GreenStableYolo: Optimizing Inference Time and Image Quality of Text-to-Image Generation

Jingzhi Gong[1][0000−0003−4551−0701], Sisi Li[2][0000−0001−5376−5144], Giordano d'Aloisio[3][0000−0001−7388−890X], Zishuo Ding[4][0000−0002−0803−5609], Yulong Ye[5][0000−0002−2478−778X], William B. Langdon[6][0000−0002−6388−4160], and Federica Sarro[6][0000−0002−9146−442X]

[1] Loughborough University j.gong@lboro.ac.uk
[2] Beijing University of Posts and Telecommunications sisili@bupt.edu.cn
[3] Università degli Studi dell'Aquila giordano.daloisio@graduate.univaq.it
[4] University of Waterloo zishuo.ding@uwaterloo.ca
[5] University of Birmingham yxy382@student.bham.ac.uk
[6] University College London {w.langdon, f.sarro}@ucl.ac.uk

Abstract. Tuning the parameters and prompts for improving AI-based text-to-image generation has remained a substantial yet unaddressed challenge. Hence we introduce GreenStableYolo, which improves the parameters and prompts for Stable Diffusion to both reduce GPU inference time and increase image generation quality using NSGA-II and Yolo. Our experiments show that despite a relatively slight trade-off (18%) in image quality compared to StableYolo (which only considers image quality), GreenStableYolo achieves a substantial reduction in inference time (266% less) and a 526% higher hypervolume, thereby advancing the state-of-the-art for text-to-image generation.

Keywords: SBSE · ANN · GenAI · Text2Image · Stable Diffusion · Yolo

1 Introduction

In recent years Generative Artificial Intelligence (GenAI) has emerged as a powerful approach encompassing various techniques that enable machines to generate new content, such as text [13], images [19], and videos [14]. Particularly, image generation and text-to-image synthesis have garnered significant attention due to their potential in bridging the gap between textual descriptions and visual representations [18]. It enables systems to understand and interpret human language and automatically translate it to a visually meaningful way, facilitating tasks such as generating accompanying images for books [4], generating product images for advertising [22], and inspiring artists to create new forms of art [12].

However, achieving optimal performance in image generation tasks involves fine-tuning various aspects of a GenAI model, such as the number of inference steps, positive and negative prompts [1,3]. These parameters play a crucial role in determining the quality and efficiency of the generated images and tuning these parameters is essential to unlock the full potential of image generation

models [3,15]. At the same time, GenAI model are energy demanding and largely contribute to the increased CO_2 emissions[8,20].

Berger et al. [3] proposed a search-based approach, dubbed StableYolo, to optimize the image quality of Stable Diffusion by assessing image quality using Yolo [17]. However, they their approach do not take into account the aspect of inference time, which is a cornerstone for both ensuring user experience and minimizing the energy consumption of GenAI models. Especially in real-world scenarios, where responsiveness and energy efficiency are vital, addressing this aspect is vital for the widespread adoption of such models [5,11,20,21].

To address this gap, we present GreenStableYolo, a novel approach that addresses the challenge of optimizing the trade-off between inference time and image quality using a search-based multi-objective optimization method, namely Non-dominated Sorting Genetic Algorithm (NSGA-II) [6]. We provide initial empirical evidence that by using GreenStableYolo Stable Diffusion achieves a satisfactory equilibrium between inference time and image quality, making it suitable for real-world applications where both factors play a crucial role.

In a nutshell, the key contributions of this work include:

- The development of a novel system that seeks for an optimal trade-off between inference time and image quality by optimizing the prompts and parameters for Stable Diffusion, dubbed GreenStableYolo;
- Empirical evidence on the effectiveness of GreenStableYolo in achieving significantly less inference time and higher hypervolume compared to Stable-Yolo, thereby advancing the state-of-the-art multi-objective optimization for text-to-image generation;
- A comprehensive analysis to understand the influence of different parameters and prompts on both inference time and image quality in Stable Diffusion.

2   Related Work

To improve image generation quality, Berger et al. [3] were the first to propose the use of a Genetic Algorithm (GA) able to simultaneously tune the prompt and parameters of Stable Diffusion. Magliani et al. [15] use GA to find the best diffusion parameters for automated image retrieval from a dataset. While some research [5,11] has been carried out to optimize inference time, from hardware design to model architecture, there has been limited work focusing on optimizing the prompts and parameters. Our work builds upon previous work by considering both inference time and image quality as optimization objectives.

3   Methodology

To mitigate the aforementioned challenge, we propose GreenStableYolo, a novel multi-objective search-based approach that, given a text prompt for image generation, searches for the optimal parameters that can strike a balance between: (1) *Inference time*, which is measured by the GPU time taken for the execution

of the StableDiffusionPipeline; and (2) *Image quality*, which is determined by performing object recognition with Yolo, then selecting objects that match the input prompt, and computing their average probabilities [3].

NSGA-II Optimization Algorithm To simultaneously enhance image quality and reduce inference time, we leverage NSGA-II, a well-known and efficient multi-objective evolutionary algorithm [10,16]. Specifically, NSGA-II works as follows: (1) Initialize a population with N individuals; (2) Perform crossover and mutation operations, generating an offspring population denoted as P_o; (3) Reassemble the parent population P_{t-1} and P_o into a temporary population with the size of $2N$, and formulate individuals into i non-inferior frontier through fast non-dominating sorting; (4) Select N individuals from the temporary population to form the next population for the tth iteration, denoted as P_t. (5) Repeat steps (2)-(4) until the termination condition is met; and (6) The algorithm ends up and returns the current Pareto-Optimal set.

Selected Parameters To make a straightforward comparison with StableYolo, we adopt the same settings as used by Berger et al. [3]. Specifically, the following parameters and prompts are tuned and searched with NSGA-II: (1) *Inference steps* (1 to 100): the AI's image generation iterations; (2) *Guidance scale* (1 to 20): the impact of the prompt on image generation; (3) *Guidance rescale* (0 to 1): rescales the guidance factor to prevent over-fitting; (4) *Seed* (1 to 512): randomization seed; (5) *Positive prompt*: used to describe images and improve their details, e.g., "photograph", "color", and "ultra real" [2]; and (6) *Negative prompt*: avoided description during image generation, e.g., "sketch", "cropped", and "low quality" [2].

4   Evaluation

To evaluate our proposal, we address the following research questions (RQs):
➤ **RQ1:** To what extent can GreenStableYolo improve image quality and inference time compared with StableYolo?
➤ **RQ2:** How do parameters/prompts of Stable Diffusion influence the inference time for image generation?
➤ **RQ3:** How do parameters/prompts of Stable Diffusion influence the quality of the generated images?

Experimental Setup To ensure a fair evaluation of the optimization effectiveness, we employed the same hyperparameter setup as StableYolo for NSGA-II, e.g., the population size was set to 25, the number of generations was set to 50, and both the mutation rate and crossover rate were set to 0.2. We selected the weights of 0.001 for image quality and -1000 for inference time based on empirical investigation of different weight combinations. In addition, we used Stable Diffusion version v2 and Yolo version v8. To assess variability, we evaluated each model 15 times using different random seeds, focusing solely on the prompt "two
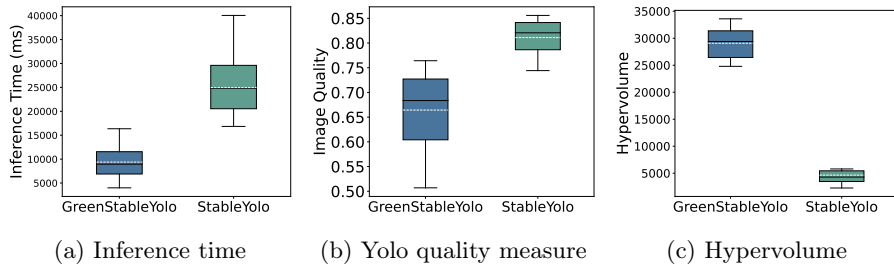
(a) Inference time   (b) Yolo quality measure   (c) Hypervolume

Fig. 1: Comparison of GreenStableYolo and StableYolo on 15 independent runs

people and a bus" due to time constraints. Any future studies can explore additional prompts. All experiments were conducted on a virtual machine hosted on Google Colaboratory, with an NVIDIA Tesla T4 GPU with 16 GB of RAM.

RQ1 Results Figure 1 presents the performance comparisons between Green-StableYolo and StableYolo. Specifically, Figure 1a reveals that GreenStable-Yolo achieves an average inference time of 9.4 seconds with an interquartile range (IQR) of 4.7 seconds. Conversely, StableYolo exhibits an average inference time of 25.0 seconds, which is 1.66 times higher than GreenStableYolo, with an IQR of 9.1 seconds. That is, GreenStableYolo generates images much faster.

This improvement in inference time comes at a slight cost to image quality. As illustrated in Figure 1b, GreenStableYolo experiences approximately an average degradation of 0.18 points in image quality. We also compute the hypervolume [9] for both models for a more comprehensive comparison[7]. Figure 1c presents the hypervolume values with the reference point set as (1, 50000), where Green-StableYolo achieves an average hypervolume of 29074.11, surpassing StableYolo's score of 4642.17 by 5.26 times. ***This substantial difference demonstrates the clear dominance of GreenStableYolo over StableYolo in this two-objective optimization problem for text-to-image generation.***

RQs2-3 Results To investigate RQs2–3, we followed previous work [7] and built two Random Forest regression models using `scikit-learn`. The features of these models include the number of iteration steps, guidance scale, guidance rescale, positive prompts, and negative prompts (excluding the random seed). The target variables are inference time and image quality score, respectively. We use the `RandomizedSearchCV` function from `scikit-learn` to find the optimal hyperparameters during model training. The `feature_importances_` function is then used to compute the importance of each parameter and prompt based on the Mean Decrease Impurity (MDI), a.k.a. as Gini importance. To ensure reliability, we repeat this process 10 times.

Figures 2a and 2b present the calculated importance of parameters and prompts based on the mean decrease in impurity, with respect to inference time

---

[7] Hypervolume is a fundamental metric used in multi-objective optimization problems that indicates the dominance of a solution in the objective space.

(a) RQ2: MDI w.r.t. inference time   (b) RQ3: MDI w.r.t. image quality
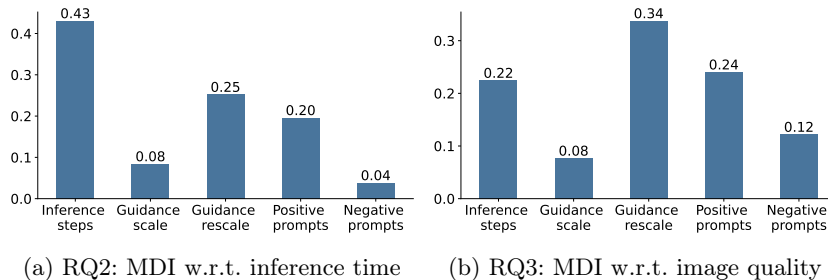
Fig. 2: Parameters/prompts importance based on the mean decrease in impurity

and image quality scores, respectively. As shown in Figure 2a, the number of *inference steps* emerges as a significant factor affecting inference time. This is expected, as more steps involve more computations, thereby resulting in higher inference time. Meanwhile, for image quality (Figure 2b), parameters like *guidance rescale* and *positive prompts* play a relatively more critical role.

These findings confirm the value of our work: ***increased computational resources do not necessarily translate to better image quality; instead, appropriate model parameter settings are more crucial***. This highlights the importance of identifying optimal parameter combinations during model inference to balance computational efficiency and output quality.

Threats to Validity The limited exploration of prompts, the randomness in the optimization process, and the specific configuration for NSGA-II may introduce internal threats. Besides, external threats may include the choice of the GenAI model, the noise when measuring the inference time, and the evaluation of image quality based on object recognition using Yolo.

5   Conclusion

In GenAI text-to-image, achieving images of high-quality is often not the only important aspect to consider, as inference time, which directly impacts user experience and energy consumption, also plays a critical role. In this work we introduced GreenStableYolo, the first approach leveraging NSGA-II to strike an optimal trade-off between these two objectives for Stable Diffusion. Experimental comparisons with StableYolo demonstrate that GreenStableYolo achieves significantly reduced inference time while maintaining a relatively high image quality. Future research can expand upon our evaluation by incorporating alternative initial prompts, optimizing different performance metrics such as energy consumption, and broadening to other GenAI systems such as DALL-E, ImageFX, or Midjourney.

References

1. Stable diffusion documentation https://huggingface.co/docs/diffusers/api/pipelines/stable_diffusion/text2img
2. Stable diffusion photorealistic prompts https://prompthero.com/stable-diffusion-photorealistic-prompts
3. Berger, H., Dakhama, A., Ding, Z., Even-Mendoza, K., Kelly, D., Menéndez, H.D., Moussa, R., Sarro, F.: StableYolo: Optimizing image generation for large language models. In: SSBSE 2023. LNCS, vol. 14415, pp. 133–139
4. Bruens, J.D., Meissner, M.: Do you create your content yourself? using generative artificial intelligence for social media content creation diminishes perceived brand authenticity. Journal of Retailing and Consumer Services **79**, 103790 (2024)
5. Cao, Y., Li, S., Liu, Y., Yan, Z., Dai, Y., Yu, P.S., Sun, L.: A comprehensive survey of AI-generated content (AIGC): a history of generative AI from GAN to ChatGPT. arXiv 2303.04226 (2023)
6. Deb, K., Agrawal, S., Pratap, A., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
7. Ding, Z., Chen, J., Shang, W.: Towards the use of the readily available tests from the release pipeline as performance tests: are we there yet? p. 1435–1446. ICSE '20
8. Georgiou, S., Kechagia, M., Sharma, T., Sarro, F., Zou, Y.: Green ai: do deep learning frameworks have different costs? In: Procs. of the ICSE'22. p. 1082–1094
9. Guerreiro, A.P., Fonseca, C.M., Paquete, L.: The hypervolume indicator: Computational problems and algorithms. ACM Computing Surveys **54**(6), 1–42 (2021)
10. Ji, B., Huang, H., Yu, S.S.: An enhanced NSGA-II for solving berth allocation and quay crane assignment problem with stochastic arrival times. IEEE Trans.
11. Kim, S., Hooper, C., Wattanawong, T., Kang, M., Yan, R., Genc, H., Dinh, G., Huang, Q., Keutzer, K., Mahoney, M.W., Shao, Y.S., Gholami, A.: Full stack optimization of transformer inference: a survey. arXiv 2302.14017 (2023)
12. Ko, H., et al.: Large-scale text-to-image generation models for visual artists' creative works. In: Intelligent User Interfaces, IUI 2023. pp. 919–933
13. Li, J., Tang, T., Zhao, W.X., Wen, J.R.: Pretrained language models for text generation: A survey. arXiv 2105.10311 (25 May 2021)
14. Long, F., Qiu, Z., Yao, T., Mei, T.: VideoDrafter: Content-consistent multi-scene video generation with LLM. arXiv 2401.01256 (2 Jan 2024)
15. Magliani, F., Sani, L., Cagnoni, S., Prati, A.: Genetic algorithms for the optimization of diffusion parameters in content-based image retrieval. In: ICDSC 2019
16. Rashid, T.A., Majidpour, J., Thinakaran, R., Batumalay, M., Dewi, D.A., Hassan, B.A., Dadgar, H., Arabi, H.: NSGA-II-DL: metaheuristic optimal feature selection with deep learning framework for HER2 classification in breast cancer. IEEE Access
17. Redmon, J., Divvala, S.K., Girshick, R.B., Farhadi, A.: You only look once: Unified, real-time object detection. In: CVPR 2016. pp. 779–788
18. Reed, S.E., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., Lee, H.: Generative adversarial text to image synthesis. In: ICML 2016. pp. 1060–1069
19. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: CVPR. pp. 10674–10685 (2022)
20. Sarro, F.: Search-based software engineering in the era of modern software systems. In: IEEE International Requirements Engineering Conference. pp. 3–5 (2023)
21. Sarro, F.: Automated optimisation of modern software system properties. In: International Conference on Performance Engineering, ICPE '23. pp. 3–4
22. Yu, T., Yang, X., Jiang, Y., Zhang, H., Zhao, W., Li, P.: TIRA in Baidu image advertising. In: International Conference on Data Engineering, ICDE 2021