Image Generation of Egyptian Hieroglyphs

Song Gao
gaoso@oregonstate.edu
Oregon State University
Corvallis, Oregon, USA

Bo Hui
bo-hui@utulsa.edu
University of Tulsa
Tulsa, Oklahoma, USA

Wanwan Li
wanwan-li@utulsa.edu
University of Tulsa
Tulsa, Oklahoma, USA

ABSTRACT

This comprehensive study explores the enduring fascination with and scholarly examination of Egyptian hieroglyphs. The investigation focuses on the writing structure of Egyptian hieroglyphs, employing image and pixel representations with the aim of achieving accurate reconstruction. The study utilizes a stable diffusion model and DeepSVG. We investigate challenges in providing precise reconstructions and evaluate the strengths and weakness of these methods. Thorough A significant contribution of the study is the presentation of a dataset comprising both pixel-based and vector-based images of Egyptian hieroglyphs. The findings contribute to ongoing discussions in linguistics, archaeology, and the interdisciplinary intersection of AI with historical studies.

CCS CONCEPTS

• **Computing methodologies → Neural networks**.

KEYWORDS

Image Synthesis, Egyptian hieroglyphs, Stable diffusion, Autoencoder, Deep Learning

ACM Reference Format:
Song Gao, Bo Hui, and Wanwan Li . 2024. Image Generation of Egyptian Hieroglyphs. In *2024 16th International Conference on Machine Learning and Computing (ICMLC 2024), February 02–05, 2024, Shenzhen, China.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3651671.3651771

1 INTRODUCTION

Egyptian hieroglyphs have a long history of interpretation and study. The study of Egyptian hieroglyphs is characterized by a rich history of interpretation, attracting individuals for diverse personal and professional reasons. Archaeologists and Egyptologists focus on deciphering hieroglyphic inscriptions found on ancient monuments, tombs, and artifacts, extracting information about buried individuals, historical events, and religious practices. The unique combination of logographic and alphabetic elements in Egyptian hieroglyphs makes them a subject of interest for linguists and language enthusiasts exploring ancient languages and linguistic evolution. Additionally, scholars delve into religious and philosophical texts, recognizing hieroglyphs as vehicles for conveying the spiritual and literary aspects of ancient Egyptian culture. Beyond a writing system, hieroglyphs serve as a system of symbols and artistic expression, fostering an appreciation for the artistic and symbolic intricacies of ancient Egyptian visual communication. Proficiency in hieroglyphs is valued in professional fields such as archaeology and museum curation, where understanding ancient inscriptions is essential. Ultimately, the study of Egyptian hieroglyphs contributes significantly to our comprehension of human history, language development, and cultural evolution, providing a connection to the enduring legacy of a rich and ancient civilization.

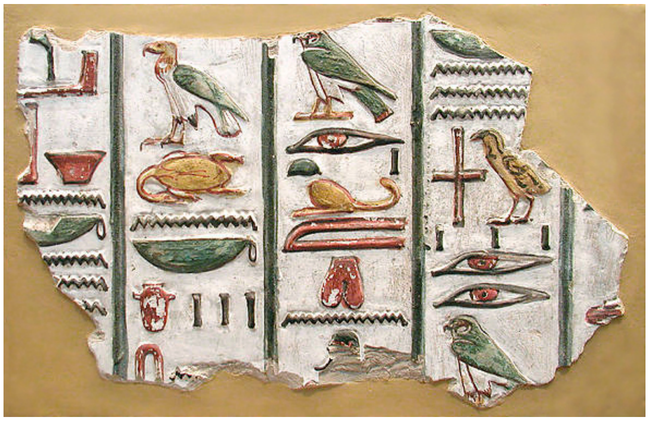

Figure 1: Example of Egyptian hieroglyphs [29]

This study investigates the structural aspects of Egyptian hieroglyphs. The exploration commences by employing the Imagen model [28] and pixel representations of Egyptian hieroglyphs, with the anticipation that this approach will yield a successful reconstruction. Subsequently, after conducting numerous iterations using a stable diffusion model[26], it became apparent that the outcomes were limited to the slow convergence speed and the possibility of failure of convergence of the model. This falls short of achieving an accurate reconstruction of Egyptian hieroglyphs.

Subsequent to these efforts, an alternative methodology, DeepSVG [4], was explored; however, it exhibited suboptimal performance. When supplied with Egyptian hieroglyphs as input data, DeepSVG demonstrated challenges in comprehending the intricate representation of symbols. This limitation is attributed to the inherent inability of DeepSVG, utilizing a Variational Autoencoder (VAE) [13], to effectively capture the logical order of symbols.

Furthermore, a comprehensive evaluation of both methodologies was conducted, resulting in valuable insights. Notably, our contribution to the field encompasses the provision of a dataset featuring both pixel-based and vector-based images of Egyptian hieroglyphs, serving as a valuable resource for further research and analysis.

ICMLC 2024, February 02–05, 2024, Shenzhen, China

ACM ISBN 979-8-4007-0923-4/24/02
https://doi.org/10.1145/3651671.3651771

2 RELATED WORKS

Neural networks play crucial roles in image generation. These artificial intelligence models, inspired by the human brain's neural architecture, excel in capturing and reproducing complex visual information. Generative models, particularly Generative Adversarial Networks (GANs) [6, 9, 15–17, 19], Autoencoders (AEs) [1, 2, 8, 14, 18], Diffusion Models (DMs) [7, 12, 22, 27, 31], have emerged as powerful tools for synthesizing realistic and diverse images. While the Imagen model is used in this work, there are other models that also achieve the goal with different strategies. In textDiffuser [5], the authors introduce a versatile diffusion model-based framework featuring two stages. Initially, a Layout Transformer locates keyword coordinates and generates character-level segmentation masks from text prompts. Subsequently, the latent diffusion model is fine-tuned in the second stage, utilizing the generated masks as conditions. A character-aware loss in the latent space enhances the quality of generated text regions. Figure 1 illustrates TextDiffuser's application for generating precise text images. TextDiffuser excels in text inpainting, reconstructing incomplete images with text. Training employs OCR tools and filtering strategies to create a dataset of 10 million high-quality image-text pairs with OCR annotations.

In the field of vector-based representation and learning, canvasVAE [30] studies the content of vector graphic documents and shows that the document can be divided into a sequence of visual elements, such as shapes, images, or texts. author trains variational autoencoders to learn the representation of the documents. In experiments, the author shows that the model named CanvasVAE contributes a strong baseline for the generative modeling of vector graphic documents. Different from a pixel-to-pixel method or vector-to-vector method, Img2Vec [25] uses a text prompt and a pixel-based image as input and outputs a vector-based image. The proposed model studies the rasterized input graphic and restores the vector output graphic image. Instead, we propose a new neural network that can generate complex vector graphics with varying topologies and only requires indirect supervision from readily available raster training images. To enable this, we use a differentiable rasterization pipeline that renders the generated and composites vector shapes onto raster canvas.

There is a pixel-to-vector approach method [11] that takes advantage of the stable diffusion model. The authors use massive datasets of captioned images, and diffusion models learn to generate raster images that can be used to generate vector representations of images like Scalable Vector Graphics (SVGs). This text-conditioned diffusion model, trained on pixel representations of images, can be used to generate SVG exportable vector graphics. The method, VectorFusion, distills abstract semantic knowledge out of a pre-trained diffusion model. Inspired by recent text-to-3D work, they learn an SVG consistent with a caption using Score Distillation Sampling. To accelerate generation and improve fidelity, VectorFusion also initializes from an image sample. Experiments show greater quality than prior work for a wider range of styles including sketches.

3 OVERVIEW

The dataset is constructed from the font family NotoSans, which contains Egyptian hieroglyphs. The fonts of NotoSans can be found at webpage of Google font, and it is also available on GitHub. This dataset contains 2,142 data samples, with half of them featuring a white background with black background and white fonts JPEG images, and the other half are vector-based SVG images. Each sample consists of a 400 by 400 pixels sized JPEG image. In the Imagen paper, authors examined expansive frozen language models, exclusively trained on textual data, exhibit remarkable efficacy as text encoders for the generation of images from text. Notably, the author's observations reveal that enhancing the scale of the frozen text encoder yields a substantial improvement in sample quality, surpassing the impact of scaling the size of the image diffusion model. Additionally, the authors present a novel diffusion sampling technique called dynamic thresholding, designed to harness elevated guidance weights. Furthermore, the authors underscore various crucial design choices in the diffusion architecture and propose a streamlined variant of U-Net to achieve faster convergence and greater memory efficiency.

DeepSVG model represents a vector approach using SVG commands as input to reconstruct vector graphics, which aligns closely with our goal of reconstructing Egyptian hieroglyphs fonts. By employing such a network, we aim to reduce memory consumption and achieve more accurate reconstructions of Egyptian Hieroglyphs.

4 TECHNICAL APPROACH

4.1 The Imagen Method

The Imagen model is a text-to-image diffusion model that takes a text prompt and a pixel-based image as input, outputting a reconstructed pixel image. This model is built on the large transformer language model, T5. The T5 language model encodes text, maintains image and text alignment on a large scale for image generation and improves performance. The Imagen model involves two main steps: first, inputting a prompt into a frozen text encoder to obtain a text embedding that encapsulates all the relevant text information; second, feeding this text embedding into a generative model to instruct it in generating images. The generative model initially creates low-resolution image, which is enhanced with two super-resolution networks. These networks take low-quality image and preceding text embedding as inputs and produce high-quality images.

The primary focus of the paper is to leverage the potent language model in Natural Language Processing (NLP) rather than employing CLIP model[23], as seen in image-text pair trained text encoders. The rationale behind this choice lies in the substantial amount of training data available for language models, surpassing that of image-text pairs. Moreover, the language model's size dwarfs current image-text models, suggesting a superior understanding of text, a prerequisite for generating high-quality images. In the deeper parsing of the process, the text encoder extracts text information, performs pooling, and adds the resulting embedding to the original image to facilitate conditional operations. While there is an option for direct cross-attention showing that applying cross-attention yields better results, as evidenced by ablation studies.

4.2 The DeepSVG Method

The DeepSVG model, as detailed in the work by Carlier et al. [4], assumes particular significance in practical applications involving Scalable Vector Graphics (SVGs), where users must perform diverse

Token	Parameters	Description
<SOS>	∅	Start of SVG token.
M (MoveTo)	x_2, y_2	Move cursor to end-point (x_2, y_2) without drawing.
L (LineTo)	x_2, y_2	Draw a line to point (x_2, y_2).
C (Cubic Bézier)	$qx_1, qy_1, qx_2, qy_2, x_2, y_2$	Draw cubic Bézier curve with control points (qx_1, qy_1), (qx_2, qy_2), and end-point (x_2, y_2).
z (ClosePath)	∅	Close the path by moving cursor back to starting position (x_0, y_0).
<EOS>	∅	End of SVG token.

Table 1: Descriptions of SVG Tokens[4].

operations on vector graphics while preserving the authenticity of their original compositions. It is noteworthy that SVG images are fundamentally distinct from pixel-based images in their construction and representation.

Scalable Vector Graphics denoted as SVG, stands as an XML-based format meticulously crafted for delineating two-dimensional graphics in a scalable manner. This format establishes a structured framework for the representation of vector images, endowing them with inherent support for interactivity and animation. The ensuing discourse provides an insightful exploration of the unique characteristics inherent in SVG images, thereby elucidating their distinctiveness from their pixel-based counterparts. The following table describes the commands inside SVG:

In addition, the DeepSVG model provides a hierarchical generative network model that studies SVG image generation from high-level shapes to low-level commands, resulting in high-quality image generation. The model employs a variational auto-encoder (VAE) [13] structure with an encoder and decoder network. The encoder incorporates a Transformer-based architecture, processing individual paths independently before aggregating their representations. The hierarchical nature of SVG images, consisting of paths with sequences of commands, is considered in both the encoding and decoding stages. The feed-forward prediction approach is used for generating commands and arguments, offering advantages in terms of reconstruction quality and interpolation smoothness. The model leverages transformer blocks, and the encoder maintains permutation invariance of the input paths. The decoder, on the other hand, does not require this invariance, employing a learned index embedding to break symmetry during generation. The reparametrization trick is utilized in obtaining the latent vector. The entire architecture is presented as a schematic representation, showcasing the multi-stage encoding and decoding process.

5 EXPERIMENTAL RESULTS

In this section, we delve into a sequence of training experiments for the Imagen model, which is followed by a transition to the DeepSVG model in the fourth experiment. Starting with the Imagen model, the parameters for the stable diffusion experiments are configured as outlined below:

For Imagen model:

- text_encoder_name = ’t5-large’
- unets = (unet0)
- output_image_sizes = 88
- inference_timesteps = 2000
- cond_drop_prob = 0.1

The default configuration incorporates two U-nets, where Unet 0 has an input size of 32, and Unet 1, operating as an upscaling model, has an input size of 96. This selection is made considering the relatively straightforward nature of the sample data in comparison to other images, leading us to opt for a simplified network structure with only one Unet. While considering the limitation of our hardware, we choose an output image size of 88, which just be a right fit to our GPU VRAM. For the encoding language model, T5-large is employed. This model generates text embeddings as float tensors with dimensions (number of training samples, 10, 1024), and Boolean tensors representing text masks with dimensions (number of training samples, 10).

The text embeddings play a pivotal role in serving as the extracted data representation of the model, projecting textual information into a high-dimensional latent space. Simultaneously, the text masks function to selectively mask certain words in a sentence, contributing to the model’s robustness by predicting the replacements for masked words.

5.1 Training iterations vs samples

We conducted a training experiment involving the inference of 8 samples at every 500 training iterations, spanning from iteration 0 to iteration 5500. The primary objective of this experiment is to gain insights into the generated samples’ quality and assess the learning stage of stable diffusion model.

Two distinct trials were executed during this experiment. The initial trial utilized the entire dataset as training samples, while the subsequent trial focused exclusively on the first 12 data samples. The initial trial did not yield successful data sample generation, as evidenced by the outcomes depicted in Table 2. Notably, even when the sampled images contained intricate details after 6000 iterations, the results indicated that the model struggled to correctly generalize and reconstruct. Conversely, the second trial demonstrated successful data sample reconstructions for all 8 data samples included. This trial exclusively employed the first 12 data samples, all of which featured human figures. Consequently, it is plausible to infer that the inclusion of diverse geometries in the full dataset may have posed challenges to the model’s learning and generalization capabilities for accurate reconstruction.

The experimental setup utilized a laptop with an RTX 3070 8GB. To accommodate the constraints of the 8GB VRAM, careful adjustments were made to batch samples and model parameters. The complete dataset training for each epoch took approximately 58 seconds. However, the extensive duration of more than 112 hours to complete training for 7000 iterations on the entire dataset is deemed less than optimal. Despite attempts to enhance speed through half-precision training, no significant improvements were observed.

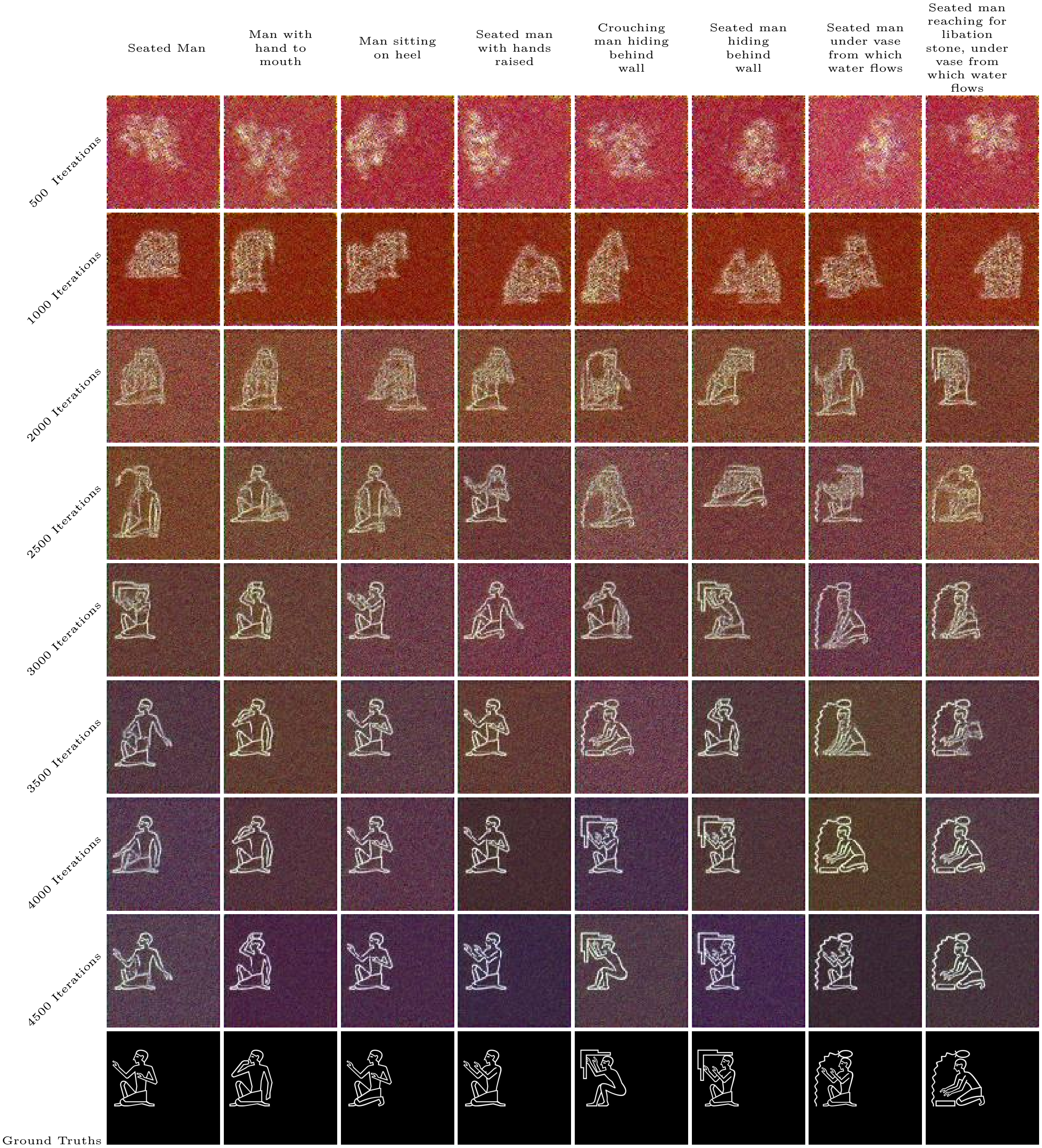
Seated Man
Man with hand to mouth
Man sitting on heel
Seated man with hands raised
Crouching man hiding behind wall
Seated man hiding behind wall
Seated man under vase from which water flows
Seated man reaching for libation stone, under vase from which water flows
500 Iterations
1000 Iterations
2000 Iterations
2500 Iterations
3000 Iterations
3500 Iterations
4000 Iterations
4500 Iterations
Ground Truths

Table 2: A 8×8 Image table for Egyptian hieroglyphs generation

5.2 Interpolation over latent space of model

Interpolation over the latent space [3] is a widely adopted technique among researchers to showcase a model's generalization capabilities. We applied a similar technique to evaluate the generalization of our modest model.

The Imagen model is conceptualized as an Encoder-Decoder structure, where initial features are encoded into latent variable and continuously denoised to generate raw features. However, the resulting latent variable lacks crucial high-level semantic information and other essential latent space properties, including interpolation and feature decoupling.

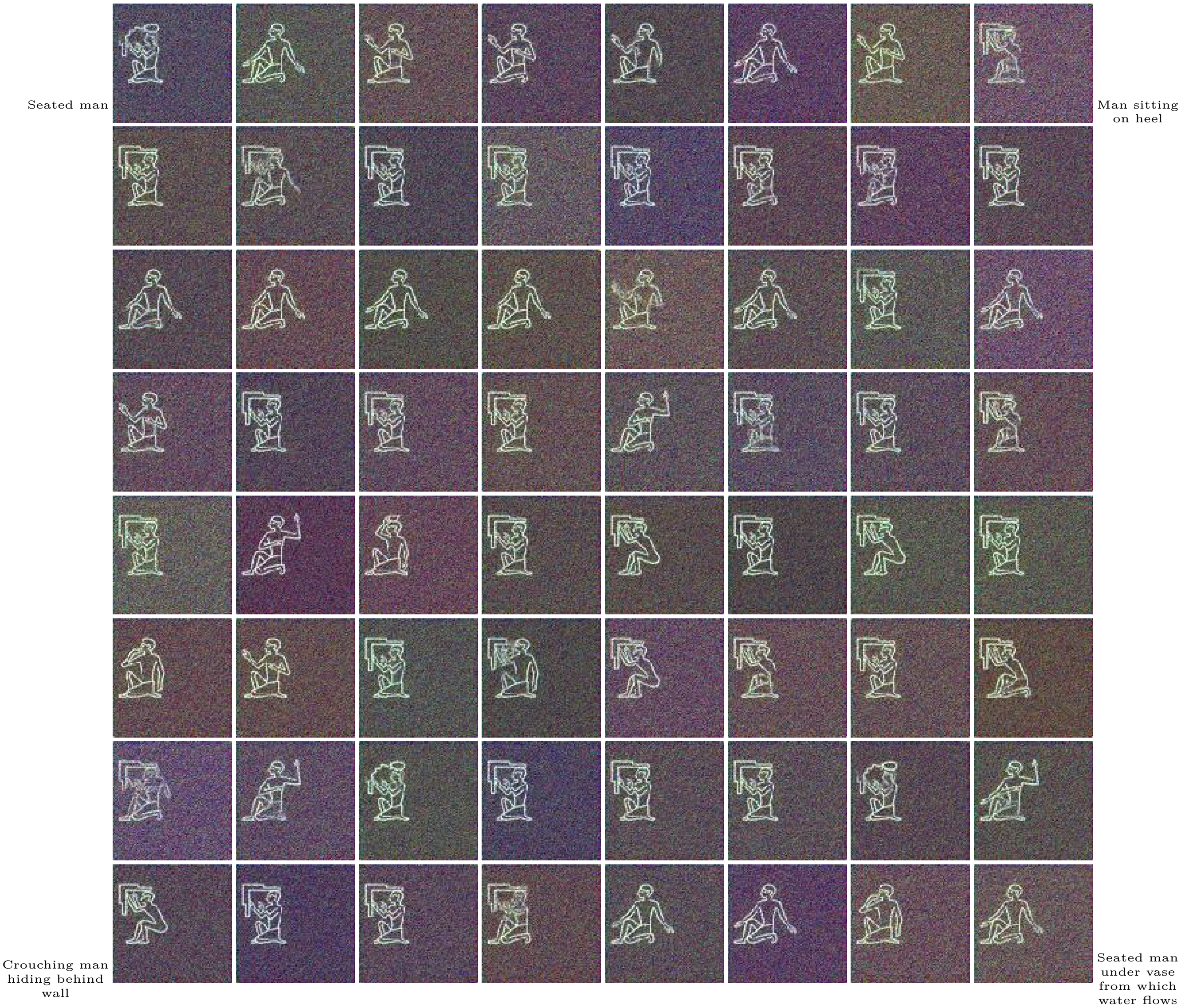
Seated man
Man sitting on heel
Crouching man hiding behind wall
Seated man under vase from which water flows

Table 3: A 8×8 table for interpolation of latent space

Ensuring interpolation and perturbation within the Latent Space align with the manifold of valid image priors is paramount. When the valid image prior resides on a manifold, perturbation, and interpolation should remain within the neighborhood of the initial value. This guarantees a theoretically seamless transition between base images, preserving good local coherence in the output image.

In implementing this method, we utilized nn.Upsampling Bilinear2d [21] from PyTorch to interpolate four selected text prompts. Leveraging the T5 large language model, we converted the text prompts into four text embeddings, reshaped the text embeddings, and employed nn.Upsampling-Bilinear2d to generate 64 text embeddings. Due to the larger text embedding, graphic memory exceeded 8GB, leading CUDA [20] to use memory as a swapping station. To mitigate this, we slice tensor into smaller sizes to fit VRAM, resulting in processing time of approximately 35 minutes to complete this inference, as opposed to anticipated 5 hours.

Two experimental trials are presented in Table 3 and Table 4, showcasing the results from diverse viewing angles. The corners of the tables represent the original text samples without any interpolation, while the intermediate samples result from combinations of different ratios of the four corner samples. Specifically, Table 3 includes samples related to "Seated man," "Man sitting on heel," "Crouching man hiding behind a wall," and "Seated man under a vase from which water flows." Conversely, Table 4 encompasses varied samples associated with "Crouching man hiding behind a wall," "Seated man," "Seated man under a vase from which water flows," and "Fatigued man." Despite differences in the experimental

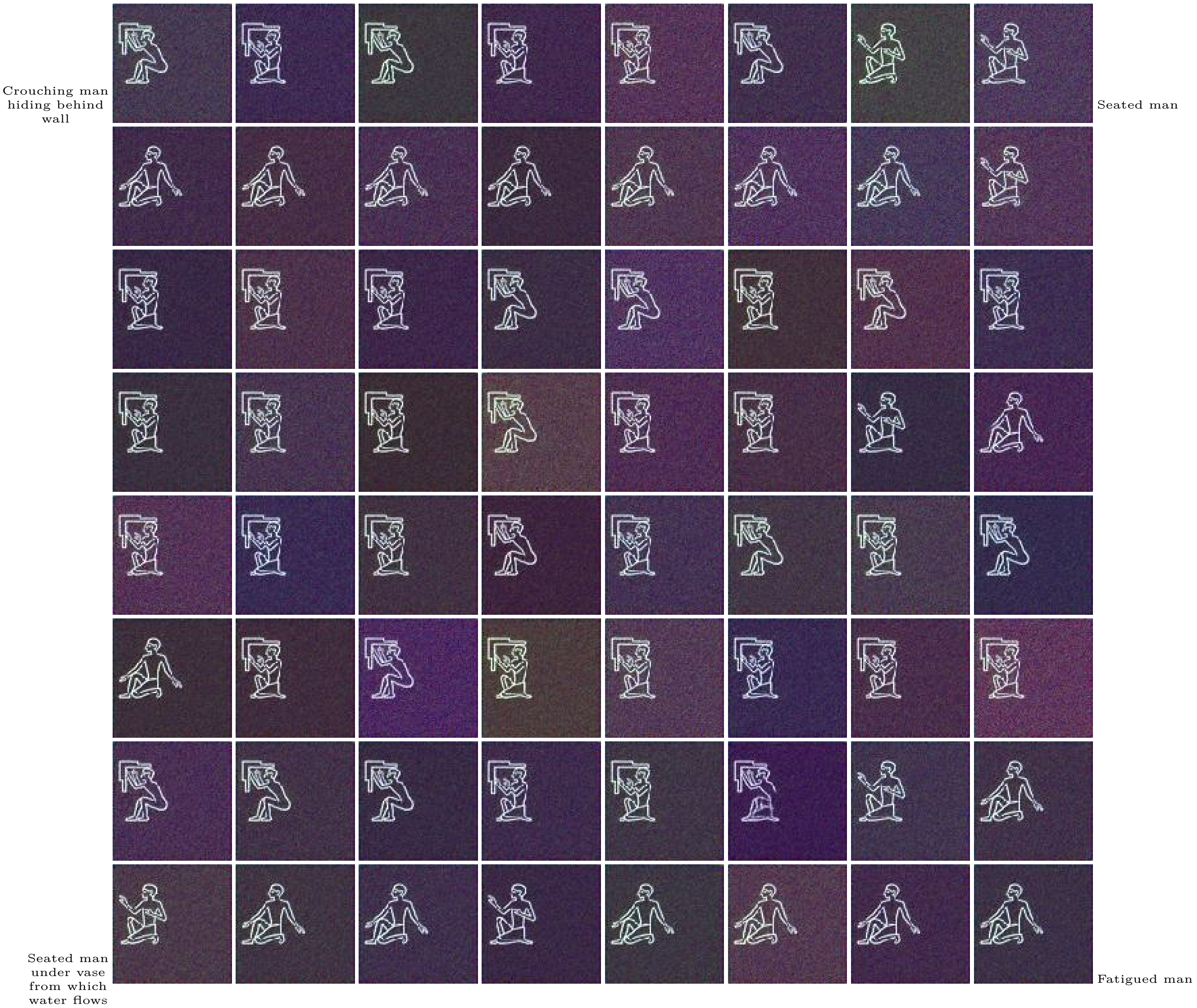
Crouching man hiding behind wall
Seated man
Seated man under vase from which water flows
Fatigued man

Table 4: A 8×8 table for interpolation of latent space

setups, a common observation emerges—the recurring and seemingly random appearance of the man behind the wall.

5.3 Clip score

The CLIP Score effectively captures meaningful relationships between natural language and image pairs through the acquisition of their semantic connections. Feature vectors are individually derived from the corresponding natural language and image pairs, followed by the computation of their cosine similarity. A higher CLIP Score signifies an elevated correlation between image-text pairs, indicating a stronger alignment in semantic content. Consequently, the CLIP Score serves as a metric for evaluating the match and correlation levels between natural language and image pairs, with larger values, approaching 1, indicating a more robust assessment.

In this part of the experiment, we tested the Clip score [10] of different iterations from 500 to 5500. In Figure 2, the result shows an improvement from 500 iterations to 2000 iterations, after 2000 iterations the Clip score starts to drop.

5.4 DeepSVG training and testing

For the DeepSVG model, a training regimen of 2000 iterations was initially planned, yet an early stop occurred at the 800th iteration. Upon scrutinizing the model's loss, it became evident that optimal hyper-parameters were achieved around the 700th iteration. In subsequent testing, we adopted a similar interpolation methodology, as employed in the Imagen model, to assess the latent space. However, a notable distinction lies in the absence of an intermediate step directly available for sampling from the DeepSVG model.

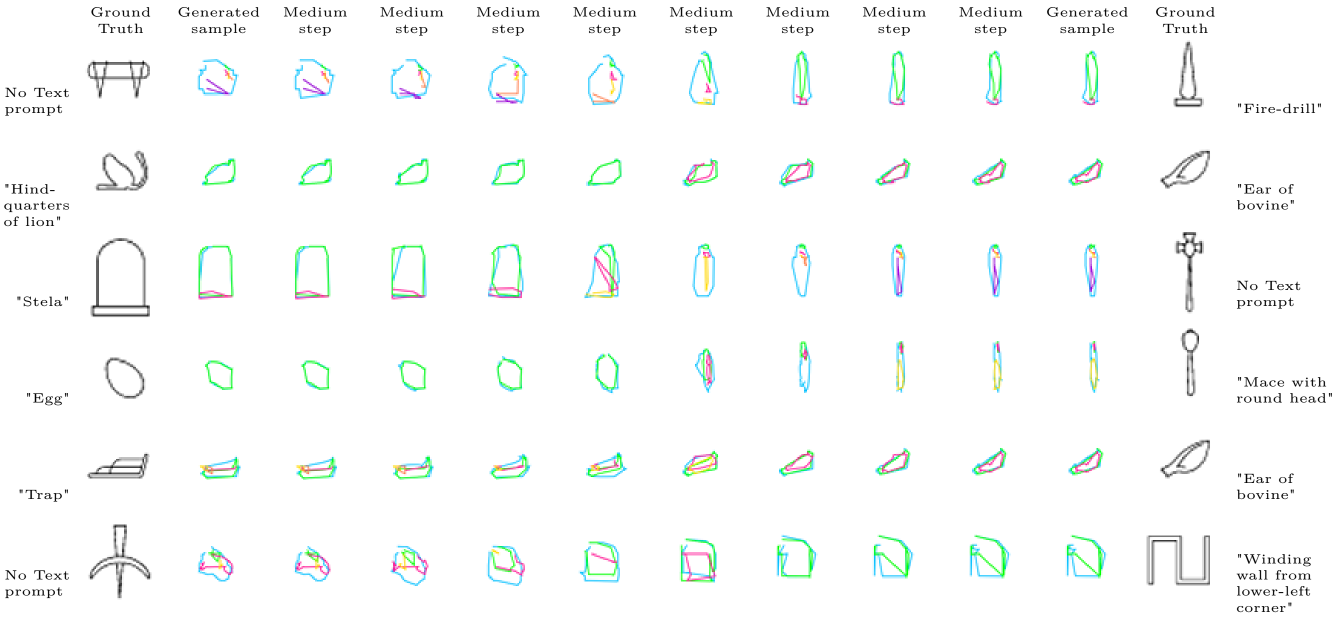

Table 5: A 12×6 Image table for DeepSVG interpolation generation

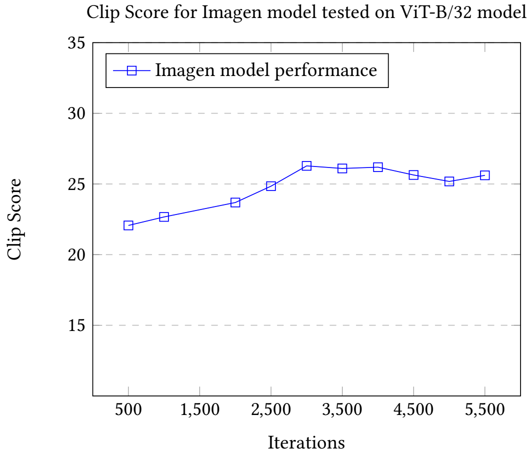

Figure 2: Clip score plot

In the context of DeepSVG, the interpolation method relies on two keyframes, aiming to generate intermediate frames through shape morphing. This iterative approach facilitates the incorporation of hand-drawn keyframes at each step until a satisfactory outcome is achieved. Notably, our findings indicate that this model predominantly generalizes to contours of Egyptian hieroglyphs.

6 DISCUSSION

An insightful observation gleaned from the Imagen training experiment is encapsulated in the 8×8 Image table presented in Table 2. This table offers a visual narrative of the standard stable diffusion model's training trajectory, gradually transforming noisy inputs into accurately reconstructed images. Notably, from iteration 2500 onward, a discernible shift occurs, with the majority of cells depicting human forms. This suggests a sudden emergence of learning, indicative of the model's rapid generalization. The speed of this learning phenomenon is such that within a mere dozen iterations, the model undergoes a substantial performance shift.

Subsequently, starting at iteration 3500, the model attains a stage of complete image generation, with pixel quality details reaching a zenith. Interestingly, by the conclusion of iteration 4500, the image quality exhibits minimal variance compared to the state at iteration 4000. However, an intriguing observation emerges between iterations 3500 and 4500, where the generated images display instability. This phenomenon is attributed to Imagen's utilization of the T5 language model [24], introducing fluctuations, particularly when the language model encounters descriptions not typically encountered during its training. Consequently, the Imagen model can encounter challenges in producing accurate responses. In the context of the latent space interpolation experiment, multiple trials were meticulously conducted to ensure the reliability of the conclusions drawn. The results presented in Table 3 were derived from a model trained with 4500 iterations, while those in Table 4 originated from a model trained with 5000 iterations. Trial associated with the model trained for 5500 iterations exhibits an overfitting pattern, manifesting in a lack of variation in latent space.

The diffusion model might not produce a correct representation, while DeepSVG, incorporating a VAE, struggles to capture the logical order of symbols. Concerns arise regarding the DeepSVG model's proficiency in capturing the semantics of SVG commands within its latent space. As the table 5 showed, it is posited that the model lacks the necessary depth in this latent space to reconstruct the intricate and abstract nature of SVG commands. Notably, SVG commands represent highly abstract ideas, characterized by non-linearity, high dimensionality, and non-convex optimization

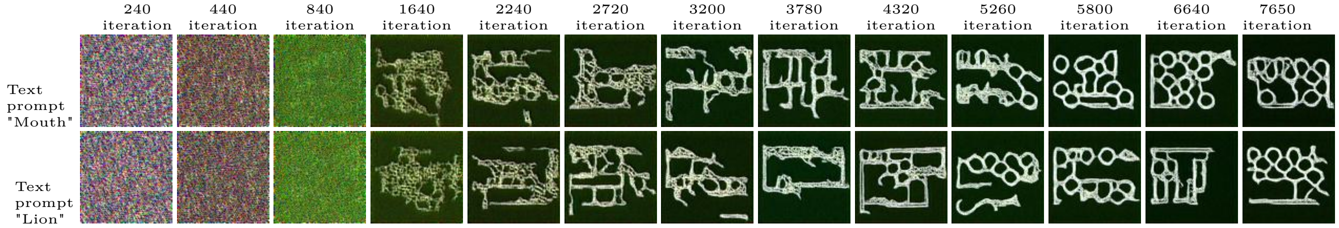

Table 6: A 13×2 Image table for Egyptian hieroglyphs generation (the failed case)

challenges. In contrast to pixel-based information, these abstract features pose difficulties for the model's accurate interpretation. Furthermore, neural networks like DeepSVG often operate in high-dimensional spaces due to their numerous parameters. Addressing the intricacies of high-dimensional spaces presents computational challenges, making the formal proof of statements in such spaces intensive. Traditional optimization proofs, primarily designed for convex problems, may not be applicable in this context. Thus, the abstract nature of SVG commands and computational complexity raises questions about model's capacity for robust reconstruction.

7 CONCLUSION AND FUTURE WORK

In this study, we delved into the intricacies of reconstructing Egyptian hieroglyphs, employing both pixel-based and vector-based methodologies. Our discernments highlight the superior performance of the pixel-based model, Imagen, in contrast to the vector-based counterpart, DeepSVG. To comprehensively evaluate the generalization capabilities and convergence of these models, we conducted a series of experiments, providing a nuanced review of the strengths and limitations inherent in each approach. As a noteworthy contribution to the research community, we furnish a dataset featuring Egyptian hieroglyph images in both pixel-based and vector-based formats.

The success demonstrated by the Imagen model in generalization prompts future exploration, where we aim to delve into advanced methodologies such as Imagen 2, Img2Vec, and VectorFusion. Imagen 2, being a recent development, intrigues us, and we seek to discern its advancements over the initial version. Furthermore, Img2Vec's unique approach of utilizing pixel-based images to reconstruct vector-based images aligns seamlessly with our project objectives, warranting a closer examination. Encouraged by positive outcomes associated with these methodologies, we plan to integrate them into our ongoing research.

REFERENCES

[1] Pierre Baldi. 2012. Autoencoders, unsupervised learning, and deep architectures. In *Proceedings of ICML workshop on unsupervised and transfer learning*. JMLR Workshop and Conference Proceedings, 37–49.

[2] Dor Bank, Noam Koenigstein, and Raja Giryes. 2023. Autoencoders. *Machine Learning for Data Science Handbook: Data Mining and Knowledge Discovery Handbook* (2023), 353–374.

[3] Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew M. Dai, Rafal Jozefowicz, and Samy Bengio. 2016. Generating Sentences from a Continuous Space. arXiv:1511.06349 [cs.LG]

[4] Alexandre Carlier, Martin Danelljan, Alexandre Alahi, and Radu Timofte. 2020. DeepSVG: A Hierarchical Generative Network for Vector Graphics Animation. arXiv:2007.11301 [cs.CV]

[5] Jingye Chen, Yupan Huang, Tengchao Lv, Lei Cui, Qifeng Chen, and Furu Wei. 2023. TextDiffuser: Diffusion Models as Text Painters. arXiv:2305.10855 [cs.CV]

[6] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath. 2018. Generative adversarial networks: An overview. *IEEE signal processing magazine* 35, 1 (2018), 53–65.

[7] Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. 2023. Diffusion models in vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2023).

[8] Carl Doersch. 2016. Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908* (2016).

[9] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. *Commun. ACM* 63, 11 (2020), 139–144.

[10] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. 2022. CLIPScore: A Reference-free Evaluation Metric for Image Captioning. arXiv:2104.08718 [cs.CV]

[11] Ajay Jain, Amber Xie, and Pieter Abbeel. 2022. VectorFusion: Text-to-SVG by Abstracting Pixel-Based Diffusion Models. arXiv:2211.11319 [cs.CV]

[12] Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. 2021. Variational diffusion models. *Advances in neural information processing systems* 34 (2021), 21696–21707.

[13] Diederik P Kingma and Max Welling. 2022. Auto-Encoding Variational Bayes. arXiv:1312.6114 [stat.ML]

[14] Diederik P Kingma, Max Welling, et al. 2019. An introduction to variational autoencoders. *Foundations and Trends® in Machine Learning* 12, 4 (2019), 307–392.

[15] Wanwan Li. 2021. Image Synthesis and Editing with Generative Adversarial Networks (GANs): A Review. In *2021 Fifth World Conference on Smart Trends in Systems Security and Sustainability (WorldS4)*. IEEE, 65–70.

[16] Wanwan Li. 2023. Synthesizing 3D VR Sketch Using Generative Adversarial Neural Network. In *Proceedings of the 2023 7th International Conference on Big Data and Internet of Things*. 122–128.

[17] Wanwan Li. 2023. Terrain synthesis for treadmill exergaming in virtual reality. In *2023 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*. IEEE, 263–269.

[18] Wanwan Li, Changyang Li, Minyoung Kim, Haikun Huang, and Lap-Fai Yu. 2023. Location-Aware Adaptation of Augmented Reality Narratives. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–15.

[19] Wanwan Li, Biao Xie, Yongqi Zhang, Walter Meiss, Haikun Huang, and Lap-Fai Yu. 2020. Exertion-aware path generation. *ACM Trans. Graph.* 39, 4 (2020), 115.

[20] John Nickolls, Ian Buck, Michael Garland, and Kevin Skadron. 2008. Scalable Parallel Programming with CUDA. *ACM Queue* 6, 2 (2008), 40–53. http://dblp.uni-trier.de/db/journals/queue/queue6.html#NickollsBGS08

[21] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. arXiv:1912.01703 [cs.LG]

[22] Ryan Po, Wang Yifan, Vladislav Golyanik, Kfir Aberman, Jonathan T Barron, Amit H Bermano, Eric Ryan Chan, Tali Dekel, Aleksander Holynski, Angjoo Kanazawa, et al. 2023. State of the art on diffusion models for visual computing. *arXiv preprint arXiv:2310.07204* (2023).

[23] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. arXiv:2103.00020 [cs.CV]

[24] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2023. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. arXiv:1910.10683 [cs.LG]

[25] Pradyumna Reddy, Michael Gharbi, Michal Lukac, and Niloy J. Mitra. 2021. Im2Vec: Synthesizing Vector Graphics without Vector Supervision. arXiv:2102.02798 [cs.CV]

[26] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-Resolution Image Synthesis with Latent Diffusion Models. arXiv:2112.10752 [cs.CV]

[27] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. 2022. Palette: Image-to-image

diffusion models. In *ACM SIGGRAPH 2022 Conference Proceedings*. 1–10.

[28] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. 2022. Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding. arXiv:2205.11487 [cs.CV]

[29] Unknown Egyptian Scribe. c. 1294 or 1290 - 1279 BC. Fragment of a Wall with Hieroglyphs from the Tomb of Seti I. The British Museum. http://www.egyptarchive.co.uk/html/british_museum_29.html (Jon Bodsworth) - https://www.egyptarchive.co.uk/.

[30] Kota Yamaguchi. 2021. CanvasVAE: Learning to Generate Vector Graphic Documents. arXiv:2108.01249 [cs.CV]

[31] Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. 2023. Diffusion models: A comprehensive survey of methods and applications. *Comput. Surveys* 56, 4 (2023), 1–39.