Towards the Evolution of Prompts with MetaPrompter

Tiago Martins, João M. Cunha(✉), João Correia, and Penousal Machado

CISUC, Department of Informatics Engineering, University of Coimbra, Coimbra, Portugal
{tiagofm,jmacunha,jncor,machado}@dei.uc.pt

Abstract. The dissemination of open-source text-to-image generative models and the increasing quality of their output has led to a growth in interest in the field. The quality of the images greatly depends on the prompt used, *i.e.* a phrase that includes descriptive terms to be used as input on text-to-image model. However, choosing the right prompt is a complex task, often relying on a trial-and-error approach. In this paper, we introduce an evolutionary approach to prompt generation where users begin by creating a blueprint for what might be a candidate prompt and then initiate an evolutionary process to interactively explore the space of prompts encoded by the initial blueprint and according to their preferences. Our work is a step towards a more dynamic and interactive way to generate prompts that lead to a wide variety of visual outputs, with which users can easily obtain prompts that match their goals.

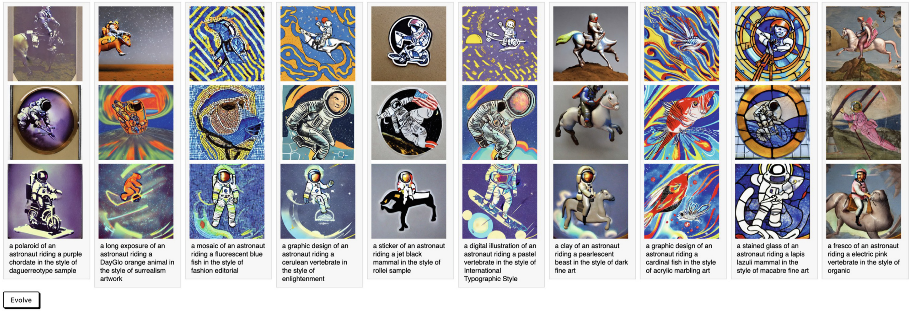
Keywords: Image Generation · Text-to-Image · Stable Diffusion · Interactive Evolutionary Computation

1 Introduction

In the past two years, we have witnessed a growing interest in text-to-image Artificial Intelligence (AI) systems caused by an increase in their performance and output quality. This wave of development can be linked to the appearance of multimodal models, such as Contrastive Language-Image Pre-Training (CLIP) [14]. CLIP is a contrastive language-visual model, trained on a dataset of 400 million text-image pairs collected from the internet. It results in the compression of two models at once (language and visual), establishing a connection between the two and allowing the estimation of semantic similarity between an image and a given text. While prior image generators were greatly limited to the classes of the datasets used in their training process (*e.g.* MS-COCO [7]), the introduction of these language-visual models enabled a larger scale text-to-image generation with few restrictions on what can be produced.

One of the first text-to-image generation models to use contrastive models is DALL-E [15]. However, the lack of public access triggered the interest in open-source alternative approaches, leading to the development of multiple openly

C. Johnson et al. (Eds.): EvoMUSART 2023, LNCS 13988, pp. 180–195, 2023.
https://doi.org/10.1007/978-3-031-29956-8_12



a polaroid of an astronaut riding a purple chordate in the style of daguerreotype sample

a long exposure of an astronaut riding a DayGlo orange animal in the style of surrealism artwork

a mosaic of an astronaut riding a fluorescent blue fish in the style of fashion editorial

a graphic design of an astronaut riding a cerulean vertebrate in the style of enlightenment

a sticker of an astronaut riding a jet black mammal in the style of rollei sample

a digital illustration of an astronaut riding a pastel vertebrate in the style of International Typographic Style

a clay of an astronaut riding a pearlescent beast in the style of dark fine art

a graphic design of an astronaut riding a cardinal fish in the style of acrylic marbling art

a stained glass of an astronaut riding a lapis lazuli mammal in the style of macabre fine art

a fresco of an astronaut riding a electric pink vertebrate in the style of organic

Fig. 1. Snapshot of a population of prompts evolved using the presented system. The images in each column are generated from different seeds using the same prompt, which is shown at the bottom.

available systems that also use CLIP for guiding image generation (*e.g.* Vector Quantized Generative Adversarial Network (VQGAN)-CLIP) [1], Stable Diffusion [17], *etc.*). These text-conditioned generative systems produce images from text inputs that are referred to as prompts. The choice of the words used in the prompt is key for producing images that match individual preferences and goals – adding specific keywords to the prompt can greatly improve results as reported by independent users (*e.g.* "unreal engine" improves images generated with VQGAN and CLIP[1]) and prior research studies [9,13]. However, the task of constructing effective prompts, known in the field of Natural Language Processing (NLP) as *prompt engineering* [16], has an open-ended nature and often consists of a highly experimental and iterative process [9]. This trial-and-error process often involves the use of prompt parts without a clear understanding of their impact on the results, leaving users with a sense of randomness [19].

The importance of prompts for producing high-quality images, both in terms of aesthetics and alignment with user goals, is reflected in the number of resources related to prompt engineering under development. One example that shows the value of prompt engineering expertise is the platform *PromptBase*,[2] which centres its business model on the monetisation of prompts, allowing users to buy and sell prompts for different generative models.

On the other hand, the growing accessibility of text-to-image generation models led to the emergence of communities, whose members (from artists to researchers) are devoted to the open-source development of these systems and to the creation and sharing of resources, such as guides (*e.g. A Traveler's Guide to the Latent Space* [18]). Researchers have focused on the development of computational approaches for the generation of prompts [13] and addressed the identification of design guidelines to write effective prompts [9,12].

In this paper, we aim to contribute to the set of resources that facilitate the creation of prompts. We propose a method where users *(i)* design a *meta prompt*

[1] https://twitter.com/arankomatsuzaki/status/1399471244760649729 accessed 2023.
[2] https://promptbase.com/ accessed 2023.

that encodes a space of different alternative prompts and then *(ii)* interactively evolve a population of prompts (see Fig. 1) that explores multiple points of this space according to their preferences. Our goal with the proposed method is to offer a more dynamic and interactive way to find and tune prompts with which users can obtain images that match their preferences.

Overall, the experimental results indicate the ability of the proposed method to evolve prompts that result in images that meet preferences expressed interactively by the user. Furthermore, we can generate a wide variety of visual outputs and also converge to prompts whose images share visual characteristics, namely their content and style.

The remainder of this paper is organised as follows: Sect. 2 summarises related work on prompt engineering, tools for prompt construction and prompt generation; Sect. 3 explains the proposed system; Sect. 4 describes the experimentation conducted to explore and analyse the possibilities created with the presented system and presents the obtained results; finally, Sect. 5 presents conclusions and directions for future work.

2 Related Work

Our work aims to facilitate the production of prompts with which users can obtain images that match their preferences. We present related work divided into three sections: prompt engineering, tools for prompt construction and generation.

2.1 Prompt Engineering

Prompt Engineering can be understood as the task of composing a textual instruction or description with the goal of obtaining a specific result from a language-based model [8]. Although the most recent interest in prompt engineering is related to text-to-image systems, the term has originated in relation to text-to-text systems [9] and researchers have investigated ways of formulating prompts for specific tasks (*e.g.* summarisation [8]).[3]

Regarding text-to-image systems, a prompt usually consists of a description of the image(s) that the user wishes to produce. Several authors have identified and studied different ways of structuring prompts (*prompt templates*), such as `[Subject] in the style of [Style]` [9], `[Medium] [Subject] [Artist(s)] [Details] [Image repository support]` [12] or `[Subject], by [Artist] (and [Artist]), [Modifier(s)],...` [18]. As observed in these templates, prompts can be divided into parts that can be ontologically categorised, from which the most important is considered to be the subject [9]. Other prompt parts are often referred to as *modifiers* and consist of words or phrases that are added to the prompt to alter the style or improve the quality of the results [11].

The discovery of new modifiers usually occurs in trial-and-error approaches and good results may lead to the widespread of modifiers in the community (*e.g.* “greg rutkowski” [6]). The identification of these modifiers has been a key

[3] See examples in https://beta.openai.com/examples accessed 2023.

task in the field, resulting in the development of multiple prompt guides by community members (*e.g.* the *Stable Diffusion Prompt Book*[4]) and modifier lists.[5]

Researchers have also devoted their attention to studying aspects of prompt engineering. Oppenlaender [12] describes an ethnographic study with online communities that resulted in the proposal of a prompt modifier taxonomy with six different types: subject terms, image prompts, style modifiers, quality boosters, repetitions, and magic terms. Liu and Chilton [9] conducted a study using nine configurations of the `[Subject] [Style]` template and propose a set of guidelines for prompt engineering – *e.g.* focus on subject and style instead of connecting words and generate between 3 to 9 seeds to get a representative set.

Some aspects of prompt engineering, *e.g.* weight assignment and repetition, despite being mentioned by multiple authors, have not yet been thoroughly studied in terms of their impact on results.

2.2 Tools for Prompt Construction

Given the difficulty of producing good-quality prompts, a number of tools have been implemented to aid in prompt engineering. An example is *Prompt Builder*,[6] which is described as a user-friendly tool that provides aid in constructing prompts and generating images. The user starts by selecting a model (*e.g.* Stable Diffusion) and then is presented with an interface for prompt building with multiple sections: *(i)* add an image prompt, *(ii)* add prompt parts, *(iii)* select a base image, *(iv)* add details and *(v)* mimic an art style or artist. Regarding prompt parts, the user is presented with an input field where a subject can be introduced but they also have the option to add more fields for extra prompt parts. Each field can be assigned a weight. The details section allows the user to select from different categories (*e.g.* "Art Medium") and sub-categories (*e.g.* "Drawing").

A common method used for producing prompts involves getting inspiration from prompts of already generated images, to identify keywords that may improve the prompts being created. There are multiple online platforms that allow users to explore generated images, for example, *Krea.ai*, *Lexica.art* and *OpenArt.ai*. In addition to these platforms, there are freely accessible datasets of prompts and generated images that can be used to investigate prompt engineering. An example is *Open Prompts* (used to build *krea.ai*), whose authors invite further research with it as an alternative to retrain models for quality improvement.[7] Another example is the work developed by Wang et al. [19], who present an open-source large-scale text-to-image prompt dataset (DIFFUSIONDB), containing 2 million images generated by Stable Diffusion. The authors suggest multiple applications for their dataset, including *prompt autocomplete* (*i.e.* keyword suggestion) and *prompt auto-replace* (*i.e.* exchanging prompt keywords for more effective ones). These functions can be considered as part of what we address in the following section: *prompt generation*.

[4] https://openart.ai/promptbook accessed 2023.

[5] https://proximacentaurib.notion.site/2b07d3195d5948c6a7e5836f9d535592 ac. 2023.

[6] https://promptomania.com/stable-diffusion-prompt-builder/ accessed 2023.

[7] https://github.com/krea-ai/open-prompts accessed 2023.

2.3 Prompt Generation

Several approaches have been explored for prompt generation. Although a great part of the work has been done within the context of text generation [8] and does not have the visual domain as a target [9], text generators can be useful for purposes of text-to-image generation. On the one hand, text generators can be used as sources of inspiration, aiding users in the production of prompts. This still holds even if they are not specifically developed for prompt generation (*e.g. drawingprompt.com*). On the other hand, text generation can be used specifically for the task of producing prompts for text-to-image generation, both directly and indirectly. An example of the latter is the work by Ge and Parikh [5]. The authors [5] implement a pipeline to generate prompts by using a Bidirectional Encoder Representations from Transformers (BERT) language model to predict masked words in templates (*e.g.* `the moon is like a [MASK]`), which are then used to establish an analogical relation between different concepts and produce a prompt. The prompt is used to produce visual conceptual blends [3] with BigSleep and DeepDaze. Regarding direct prompt production, an example is the *Stable Diffusion Prompt Generator*,[8] which is a GPT-2 model (Generative Pre-trained Transformer) that generates prompts from text input, trained with data retrieved from *Lexica.art* Stable Diffusion image repository.

Text generation can also be integrated as part of a bigger system. An example is the work by Liu et al. [10], who propose a system that brings together DALL-E, GPT-3 and CLIP within a computer-aided design software. The system uses 3D keywords sampled from a set of high-frequency words, styles and design parts generated with GPT-3 and keywords given by the user to produce text prompts, which are then used in DALL-E to obtain 3D designs.

Other approaches focus on prompt optimisation, for example by using nature-inspired algorithms. Pavlichenko and Ustalov [13] follow a human-in-the-loop approach and use a Genetic Algorithm (GA) to find the keyword set that produces the most aesthetically appealing images with Stable Diffusion, from a list of 100 keywords. They use 60 different image descriptions to produce prompts with the following template: `[keyword,...] [description] [keyword,...]`. They report that their keyword sets produce better results than the most popular keywords used by the community. A different approach is used in the system *EvoGen*[9], which combines an evolutionary algorithm, Stable Diffusion and an aesthetics model. An initial prompt population is randomly produced and evolved using the highest-rated prompts based on the aesthetic quality of the images generated with them. To produce prompts, they sample from different lists, *e.g.* artists and genres keywords, an English dictionary, among others.

Although our work aligns with some of the described work, *e.g.* by using nature-inspired computation, our approach differs from the ones described. Our goal is to facilitate the task of finding appropriate prompts for a given user. Instead of using metrics of aesthetics, we explore an interactive approach in which users can evolve prompts according to their preferences.

[8] https://huggingface.co/Gustavosta/MagicPrompt-Stable-Diffusion accessed 2023.

[9] https://github.com/MagnusPetersen/EvoGen-Prompt-Evolution accessed 2023.

3 Approach

The generation of images using a text-to-image generative model usually begins with the writing of a prompt that is then passed to the model to create images. Although there are several resources with numerous examples of tested prompts that anyone can modify and use, we believe that the process of finding and tuning prompts can be more dynamic and interactive. First, instead of dealing with prompts individually, which requires the selection and sequencing of individual terms and then the testing of each possibility by passing it to the model to generate images, we suggest a method where the user designs a blueprint for what might be a candidate prompt. As a result, rather than writing a specific prompt, the user writes a *meta prompt* which encodes a space of prompts. This allows variation using a set of terms given directly as input as well as terms automatically obtained through a method of conceptual extension [2] (*i.e.* from an initial term, *e.g.* "animal", obtaining others, *e.g.* "dog"). Second, instead of having the user manually tune the prompt, we propose an interactive approach in which an evolutionary system promotes solutions based on user feedback.

The presented method integrates two core modules. The first module takes as input a special type of prompt, which we call meta prompt, and translates it into a set of individual prompts. The second module provides an interactive way to explore and test the prompts produced by the first module. In the following subsections, we describe these two modules in more detail.

3.1 Creating Meta Prompts to Represent Spaces of Prompts

Our approach is based on a strategy in which the user does not input a specific prompt but a blueprint that can be used to produce multiple prompts. The creation of meta prompts is achieved using a syntax made especially for this task. To inform the design of a functional and flexible syntax, we analysed prompt examples retrieved from different sources and studied prompt taxonomies by other authors. Differently from other taxonomies, *e.g.* [12], we distinguish between *components* (*e.g. subject*) and *functions* (*e.g. repetition*). Moreover, we refer to different options of a given component as *terms* instead of the commonly used expression "modifiers", which we believe is not suitable for all components (*e.g.* different *subject* options are not exactly "modifiers"). The proposed syntax is based on four main functions that are described in the following paragraphs.

Define Meta Prompt Components — Like in any prompt or sentence, the creation of a meta prompt requires the definition of a structure or pattern consisting of a sequence of components, *e.g.* subject, verb and then object. Each component can contain one or more words. In our meta prompt syntax, a dynamic component can be identified by enclosing it inside a less-than sign (`<`) and a greater-than sign (`>`). For example, the meta prompt `<person> eating <fruit>` explicitly identifies two dynamic components and a static one (`eating`), which is directly printed to the final prompts. This process of identifying the components of the meta prompt is essential for the next function.

Enumerate Possible Terms for Each Meta Prompt Component — For each dynamic component defined in the meta prompt, we can enumerate options (terms) that can be selected and used to create prompt variations from the initial meta prompt. For example, the meta prompt `<farmer|policeman> eating <banana|kiwi|orange>` encodes two possible terms for the first dynamic component and three terms for the second one. By recombining these options, we can obtain six different prompts. This example illustrates the specification of different terms made directly in the meta prompt using vertical bars (`|`) to separate them. In addition to this method, we can link the dynamic component to an external list of terms. This method, which is illustrated in the example presented at the end of this subsection as lists `B`, `C` and `D`, not only facilitates the input of larger sets of terms but more importantly enables the dynamic creation and modification of such sets. For example, we can make use of existing sources or tools to retrieve related terms and use them as options for a dynamic component (*i.e.* conceptual extension [2]).

Combine Terms in a Prompt Component — In addition to specifying the set of terms that can be used in a given component, we can indicate that multiple terms can be combined. Specifically, we can set the number or an interval (minimum and maximum) of terms that should be selected and combined (the default number is 1). It is also possible to set the text that is used to join multiple terms (the default join text is a space). For example, the meta prompt `god eating <banana|kiwi|orange:1-2: and >` presents a dynamic component that has four possible terms and specifies the minimum and the maximum number of terms that can be used (1 and 2, respectively) as well as the text that should be used to join the selected terms (` and `). This information is indicated inside the component and is separated by colons (`:`). There is an extra option related to the possible repetition of the terms. By default, the system will try not to repeat the selected terms. However, we can make the system skip this check by inserting an asterisk (`*`) after the interval or number of terms.

Repeat Prompt Components — We can specify in the meta prompt the number of times that a given dynamic component should be repeated in the resulting prompts. This function will repeat the group of one or more terms selected for that dynamic component. For example, the meta prompt `astronaut riding a <<yellow|green>4> horse` may result in two possible prompts where the selected colour will be repeated 4 times. This function can be useful to give more weight to a given prompt component. Repetition and weight assignment are functions that we identified when analysing existing prompts, although there is scarce information on how they exactly work and impact the resulting images. Despite this, we considered that they should be included in our syntax.

With the proposed meta prompt syntax, we are not limited to any type of predefined grammar or sentence constructions. On the contrary, it is flexible by allowing the encoding of varied types of prompts, which in turn can spawn a vast set of alternative prompts.

To illustrate the functions explained above, we present a simplified example of a meta prompt below. The first text line is the meta prompt and the second shows the lists of terms that can be used to fill (replace) specific components of the meta prompt.

```
<A1|A2> of <B> with <C:1-2: and >, <D:2-5*:, >
B=[B1,B2]   C=[C1,C2,C3,C4]   D=[D1,D2,D3,D4,D5,D6]
```

From the meta prompt above, we can create several different prompts (over half a million prompts). Some examples of these prompts are presented below.

```
A1 of B1 with C3 and C2, D3, D4, D2, D3, D2
A2 of B2 with C4 and C2, D4, D2, D2, D2
A1 of B1 with C4 and C3, D3, D5
A2 of B2 with C3, D5, D4
A2 of B2 with C3 and C1, D3, D6, D2
A1 of B2 with C2, D1, D3
```

3.2 Exploring Spaces of Prompts in an Interactive Fashion

The range of individual prompts that can be generated from a meta prompt can easily grow and might result in a wide range of imagery. However, this also poses the challenge of finding the prompts that result in images that please the user. To facilitate this search process, we use an Interactive Evolutionary Computation (IEC) method, in particular an Interactive Genetic Algorithm (IGA), to evolve a population of prompts and this way interactively explore the space of prompts created from an input meta prompt.

Each individual in the population represents a prompt. Each prompt is encoded as a list of lists, where each inner list relates to a component specified in the meta prompt and stores integers representing indexes of selected terms for that component. Using these indexes, and applying the functions that may be specified in the meta prompt (see Sect. 3.1), we create each individual prompt.

The recombination of evolving prompts is achieved with a crossover operator which exchanges inner lists corresponding to the same component of the meta prompt. In the presented version of the approach, we use a two-point crossover operator. Regarding the mutation of prompts, we use an operator that is capable of deleting, replacing or/and inserting indexes in the inner lists. Each one of these procedures can occur independently with preset rates and according to the meta prompt configuration (*e.g.* minimum and maximum number of terms; or the possibility of repeating terms).

The initial population is seeded with random prompts. For each prompt in the population, a preset number of images is generated and displayed. In this process, we use fixed random seeds so that the first image of each prompt is created from a given seed, the second image of each prompt is created from another seed, and so on. This way, we can re-create any image produced during the evolutionary process. Furthermore, it facilitates the comparison of images generated with different evolved prompts.

The user takes a key role in the evolutionary process by looking at the images generated with the evolved prompts and selecting the preferred sets (prompts) to create the next generation, *i.e.* the fitness of the evolved prompts is determined by the user selection. The idea is that users can regard the population of prompts as a dynamic repository of images, and their prompts, and interactively generate variations of the preferred ones.

3.3 Implementation

One of our initial goals was to make the approach easy to use by anyone, ideally without the need for complicated technical configurations and installation of necessary dependencies on the computer for the approach to work. This way, we created a Google Colaboratory notebook that enables anyone to run the presented approach, which is implemented in Python, using a web browser. The source code of the project is publicly available.[10]

To get our approach running on a Colab notebook, we had to come up with a way to allow users to visualise the population of prompts being evolved, select the preferred ones, and ask the system to evolve the next generation of prompts, all this in a notebook. The result is a graphical interface implemented as a web page which is dynamically created and embedded in an output cell of the Colab notebook (see Fig. 1). Once each new generation of prompts is generated, the population is displayed to the user to select the preferred prompts and continue to the next generation. For each prompt in the population, we present the prompt string and a preset number of images generated using that prompt.

4 Experimentation

To assess the validity of the developed approach and its generative potential, we tested it under different conditions, divided into two experimental scenarios. For the tests hereon the system is deployed as an IEC system, *i.e.*, users guide evolution by selecting the individuals they like the most. As explained in the previous section, our system is designed to allow the input of external lists of terms to be used as options for the meta prompt components. In our experiments, we used two data sources. First, we produced a dataset of terms by collecting a total of 1,725 "modifiers" from different sources (*e.g.* lists of stable diffusion modifiers). Then, we removed the terms for which we had no class information and manually selected one class for those that had more than one, resulting in a dataset with a total of 1,237 terms, belonging to 25 different classes (*e.g. medium*, *material*, *style*, *etc.*). When we define the prompt component `<MEDIUM>`, we access the list of terms from the *medium* class (if it is given as input). Second, we implemented a method of conceptual extension that automatically retrieves related terms from the platform *relatedwords.org* using an initial input.

[10] The source code of the presented approach can be found at: https://cdv.dei.uc.pt/metaprompter.

4.1 Scenario 1: Study with Users

For a first experimental scenario, we designed and conducted a user study with the goal of assessing the potential of the approach. We used the platform *Drawing Prompt Generator* to generate random prompt-like phrases (*e.g.* "Naughty dog stealing a piece of pizza off the table"). From these phrases, we produced three different meta prompts (A, B and C):

```
a <MEDIUM> of a <COLOR> <ANIMAL>, exploring a pirate shipwreck
a <MEDIUM> of a <ANIMAL> stealing a piece of <VEGETABLE> off the table
a <ANIMAL> in their <HOUSE> in the style of <STYLE>
```

For these meta prompts, `<MEDIUM>`, `<COLOR>` and `<STYLE>` use terms from the produced dataset, while `<ANIMAL>`, `<VEGETABLE>` and `<HOUSE>` use terms obtained through conceptual extension.

We asked participants with background in graphic design to use the system to evolve prompts. In each run of the system, a set of three random seeds is produced – the seeds stay fixed and are used to generate the phenotype of each individual (composed of three images generated with the same prompt). As such, the individuals were always based on the same seeds, allowing image comparison. In each generation, the participant would identify the individual that they considered most aesthetically pleasing, select it and produce a new generation. We established a limit of ten generations. In the end, they would save the best individual and conduct the following tasks: (T1) identify a style shared by the three images; (T2) rate the ability of the system to evolve according to their taste from 1 (very bad) to 5 (very good); (T3) rate the aesthetic quality of the selected set of images from 1 (very bad) to 5 (very good); (T4) rate how well the selected set of images represent the corresponding prompt from 1 (very bad) to 5 (very good). The participants were also asked for comments.

For the IGA setup, we used the setting of Table 1, with the exception of population size (set to 6) and tournament size (set to 2).

Results

In total, ten users participated in the study, three with meta prompt A, three with B and four with C. All participants reached the tenth generation, except for two (one did an extra generation and one only reached the eighth). Regarding the tasks, one of the participants did not provide answers.

To assess if the system was able to converge using the interactive feedback of the users, we calculated the number of different words ($\neq$W) in the prompts of the individuals in each generation and the standard deviation (σ) of this value among generations. For all runs, $\sigma <= 1$ considering all generations and $\sigma <= 1.24$ considering only the first and the last (note that $\sigma < 0.86$ in all runs except one). As such, this shows that prompts had a constant word length and, consequently, a reduction of $\neq$W throughout the run would indicate a convergence. We observe a reduction in all runs – in 5 out of the 10 runs $\neq$W reduced to values between 52–54% of the value calculated for the initial population; in the other 5 runs, this percentage was higher but below 67%. Interestingly, the minimum of $\neq$W was achieved in one of the last two generations only in six of the runs; in the other four, the minimum $\neq$W was reached around generation 4–7, which suggests that the system converged but then the user preferences changed. This result is aligned with the comment provided by a participant, who reported having changed their style goal during the run.

Regarding T1, all participants indicated that they identify a shared style; some of the participants even identified a specific style, *e.g.* "crayon-like". For T2–T4, we calculated mode (mo), median ($\tilde{x}$), mean ($\overline{x}$) and standard deviation (σ), obtaining the following results: T2 (evolution) $mo = 2$, $\tilde{x} = 3$, $\overline{x} = 3$, $\sigma = 1$; T3 (aesthetics) $mo = 4$, $\tilde{x} = 4$, $\overline{x} = 3.88$, $\sigma = 0.78$; T4 (representation) $mo = 3$, $\tilde{x} = 3$, $\overline{x} = 2.88$, $\sigma = 1.166$. Regarding T2, the results indicate that the users do not fully perceive the system adaptation to their preferences. We believe this result may be related to the setup of the IGA for this experience, specifically a low tournament size (2), which does not foster selection for recombination and mutation of the individual(s) selected by the user as the best. Regarding the images produced, these were considered of good aesthetic quality (T3) but of only medium representation quality (T4), meaning that the images were not considered good representations of the prompt. This latter value may be related to the low number of inferences steps (10), which was chosen to reduce the time of the image generation but consequently has an impact on the connection between text and image.

4.2 Scenario 2: Variety and Convergence

In a second experimental scenario, the system is used to converge into a visual style. Thus, we perform a fixed number of generations and aim for convergence, obtaining a prompt or a set of prompts that produce images in a style that matches our preferences. We took into consideration the conclusions and user comments from Scenario 1 in the following experimentation with the system. The setup used to conduct this experiment can be viewed in Table 1.

Table 1. IGA setup parameters.

Parameter	Setting
population size	10
elite size	1
tournament size	5
crossover rate	0.7
delete term	0.1
replace term	0.25
insert term	0.1
generations	10
image size	512×512
inference steps	20
images per individual	3

For this experiment, we were inspired by a widely known *Openai*'s DALL-E prompt: "a photo of an astronaut riding a horse". From this prompt, we defined the following meta prompt and lists:

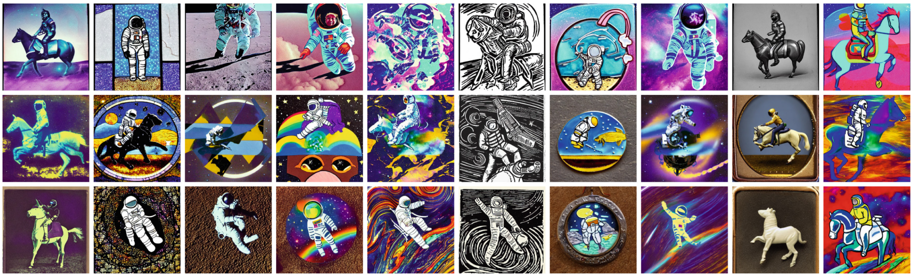
```
<M_of|SM_of|STYLE:0-1> astronaut riding a <COLOR:0-1> <horse|ANIMAL>
<of_S:0-1>
STYLE = ['3D',...]   MEDIUM = ['cartoon',...]   COLOR = ['cinnamon',...]
ANIMAL = get_related_terms('animal')
M_of = ['a {} of an'.format(m) for m in MEDIUM]
of_S = ['in the style of {}'.format(s) for s in STYLE]
SM_of = ['a {} {} of an'.format(s, m) for m in MEDIUM for s in STYLE]
```

The lists with terms used in the dynamic components of the meta prompt are composed as follows: MEDIUM contains 121 types of medium (*e.g.* acrylic painting, cartoon), ANIMAL contains 6 terms related with animals (*e.g.* mammal, vertebrate, fish), COLOR contains 309 different colours described with text (*e.g.* CMYK, cinnamon), and STYLE contains 228 different visual styles (*e.g.* fractal, acrylic artwork, pixel art). In total, the search space is composed of over 5 million possible prompts, among which is the initial *Openai*'s prompt.

Results

Figure 2 shows the initial population generated with the experimental settings shown in Table 1. It is possible to observe the diversity of prompts and images that are generated. As shown in Fig. 2, the dynamic nature of the meta prompt enables the generation of multiple prompts – all prompts are distinct. We can also observe that we get different results even with the same prompt when rendered with different random seeds.

The process is carried out with the interactive evaluation of the generated prompts and corresponding images. In this experiment, we aimed at convergence while picking individuals we considered fit. Note that we can select more than



a octane cyanotype of an astronaut riding a fluorescent purple horse

a pointillisim stained glass of an astronaut riding a jet black mammal

a ektachrome sample stop motion of an astronaut riding a chordate in the style of geometric abstraction

astronaut riding a rainbow beast

a nerikomi design multiple exposure of an astronaut riding a beast in the style of acrylic marbling art

a contour drawing linocut of an astronaut riding a firebrick animal in the style of Edwardian

astronaut riding a fish in the style of plique-a-jour enameling technique

a International Typographic Style long exposure of an astronaut riding a blue-violet omnivore in the style of deviantart

a isometric 3d ambrotype of an astronaut riding a pearlescent animal

astronaut riding a horse in the style of vibrant color scheme

Fig. 2. Prompts created in the initial generation. The images in each column are generated from different seeds using the same prompt, which is shown at the bottom.

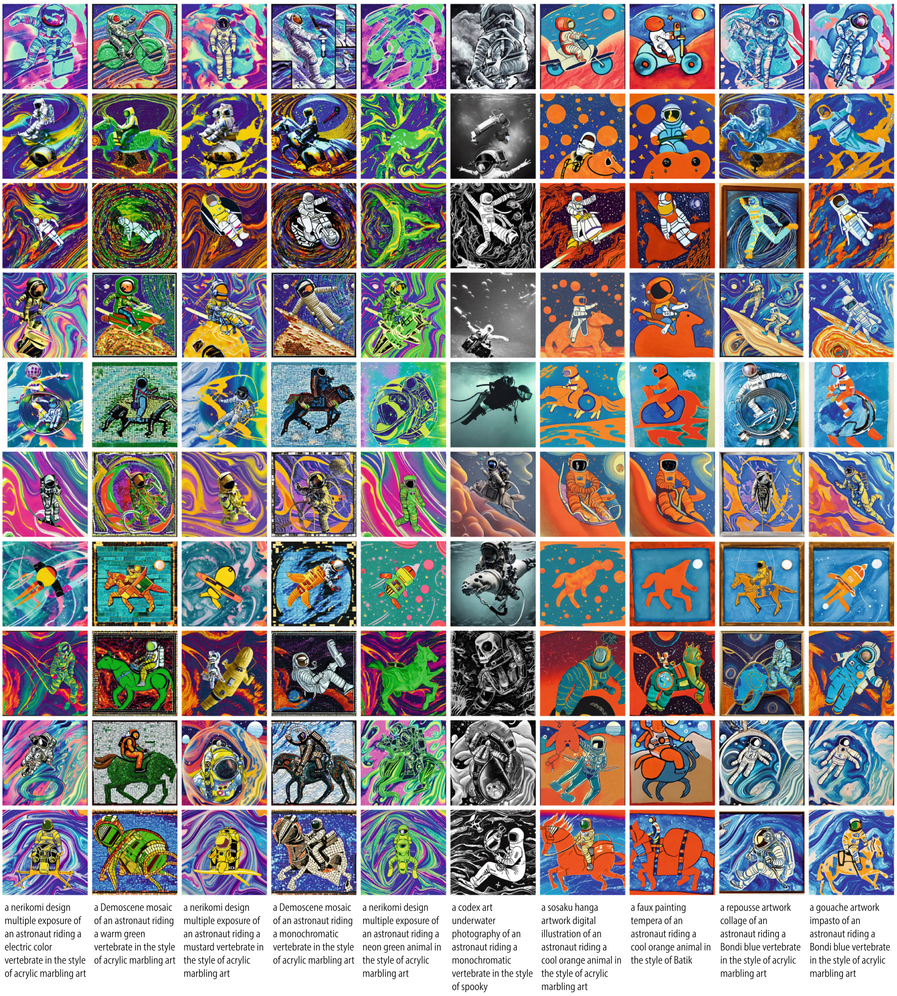

a nerikomi design multiple exposure of an astronaut riding a electric color vertebrate in the style of acrylic marbling art
a Demoscene mosaic of an astronaut riding a warm green vertebrate in the style of acrylic marbling art
a nerikomi design multiple exposure of an astronaut riding a mustard vertebrate in the style of acrylic marbling art
a Demoscene mosaic of an astronaut riding a monochromatic vertebrate in the style of acrylic marbling art
a nerikomi design multiple exposure of an astronaut riding a neon green animal in the style of acrylic marbling art
a codex art underwater photography of an astronaut riding a monochromatic vertebrate in the style of spooky
a sosaku hanga artwork digital illustration of an astronaut riding a cool orange animal in the style of acrylic marbling art
a faux painting tempera of an astronaut riding a cool orange animal in the style of Batik
a repousse artwork collage of an astronaut riding a Bondi blue vertebrate in the style of acrylic marbling art
a gouache artwork impasto of an astronaut riding a Bondi blue vertebrate in the style of acrylic marbling art

Fig. 3. Images generated from a population of evolved prompts. Each column of images is obtained from the same prompt, which is shown at the bottom. Each row of images is generated from the same random seed.

one individual per generation. After the evolutionary process, we can pick up and generate more images based on the evolved prompts via the modification of the random seed of the generator prior to the stable diffusion rendering process. In Fig. 3, we can see a large sample from the final population of prompts. During the evolutionary process, the user only sees a preset number of images per prompt. By observing the produced images at the end of the defined generations, one can see that images for each individual have the same overall style. Moreover, it is possible to see that the individuals of the population share visual characteristics, suggesting convergence. To further assess convergence, we analysed the prompts using the metric based on the number of different words ($\neq$w) used in Scenario 1. One difference that we observe is that a direct comparison of $\neq$w value does not suffice (59 in the first generation and 50 in the last), as there is a higher variation of prompt length – standard deviation of 2.76 considering all runs and 3.38 considering first and last. As such, we also calculated the percentage of words that are unique in the population prompts, obtaining 49.5% in the first generation and 31.4% in the last (more than 2/3 were repeated words), showing that we ended up with a converged population where the prompts have similar terms. Moreover, we note that there are still terms of the prompt present in the final population that were in the initial population. The results further the idea of guidance and convergence as well as support the validation and utility of the process of evolving prompts with multiple terms.

In summary, we were able to guide evolution towards a point of convergence in terms of style despite having different textual outputs. Visuals are distinct in some cases, but we can see the influence of the `<COLOR>` and `<STYLE>`, which conveys the idea of convergence in this experiment. From the set of images from Fig. 3 we can notice the impact of the random seeds, leaving a trail of common artefacts and objects across all prompts from the last generation (namely, in the 1st, 2nd, 3rd, 6th, 8th and 9th rows). If we ignore the colour, we can see similar styles and objects between individuals, even though the prompts differ. Overall, this showcases the ability of the approach to generate several visual outputs with prompts that can be diverse but convey the same visual traits.

5 Conclusions and Future Work

Recent development and increase in the output quality of text-to-image computational approaches have led to a growing interest in the field. Multiple communities have emerged and are dedicated to the development of open-source resources for text-to-image generative models, *e.g. Stable Diffusion*. One key aspect is the relation between the input prompt and the quality of the output. For this reason, there is a general venture to identify better prompts as well as better ways of conducting this search.

We presented an approach in which users can define a prompt blueprint (meta prompt), which is used to produce prompts, and interact with the system in order to produce solutions that match their preferences. We have tested our approach with two experimental scenarios: *(i)* a user study using three meta prompts, and *(ii)* a study using a meta prompt inspired by a widely known prompt example. The results show that our approach allows users to converge to specific styles, obtaining prompts that can be further used to produce images in the same style. It is important to mention that although there are differences between generative models and prompt syntax, obtaining results of different quality with the same prompt [19], our meta prompt approach is flexible and can be easily adapted to work with different models.

Our experimentation also allowed us to identify future research directions. First, we have mostly used terms that have been previously experimented with, being retrieved from other authors' works. However, there is potential to be explored in the identification of new terms that may lead to "hotspots" in the latent space. For example, Daras and Dimakis [4] investigate the existence of a "hidden vocabulary" in DALL-E-2 – apparently nonsensical prompts result in a given type of visual output (*e.g.* "apoploe vesrreaitais" is reported to produce birds). Second, there is still work to be done in identifying the best terms to use based on the prompt subject, *e.g.* 3DALL-E [10] uses terms specifically related to 3D modelling. Third, the system could be coupled with a prompt validator based on NLP approaches, aimed to analyse parts of speech and improve the quality of the prompts before generating the images. Additionally, the approach could also be further developed to also allow the evolution of meta prompts with the goal of finding the optimal configuration for each user.

The current approach has similarities with Grammatical Evolution (GE) approaches despite being a conventional IGA system. Therefore, the evolutionary engine can be enhanced by GE mechanisms, such as variation operators or initialisation methods and genotype-to-phenotype mapping approaches.

Another avenue of research is the automation of evaluation to explore a different dimension of the presented approach. Methods to automatically evaluate the generated images and the prompts could be used to build enhanced and automatic fitness function schemes, *e.g.*, use aesthetic evaluation models to evolve visually appealing images. Mechanisms to measure and improve the distinctness of generated outputs are also a hypothesis.

Our main goal is to facilitate the process of finding the best prompts, which is aligned with attempts of strengthening the relationship between the user and AI, fostering a collaborative interaction. In this sense, future developments can be made to the interface to improve this interaction and increase usability.

Acknowledgements. This research was partially funded by the FCT - Foundation for Science and Technology, I.P./MCTES through national funds (PIDDAC), within the scope of CISUC R&D Unit - UIDB/00326/2020 or project code UIDP/00326/2020.

References

1. Crowson, K., et al.: VQGAN-CLIP: open domain image generation and editing with natural language guidance. In: Avidan, S., Brostow, G., Farinella, G.M., Hassner, T. (eds.) Computer Vision – ECCV 2022. ECCV 2022. LNCS, vol. 13697, pp. 88–105. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-19836-6_6
2. Cunha, J.M.: Generation of concept representative symbols: towards visual conceptual blending, Ph. D. thesis, University of Coimbra (2022)
3. Cunha, J.M., Martins, P., Machado, P.: Let's figure this out: a roadmap for visual conceptual blending. In: Proceedings of the Eleventh International Conference on Computational Creativity (2020)
4. Daras, G., Dimakis, A.G.: Discovering the hidden vocabulary of DALLE-2. CoRR abs/2206.00169 (2022)
5. Ge, S., Parikh, D.: Visual conceptual blending with large-scale language and vision models. In: Proceedings of the 12th International Conference on Computational Creativity (2021)

6. Heikkila, M.: This artist is dominating AI-generated art. And he's not happy about it. https://www.technologyreview.com/2022/09/16/1059598/ (2022). Accessed Jan 2023
7. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
8. Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., Neubig, G.: Pre-train, prompt, and predict: a systematic survey of prompting methods in natural language processing. CoRR abs/2107.13586 (2021). https://arxiv.org/abs/2107.13586
9. Liu, V., Chilton, L.B.: Design guidelines for prompt engineering text-to-image generative models. In: Barbosa, S.D.J., Lampe, C., Appert, C., Shamma, D.A., Drucker, S.M., Williamson, J.R., Yatani, K. (eds.) CHI '22: CHI Conference on Human Factors in Computing Systems, New Orleans, LA, USA, 29 April 2022–5 May 2022, pp. 1–23. ACM (2022)
10. Liu, V., Vermeulen, J., Fitzmaurice, G., Matejka, J.: 3DALL-E: Integrating text-to-image AI in 3D design workflows. arXiv preprint arXiv:2210.11603 (2022)
11. Oppenlaender, J.: The creativity of text-to-image generation. In: 25th International Academic Mindtrek conference, Academic Mindtrek 2022, Tampere, Finland, 16–18 November 2022, pp. 192–202. ACM (2022)
12. Oppenlaender, J.: A taxonomy of prompt modifiers for text-to-image generation. arXiv preprint arXiv:2204.13988 (2022)
13. Pavlichenko, N., Ustalov, D.: Best prompts for text-to-image models and how to find them. CoRR abs/2209.11711 (2022)
14. Radford, A., et al.: Learning transferable visual models from natural language supervision. In: Meila, M., Zhang, T. (eds.) Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18–24 July 2021, Virtual Event. Proceedings of Machine Learning Research, vol. 139, pp. 8748–8763. PMLR (2021)
15. Ramesh, A., et al.: Zero-shot text-to-image generation. In: Meila, M., Zhang, T. (eds.) Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18–24 July 2021, Virtual Event. Proceedings of Machine Learning Research, vol. 139, pp. 8821–8831. PMLR (2021)
16. Reynolds, L., McDonell, K.: Prompt programming for large language models: beyond the few-shot paradigm. In: Kitamura, Y., Quigley, A., Isbister, K., Igarashi, T. (eds.) ACM CHI Conference on Human Factors in Computing Systems. ACM (2021)
17. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, 18–24 June 2022, pp. 10674–10685. IEEE (2022)
18. Smith, E.: A traveler's guide to the latent space. https://sweet-hall-e72.notion.site/A-Traveler-s-Guide-to-the-Latent-Space-85efba7e5e6a40e5bd3cae980f30235f (2022). Accessed Jan 2023
19. Wang, Z.J., Montoya, E., Munechika, D., Yang, H., Hoover, B., Chau, D.H.: DiffusionDB: a large-scale prompt gallery dataset for text-to-image generative models. CoRR abs/2210.14896 (2022)