conference on Learning Representations 2016* (cit. on pp. 19, 37).
- [168] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. (2019). Language models are unsupervised multitask learners [Accessed 11 Jan 2024]. (Cit. on pp. 17, 19, 35, 70, 71).
- [169] Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1), 5485–5551 (cit. on pp. 19, 36).
- [170] Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., & Sutskever, I. (2021). Zero-Shot Text-to-Image Generation. *Proceedings of the 38th International Conference on Machine Learning*, 8821–8831 (cit. on pp. 21, 42).
- [171] Reed, S., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., & Lee, H. (2016). Generative Adversarial Text to Image Synthesis. *Proceedings of The 33rd International Conference on Machine Learning*, 1060–1069 (cit. on pp. 19, 42).
- [172] Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, 3982–3992. <https://doi.org/10.18653/v1/D19-1410> (cit. on pp. 18, 50, 80).
- [173] Ritchie, G. (2007). Some Empirical Criteria for Attributing Creativity to a Computer Program. *Minds and Machines*, 17, 76–99 (cit. on pp. 2, 5, 7, 32, 52, 55, 73).
- [174] Rogers, C. (1951). *Client-centered therapy: Its current practice, implications and theory*. Houghton Mifflin. (Cit. on p. 34).

- [175] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., & Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. *Proceedings of the Conference on Computer Vision and Pattern Recognition* (cit. on pp. 2, 24, 38, 41, 42, 73, 159).
- [176] Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), 386–408. <https://doi.org/10.1037/h0042519> (cit. on p. 15).
- [177] Rossi, S., Rossi, M., Mukkamala, R. R., Thatcher, J. B., & Dwivedi, Y. K. (2024). Augmenting research methods with foundation models and generative AI. *International Journal of Information Management*, 77, 1–8. <https://doi.org/10.1016/j.ijinfomgt.2023.102749> (cit. on p. 11).
- [178] Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088), 533–536. <https://doi.org/10.1038/323533a0> (cit. on p. 15).
- [179] Saleh, B., & Elgammal, A. (2016). Large-scale classification of fine-art paintings: Learning the right metric on the right feature. *International Journal for Digital Art History*, 2, 70–93. <https://doi.org/10.11588/dah.2016.2.23376> (cit. on p. 36).
- [180] Scao, T. L., Fan, A., Akiki, C., Pavlick, E., Ilić, S., Hesslow, D., Castagné, R., Luccioni, A. S., Yvon, F., Gallé, M., Tow, J., Rush, A. M., Biderman, S., Webson, A., Ammanamanchi, P. S., Wang, T., Sagot, B., Muennighoff, N., del Moral, A. V., ... Wolf, T. (2023). BLOOM: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*. <https://doi.org/10.48550/arXiv.2211.05100> (cit. on pp. 19, 36).
- [181] Schaeffer, R., Miranda, B., & Koyejo, S. (2023). Are emergent abilities of Large Language Models a mirage? *Proceedings of the 37th International Conference on Neural Information Processing Systems* (cit. on p. 18).
- [182] Schaul, T. (2013). A video game description language for model-based or interactive learning. *Proceedings of the 2013 IEEE Conference on Computational Intelligence in Games*. <https://doi.org/10.1109/CIG.2013.6633610> (cit. on pp. 46, 160).
- [183] Schneider, F., Kamal, O., Jin, Z., & Schölkopf, B. (2024). Moûsai: Efficient text-to-music diffusion models. *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, 8050–8068. <https://doi.org/10.18653/v1/2024.acl-long.437> (cit. on p. 25).
- [184] Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., Schramowski, P., Kundurthy, S., Crowson, K., Schmidt, L., Kaczmarczyk, R., & Jitsev, J. (2022). LAION-5B: An open large-scale dataset for training next generation image-text models.

- Proceedings of the Conference on Neural Information Processing Systems*, 25278–25294 (cit. on p. 42).
- [185] Secretan, J., Beato, N., D Ambrosio, D. B., Rodriguez, A., Campbell, A., & Stanley, K. O. (2008). Picbreeder: Evolving pictures collaboratively online. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1759–1768. <https://doi.org/10.1145/1357054.1357328> (cit. on p. 39).
 - [186] Sfikas, K., Liapis, A., & Yannakakis, G. N. (2021). Monte Carlo elites: Quality-diversity selection as a multi-armed bandit problem. *Proceedings of the Genetic and Evolutionary Computation Conference*, 180–188. <https://doi.org/10.1145/3449639.3459321> (cit. on pp. 29, 30, 76).
 - [187] Shaker, N., Liapis, A., Togelius, J., Lopes, R., & Bidarra, R. (2016). Constructive generation methods for dungeons and levels. In *Procedural content generation in games: A textbook and an overview of current research* (pp. 31–55). Springer International Publishing. https://doi.org/10.1007/978-3-319-42716-4_3 (cit. on p. 44).
 - [188] Shaker, N., Togelius, J., & Nelson, M. J. (2016). Fractals, noise and agents with applications to landscapes. In *Procedural content generation in games: A textbook and an overview of current research* (pp. 57–72). Springer International Publishing. https://doi.org/10.1007/978-3-319-42716-4_4 (cit. on p. 36).
 - [189] Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27, 379–423 (cit. on pp. 34, 112).
 - [190] Simon, N. (2024). Does Robin Hood use a lightsaber?: Automated planning for storytelling. *Proceedings of the AAAI Conference on Artificial Intelligence*, 23421–23422. <https://doi.org/10.1609/aaai.v38i21.30411> (cit. on pp. 45, 46).
 - [191] Singh, A., Hu, R., Goswami, V., Couairon, G., Galuba, W., Rohrbach, M., & Kiela, D. (2022). FLAVA: A foundational language and vision alignment model. *Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15617–15629. <https://doi.org/10.1109/CVPR52688.2022.01519> (cit. on pp. 22, 110, 123).
 - [192] Sohl-Dickstein, J., Weiss, E. A., Maheswaranathan, N., & Ganguli, S. (2015). Deep unsupervised learning using nonequilibrium thermodynamics. *Proceedings of the 32nd International Conference on International Conference on Machine Learning*, 2256–2265 (cit. on p. 23).
 - [193] Stanley, K. O., & Miikkulainen, R. (2002). Evolving neural networks through augmenting topologies. *Evolutionary Computation*, 10, 99–127. <https://doi.org/10.1162/106365602320169811> (cit. on pp. 27, 40).

- [194] Stiennon, N., Ouyang, L., Wu, J., Ziegler, D. M., Lowe, R., Voss, C., Radford, A., Amodei, D., & Christiano, P. (2020). Learning to summarize from human feedback. *Proceedings of the 34th International Conference on Neural Information Processing Systems* (cit. on p. 18).
- [195] Sun, Q., Luo, Q., Ni, Y., & Mi, H. (2024). Text2AC: A framework for game-ready 2D agent character(AC) generation from natural language. *Proceedings of the CHI Conference on Human Factors in Computing Systems*. <https://doi.org/10.1145/3613905.3651049> (cit. on pp. 45, 46).
- [196] Sutskever, I., Martens, J., & Hinton, G. (2011). Generating text with recurrent neural networks. *Proceedings of the 28th International Conference on Machine Learning*, 1017–1024 (cit. on p. 16).
- [197] Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. *Proceedings of the 28th International Conference on Neural Information Processing Systems*, 3104–3112 (cit. on p. 35).
- [198] Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press. (Cit. on p. 44).
- [199] Tan, W. R., Chan, C. S., Aguirre, H., & Tanaka, K. (2019). Improved ArtGAN for Conditional Synthesis of Natural Image and Artwork. *IEEE Transactions on Image Processing*, 28(1), 394–409. <https://doi.org/10.1109/TIP.2018.2866698> (cit. on pp. 19, 37, 52, 54).
- [200] Togelius, J., Shaker, N., & Nelson, M. J. (2016). Introduction. In *Procedural content generation in games* (pp. 1–15). Springer International Publishing. https://doi.org/10.1007/978-3-319-42716-4_1 (cit. on pp. 15, 122).
- [201] Togelius, J., Yannakakis, G. N., Stanley, K. O., & Browne, C. (2010). Search-based procedural content generation. *Proceedings of the Conference on Applications of Evolutionary Computation*, 6024, 141–150. https://doi.org/10.1007/978-3-642-12239-2_15 (cit. on p. 11).
- [202] Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., & Lample, G. (2023). LLaMA: Open and Efficient Foundation Language Models. *arXiv preprint arXiv:2302.13971*. <https://doi.org/10.48550/arXiv.2302.13971> (cit. on pp. 18, 19, 70).
- [203] Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Ferrer, C. C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., ... Scialom, T. (2023). Llama 2: Open Foundation and Fine-Tuned Chat Models. *arXiv preprint*

- arXiv:2307.09288*. <https://doi.org/10.48550/arXiv.2307.09288> (cit. on pp. 18, 19, 70, 97).
- [204] Treanor, M., Schweizer, B., Bogost, I., & Mateas, M. (2012). The micro-rhetorics of Game-o-Matic. *Proceedings of the International Conference on the Foundations of Digital Games*, 18–25. <https://doi.org/10.1145/2282338.2282347> (cit. on p. 44).
 - [205] Turner, S. R. (1993). *MINSTREL: A computer model of creativity and storytelling* [Doctoral dissertation, University of California]. (Cit. on p. 35).
 - [206] Umbrello, S. (2021). AI winter. In *Encyclopedia of artificial intelligence*. ABC-CLIO. (Cit. on p. 15).
 - [207] van den Oord, A., Kalchbrenner, N., & Kavukcuoglu, K. (2016). Pixel recurrent neural networks. *Proceedings of The 33rd International Conference on Machine Learning*, 1747–1756 (cit. on p. 37).
 - [208] van den Oord, A., Kalchbrenner, N., Vinyals, O., Espeholt, L., Graves, A., & Kavukcuoglu, K. (2016). Conditional image generation with PixelCNN decoders. *Proceedings of the 30th International Conference on Neural Information Processing Systems*, 4797–4805 (cit. on p. 37).
 - [209] van den Oord, A., Vinyals, O., & Kavukcuoglu, K. (2017). Neural discrete representation learning. *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 6309–6318 (cit. on p. 20).
 - [210] Värtinen, S., Hämäläinen, P., & Guckelsberger, C. (2024). Generating role-playing game quests with GPT language models. *IEEE Transactions on Games*, 16(1), 127–139. <https://doi.org/10.1109/TG.2022.3228480> (cit. on p. 46).
 - [211] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is All You Need. *Proceedings of the Conference on Neural Information Processing Systems*, 5998–6008 (cit. on pp. 16, 35).
 - [212] Verma, P., & Chafe, C. (2021). A generative model for raw audio using transformer architectures. *2021 24th International Conference on Digital Audio Effects*, 230–237. <https://doi.org/10.23919/DAFx51585.2021.9768298> (cit. on p. 17).
 - [213] Viana, B. M. F., Pereira, L. T., & Toledo, C. F. M. (2022). Illuminating the space of enemies through MAP-Elites. *Proceedings of the IEEE Conference on Games*. <https://doi.org/10.1109/CoG51982.2022.9893621> (cit. on p. 29).
 - [214] Victory, S. (n.d.). Cyclic dungeon generation [Accessed 11 Oct 2024]. (Cit. on p. 106).
 - [215] Wah, C., Branson, S., Welinder, P., Perona, P., & Belongie, S. (2011). The Caltech-UCSD Birds-200-2011 dataset. (Cit. on p. 42).

- [216] Wang, Z., Bovik, A., Sheikh, H., & Simoncelli, E. (2004). Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4), 600–612. <https://doi.org/10.1109/TIP.2003.819861> (cit. on pp. 48, 80).
- [217] Wei, H., Bizzocchi, J., & Calvert, T. (2010). Time and space in digital game storytelling. *International Journal of Computer Games Technology*, 2010(1). <https://doi.org/10.1155/2010/897217> (cit. on p. 98).
- [218] Wei, J., Tay, Y., Bommasani, R., Raffel, C., Zoph, B., Borgeaud, S., Yogatama, D., Bosma, M., Zhou, D., Metzler, D., Chi, E. H., Hashimoto, T., Vinyals, O., Liang, P., Dean, J., & Fedus, W. (2022). Emergent abilities of Large Language Models. *arXiv preprint arXiv: 2206.07682*. <https://doi.org/10.48550/arXiv.2206.07682> (cit. on p. 18).
- [219] Weizenbaum, J. (1966). ELIZA - a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, 9(1), 36–45. <https://doi.org/10.1145/365153.365168> (cit. on p. 34).
- [220] Werbos, P. J. (1974). *Beyond regression: New tools for prediction and analysis in the behavioral sciences* [Doctoral dissertation, Harvard University]. (Cit. on p. 16).
- [221] West, P., Lu, X., Holtzman, A., Bhagavatula, C., Hwang, J. D., & Choi, Y. (2021). Reflective decoding: Beyond unidirectional generation with off-the-shelf language models. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*. <https://doi.org/10.18653/v1/2021.acl-long.114> (cit. on p. 87).
- [222] Wiggins, G. A. (2006). A preliminary framework for description, analysis and comparison of creative systems. *Knowledge-Based Systems*, 19(7), 449–458. <https://doi.org/10.1016/j.knosys.2006.04.009> (cit. on p. 32).
- [223] Wiher, G., Meister, C., & Cotterell, R. (2022). On decoding strategies for neural text generators. *Transactions of the Association for Computational Linguistics*, 10, 997–1012. https://doi.org/10.1162/tacl_a_00502 (cit. on p. 26).
- [224] Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6), 80–83. <https://doi.org/10.2307/3001968> (cit. on pp. 80, 127).
- [225] Winograd, T. (1972). Understanding natural language. *Cognitive Psychology*, 3(1), 1–191. [https://doi.org/10.1016/0010-0285\(72\)90002-3](https://doi.org/10.1016/0010-0285(72)90002-3) (cit. on p. 34).
- [226] Woods, W. A. (1970). Transition network grammars for natural language analysis. *Communications of the ACM*, 13(10), 591–606. <https://doi.org/10.1145/355598.362773> (cit. on p. 34).

- [227] Wu, B., Xu, C., Dai, X., Wan, A., Zhang, P., Tomizuka, M., Keutzer, K., & Vajda, P. (2020). Visual transformers: Token-based image representation and processing for computer vision. *arXiv preprint arXiv:2006.03677*. <https://doi.org/10.48550/arXiv.2006.03677> (cit. on p. 20).
- [228] Wu, Z., Chen, Z., Zhu, D., Mousas, C., & Kao, D. (2025). A systematic review of generative ai on game character creation: Applications, challenges, and future trends. *IEEE Transactions on Games, Early Access*. <https://doi.org/10.1109/TG.2025.3564869> (cit. on p. 92).
- [229] Xie, S., & Tu, Z. (2015). Holistically-nested edge detection. *Proceedings of the IEEE International Conference on Computer Vision*. <https://doi.org/10.1109/ICCV.2015.164> (cit. on pp. 47, 75).
- [230] Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., & Le, Q. V. (2019). XLNet: Generalized autoregressive pretraining for language understanding. *Proceedings of the 33rd International Conference on Neural Information Processing Systems* (cit. on p. 17).
- [231] Yannakakis, G. N., & Togelius, J. (2018). *Artificial intelligence and games*. Springer. (Cit. on p. 6).
- [232] Ye, H., Zhang, J., Liu, S., Han, X., & Yang, W. (2023). IP-Adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721* (cit. on pp. 25, 41, 92, 100).
- [233] Zammit, M., Liapis, A., & Yannakakis, G. N. (2022). Seeding Diversity into AI Art. *Proceedings of the International Conference on Computational Creativity* (cit. on pp. 54, 87).
- [234] Zammit, M., Liapis, A., & Yannakakis, G. N. (2024). CrawLLM: Theming Games with Large Language Models. *Proceedings of the IEEE Conference on Games (IEEE CoG) 2024* (cit. on p. 44).
- [235] Zarrieß, S., Voigt, H., & Schüz, S. (2021). Decoding methods in neural language generation: A survey. *Information*, 12(9). <https://doi.org/10.3390/info12090355> (cit. on p. 26).
- [236] Zellers, R., Holtzman, A., Rashkin, H., Bisk, Y., Farhadi, A., Roesner, F., & Choi, Y. (2019). Defending Against Neural Fake News. *Advances in Neural Information Processing Systems* 32, 9051–9062. <https://doi.org/10.5555/3454287.3455099> (cit. on p. 18).
- [237] Zhang, L., & Agrawala, M. (2024). Transparent image layer diffusion using latent transparency. *ACM Transactions on Graphics*, 43(4). <https://doi.org/10.1145/3658150> (cit. on p. 160).

- [238] Zhang, L., Rao, A., & Agrawala, M. (2023). Adding Conditional Control to Text-to-Image Diffusion Models. *arXiv preprint arXiv:2302.05543* (cit. on pp. 25, 41, 92, 100).
- [239] Zhang, R., Isola, P., Efros, A. A., Shechtman, E., & Wang, O. (2018). The unreasonable effectiveness of deep features as a perceptual metric. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 586–595 (cit. on pp. 49, 65, 80).
- [240] Zhang, S., Roller, S., Goyal, N., Artetxe, M., Chen, M., Chen, S., Dewan, C., Diab, M., Li, X., Lin, X. V., Mihaylov, T., Ott, M., Shleifer, S., Shuster, K., Simig, D., Koura, P. S., Sridhar, A., Wang, T., & Zettlemoyer, L. (2022). OPT: Open Pre-trained Transformer Language Models. *arXiv preprint arXiv:2205.01068*. <https://doi.org/10.48550/arXiv.2205.01068> (cit. on p. 19).
- [241] Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2020). BERTScore: Evaluating text generation with BERT. *International Conference on Learning Representations* (cit. on p. 50).
- [242] Zhang, Y., Sun, S., Galley, M., Chen, Y.-C., Brockett, C., Gao, X., Gao, J., Liu, J., & Dolan, B. (2020). DIALOGPT : Large-scale generative pre-training for conversational response generation. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 270–278. <https://doi.org/10.18653/v1/2020.acl-demos.30> (cit. on p. 36).
- [243] Zhu, J.-Y., Park, T., Isola, P., & Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. *Proceedings of the 2017 IEEE International Conference on Computer Vision*, 2242–2251. <https://doi.org/10.1109/ICCV.2017.244> (cit. on p. 41).