Court File No. T-1717-24

FEDERAL COURT

B E T W E E N:

SAMUELSON-GLUSHKO CANADIAN INTERNET POLICY AND PUBLIC INTEREST CLINIC

Applicant

- and -

ANKIT SAHNI

Respondent

AFFIDAVIT OF PHILLIP MITCHELL WILLIAMS

I, Phillip Mitchell Williams, of the City of Ottawa in the Province of Ontario, AFFIRM:

I. Statement of the Issues Addressed in the Report

1. This affidavit expresses my opinion with respect to image generation and image style transfer using artificial intelligence (AI) models and techniques. Except as otherwise indicated, I have personal knowledge of the matters to which I depose in this affidavit. Where I lack such personal knowledge, I have indicated the source of my information, and I believe such information to be true.

II. Qualifications and Areas of Expertise for Qualification as an Expert

2. I graduated from the University of Ottawa in 2019 with a B.Sc. degree in Chemical Engineering and Computing Technology. I received my M.Sc. in Machine Learning from University College London (UK) in 2024.

3. I am employed as a software engineer and AI researcher at Kinaxis, a supply chain management and sales and operation planning software company based in Kanata, Ontario, Canada. My current *curriculum vitae* is attached as **Exhibit A**.

4. I have published multiple papers on the use of AI in visual computing and image classification. A copy of my Google Scholar profile, including my patents and publications, is attached as **Exhibit B**.

III. Introduction

5. Machine Learning refers to a wide range of mathematical and statistical techniques that are used to automatically model relationships in a given dataset with minimal human oversight. Whereas rule-based systems apply human-written rules to perform a task, Machine Learning systems rely on patterns learned directly from data. In the context of Machine Learning, generative models refer to techniques that generate data, such as images or audio, based on training data and user inputs. This affidavit will examine the theoretical and practical aspects of generative models, as well as "style transfer," which is the specific technique of interest for this case and a particular type of generative model.

IV. Overview of Machine Learning

6. In this section, I will examine the motivation behind using Machine Learning models, how they function from a theoretical point of view, and how they are used in practice. First, I will examine the motivation behind Machine Learning models to better understand how and why Machine Learning is used and to better distinguish Machine Learning techniques from more conventional tools used for image generation.

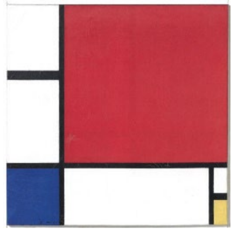
V. Motivation

7. Rule-based (or expert) systems are programs which use a set of human-defined rules, heuristics, or known facts to reason about a given task. This is an extremely powerful and useful paradigm when dealing with well-defined problems, but it typically fails when the problem is vague and hard to define in exact terms. Filters in Adobe Photoshop are an example of a rule-based system and use rules to modify an existing image in consistent ways to achieve various visual effects such as blurring. Notably, re-applying the same filter to the same input image with the same settings results in the same output.

8. In contrast, Machine Learning algorithms automatically extract relationships from sufficiently large datasets and apply these relationships to new input images. They are useful in settings where exact rules are hard to define, but a large collection of known samples is available. Image generation using Machine Learning algorithms uses probabilistic relationships to create *new* images—re-generating from the same input image or prompt with the same settings results in a new output.

9. In the context of generative models, image generation, and style transfer, I will examine the works of Piet Mondrian and Vincent Van Gogh to better illustrate rule-based and Machine Learning-based approaches.

(a) Composition with large red plane, yellow, black, gray and blue, 1921



(b) Composition no. II, 1930

(c) Composition with Yellow, Blue and Red, 1937-1943

Figure 1: Various works of artist Piet Mondrian[1]

(a) Starry Night, 1889

(b) Starry Night, 1888

(c) Country Road in Provence by Night, 1890

Figure 2: Various works of artist Vincent van Gogh[2]

10. First, the style of the artist Piet Mondrian (as shown in Figure 1) can be emulated by a rule-based system. One could imagine a set of rules for randomly generating squares and rectangles, filling them with various colors and separating them by thick black lines. While not creative like the originals, the output would, nevertheless, emulate Mondrian's visual style.

11. However, the style of Vincent van Gogh (as shown in Figure 2) is not so easily emulated by a rule-based system. While a distinct visual style is clearly present in van Gogh's works, it is not easily broken into its constituent components. We cannot express the style of van Gogh in a

[1] Piet Mondrian, *Composition with large red plane, yellow, black, gray and blue*, (1921) held at the Kunstmuseaum Den Haag [oil on canvas], online: < https://www.kunstmuseum.nl/en/collection/composition-large-red-plane-yellow-black-gray-and-blue>; Piet Mondrian, *Composition no II*, (1930) held in private collection [oil on canvas], online: < https://rkd.nl/images/253672>; Piet Mondrian, *Composition with Yellow, Blue and Red*, (1937-1943) held at the Tate Gallery [oil on canvas], online: < https://www.tate.org.uk/art/artworks/mondrian-composition-with-yellow-blue-and-red-t00648>.

[2] Vincent van Gogh, *Starry Night*, (1889) held at The Museum of Modern Art [oil on canvas], online: <https://www.moma.org/collection/works/79802>; Vincent van Gogh, *Starry Night*, (1888) held at the Musée D'Orsay [oil on canvas], online: <https://www.musee-orsay.fr/en/artworks/la-nuit-etoilee-78696>; Vincent van Gogh, *Country Road in Provence*, (1890) held at the Kröller-Müller Museum [oil on canvas], online: <https://krollermuller.nl/en/vincent-van-gogh-country-road-in-provence-by-night-1>.

set of simple rules that can be followed, as it is a combination of texture, palette, and other elements not easily summarized. However, we do have a comprehensive set of examples of van Gogh's works. Consequently, I would likely need to make use of Machine Learning techniques to generate new images in the style of van Gogh.

12. Machine Learning offers a versatile approach to tackling complex tasks such as generating images in the style of van Gogh. In the next section, I will explore generative models in more detail and provide insight into how models generate images.

VI. Generative Models

13. Generative Models are a broad class of Machine Learning techniques used to generate new data (such as images, text or audio) as a function of some input data. For example, Large Language Models and Diffusion Models are well-known categories of generative models. Furthermore, generative models are probabilistic, meaning that they do not generate the same output every time. For a given input, the generative model produces a range of similar but distinct outputs based on some underlying randomness within the model. The technical term for generating a new random output from a generative model is known as "sampling." As shown in Figure 3, the same human inputs can result in a wide variety of related images when sampling.

Figure 3: Multiple images sampled from Stable Diffusion 2.1 using the prompt "An astronaut riding a horse."

14. To obtain a useful model that can be used to sample interesting data or, in this case, aesthetically pleasing images, several elements are needed. First, we must choose the model architecture. The model architecture refers to the exact mathematical formulation and

specification for the machine learning model. The model architecture has profound implications for the quality and performance of the model. For example, more complex model architectures may generate higher-quality images at the cost of requiring more computational resources.

15. However, the model architecture is merely a blueprint for producing an actual model. The second element needed is a dataset on which to train the model. The dataset defines the task that the model will accomplish (such as text generation or image generation). Additionally, the size and quality of the dataset will largely determine the performance of the final model. For example, when performing image generation, a large dataset would provide a wide variety of samples, allowing the model to produce a diverse set of output images. Conversely, a small dataset would limit the model to a smaller set of visual aesthetics and scenes, and so a model trained on the smaller dataset would likely display a more constrained set of generated images compared to the same model architecture on a much larger dataset.

16. Finally, we must understand the training process. Building an effective AI system starts with a well-designed blueprint and a diverse, high-quality dataset, followed by a training process that fine-tunes its internal settings to consistently achieve the desired performance. Training is a colloquial term for the mathematical optimization of a model architecture and a collection of model parameters on a specific dataset or task. Put plainly, a model consists of an architecture and a set of numerical parameters. We then define a quantitative measure of how well or poorly a model performed on the task at hand, known as the loss function. A common example of a loss function is the deviation between the output of the model and the desired output. In the context of the dataset, the loss function, and the model architecture, the role of the training process is to produce a set of numerical parameters which provide the “best” behaviour as defined by the loss

function. When using the term "model," it often refers to the combination of model architecture and optimized model parameters.

17. In other words, the usefulness of the model is directly derived from the input dataset. The optimized model parameters, which are necessary for the generation and sampling of data from the model, are directly produced via mathematical and statistical techniques collectively referred to as training.

18. Next, I will examine the specific technique of interest for this court case: style transfer.

VII. Style Transfer

19. Style transfer refers to a family of generative models that can take the aesthetic style of one image and apply it to the content of another image. The style image refers to the image whose aesthetic style is copied, while the content image refers to the image whose content we would like to represent in the artistic style of the style image. To illustrate this, several artistic styles are applied to the same content image in Figure 4.

20. In his submission to the US Copyright Office, Mr. Sahni refers to the seminal paper "Exploring the structure of a real-time, arbitrary neural artistic stylization network" as the method used to create RAGHAV.[3] However, the authors prepared this paper within a specific research context, and it may be useful to examine several publications leading up to the final technique to better understand how the methodology came about. The copy of Mr. Sahni's response to the US Copyright Office that I was provided and reviewed is included as **Exhibit C**.

[3] Golnaz Ghiasi et al, "Exploring the structure of a real-time, arbitrary neural artistic stylization network" (Accepted as an oral presentation at British Machine Vision Conference, 2017) online: <https://arxiv.org/abs/1705.06830>, cited by Mr. Sahni on p 6 of Exhibit C.

Figure 4: Examples of style transfer from the paper; The top left image represents the content image, while the small thumbnails represent the style images. The final style transferred images for various artistic styles are shown.[4]

VIII. Background Information

21. The first relevant work that I will explore in this affidavit is "Image Style Transfer using Convolution Neural Networks."[5] This paper utilizes a particular type of model called a Convolutional Neural Network to perform style transfer. In this section, I will explore three topics central to style transfer: (1) convolutions, (2) Convolutional Neural Networks (CNNs) and (3) embeddings.

[4] Adapted from Leon A Gatys, Alexander S Ecker & Matthias Bethge, "Image Style Transfer Using Convolutional Neural Networks" in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Las Vegas, 2016) at 2414-2423.

[5] *Ibid*.

22. First, "convolutions" are a particular family of mathematical operations typically used in image or audio processing settings. In the case of images, a convolution is a small grid of numbers (typically 3x3 or 5x5) that is applied to small patches of the image at a time. Depending on the choice of convolution, different effects can be achieved, such as blurring, sharpening, or highlighting the edges in an image. Mathematically, $\mathcal{F}$ represents the input image, ω the convolution matrix, n the size of the kernel, i and j the vertical and horizontal positions within the grid being processed, and $\mathcal{G}$ the output image:

$$\mathcal{G} = \sum_{i=-n}^{n} \sum_{j=-n}^{n} \mathcal{F}_{i,j} \cdot \omega_{i,j}$$

23. In Figure 5, I demonstrate an example of a 2-dimensional convolution, while Figure 6 demonstrates the visual effects of several common types of convolutions.

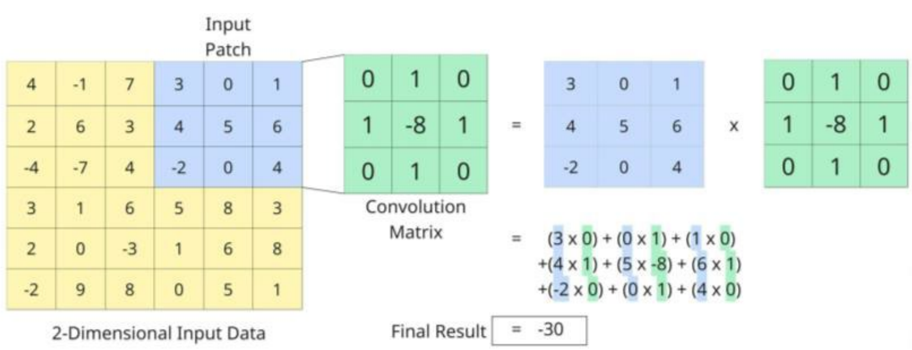


Figure 5: Worked example of a 2-dimensional convolution

(a) Original Image *(b) Blur Convolution* *(c) Sharpen Convolution*

$$\omega = \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \qquad \omega = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

Figure 6: Examples of a Blur and a Sharpen convolution. Note that selecting different values for the entries of the convolution matrix allows for a variety of visual effects to be captured.[6]

24. Second, CNNs are a particular type of Machine Learning model typically used for computer vision tasks. CNNs are constructed from stacked convolutions, and the training process optimizes the values in the convolution matrices for the task at hand. Individually, convolutions will only capture small-scale details of the images (such as edges). However, by stacking convolutions, the subsequent layers of the neural network will capture increasingly higher-level detail, starting with low-level texture, then shapes and contours, objects, and finally capturing the overall content of an input image. Put plainly, the sequential application of convolutions allows the content of the image to be represented at different levels of granularity. See Figure 7 for an illustration of a typical CNN architecture.

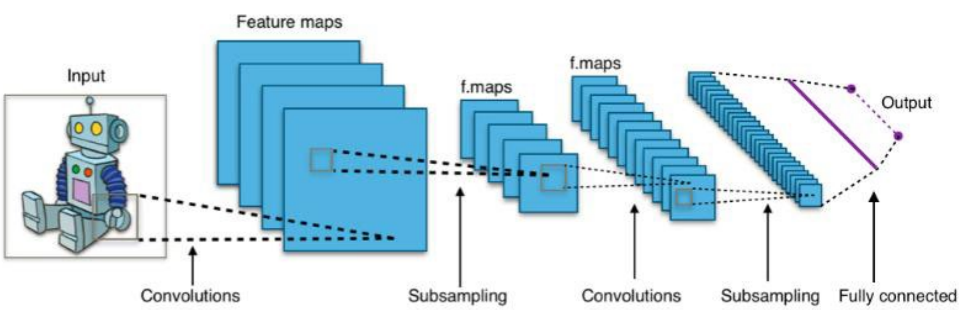


Figure 7: An example of a typical Convolutional Neural Network Architecture.[7]

---

[6] Michael Plotke, "Vd-Orig.png" (2013) online (image): <https://commons.wikimedia.org/wiki/File:Vd-Orig.png>; Michael Plotke, "Vd-Blur1.png" (2013) online (image): <https://commons.wikimedia.org/wiki/File:Vd-Blur1.png>; Michael Plotke, "Vd-Sharp.png" (2013) online (image): <https://commons.wikimedia.org/wiki/File:Vd-Sharp.png>.
[7] Aphex34, "Typical cnn.png" (2015), online (image): <https://en.m.wikipedia.org/wiki/File:Typical_cnn.png>.

25. Third, the intermediate representations obtained by successive applications of convolutions are often referred to as "embeddings". In general, an embedding is a numerical representation of the semantic content of the input data that includes surface-level patterns such as edges or colours, as well as high-level concepts and meaningful content. In the case of an image, the embedding would capture visual elements (such as "colour" and "texture"), as well as semantic content, such as specific objects present in the input image ("this is a cat"). Since the embeddings capture the semantic information of the input data in a quantitative manner, the similarity of two embeddings can be used as a measure of similarity between corresponding input images. If two embeddings are quite similar, then the two input images share many common elements. Alternatively, if two embeddings are very different, the input images differ significantly. Embeddings are a key aspect of style transfer.

IX. How Style Transfer Works

26. Plainly put, style transfer uses machine learning models to apply the aesthetic style of one image (called the style image) to the content of another image (called the content image). The goal is to produce a third image that matches the style of the style image while also matching the content of the content image.

27. Towards this end, researchers devised the concept of a "style-loss" and a "content-loss."[8] Recall that the intermediate outputs of a CNN produce embeddings, which capture the semantic meaning of the input image at various levels of granularity. Thus, if two images have similar embeddings in the lower layers of a CNN, they can be said to have similar texture or colour.

[8] Gatys et al, *supra* note 4 at 2417.

Meanwhile, if the images have similar embeddings at the top levels of a CNN, they have similar overall content or composition. Gatys et al. define the style loss as a function of the difference between embeddings of a pre-trained CNN at several layers and the content loss as the difference between top-layer embeddings of a pre-trained CNN.[9] We can express a style transfer loss (denoted as L_{total}), which is a combination of the style loss between the style image and the generated image and the content loss between the content image and the generated image:

$$L_{total} = \alpha L_{style} + \beta L_{content}$$ [10]

28. Subsequently, the process of style transfer becomes a straightforward matter of optimizing L_{total} via well-known mathematical techniques such as gradient descent. Additionally, by choosing the value of α and β, a user can determine how much or how little they would like to maintain style and content. See Figure 8 for an example of increasing the β value, resulting in a progressively more stylized output image.

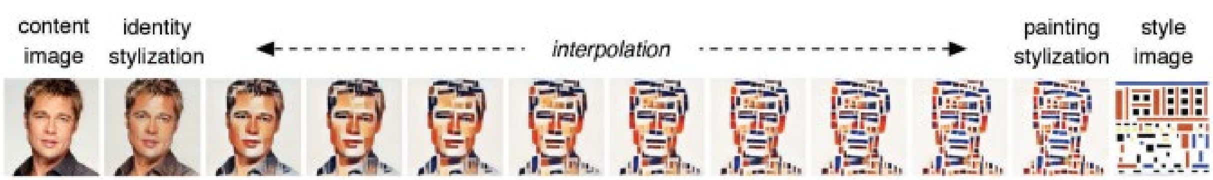


Figure 8: An example of varying the α and β parameters to increase the amount of style transfer.[11]

29. However, this adjustment is purely mathematical—it alters the weighted contribution of pre-trained embeddings in a loss function. Changing α and β adjusts the balance between statistical representations of style and content; it does not give the user creative control over the placement of visual elements, colour choices, or structural composition. This is not analogous to creative decisions like applying brushstrokes or altering composition manually. The generation process remains

[9] Gatys et al, *supra* note 4 at 2417.
[10] Nicholas Carlini et al, "Extracting Training Data from Diffusion Models" in *Proceedings of the 32nd USENIX Security Symposium* (Anaheim, 2023) at 5253-5270.
[11] Ghiasi et al, *supra* note 3 at 10.

automated and opaque: the internal calculations of the model, not the user's aesthetic judgment, determine the final output. The process is like planting a seed: while the user may choose the seed and the conditions, they do not control what grows or how it looks. Indeed, given the same content image and the same style image, small variations in the α and β values may lead the model to produce significantly different outputs.

30. The process of optimizing L_{total} directly via gradient descent techniques was found to be slow and computationally expensive.[12] As such, subsequent research identified several optimizations and novel applications of generative models to accelerate the process.

31. In 2016, Johnson et al. demonstrated that generative models, called style transfer networks, can directly produce a style-transferred image without the need for the iterative optimization process used in earlier gradient descent techniques.[13] This is accomplished by training a generative model to minimize the same L_{total} by outputting the style-transferred image directly in one shot. The limitation of this method is that a style transfer network needs to be trained for each desired aesthetic style. For example, if one wanted to perform style transfer in the style of van Gogh, then one would need to have trained a van Gogh-style transfer network. Consequently, there is a collection of known style networks that can be applied, and transferring styles outside of the catalogue of supported styles would require a lengthy training process.

32. To overcome the limitation of needing a model for each artistic style, Dumoulin et al. showed that a single style transfer network can be trained across all styles.[14] Once the shared style transfer network is trained, a small number of style transfer parameters are computed for

---

[12] Gatys et al, *supra* note 4 at 2421.

[13] Justin Johnson, Alexandre Alahi & Li Fei-Fei, "Perceptual Losses for Real-Time Style Transfer and Super-Resolution" in *Computer Vision – ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II* (Springer, 2016) at 694-711.

[14] Vincent Dumoulin, Jonathon Shlens & Manjunath Kudlur, "A Learned Representation for Artistic Style" in 5th International Conference on Learning Representations (France, 2017) online: <https://openreview.net/forum?id=BJO-BuT1g>.

each aesthetic style. The style transfer parameters can be thought of as a distillation of an artistic style and are derived from a collection of reference images. Thus, given a collection of sample images, one can extract the style parameters and then perform style transfer in that aesthetic style without the need to retrain a new generative model.

33. Finally, in the seminal paper referenced by Mr. Sahni in his submission to the US Copyright Office, "Exploring the structure of a real-time, arbitrary neural artistic stylization network," the authors Ghiasi et al. introduced a "style prediction network."[15] This model is trained to take a reference style image and estimate the style transfer parameters required by the style transfer network. This addition allows for a single arbitrary image to be used for style transfer. By passing a reference-style image into the style prediction network, the aesthetic style can be distilled from a single image and then used for arbitrary style transfer. See Figure 9 for a diagram of the style transfer architecture.

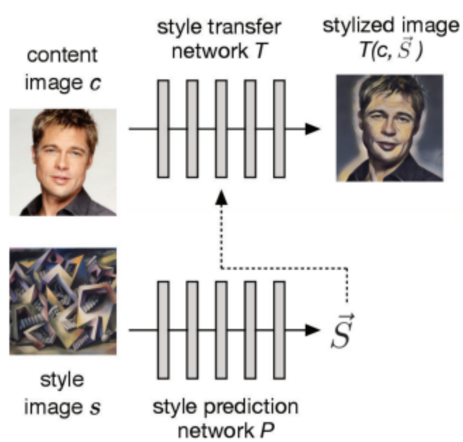


Figure 9: The style transfer architecture.[16]

34. This section described how style transfer uses convolutional neural networks to break down images into numerical summaries that capture key visual features, allowing artistic styles to be transferred between images by balancing style and content. This section also outlined

[15] Ghiasi et al, *supra* note 11.
[16] *Ibid*.

advancements that enable efficient, one-shot style transfer through improved generative models and prediction networks. In the next section, I will discuss how a user interacts with these kinds of generative models before concluding with a section on practical challenges and limitations when attributing the output of a generative model.

X. How a User Interacts with the Model

35. To understand how images generated with style transfer systems differ from the creative expressions of users, it is important to examine how users interact with such systems.[17]

36. The process begins with a user choosing two input images: a content image (typically a photograph) and a style image (typically a painting or texture). These inputs do not get combined in any direct way by the user. The user does not edit the image or apply a brushstroke. Instead, the style image is passed to a pre-trained neural network to extract the style parameters. Subsequently, the content image and the style parameters are passed to a second pre-trained neural network, which produces a third image: a new output that combines the visual "style" of one with the "content" of the other.

37. The important thing to understand is that the AI model generates a new image from scratch every time based on probabilities it has learned from its training data. The generation process is opaque to the user, and minor modifications to the input images or loss parameters can lead to significantly different results. The final output is, therefore, neither a modification of the content image nor a rearrangement of style image elements, but rather a new output generated entirely by the AI through internal algorithmic operations.

[17] Such as the one identified by Ghiasi et al., *supra* note 3.

38. In this way, the content and style images serve more like inspiration or direction—they are the idea, not the expression. To use a simple analogy: if the user says, "paint me a picture of my cat in the style of van Gogh," the user has given an idea, but the AI determines the location of the eyes, the rendering of the fur, and the appearance of brushstrokes and shapes. The user cannot specify those things.

39. That's the key point: while the user selects the inputs—and that selection may involve taste, judgment, or even artistic intuition—the final arrangement of pixels in the generated image is determined by the internal mechanics of the AI model. The user does not and cannot control how the final image will be composed or what specific elements it will contain.

40. As a result, the user's input may influence the general direction of the result but does not determine its output. The output is generated by the model, using learned patterns and statistical representations of images in its training data.

XI. Challenges and Limitations

41. First, the final output of a generative model is entirely dependent on the underlying dataset that was used to produce the trained model weights. Consequently, when using such generative models to generate art, the provenance of the model weights is a key concern. In particular, the trained model is essentially derived from the initial dataset, and so the provenance of the underlying data may be a concern.

42. Additionally, generative models used to create art can sometimes fail in unexpected ways, making it hard to determine if a generated output is truly novel. For instance, diffusion models can, under the right conditions, produce an image that is nearly an exact copy of one found in their

training data, meaning that a user might unknowingly recreate an existing work.[18] Although the style transfer method discussed by Ghiasi et al. does not use diffusion models, recent developments have applied diffusion models to style transfer as well, raising similar concerns.[19] As a result, extra care must be taken to ensure that AI-generated art is not simply reproducing existing works, especially since there are currently no efficient methods to automatically flag such reproductions without manually comparing the output to a corpus of known artworks.

43. Finally, the probabilistic nature of generative models may make it difficult to attribute creativity to a human-driven process versus the underlying randomness of the models. As described in the section *How a User Interacts with the Model*, even when a user selects particular inputs, they cannot control how the model interprets those inputs in the final output. The specific arrangement of visual elements is determined by the model's internal statistical processes, not by human instruction. For example, if a user were to craft a very particular set of input images and style images to achieve a particular outcome, it can be difficult to ascertain if the curation of the inputs truly led to the desired output as the model may have produced the desired output purely by chance without any impact of the curation process. In fact, "Prompt Engineering" is a field of study dedicated to crafting inputs to produce the best outputs from generative models. However, the current consensus is that the process is largely driven by trial-and-error and heuristics as we currently lack mechanisms to precisely drive the outputs of generative models via user input. This area touches upon the challenges of alignment. Alignment means ensuring that AI-driven systems act in accordance with a user's goals or preferences and is an open research problem.

[18] Carlini et al, *supra* note 10.
[19] Ghiasi et al, *supra* note 3; Jiwoo Chung, Sangeek Hyun & Jae-Pil Heo, "Style Injection in Diffusion: A Training-free Approach for Adapting Large-scale Diffusion Models for Style Transfer" accepted by *The IEEE/CVR Conference on Computer Vision and Pattern Recognition 2024*, online: <https://arxiv.org/pdf/2312.09008>.

44. The output image of a program like that used by Mr. Sahni is not the result of any human exercise; rather, the generation process is governed by statistical optimization and learned weights. While a user may select inputs, the actual arrangement of visual elements in the output is determined by the model's internal processes, not by the human operator.

XII. Conclusion

45. In conclusion, I have examined in detail how style transfer is performed using Machine Learning models. I have explored some of the theoretical and practical aspects of such models, as well as highlighting some of the challenges and limitations of the current state of the art.

Sworn remotely by Phillip Mitchell Williams at the City of Ottawa in the Province of Ontario, before me on Friday, April 11, 2025 in accordance with O. Reg. 431/20, Administering Oath or Declaration Remotely. Signed by: David Fewer 168BAE5D783749E... Commissioner for Taking Affidavits	Signed by: Phillip Williams EF73AAB6FA4C458... Phillip Mitchell Williams

LIST OF AUTHORITIES

Secondary Sources		
1.	Nicholas Carlini et al, "Extracting Training Data from Diffusion Models" in *Proceedings of the 32nd USENIX Security Symposium* (Anaheim, 2023) at 5253-5270	
2.	Jiwoo Chung, Sangeek Hyun & Jae-Pil Heo, "Style Injection in Diffusion: A Training-free Approach for Adapting Large-scale Diffusion Models for Style Transfer" accepted by *The IEEE/CVR Conference on Computer Vision and Pattern Recognition 2024*	
3.	Vincent Dumoulin, Jonathon Shlens & Manjunath Kudlur, "A Learned Representation for Artistic Style" in 5th International Conference on Learning Representations (France, 2017)	
4.	Leon A Gatys, Alexander S Ecker & Matthias Bethge, "Image Style Transfer Using Convolutional Neural Networks" in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Las Vegas, 2016) at 2414-2423	
5.	Golnaz Ghiasi et al, "Exploring the structure of a real-time, arbitrary neural artistic stylization network" (Accepted as an oral presentation at British Machine Vision Conference, 2017)	
6.	Justin Johnson, Alexandre Alahi & Li Fei-Fei, "Perceptual Losses for Real-Time Style Transfer and Super-Resolution" in *Computer Vision – ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II* (Springer, 2016) at 694-711	
Images and Artwork		
7.	Aphex34, "Typical cnn.png" (2015)	Figure 7
8.	Piet Mondrian, *Composition with large red plane, yellow, black, gray and blue*, (1921) held at the Kunstmuseaum Den Haag [oil on canvas]	Figure 1
9.	Piet Mondrian, *Composition no II*, (1930) held in private collection [oil on canvas]	Figure 1
10.	Piet Mondrian, *Composition with Yellow, Blue and Red*, (1937-1943) held at the Tate Gallery [oil on canvas]	Figure 1
11.	Michael Plotke, "Vd-Orig.png" (2013)	Figure 6
12.	Michael Plotke, "Vd-Blur1.png" (2013)	Figure 6
13.	Michael Plotke, "Vd-Sharp.png" (2013)	Figure 6
14.	Vincent van Gogh, *Starry Night*, (1889) held at The Museum of Modern Art [oil on canvas]	Figure 2
15.	Vincent van Gogh, *Starry Night*, (1888) held at the Musée D'Orsay [oil on canvas]	Figure 2
16.	Vincent van Gogh, *Country Road in Provence*, (1890) held at the Kröller-Müller Museum [oil on canvas]	Figure 2

TAB

A

This is **Exhibit "A"** to the

Affidavit of **Phillip Williams**, solemnly affirmed remotely this

11th day of April 2025

Signed by:

David Fewer

168BAF5D783749E...

David Fewer, Commissioner for Taking Oaths

Phillip Williams

Machine Learning Researcher and Data Engineer

phillip.mitchel.williams@gmail.com Ottawa, CAN in/phillipwilliamsML philliams.com

Summary

I am a software engineer and AI Researcher with 5 years industry experience, a Master's of ML with a thesis in Generative AI from University College London and a double undergrad in Chemical Engineering and Computer Science from the University of Ottawa. Generative AI has shown a lot of promise in fuzzy domains such as images. However, fields such as material science and medicine may benefit from more precise formalisms. My thesis explores a novel way to frame generative AI as a constraint satisfaction problem using Signed Distance Functions, providing more exact conditioning and stronger guarantees when generating novel samples. In fact, the approach uses SQL-style queries to perform multi-objective conditioning and has shown success in generating both images, and novel antibiotics that will be tested in-vivo.

Education

Msc Machine Learning (In Progress) *University College London* **London, UK** *2023-2024*

Masters Thesis: Computational Design of Small Molecules and Drugs via Constraint Queries.

BASc Chem. Eng and BSc Computing Technology *University of Ottawa* **Ottawa, Canada** *2014-2019*

Undergraduate Thesis: Computational Design of Chemicals using Deep-Learning and High-Throughput Screening

Skills

Leadership: Scrum/Agile, public speaking (conferences & guest lectures), mentorship, project management
ML: Julia, Python, Low-Level optimization, GPU-acceleration, Computer Vision, Generative Models, Material Science
Cloud Computing: Kubernetes, Docker, Argo Workflow, Postgres, Dask, Spark, Distributed Computing, Parallel Computing

Experience

ML Team Lead, *(Kinaxis)* **Ottawa, ON** *03/2022 - 09/2023*

- Responsible for improvement and development of keystone Feature Generation service. Re-designed legacy service architecture to make better use of spark clusters while improving code quality by using a functional-inspired design. Changes were implemented as a full internal re-write, which led to a 100x improvement in compute usage and runtime, an increase of test code coverage to >85% as well as implementing additional features requested by large customers.
- Managed team of 8 developers. Performed epic planning and breakdown, as well as day-to-day SCRUM tasks such as sprint planning and running stand-ups and retrospectives. Provided technical direction, career guidance, mentorship to developers within the ML department as well as conducting interviews and hiring process for open positions
- Participated in technical leadership and evangelism internally, running a Python Guild to disseminate cutting edge ideas and best practices within the organization. Popular presentations included "Property-Based Testing". Key member of the academic outreach team, giving guest lectures on ML in Supply Chain at various universities across North America.

ML Developer II, *(Kinaxis)* **Ottawa, ON** *05/2019 - 03/2022*

- Worked with distributed and cloud computing for large scale time series forecasting. Designed and implemented a chunk-based embarrassingly parallel forecasting system using Dask and LightGBM, and deployed to Kubernetes with Docker and Helm. I was additionally responsible for the design and implementation an Automatic Machine Learning pipeline to flexible compute features for time series forecasting using custom table metadata.
- Proposed novel chunk-based performance optimization, reducing the total runtime of the forecasting solution from >12 hours per run to < 30 mins by re-formulating the problem to be embarrassingly parallel, eliminating costly blocking operations. Additionally, I identified and fixed a memory leak in Dask, leading to a 5x reduction of memory usage of long running pods.
- Implemented General Purpose solver that allowed for Demand Plans to be optimized, yielding on time demand improvements of more than 10% on real-world datasets, winning second place at the Kinaxis R&D Hackathon
- Responsible for mentoring other developers as well as interns, presenting results to technical and non-technical stakeholders, holding knowledge sharing sessions and prepared workshops to demonstrate various ML and design concepts across ML teams

Machine Learning Intern, *(Kinaxis)* **Ottawa, ON** *Summer 2018*

Data Science Intern, *(Shopify)* **Ottawa, ON** *Summer 2017*

Extreme Blue Intern - Technical, *(IBM)* **Ottawa, ON** *Summer 2016*

Research Assistant, *(Lessard Resarch Group)* **Ottawa, ON** *10/2015 - 05/2016*

Notable Projects

1. **Master's Thesis in Generative AI for Drug and Chemical Design:** Demonstrated a novel approach to generative AI that allows for SQL-style queries to be used to condition generative models. Publication involved several novel mathematical innovations, implementing a custom high-dimensional numerical solver, training custom deep models using PyTorch, and building a query language compiler. Will be submitting to NeurIPS with my supervisor prof. Brooks Paige from UCL. Further work includes in-vivo testing in collaboration with the Machine Biology Group at uPenn.

2. **Technical Lead for large-scale industrial project:** Led implementation of a terabyte scale automatic Machine Learning service for time-series forecasting. Designed architecture and provided technical direction for implementation. Managed team of 8 developers, negotiated with multiple teams and stake holders, coordinated efforts across 50+ developers and lead to a 100x improvement in compute while implementing additional customer features.

3. **Multiple publications and patents:** Have published multiple papers in pure ML and applying ML to nanotechnology. Have been granted patents for my work in industry. I also often give talks and guest lectures. Google Scholar Profile.

4. **Material Science focused ML Research:** During my undergrad, I completed an undergraduate thesis with the Lessard Research Group on using ML to accelerate material discovery. I also participated in high-impact publications using Computer Vision to characterize crystalline structures. Presented results at Materials Research Society Symposium.

Publications

1. Dindault, C., King, B., **Williams, P.**, Absi, J. H., Faure, M. D., Swaraj, S., & Lessard, B. H. (2022). Correlating Morphology, Molecular Orientation, and Transistor Performance of Bis (pentafluorophenoxy) silicon Phthalocyanine Using Scanning Transmission X-ray Microscopy. Chemistry of Materials.

2. Mirka, B., Rice, N. A., **Williams, P.**, Tousignant, M. N., Boileau, N. T., Bodnaryk, W. J., ... & Lessard, B. H. (2021). Excess Polymer in Single-Walled Carbon Nanotube Thin-Film Transistors: Its Removal Prior to Fabrication Is Unnecessary. ACS nano, 15(5), 8252-8266.

3. Smith, K. E., & **Williams, P.** (2019, May). A Shallow Learning-Reduced Data Approach for Image Classification. In Canadian Conference on Artificial Intelligence (pp. 345-351). Springer, Cham.

4. Smith, K. E., **Williams, P.**, Bryan, K. J., Solomon, M., Ble, M., & Haber, R. (2018, July). Shepard interpolation neural networks with k-means: a shallow learning method for time series classification. In 2018 International Joint Conference on Neural Networks (IJCNN) (pp. 1-6). IEEE.

5. **Williams, P.** (2016, December). SINN: shepard interpolation neural networks. In International Symposium on Visual Computing (pp. 349-358). Springer, Cham.

Full list of publications available on my google scholar profile.

Patents

1. SYSTEMS AND METHODS FOR PARAMETER OPTIMIZATION – Kinaxis, granted November 29th, 2022

2. ANALYSIS AND CORRECTION OF SUPPLY CHAIN DESIGN THROUGH MACHINE LEARNING – Kinaxis, granted May 3rd, 2020

Conference Talks and Guest Lectures

1. "AI/ML in Retail Supply Chain Management", **Williams P.**, Sengupta O., Petosa Z., University of Dalhousie, Fall 2022

2. "Supply Chain Resilience Requires Robust Information", **Williams P.**, Mitchell-Guthrie P., University of South Carolina, Spring 2022

3. "ML for Demand Sensing in Supply Chain", **Williams P.**, Perryman O., University of Waterloo, Spring 2022

4. "Interactive Graphical Software for the Automatic Characterization of Nanoscale Objects Using Computer Vision", **Williams P.**, Materials Research Society Symposium, Fall 2021

5. "Automatic Characterization of Single-Walled Carbon Nanotube Film Morphologies Using Computer Vision", **Williams P.**, Materials Research Society Symposium, Spring 2021

TAB

B

This is **Exhibit "B"** to the

Affidavit of **Phillip Williams**, solemnly affirmed remotely this

11th day of April 2025

Signed by:

David Fewer

168BAF5D783749E...

David Fewer, Commissioner for Taking Oaths

Phillip Williams

Department of Computer Science, University College London

Machine Learning
Materials Science
Nanotechnology

	All	Since 2020
Citations	76	62
h-index	5	3
i10-index	2	2

0 articles	2 articles
not available	available

Based on funding mandates

TITLE	CITED BY	YEAR
Excess Polymer in Single-Walled Carbon Nanotube Thin-Film Transistors: Its Removal Prior to Fabrication Is Unnecessary B Mirka, NA Rice, P Williams, MN Tousignant, NT Boileau, WJ Bodnaryk, ... ACS nano	36	2021
Correlating Morphology, Molecular Orientation, and Transistor Performance of Bis (Pentafluorophenoxy) Silicon Phthalocyanine Using Scanning Transmission X-Ray Microscopy C Dindault, B King, P Williams, JH Absi, MDM Faure, S Swaraj, ... Chemistry of Materials 34 (10), 4496-4504	10	2022
Shepard interpolation neural networks with k-means: a shallow learning method for time series classification KE Smith, P Williams, KJ Bryan, M Solomon, M Ble, R Haber 2018 International Joint Conference on Neural Networks (IJCNN), 1-6	9	2018
SINN: shepard interpolation neural networks P Williams International Symposium on Visual Computing, 349-358	8	2016
Time series classification with shallow learning shepard interpolation neural networks KE Smith, P Williams Image and Signal Processing: 8th International Conference, ICISP 2018 …	6	2018
Deep convolutional-shepard interpolation neural networks for image classification tasks KE Smith, P Williams, T Chaiya, M Ble Image Analysis and Recognition: 15th International Conference, ICIAR 2018 …	5	2018
A shallow learning-reduced data approach for image classification KE Smith, P Williams Advances in Artificial Intelligence: 32nd Canadian Conference on Artificial …	2	2019
Systems And Methods For Parameter Optimization S Ouellet, P Williams, N Stanley, J Downing, L Hebert US Patent US-11514328-B2		2022
Analysis and correction of supply chain design through machine learning Phillip Williams, Marcio Oliveira Almeida, Zhen Lin, Behrouz Haji Soleimani ...		2020

TITLE	CITED BY	YEAR
US Patent US10,846,651		
Solving Composable Constraints for Inverse Design Tasks P Williams, B Paige		

TAB
C

This is **Exhibit "C"** to the

Affidavit of **Phillip Williams**, solemnly affirmed remotely this

11th day of April 2025

Signed by:

David Fewer

168BAF5D783749E...

David Fewer, Commissioner for Taking Oaths

Dated: April 14, 2022

Dear Madam/Sir,

[THREAD ID:1-55FC7US]

I am deeply appreciative and grateful for the time that you may have spent on thinking through the circumstances of this peculiar copyright application, and for your detailed and precise feedback. Please find my response to your questions below, for your kind consideration:

Part 1: RAGHAV Artificial Intelligence Painting App: Background, underlying technology and operational mechanism

1.1 Biological parallel between Neural Networks and brain

One of the main differences between Machine Learning and other computer algorithms is that Machine Learning learns a vast set of rules based on the data fed into it. On the other hand, other algorithms have to rely on the programmer to type in a set of predefined rules.

The machine learning algorithm behind RAGHAV is based on the Machine Learning subfield called Neural Networks.

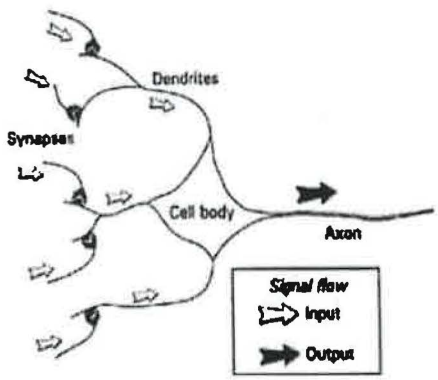


Figure 1.1 Essential components of a neuron shown in stylized form.

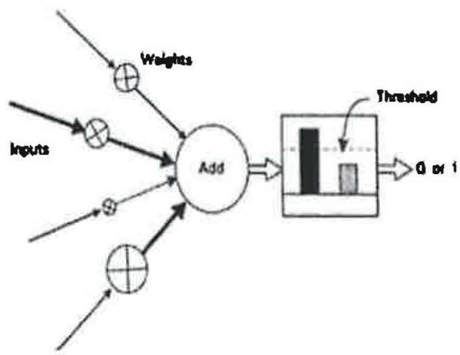
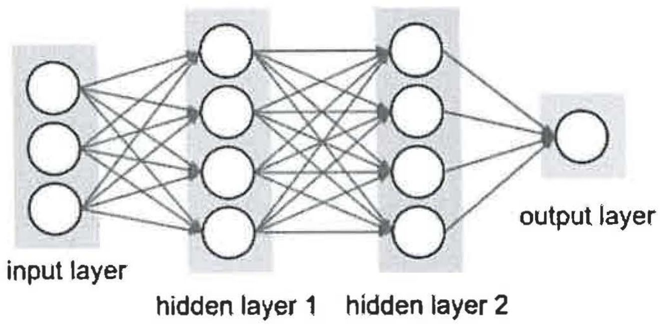


Figure 1.2 Simple artificial neuron.

Figure 1: Biological and Artificial Neurons [ref]

Neural Networks are programmed structures inspired from biological neurons of the nervous system. A biological neuron (Figure 1, Left) takes several incoming signals through synapses, electrochemical junctions located on dendrites, branches of the neuron cell. The cell body processes all the signals and generates a resulting signal based on a threshold which gets transmitted to other neurons through the axon. Similarly, an Artificial Neuron (Figure 1, Right) takes values as inputs from multiple artificial neurons, processes them using matrix multiplications (using values called weights) and other operations, and outputs the resulting signals to other artificial neurons.

1.2. Neuron, layer, CNNs, feature extraction



A 3-layer neural network with three inputs, two hidden layers of 4 neurons each and one output layer.

Figure 2: Neural Network [ref]

Many artificial neurons form a layer, and many layers form a Neural Network (Figure 2). An input layer can be pixel values of an image, numerical representations of words in a text, descriptive values in tabular data etc. The output layer can be a label predicting a category like 'dog' in an image, 'price' of a house given descriptive feature values of the house, next word prediction given a sequence of words, etc. The hidden layers are latent representations which form learnt intermediate features required to predict the output from given input. With each pair of input and output training data provided to the neural network, it updates its weights in the layers such that the output can be generated for the given input. When it learns using all data in the entire training set, we hope for the neural network to have generalized, that is learnt enough representations to produce a correct output for any new unseen input.

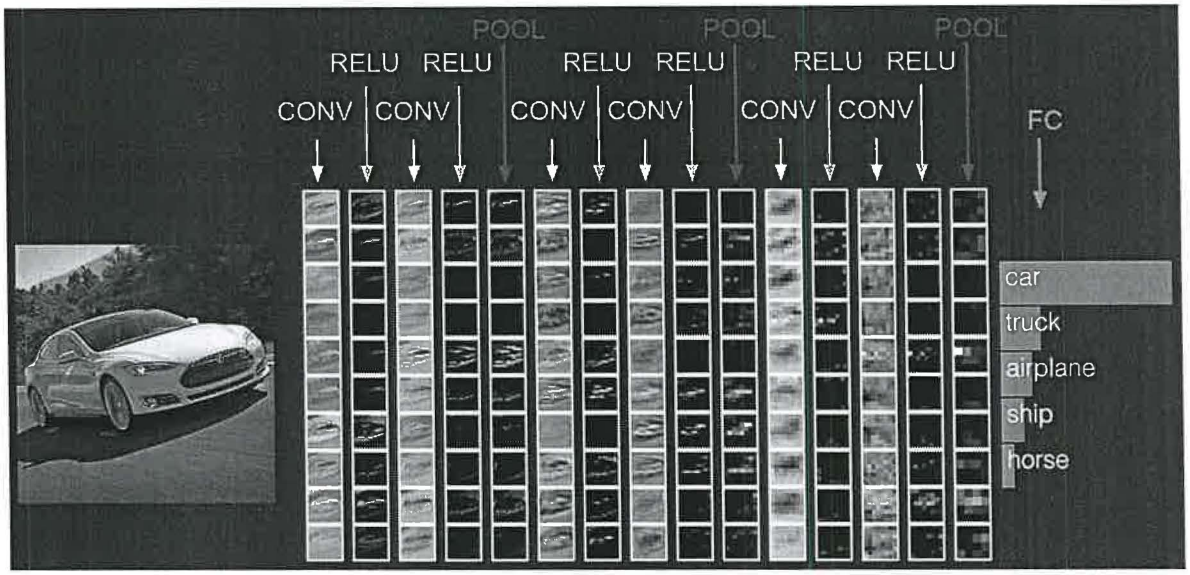


Figure 3: Convolutional Neural Network [ref]

A Convolutional Neural Network (Figure 3) (CNN) is a neural network with efficient operations in its layers for learning features from input images. It learns basic features like edge detection in its initial layers, then using these simple features, learns more complex features like textures and patterns in subsequent layers, and then using those learns features of complex objects like dog noses, human eyes and flowers in later layers of the network. (Figure 4)

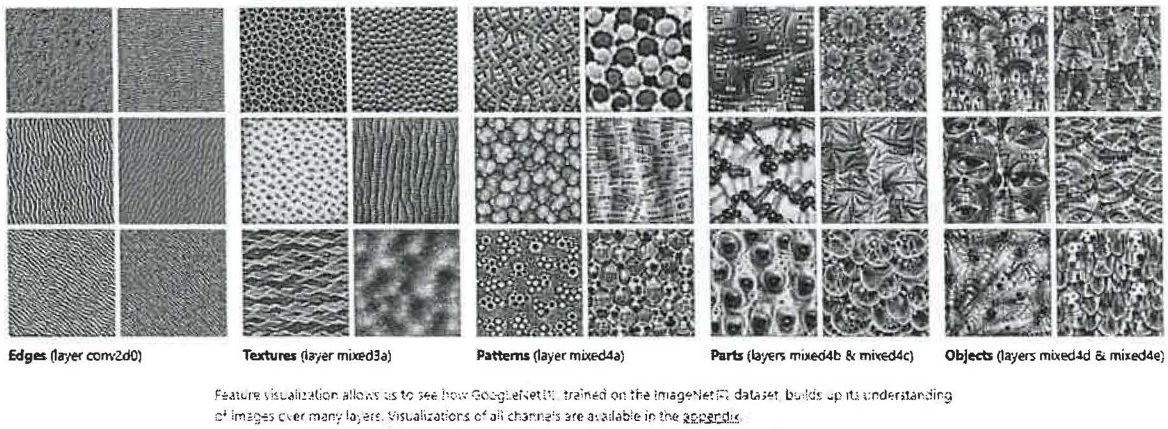


Figure 4: Features which layers learn in a CNN [ref]

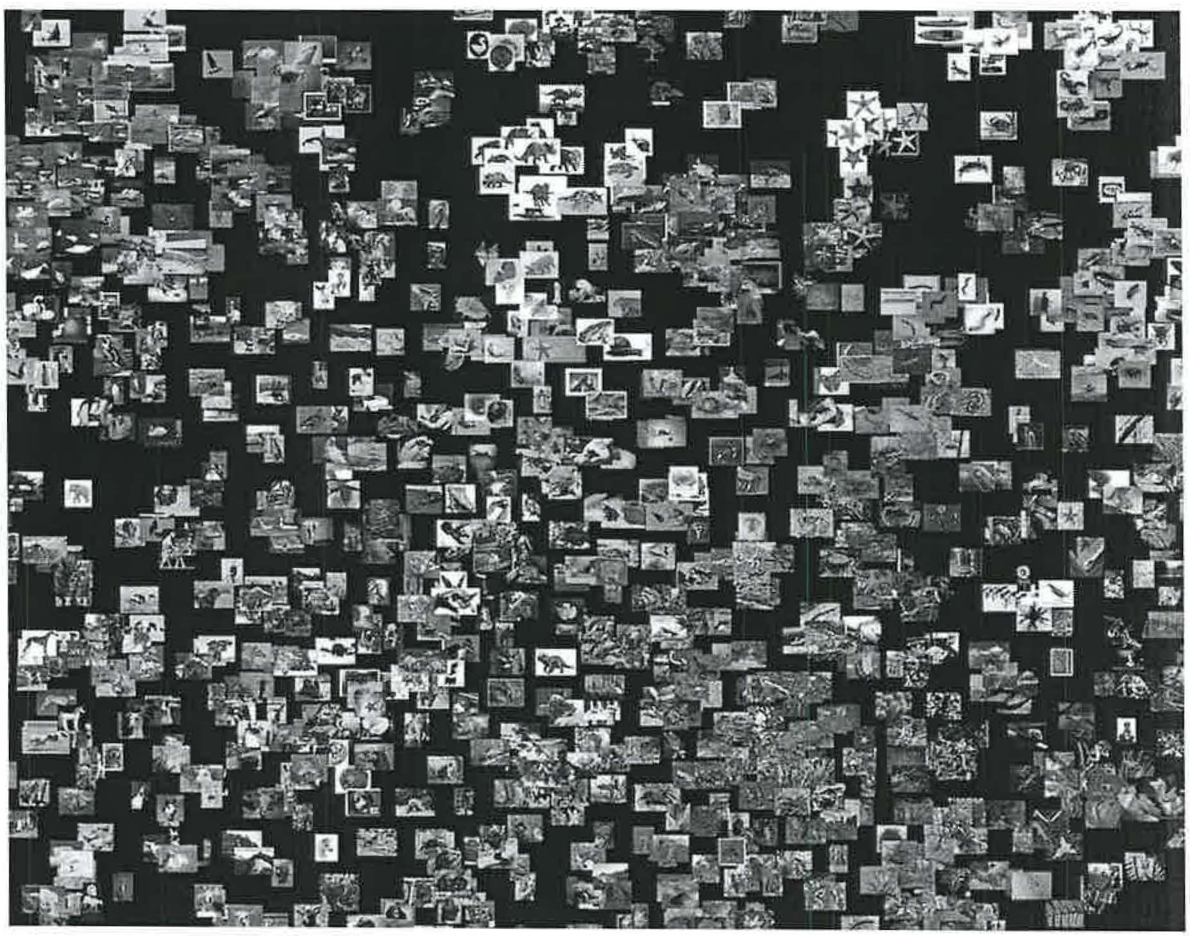


Figure 5: Feature representation space (projected to 2D space) learnt by a CNN [ref]

A CNN can learn a multi-dimensional representation space where similar images are closer together. If we pass input images to the CNN, and extract the representational vectors from its penultimate layer, then vectors from similar images (or images belonging the same category) will cluster together. (Figure 5)

1.3 **Neural artistic style transfer**

RAGHAV is based on a technique called **Neural Style Transfer** which is built using **CNNs**. Neural Style Transfer is a technique that allows us to generate an image with the same "content" as a base image, but with the "style" of our chosen picture.

Specifically, RAGHAV is built with a variant of Neural Style Transfer using the research paper "Exploring the structure of a real-time, arbitrary neural artistic stylization network" [ref]. A CNN is used to extract the features for the content and style images. (Figure 6).

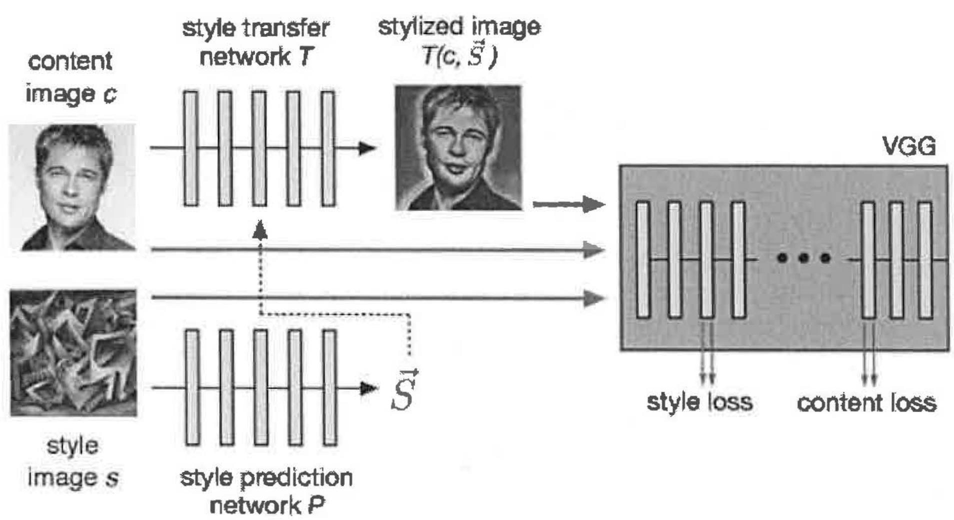


Figure 6: Architecture for Neural Style Transfer [ref].

It is based on the following definitions:

1. Two images are similar in content if their high-level features as extracted by an image recognition system are close in Euclidean distance.

2. Two images are similar in style if their low-level features as extracted by an image recognition system share the same spatial statistics. This is motivated by the hypothesis that a painting style may be regarded as a visual texture. Literature suggests that repeated motifs representative of a visual texture may be characterized by lower order spatial statistics. Images with identical lower-order spatial statistics appear perceptually identical and capture a visual texture.

Here, the image recognition system refers to a CNN, trained with image recognition task on a large corpus of 14M images called ImageNet. It is then trained for the task of Neural Style Transfer with training content and style images. After training, <u>any new unseen image</u> can be used as a style image. (Figure 7)

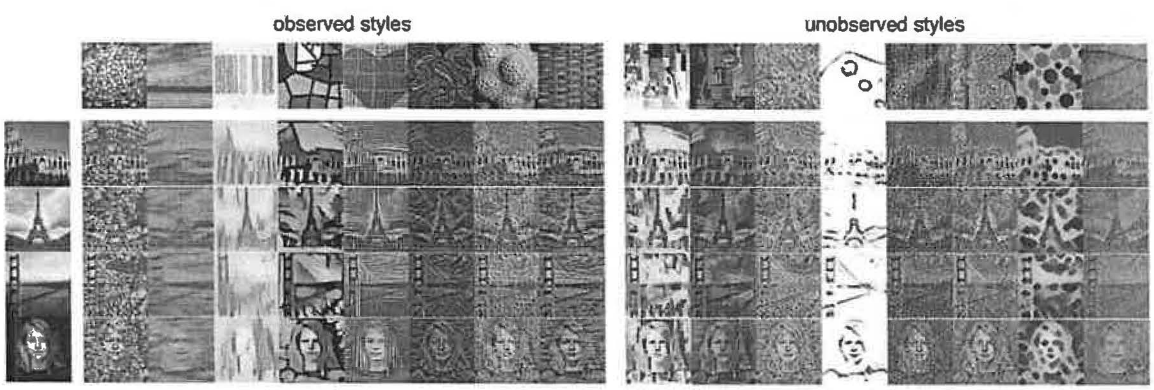


Figure 1: Stylizations produced by our network trained on a large corpus of paintings and textures. The left-most column shows four content images. Left: Stylizations from paintings in training set on paintings (4 left columns) and textures (4 right columns). Right: Stylizations from paintings never previously observed by our network.

Figure 7: New unseen styles can be used for Neural Style Transfer <u>[ref]</u>

1.4 **Creative Aspects of the specific Neural Style Transfer algorithm**

A variable value determining the amount of style transfer between content and style can also be specified which leads to different outputs for different values. (Figure 8)

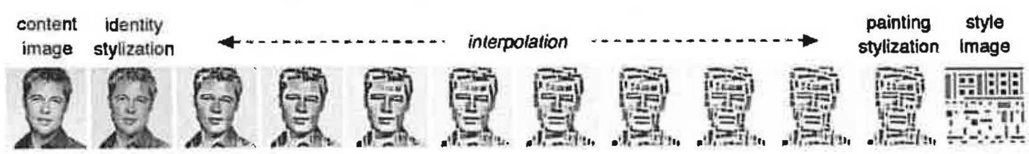


Figure 8: Linear interpolation between identity transformation and unobserved painting. Note that the identity transformation is performed by feeding in the content image as the style image.

Figure 8: Amount of Content and Style Transfer can be controlled

Using multiple styles is also possible for a single content image by extracting and using the features from all of the style images.

The embedding representation space learnt by the CNN captures semantic structure of styles, that is, semantically similar styles would be clustered together.

The structure of the embedding representation space learnt by the CNN also permits novel exploration. The CNN model can capture a local manifold from an individual artist or painting style (Figure 9). The embedding space can be explored and new stylizations can be generated by varying local style changes for a specific painting style. Thus, new styles can be used (either entirely different or a variation of a given style image) for a different output each time for the same content image.

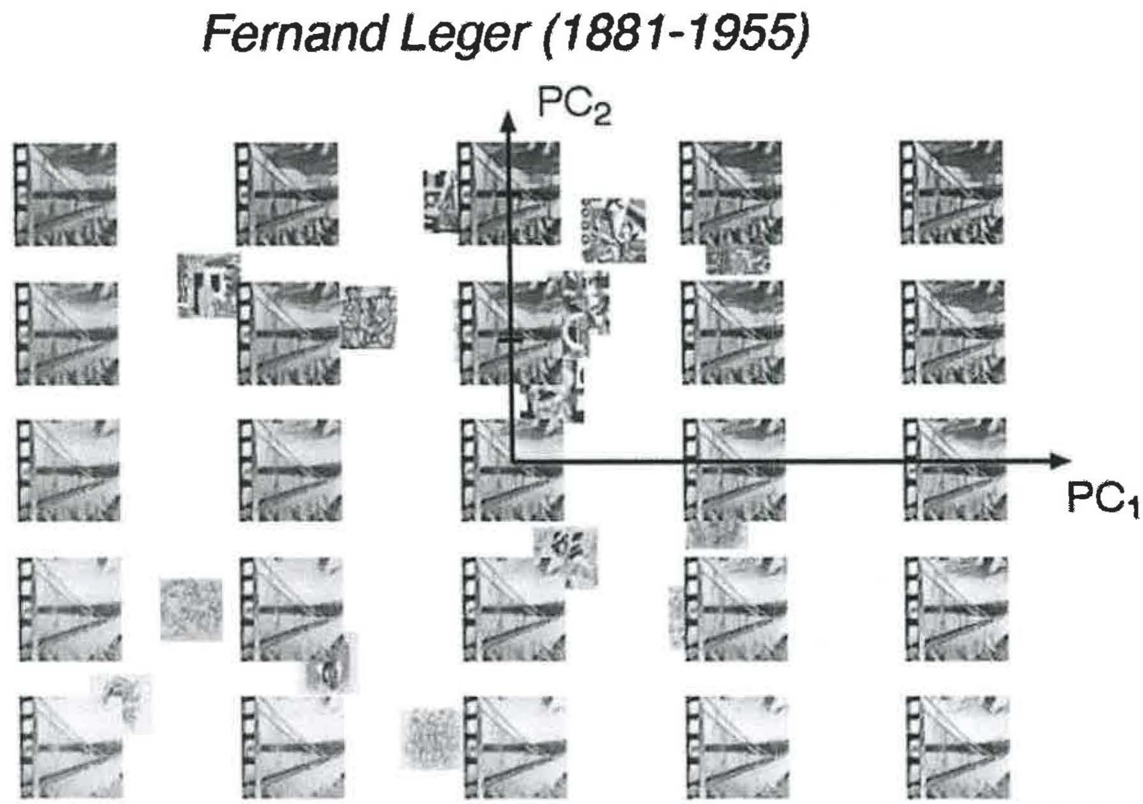


Figure 7: Exploring the artistic range of an artist using the embedding representation. Calculated two-dimensional principal components for a given artist and plotted paintings from artist in this space. The principal component space is graphically depicted by the artistic stylizations rendered on a photograph of the Golden Gate Bridge. The center rendering is the mean and each axis spans ±4 standard deviations in along each axis. Each axis tick mark indicates 2 standard deviations. Left: Paintings and principal components of Janos Mattis-Teutsch (1884-1960). Right: Paintings and principal components of Fernand Leger (1881-1955). Please zoom in on electronic version for details.

Figure 9: New styles based on an artist's artistic range can be used on the fly

Part 2: SURYAST: Step-by-step walkthrough

2.1 In the context of the abovementioned description of how RAGHAV works, the following is a step-by-step walkthrough of how the subject artwork 'SURYAST' was created using RAGHAV.

2.2 As explained above, RAGHAV accepts two inputs from the user. One input image is the style input, and the other is the content input. For the content input, I used an original photograph clicked by me using my phone's camera. The photograph is provided below for reference. As the author thereof, I am the sole owner of all rights (including copyright) in the photograph.

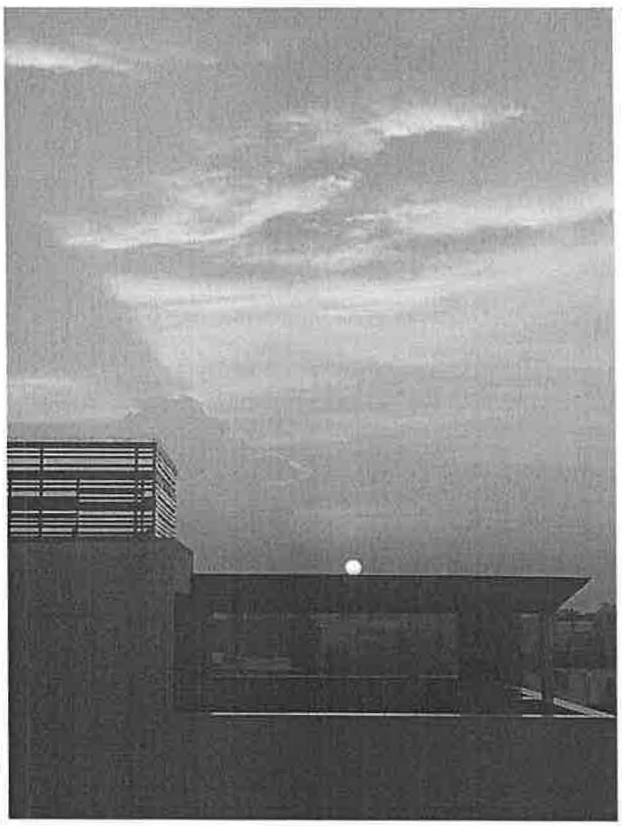


Figure 10: Photograph clicked and owned by Ankit Sahni; provided as the content input to RAGHAV

2.3 For the style input, I selected Vincent van Gogh's *The Starry Night*. The said painting was created in 1889. The original painting is currently on display at the Museum of

Modern Art, New York City. Notably, the artist Vincent van Gogh died in 1890, and therefore as of the date of creation of SURYAST, the copyright in the said painting titled *The Starry Night* had lapsed and the painting had become *publici juris*.

Figure 11: The Starry Night by Vincent van Gogh; used by Ankit Sahni as the style input to RAGHAV

2.4 Thereafter, I exercised my discretion to select a variable value determining the amount of style transfer between content and style images on RAGHAV Artificial Intelligence Painting Application (as illustrated under paragraph 1.4 above). The acts of selecting a specific variable value determining the amount and manner of style transfer between content and style images, selection of the style image (keeping into consideration the particular patterns and brushstrokes that the style image contains, the ability of RAGHAV to learn them, and the similarity of features such as the sky, buildings etc. in both content and style images), conceiving, creating and selecting an original content image (whose copyright belongs to me – Figure 10) cumulatively resulted in the output (below), which is the direct outcome of my creative expression and contribution. The selection of the specific variable value, the style input and the content input are completely arbitrary decisions, and are a culmination of my independent artistic expression and discretion. This outlines my independent, original and creative authorship in the subject artwork.

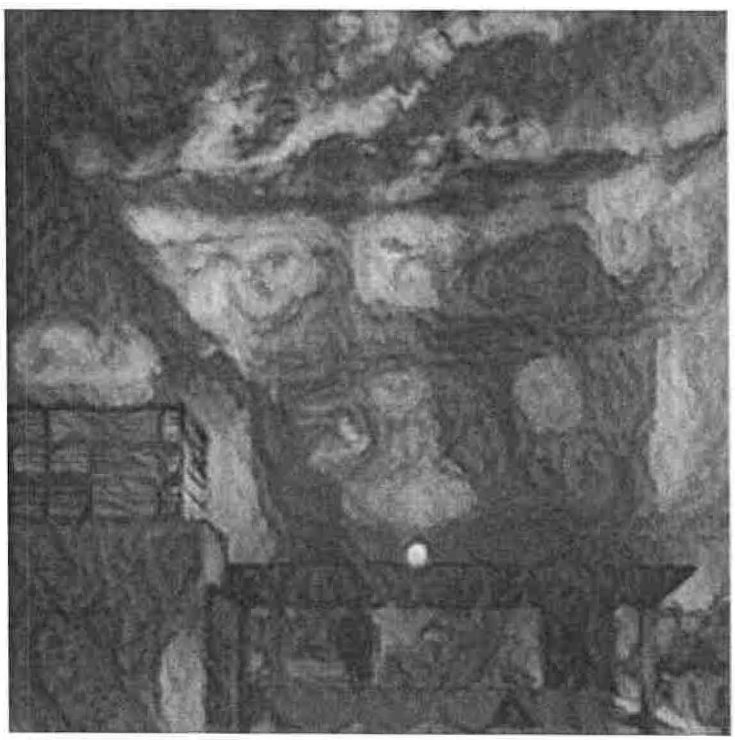


Figure 12: SURYAST (Hindi word for sunset); generated with the assistance of RAGHAV