Image Generation with Interactive Evolutionary System using Bayesian Optimization

1th Y. Dianey Rueda-Arango
Division of Education,
Universidad del Tolima,
Bogotá, CO
yenlhyn24@gmail.com

2st David Rojas-Velazquez
Division of Pharmacology,
Utrecht University,
Department of Data Science,
Julius Center for Health Sciences
and Primary Care, University Medical
Center Utrech
Utrecht, NL
e.d.rojasvelazquez@uu.nl

3st Aleksandra V. Gorelova
Division of Pharmacology,
Utrecht University,
Department of Data Science,
Julius Center for Health Sciences
and Primary Care, University Medical
Center Utrech
Utrecht, NL
A.v.gorelova@student.vu.nl

4th Johan Garssen
Division of Pharmacology,
Utrecht University,
Global Centre of Excellence
Immunology Danone
Nutricia Research
Utrecht, NL
J.Garssen@uu.nl

5th Alberto Tonda
UMR 518 MIAPS, INRAE,
Université Paris-Saclay,
Institut des Systèmes
Complexes de Paris
ÎledeFrance (ISC-PIF)
- UAR 3611 CNRS
Paris, FR
alberto.tonda@inrae.fr

6th Alejandro Lopez-Rincon
Division of Pharmacology,
Utrecht University,
Department of Data Science,
Julius Center for Health Sciences
and Primary Care, University Medical
Center Utrecht
Utrecht, NL
a.lopezrincon@uu.nl

2024 16th International Conference on Human System Interaction (HSI) | 979-8-3503-6291-6/24/$31.00 ©2024 IEEE | DOI: 10.1109/HSI61632.2024.10613596

***Abstract*—Interactive Evolutionary Systems (IES) can generate several designs based on a handful of input parameters. Nevertheless, the choice of the parameters is an open problem and it is limited to a few evaluations as they require human input. As a solution, Bayesian Optimization (BO) can be used to tune IES parameters. BO is a statistical method that efficiently models and optimizes expensive black-box derivative-free functions in few evaluations. In the context of creative IES, such as image generators, it can be used in conjunction with user preferences to optimize a complex-structured input space, such as variations of artistic images with uniqueness and creativity that follow the original concept and the artistic intention. Therefore, for this objective, we propose an implementation of BO-IES with a metric based on user preferences that interactively evaluates a batch of images to evolve a set of parameters in Stable Diffusion to create variations with a given human-made artwork. Our results proved better than baseline, and against generated images using Neural Style Transfer (NST). The resulting images were consistent in terms of uniqueness, quality, and following a given concept.**

***Index Terms*—Interactive Evolutionary Systems, Bayesian Optimization, Human-computer Interaction, Generative Art.**

I. Introduction

Interactive Evolutionary Systems (IES) can create large numbers of alternative designs [1] based typically on surrogate models controlled by a small set of parameters. After Dawkins created the program *Biomorphs* [2], where a series of designs evolved from a minimum set of parameters based on evolution by random mutation followed by nonrandom survival, the field has expanded to different applications based on user preferences, e.g. parametric tuning of facial characters [3], fashion design [4], and melody composition [5], to name a few. Nevertheless, IES generate new designs significantly faster than a human's ability to evaluate them, considering the capacity of human judgment is often constrained by fatigue [6]. Consequently, IES usually have to deal with relatively small populations and a limited number of generations. As a solution, Bayesian Optimization can assist users in finding favorable parameter settings in as few steps as possible [7]–[10].

Bayesian Optimization is an optimization method that uses a surrogate model with a Gaussian Process to predict where the global optima could be found based on an acquisition function characterized by the mean and the variance [11]. In contrast to other optimization methods, Bayesian Optimization is used where the main function is costly or difficult to evaluate, and thus only a limited set of evaluations can be done [12]. Bayesian Optimization has been used to optimize hyperparameters of vast deep learning models successfully, e.g. number of layers, inputs, learning rate, etc [12]. In addition, Bayesian Optimization has been used in different areas such as medicine [13], material design [14] and robotics [15], amongst others. However, standard Bayesian Optimization may not always be feasible for practical problems due to the absence of a straightforward metric to evaluate the cost function. Thus, for such complex systems as aesthetic evaluation in IES we can apply user preference as a metric or *human-in-the-*

loop techniques [16], [17]. These systems involve the user's preferences to select or to grade certain solutions sequentially as an alternative.

Bayesian Optimization has been applied in creative IES, e.g. in [7]. In this study, the system generates a set of animations. Through user input, it selects a set of parameters, choosing between a small number of options $2 \geq n \geq 4$, and treats each of the parameters independently. A similar approach in [5], shows $n = 4$ generated melodies, where the user selects the best one. Nevertheless, even experts may occasionally encounter difficulty in discerning a small number of alternatives based on preference. This problem is tackled in [18], where they consider incrementing options at each iteration for selecting coloring in artworks by giving an extra option. This idea is further extended in [8] by using *plane search* for visual design, where the user has to select an option from a grid $n = 9$ at each iteration to guide the selection of the parameters with as few evaluations from the user as possible.

These solutions do raise the question of how to evaluate the aesthetics of the designs, as mentioned in [6]. Over time, users' choices may increasingly prioritize novelty over quality. An alternative to this is to sample several individuals to generate a metric, but this loses the uniqueness of a generated artwork/design [6], [19]. As time passes, the task of choosing becomes increasingly challenging in comparison. Therefore, we propose selecting the parameters of an IES by deciding on how many generated designs in a fixed-size batch n are acceptable in terms of style and quality. For this, we use an image generator and optimize the parameters with Bayesian Optimization to reduce the number of interactive evaluations (BO-IES) as much as possible.

II. Background

A. Image Generation

Artificial Intelligence (AI) generated art has been a controversial topic in recent years in terms of authorship, copyright, and ethical issues [20]. A prominent advance in AI generative art was the use of Neural Style Transfer (NST). In an NST, a Convolutional Neural Network (CNN) can generate images by merging a reference image with a *style* [21], [22]. Nevertheless, NST came across as a combination of the image inputs and therefore loses the uniqueness expected in an artwork [20].

Nowadays, we have systems such as Stable Diffusion [23], Midjourney [24], and DALL-E 2 [25] that can generate hundreds of images from text prompts, with quality close to artworks and in different art styles [26]. However, studies such as *Hong and Curran* [27] do point out that human-created artworks are preferred by participants ($n = 288$) when compared against each other. Some artists have found AI image generators to be a valuable tool for creative expression, while others worry that these tools can convincingly replicate their style and produce multiple images in seconds, a feat impossible for a human to achieve [28]. However, AI image generators should be seen as a tool to enhance human creativity and enable artists to explore new possibilities, particularly alternatives to existing designs, improving the creative process [29].

Selecting the appropriate image generator system and adjusting its parameters to achieve the desired results is not a trivial task. In our study, we chose Stable Diffusion, which, compared to Midjourney and DALL-E, demonstrated better quantitative results generating faces [26]. Stable Diffusion is an open-source machine learning model that can generate images from text, modify images based on text, or fill in details on low-resolution or low-detail images. It has been trained on billions of images and can produce results that are comparable to DALL-E 2 and MidJourney [23]. The Stable Diffusion model was originally trained on 512x512 images from a subset of the LAION-5B database [30]. Our objective is to use a human-created artwork, employing Stable Diffusion as an image generator, in order to follow the artist's preferences and identify a set of optimal parameters that generate variations of the original artwork, ensuring no loss of uniqueness and quality.

B. Bayesian optimization

Bayesian Optimization (BO) is a technique for finding a global optimum in derivative-free black-box functions that are expensive to evaluate [31]. In contrast to other optimization techniques such as Evolutionary Algorithms (EA), it does not require as many evaluations of the cost function. BO has two components: firstly, the application of a Bayesian statistical model (Gaussian Process) to the data as a prior probability distribution to infer values. Secondly, an acquisition function is used to assess those values and determine the next point for evaluation within the original function [32], [33]. Additionally, sampling points with an acquisition function in the surrogate model helps find a *probable* optimum to reduce the number of function evaluations in the original cost function. The acquisition function is characterized by a covariance matrix (or kernel), μ, σ^2, and the information already gathered on the original model (observed samples) (x_0, y_0), (x_1, y_1), ... , (x_n, y_n) where $y_i = f(x_i)$, X is the vector of inputs, and y the vector of outputs [31].

As mentioned, the objective is to predict the next point x^* that maximizes the acquisition function in the surrogate model,

$$x^* = argmax\, f(x). \tag{1}$$

Using a Gaussian Process (GP) the predicted mean $\mu(x_*)$ and predicted variance $\sigma(x_*)$ of the point x_* are given by ([33], [34]):

$$\mu(x_*) = k_*^T (K + \sigma_{noise}^2 I)^{-1} y \tag{2}$$

$$\sigma^2(x_*) = k(x_*, x_*) - k_*^T (K + \sigma_{noise}^2 I)^{-1} k_*, \tag{3}$$

where $K = K(X, X)$ denotes the covariance matrix computed for each pair of observed inputs, σ_{noise}^2 is the noise level, I the identity matrix, k_* is the vector of covariances between the test point x_* and each of the n observed inputs. Examples of covariance functions include the exponential kernel (Eq. 4) and the Matérn $5/2$ (Eq. 5) [32]:

$$K_{sq-exp}(x, x') = \theta_0^2 exp(-\frac{1}{2}r^2) \quad (4)$$

$$K_{M52}(x, x') = \theta_0^2 exp(-\sqrt{5}r)(1 + \sqrt{5}r + \frac{5}{3}r^2) \quad (5)$$

with r given by:

$$r^2 = \sum_{d=1}^{D} \theta_d^2 (x_d - x'_d)^2 \quad (6)$$

with $\theta_d \in$ [0,1] (parameterized) [35]. The values of θ_i are estimated by the log marginal likelihood,

$$\log P(y|x, \theta) = -\frac{1}{2}\log|K| - \frac{1}{2}y^T K^{-1} y - \frac{N}{2}\log 2\pi. \quad (7)$$

which gives a measure of how well the model adjusts to the data [33]. The log marginal likelihood is usually calculated using gradient methods [34], such as the LBFGS-B [36].

There are different options for acquisition functions, such as Upper Confidence Bound (UCB) and Expected Improvement (EI) that select a sample point based on an *exploration-exploitation* scheme [32]. In addition, the acquisition function in itself needs to be optimized, for this we can use a gradient-based algorithm technique such as DIRECT in the BayesOpt library [37], or a global non-linear optimizer such as CMA-ES [38], [39] in the Limbo library [40]. With a given covariance function k, a set of inputs X, a set of outputs y, and a σ^2_{noise} noise level, we can optimize an acquisition function, e.g. UCB [35], [41]:

$$UCB(x) = \mu(x_*) + \kappa\sigma(x_*), \quad (8)$$

where κ is a parameter to tune the *exploration-exploitation* trade-off, a typical value is $\kappa = 1.96$ [42]. The overall algorithm is summarized in Alg. 1.

Algorithm 1: Bayesian Optimization algorithm.

Create n random initial points ;
for *Number of Evaluations* **do**
  Optimize $\log P(y|x, \theta)$;
  Optimize acquisition function ;
  Evaluate selected point x_* in the original $f(x)$;
  Update sets X, y ;

III. Methods

Our methodology consists of two modules, the Bayesian Optimization (BO) and the Interactive Evolutionary System (IES) with an image generator.

A. Bayesian Optimization

For our implementation of BO, we use a sequential scheme with UCB as the acquisition function, CMA-ES to find the optimum in the surrogate model, and LBFGS-B with explicit gradients to optimize the parameters of the log marginal likelihood with the Matérn $5/2$ kernel. This implementation has

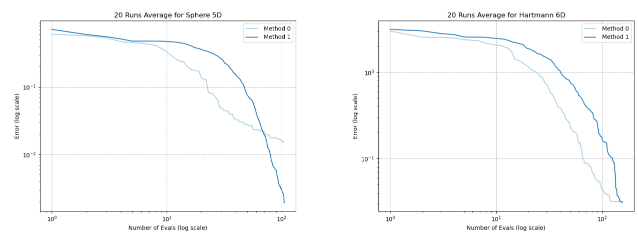

Fig. 1. Comparison between the implementation from [43] (Method 0) and our implementation (Method 1) for the Sphere 5-Dimensional function (Left) and Hartmann 6-Dimensional function (Right).

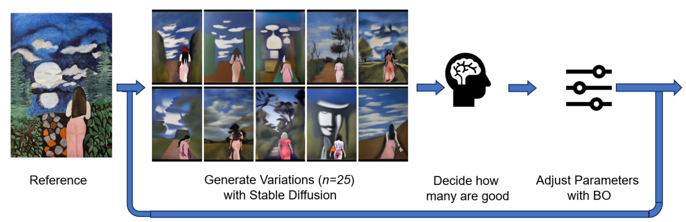

Fig. 2. Pipeline of BO-IES. Using a reference image, we create variations with the image generator with given parameters. Then the user decides based on preferences. Finally, BO adjusts the parameters and the process is repeated.

been proven successful in other applications with expensive cost functions [44].

As a reference, we compared our implementation with [43] for the Sphere 5-Dimensional function centered around a random point and the Hartmann 6-Dimensional function [45] (Fig. 1). In the case of the Sphere function, the difference is of an order of magnitude, whereas for Hartmann it is just slightly better when compared to 20 runs (Table I). Nevertheless, in both cases, the implementation from [43] was faster than ours.

TABLE I
RESULTS COMPARISON BETWEEN THE IMPLEMENTATION FROM [43] (METHOD 0) AND OUR IMPLEMENTATION (METHOD 1).

	Error Method 0	Error Method 1
Sphere 5D μ 20 Runs	0.0153	0.0019
Time Sphere (s)	20.1802	97.9991
Hartmann 6D μ 20 Runs	0.0316	0.0310
Time Hartmann (s)	60.5145	163.9318

B. Interactive Evolutionary System

For the image generation we use the NMKD Stable Diffusion GUI [46]. NMKD Stable Diffusion GUI simplifies the installation of Stable Diffusion as it comes with all necessary dependencies, and can run on a personal computer [46]. In the GUI, when using a prompt and a reference image, we have 3 parameters to optimize:

- *Initialization Image Strength (Influence)*, range 0.1-0.9,
- *Generation Steps*, range 5-120,
- *Prompt Guidance (CFG Scale)*, range 0-25.

Influence refers to how close the images are to the reference image. *Generation Steps* are the evaluations the inner system does to generate each image. *Prompt Guidance* is the effect of the text prompt in the generated images. Additionally, we fixed the *Amount of Images To Generate* to 25 based on preliminary testing. We selected *Influence*, *Generation Steps* and *Prompt Guidance* for optimization parameters as they have a direct effect on the generated images. Thus, at each evaluation, the user has to decide how many of the generated images ($n = 25$) follow the user's preferences in uniqueness and quality. A similar technique was explored in [8], [9], where the system suggests possible design options. Despite this, instead of choosing the best in a batch, we propose to quantify how many are in a batch and follow the user's preferences. This approach serves as an alternative to avoiding *overchoice*, a situation where selecting from a large array of similar options can complicate decision-making and lead to decision paralysis [47].

We start with an image created by an artist with a given concept. Then, we use the image generator (Stable Diffusion GUI) with the same concept as the prompt text and the reference image. At each evaluation step, we generate 25 images with a given set of parameters (*Influence*, *Generation Steps*, and *Prompt Guidance*). The user then decides which of the 25 generated images are of good quality, creative and capture the aesthetic design of the original artwork. Thus, the error is given by:

$$error = (n - \text{satisfactory images})/n \tag{9}$$

where $n = 25$. In this context, the error value is the result of the main cost function. After each interactive evaluation, we adjust the parameters with BO and generate a new batch for the user to evaluate (Alg. 2).

Algorithm 2: BO-IES.

Create n random initial batch of images with different parameters;
for *Number of Evaluations* **do**
  Optimize $\log P(y|x, \theta)$;
  Optimize acquisition function ;
  Generate Images with the best parameters ;
  Evaluate Images x_* by user preferences ;
  Calculate the error;
  Update sets X, y ;

IV. Results

A. Comparison to Baseline

We asked an artist to create an artwork based on the concept "*woman walking*" (Fig. 3). Then using the image generator, we use the same concept (*woman walking*) in the prompt text and generated several batches ($n = 25$) with random parameters (*Influence*, *Generation Steps*, and *Prompt Guidance*) to use as a baseline comparison. Next, we ran the BO-IES, with 3 starting sets of parameters, $n = 25$, and used the same reference image and prompt text for 50 interactive evaluations. For each evaluation, the set of parameters was selected using BO. The summary of the pipeline is in Fig. 2.

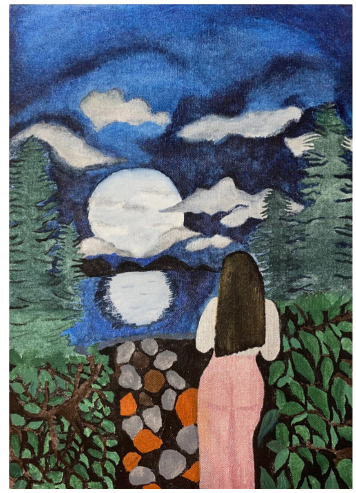

Fig. 3. Original painting made by the artist, following the concept *woman walking*.

For 50 evaluations, this generated 1,250 images in total. Nevertheless, the average evaluation time for each batch of images was approximately 47 seconds, while choosing the best image could take up to two minutes in preliminary testing, as expected from *overchoice*. We calculated the error by the results given by the user's preferences (Fig. 4). BO-IES gave better results than the baseline. The best set of parameters were:

- 0.6 for *Influence*
- 115 for *Generation Steps*
- 21.5 for *Prompt Guidance*

The best-calculated error is 0.24, which means that the user liked 19 out of the 25 generated images at evaluation step 45. An example of the generated images, using BO-IES at different evaluation steps are in the following figures: Step 1 (Fig. 5), Step 20 (Fig. 6), Step 30 (Fig. 7), and Step 45 (Fig. 8).

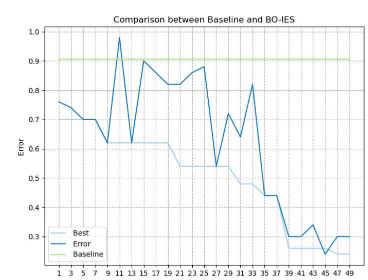

Fig. 4. Comparison between baseline and BO-IES, for 50 evaluations.

As expected, not all the images generated were considered good or bad solely based on quality by the artist. We did notice that when the parameter *Influence* is set too high there is insufficient space to generate a diverse set of images. Consequently, errors increased as creativity diminished in each

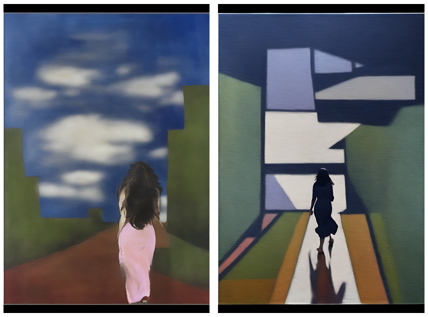

Fig. 5. Sample of Generated Images at Evaluation Step 1.

Fig. 6. Sample of Generated Images at Evaluation Step 20. It is interesting to mention that for an unknown reason, in several images a purse was added to the generated images by the system.

Fig. 7. Sample of Generated Images at Evaluation Step 30.

Fig. 8. Sample of Generated Images at Evaluation Step 45 (best Solution).

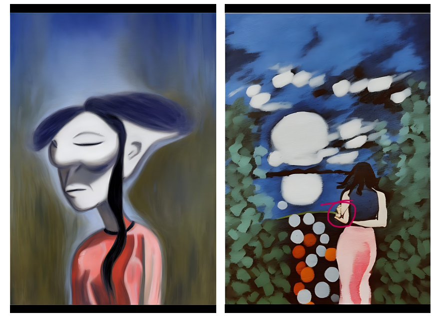

Fig. 9. Comparison of the considered worst visual error at generation 1 (left) and at generation 45 (right) based on quality. Do notice that at evaluation Step 45, the error is only a partial error.

of the variations. In contrast, when *Influence* is set too low, the resulting image generated was no longer a variation of the reference image. This is in line with [6], [19] that mention that uniqueness plays an important role in creative IES.

In addition, sometimes there were partial and complete errors in the image generator. We considered a partial error just a small non-logical detail in the generated image, e.g. a non-proportioned hand. In comparison, a complete error was when the generated image had no connection with the text prompt or the reference figure. Furthermore, as the system evolved, the errors were smaller. A comparison between what the user considered the worst visual errors at first generation and at best generation is in Fig. 9. While in the first generation, the generated image had no connection with the reference image or the text prompt, and only the colors had a connection, the worst visual error in the best evaluation step is a partial error related to the position of the arms and hands.

B. Comparison to Neural Style Transfer

As mentioned in [6], the aesthetic evaluation of AI-generated art involves the quality and uniqueness of an art-work. Thus, as a comparison, we generated a batch ($n = 25$) using an NST system with different art styles including art-works from different artists such as Vincent van Gogh, Gustav Klimt, Claude Monet, Rembrandt, Giovanni Boldini, among others. Then, we combined the reference image (Fig.3) with the different styles.

We used the implementation from [48], for 1,250 iterations for each different style with default parameters, and LBFGS as a solver. Then, we asked the user to evaluate the images, as an evaluation step in the BO-IES ($n = 25$). The result was an error of 0.48, 13 out of 25 were acceptable by the user in terms of quality and uniqueness (Fig. 10). There were no visual errors, like the ones generated by Stable Diffusion (Fig. 9). This is explained that NST only transfers the style to an existing reference. However, in some images, the aesthetic quality decreased (Fig. 11). In contrast, as expected [20], some images, although with different styles, seem to lack uniqueness. For example, in Fig. 12 the images were done with different styles, but the result is visually close with just some tones changed. The decrease in the quality and uniqueness of the generated images are explained by the selected *style*.

Fig. 10. Sample of selected generated images with NST, using the style of Giovanni Boldini (left) and the style of Gustav Klimt (right).

Fig. 11. Sample of non-selected generated images with NST, using the style of Anthony van Dyck (left) and the style of Rembrandt (right) based on aesthetic quality.

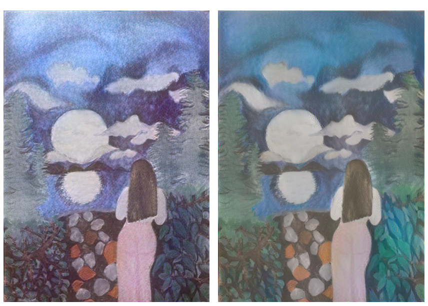

Fig. 12. Sample of non-selected generated images with NST, using the style of Zdzisław Beksiński (left) and the style of Marie Laurencin (right) based on uniqueness.

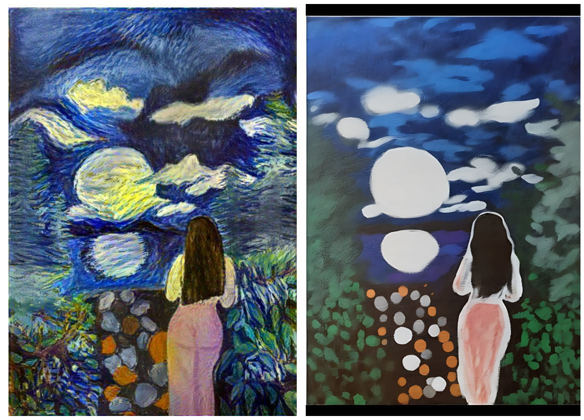

Fig. 13. Comparison between the best-selected images by the user. Left, using NST with the style of Vincent van Gogh. Right, from the batch of the best solution BO-IES at $step = 45$.

V. Conclusions and Discussion

In this study, we used Bayesian Optimization to find the best parameters in an Interactive Evolutionary System (BO-IES) to generate images. The generated images were variations of a reference image done by an artist. Then, the artist interactively evaluated the system by preference, deciding how many images were acceptable in each batch from an image generator, until most images reflected the original idea, highlighting creativity and quality.

This study works as a proof of concept: although the results are promising against baseline and NST, further testing is necessary to validate the methodology with other artworks, users, and different settings. Furthermore, it is essential to compare the results with other algorithms, e.g. GANs (Generative Adversarial Networks) [49]. In contrast to other studies, we propose a metric focusing on how many in a batch are acceptable in a *human-in-the-loop* scheme, rather than selecting the best option. In preliminary testing, our results showed this could help with the problem of *overchoice* [47]. In addition, we propose a Bayesian Optimization (BO) implementation aimed at reducing the number of evaluations required by the user, which has proven successful in standard benchmarks.

The intersection of AI and art prompts philosophical questions. Can algorithms truly create art? How does human intervention shape the creative process? These inquiries challenge our understanding of art's value in the digital era. The relationship between technology and artistic expression generates profound approaches to innovation and the evolution of an artistic work. For example, if we compare the original artwork made by an artist (Fig. 3) in contrast to AI generated by NST or BO-IES (Fig. 13), we can see that all three images are similar in figures, colors, positions. However, there is a slight difference in the style, where deciding which one is better is a matter of personal preference.

In summary, AI-generated art presents both creative possibilities and ethical dilemmas. The use of AI to create art could be a novel approach to take artwork to the next level. The possibility of viewing the same design in multiple images offers a different perspective of the original artwork uniquely. This blends art and reinterprets the artistic styles of the original artwork. Artificial intelligence can also generate original proposals, reflections, and expand the creativity of new ideas in which an artwork can be created, thereby evolving the art form. It is fascinating to observe how several images can be created from a sole painting based on the preferences of the person selecting them, improving them by creating visually appealing images based on their tastes. The generated image is based on the original artwork, yet it is unique and created with quality, following what we would expect in art preferences [6].

References

[1] Y. Li and C.-J. Hu, "Aesthetic learning in an interactive evolutionary art system," in *Applications of Evolutionary Computation: EvoApplications 2010: EvoCOMNET, EvoENVIRONMENT, EvoFIN, EvoMUSART, and EvoTRANSLOG, Istanbul, Turkey, April 7-9, 2010, Proceedings, Part II*, pp. 301–310, Springer, 2010.

[2] R. Dawkins, "The evolution of evolvability," in *Artificial life*, pp. 201–220, Routledge, 1989.

[3] Z. Gu, M. X. Tang, and J. H. Frazer, "Capturing aesthetic intention during interactive evolution," *Computer-Aided Design*, vol. 38, no. 3, pp. 224–237, 2006.

[4] S.-B. Cho, "Towards creative evolutionary systems with interactive genetic algorithm," *Applied Intelligence*, vol. 16, pp. 129–138, 2002.

[5] Y. Zhou, Y. Koyama, M. Goto, and T. Igarashi, "Generative melody composition with human-in-the-loop bayesian optimization," *arXiv preprint arXiv:2010.03190*, 2020.

[6] P. Galanter, "The problem with evolutionary art is...," in *European Conference on the Applications of Evolutionary Computation*, pp. 321–330, Springer, 2010.

[7] E. Brochu, T. Brochu, and N. De Freitas, "A bayesian interactive optimization approach to procedural animation design," in *Proceedings of the 2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pp. 103–112, 2010.

[8] Y. Koyama, I. Sato, and M. Goto, "Sequential gallery for interactive visual design optimization," *ACM Transactions on Graphics (TOG)*, vol. 39, no. 4, pp. 88–1, 2020.

[9] Y. Koyama and M. Goto, "Bo as assistant: Using bayesian optimization for asynchronously generating design suggestions," in *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology*, pp. 1–14, 2022.

[10] H. L. Admin, "Parameter selection for generative art with bayesian optimisation," Dec 2020.

[11] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*, vol. 2. MIT press Cambridge, MA, 2006.

[12] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. De Freitas, "Taking the human out of the loop: A review of bayesian optimization," *Proceedings of the IEEE*, vol. 104, no. 1, pp. 148–175, 2015.

[13] R. J. Borgli, H. K. Stensland, M. A. Riegler, and P. Halvorsen, "Automatic hyperparameter optimization for transfer learning on medical image datasets using bayesian optimization," in *2019 13th International Symposium on Medical Information and Communication Technology (ISMICT)*, pp. 1–6, IEEE, 2019.

[14] P. I. Frazier and J. Wang, "Bayesian optimization for materials design," in *Information science for materials discovery and design*, pp. 45–75, Springer, 2015.

[15] F. Berkenkamp, A. Krause, and A. P. Schoellig, "Bayesian optimization with safety constraints: safe and automatic parameter tuning in robotics," *Machine Learning*, vol. 112, no. 10, pp. 3713–3747, 2023.

[16] W. Chu and Z. Ghahramani, "Preference learning with gaussian processes," in *Proceedings of the 22nd international conference on Machine learning*, pp. 137–144, 2005.

[17] J. González, Z. Dai, A. Damianou, and N. D. Lawrence, "Preferential bayesian optimization," in *International Conference on Machine Learning*, pp. 1282–1291, PMLR, 2017.

[18] M. McCourt and I. Dewancker, "Sampling humans for optimizing preferences in coloring artwork," *arXiv preprint arXiv:1906.03813*, 2019.

[19] P. Galanter, "Computational aesthetic evaluation: past and future," *Computers and creativity*, pp. 255–293, 2012.

[20] E. Cetinic and J. She, "Understanding and creating art with ai: Review and outlook," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 18, no. 2, pp. 1–22, 2022.

[21] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2414–2423, 2016.

[22] Y. Jing, Y. Yang, Z. Feng, J. Ye, Y. Yu, and M. Song, "Neural style transfer: A review," *IEEE transactions on visualization and computer graphics*, vol. 26, no. 11, pp. 3365–3385, 2019.

[23] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

[24] D. Holz.

[25] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents," *arXiv preprint arXiv:2204.06125*, vol. 1, no. 2, p. 3, 2022.

[26] A. Borji, "Generated faces in the wild: Quantitative comparison of stable diffusion, midjourney and dall-e 2," *arXiv preprint arXiv:2210.00586*, 2022.

[27] J.-W. Hong and N. M. Curran, "Artificial intelligence, artists, and art: attitudes toward artwork produced by humans vs. artificial intelligence," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 15, no. 2s, pp. 1–16, 2019.

[28] C. Vallance, "Ai image creator faces uk and us legal challenges," Jan 2023.

[29] S. Magazine, "Are a.i. image generators violating copyright laws?," Jan 2023.

[30] C. Schuhmann, R. Beaumont, R. Vencu, C. Gordon, R. Wightman, M. Cherti, T. Coombes, A. Katta, C. Mullis, M. Wortsman, P. Schramowski, S. Kundurthy, K. Crowson, L. Schmidt, R. Kaczmarczyk, and J. Jitsev, "Laion-5b: An open large-scale dataset for training next generation image-text models," *CoRR*, vol. abs/2210.08402, 2022.

[31] P. I. Frazier, "A tutorial on bayesian optimization," *arXiv preprint arXiv:1807.02811*, 2018.

[32] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas, "Taking the human out of the loop: A review of bayesian optimization," *Proceedings of the IEEE*, vol. 104, no. 1, pp. 148–175, 2016.

[33] C. E. Rasmussen and C. K. Williams, *Gaussian processes for machine learning*, vol. 1. MIT press Cambridge, 2006.

[34] E. Schulz, M. Speekenbrink, and A. Krause, "A tutorial on gaussian process regression with a focus on exploration-exploitation scenarios," *bioRxiv*, p. 095190, 2017.

[35] J. Snoek, H. Larochelle, and R. P. Adams, "Practical bayesian optimization of machine learning algorithms," in *Advances in neural information processing systems*, vol. 25, pp. 2951–2959, 2012.

[36] R. Fletcher, *Practical methods of optimization*. John Wiley & Sons, 2013.

[37] R. Martinez-Cantin, "Bayesopt: A bayesian optimization library for nonlinear optimization, experimental design and bandits," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 3735–3739, 2014.

[38] N. Hansen and A. Ostermeier, "Completely derandomized self-adaptation in evolution strategies.," *Evolutionary computation*, vol. 9, no. 2, pp. 159–195, 2001.

[39] N. Hansen, S. D. Müller, and P. Koumoutsakos, "Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (cma-es)," *Evolutionary computation*, vol. 11, no. 1, pp. 1–18, 2003.

[40] A. Cully, K. Chatzilygeroudis, F. Allocati, and J.-B. Mouret, "Limbo: A fast and flexible library for bayesian optimization," *arXiv preprint arXiv:1611.07343*, 2016.

[41] E. Brochu, V. M. Cora, and N. De Freitas, "A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning," *arXiv preprint arXiv:1012.2599*, 2010.

[42] C. Bian, X. Wang, C. Liu, X. Xie, and L. Haitao, "Impact of exploration-exploitation trade-off on ucb-based bayesian optimization," in *IOP Conference Series: Materials Science and Engineering*, vol. 1081, p. 012023, IOP Publishing, 2021.

[43] F. Nogueira *et al.*, "Bayesian optimization: Open source constrained global optimization tool for python," 2014.

[44] A. Lopez-Rincon, D. Rojas-Velazquez, J. Garssen, S. W. van der Laan, D. Oberski, and A. Tonda, "Bayesian optimization for the inverse problem in electrocardiography," in *2023 IEEE Symposium Series on Computational Intelligence (SSCI)*, pp. 1593–1598, IEEE, 2023.

[45] L. C. W. Dixon, "The global optimization problem: an introduction," *Towards Global Optimiation 2*, pp. 1–15, 1978.

[46] N00MKRAD, "Nmkd stable diffusion gui - ai image generator by n00mkrad."

[47] A. Manolică, A.-S. Guţă, T. Roman, and L. M. Dragăn, "Is consumer overchoice a reason for decision paralysis?," *Sustainability*, vol. 13, no. 11, p. 5920, 2021.

[48] J. Johnson, "neural-style." https://github.com/jcjohnson/neural-style, 2015.

[49] S. Porkodi, V. Sarada, V. Maik, and K. Gurushankar, "Generic image application using gans (generative adversarial networks): A review," *Evolving Systems*, vol. 14, no. 5, pp. 903–917, 2023.