COMP4422 Computer Graphics Project Report
Painterly Rendering Methods for Visual Arts

Ng Emily 20047523d
Chan Ho Yin 19071028d
Gao Zhilin 20104379d

Abstract
With the advance of technologies, painterly rendering techniques have been receiving recognition in the computer science field and developed to reconstruct classic paintings with fine-tuned photorealistic images. Techniques like stroke-based rendering construct an image in a controlled manner, resulting in the appearance of visible brushstrokes within the finished painting. Our project aims to combine existing painterly rendering technologies and explore the feasibility of such combinations. Techniques like stroke-based rendering algorithms, line drawing techniques and Neural Style Transfer are reviewed and combined for different results. We propose two novel painterly rendering methods, namely brush painting and edge-enhanced style transfer. This report shows the details of our research and studies, the implementation details of the combinations, examines the experimental results and evaluates the effectiveness based on our user study, which is done to understand users' preferences for the rendered pictures and the difficulty recognizing the source image from the rendered one. After gathering insightful and actionable information from the analysis of user studies, we propose some future work on the two painterly rendering methods for further improvement.

1. Introduction
Painterly rendering has been a popular field of study in non-photorealistic rendering since the 1960s[1]. It refers to the technique that renders painterly-style artistic pictures or videos with source images or 3D meshes. Multiple studies have been conducted to improve the artistic presentation, stroke texture and interactiveness of painterly rendering[2]. As a result of the advancement of GPU in recent years, a number of research and works have been devoted to the study of arbitrary style transfer, and real-time painterly rendering.

As mentioned by Lansdown and Schofield[1], multiple stroke-related rendering algorithms, and style transfer algorithms through filtering have been proposed in the early stage of painterly rendering, to generate non-photorealistic artistic paintings from meshes and images. Lee and her colleagues[3] mentioned that painterly rendering techniques evolved from single-scale stroke-based rendering that uses straight lines and fixed polygons as brush shapes, to multiscale stroke-based rendering algorithms, using multiple-sized curved strokes to simulate hand drawing.

The technique of applying line drawing to produce stylized images emerged in the early 21^{st} century[4].

Painterly rendering has been broadly adopted in different industries of visual arts, such as graphic design, video games and animation, to help individuals, designers and animators generate artistic illustrations. It has also been embedded in modern citizen's life as a form of entertainment, for instance, the painterly effect filters on social media and camera apps, and all forms of visual art productions. In view of the close relationship between painterly rendering and our daily life, we are motivated to study painterly rendering.

This report first reviews 3 types of painterly rendering methods, namely Stroke-based painterly rendering, Flow-based line drawing and Style Transfer. Next, it presents some experiments that study the effects of different combinations of existing techniques. Then, it introduces the user study to evaluate experiment outcomes. Based on the results, this report proposes two novel painterly rendering methods, Brush painting and Edge-enhanced style transfer. Finally, the report discusses the possible directions for future work.

2. Related work

2.1. Stroke-based painterly rendering

Stroke-based painterly rendering (SBR) refers to algorithms that generate digital artistic images by placing discrete stroke elements, such as brush strokes, lines, and stipples[5]. Multiple stroke-based painterly rendering algorithms have been proposed to simulate different styles and pen-and-ink drawings. It is generally adopted in imitating traditional brush-based oil painting styles. A typical stroke-based painterly rendering image structure consists of an ordered list of strokes and a canvas.

There are two main categories of stroke-based painterly rendering methods: optimization algorithms and greedy algorithms. The latter is the most commonly applied Stroke-based rendering technique. In the Greedy algorithm, strokes are created and added to the image structure in run time and cannot be modified after creation. A typical stroke-based painterly rendering algorithm after the 90s makes use of Hertzmann's Multi-resolution Painting with Curved Strokes approach(aka Multi-resolution painterly rendering)[6].

Multi-resolution painterly rendering paints the canvas iteratively over multiple layers of brush size with long, curved strokes[7]. Brushes with larger radii are applied to illustrate the contour of the image objects, and the composition of the source picture. Meanwhile, smaller brushes are employed to refine the output and highlight the source image details. The algorithm first creates and divides a canvas into brush grids. Then, it paints the grids in the order of large brush size to smaller brush size to represent different levels of visual information. Each brush size corresponds to a layer.

In each layer, the algorithm creates a reference image by applying a Gaussian filter to the source image according to the brush size. Then, it compares the colour difference between the canvas and the reference image on each grid. For grids having a colour difference exceeding a user-defined threshold, the algorithm creates strokes based on the reference image colour and stores them in the image structure. After constructing all the necessary strokes for the layer, the algorithm paints the canvas with the generated strokes.

Stroke-based painterly rendering method is a natural and comprehendible algorithm that mimics the process of real-life painting. By increasing the number of layers, the algorithm is able to present a high level of image details, and generate a satisfying result of realism and aesthetics of the resulting image. The major drawback of this algorithm is the long computation time required to generate a high-quality artistic image, as the computation cost increases with the number of layers. Although users may modify the style parameters, there are limited options and space for style modification. There are insufficient tools for users to control a Stroke-based painterly rendering output image.

2.2 Line drawing

Line drawing algorithms are image-based painterly rendering methods that generate line sketches from a picture highlighting its important edges, inspired by artists who use lines as a both simple and effective way of visual communication. From the time researchers started to focus more on non-photorealistic rendering techniques, line drawing has been an important problem in the field, because of its essentiality and ubiquity.

So far, attempts to tackle the problem of line extraction mainly fall into two categories: stroke generation and flow-based filtering. The former resembles stroke-based methods in the idea of simulating how strokes are created by artists, based on a number of attributes, while the latter focuses more on the nature of the source image, in particular, how edges are represented.

One example of stroke generation is [4]. In this study, the authors compute a likelihood function for each point to be on a genuine edge from the gradient field, taking into account the contrast of colour as well as edge directions in its neighbourhood. Then, the high-likelihood segments are linked together to be stroked. In this way, the output image consists of strokes that look more natural and are more robust against noise. Another research [8] proposes a more stable method based on similar ideas. The method uses convolution to select the most appropriate direction for edges and generate lines along the direction.

Meanwhile, an example of flow-based filtering is Coherent Line Drawing [9]. This method uses the gradient to calculate the Edge Tangent Flow (ETF), which approximately reflects the direction of edges. The final edges are derived from the flow. To reduce the influence of severe noise in the gradient, the ETF is adjusted by aligning vectors with their neighbours beforehand. Finally, thick edges are filtered using the Difference of Gaussians (DoG) approach into thinner and more concentrated edges. A number of studies extend this method to enhance its quality. For example, [10] proposes several edge detection standards other than DoG, and [11] utilises the saliency map to enhance the quality of edge extraction by emphasising visually important areas with low contrast and vice versa, which breaks the limitation of a gradient map.

Both methods have some advantages as well as limitations. In summary, stroke generation algorithms tend to produce more natural-looking lines because they draw the picture stroke by stroke. However, the abovementioned algorithms are based on many run-time predictions that may affect the accuracy of the sketch. Flow-based filtering methods usually reflect the border of the source image more precisely, but the output edges may have a certain degree of artefacts since they are drawn as clusters pixel by pixel.

Meanwhile, line drawing methods have common pros and cons in general. The good thing is that they convey the most important information about the sketch very efficiently, requiring less time for readers to glance and understand compared to the original image. However, there is also information that line drawings cannot embed. For example, the colour, shade and texture cannot be preserved. This becomes a major limitation that prevents line drawings from being more useful.

2.3. Style transfer

Neural Style Transfer (NST) is a method that combines the content of an image and the style of another using neural networks. The process generally includes network training, feature encoding and decoding and machine learning driven by some loss function.

This technique is initially proposed by Gatys et al. [12] who notice that image and style features can be separated in a pre-trained VGG network. The authors use Gram matrices to capture style representation which is integrated into the content, and finally convert style transfer to an optimization problem. The major limitation of this method is that its computation speed is extremely slow.

Some studies such as [13, 14] adopt feed-forward networks and successfully improved the efficiency of NST. However, this comes at the cost of being unable to handle arbitrary styles beyond a small pre-defined set.

The problem of efficient arbitrary style transfer remains mostly unsolved until methods such as AdaIN [15] and AdaAttN [16] were proposed. These methods align second-order statistics of the content image to the style image instead of capturing and optimising them as in [12], which improves the speed while allowing arbitrary styles. Despite their effective performance, there are still some drawbacks. For example, the use of statistics as an overall descriptor may cause distortion since these parameters are unable to represent local details.

Recently, another study [17] further improves the quality by solving this problem. Instead of using second-order statistics, it directly aligns the style features themselves to generate more vivid details.

Neural Style Transfer might use different algorithms like CNN or GAN. They successfully abstract and distinguish the content and style features of an image. Both algorithms optimise the NST performance in transferring, generating and reconstructing realistic images. Moreover, NST can be controlled with different inputs, for instance, stroke size and colour, which can therefore produce a variety of results. However, NST might lack stroke size and spatial style control. It would be difficult to abstract various features from the content image and style image. Thus, NST might not be able to construct pictures with complex style patterns.

3. Observation and insight

Although there has been a wide range of in-depth studies in painterly rendering, due to the diversity of artworks, most algorithms fall into just a few categories with different approaches and styles. Except for arbitrary style transfer, most algorithms depend on the dominant features of the specific type of painting. Since paintings (such as line drawings and oil paintings) can be created in totally different ways, rendering algorithms may take very different approaches as well. This actually implies that painterly rendering is a very diverse topic that assembles a number of subfields that are very different in nature, and only comparable in terms of their purpose.

However, despite the lack of cross-discipline studies between these subfields, we believe in the possibility of combining different image-based methods with different advantages and disadvantages to enhance output quality. For example, neural style transfer may overlook important edges due to low contrast. Meanwhile, line drawings are good at extracting such edges, although they fail to preserve colour and texture information. Therefore, these two methods may compensate for each other's limitations when combined.

4. Experiment methodology

To form combinations of different categories of methods, we pick one representative from each with publicly available implementations on the internet, namely *Painterly Rendering with Curved Brush Strokes of Multiple Sizes* [7] for stroke-based methods, *Coherent Line Drawing* [9] for line drawing methods, and *Attention-aware Multi-stroke Style Transfer* (AAMS) [18] for neural style transfer methods. Although they are not the current state-of-the-art methods, they suffice to demonstrate the obvious effects of combination and are capable of achieving the outcomes of this project.

Since combinations may occur in various forms, it is necessary to decide the exact way to perform them according to the properties of the components.

For SBR and line drawing methods, we simply feed the output of one into another because their purposes are conflicting. While SBR aims to blur a clear image to conform to an artistic style, line drawing methods extract important dense lines from a more complex image, which is exactly the opposite. As one can imagine, extracting line sketches from an oil painting may not be a good idea. However, feeding the line drawing into SBR produces what we call "brush painting", and we further use it in our next round of combination.

Since NST may produce an output of arbitrary styles, which is difficult to assess in general, it may not be very useful to feed its output into other methods. Instead, we try to use other methods to enhance its quality. For example, line drawings, SBR paintings and the "brush paintings" of some source images are fed into NST and compared to the original outcomes using the same style images. In addition, as we observe that NST sometimes omits important boundaries and paints colour across them, we use line drawing to alleviate this problem. More specifically, we overlay the line sketch onto the content image before starting NST. We name this algorithm "edge-enhanced NST".

After some experiments, we notice one limitation of edge-enhanced NST is that large clusters of black pixels from the line drawing may be transferred into undesired patterns. To address this issue, we adopt the idea of colourized line drawing [19], which sets the pixel value in the line drawing to its corresponding pixel in the source image at the same position. Note that the colourized line drawing needs to be adjusted before being overlaid on the source image, otherwise the combination has no influence on the source at all. Hence, we introduce a parameter c (coefficient of colourization) where 0<=*c*<=1 to linearly interpolate the appropriate line colour between the source colour and pure black. That is, pixel values on the line drawing are set by the formula

$$P_{line} = cP_{source} + (1 - c)P_{black}$$

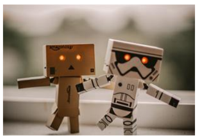
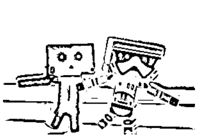
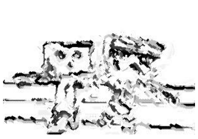
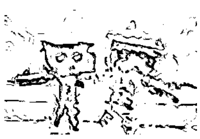
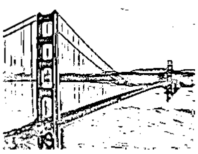
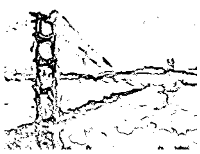
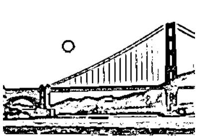
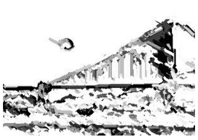
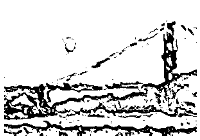
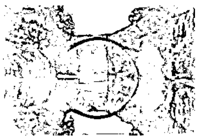
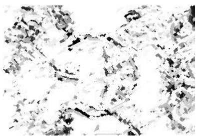
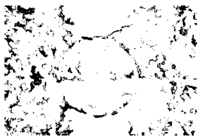
which can be further simplified in the RGB system where pure black is represented by the value of zero. Then, we overlay this adjusted colourized line drawing onto the source image and feed it into NST. We call this “colourized-edged-enhanced NST”. Note that the above-mentioned edge-enhanced NST is a special case of this more generalised method with c=0, and the original NST method is another special case with c=1. In our experiments, c is set to be 2/3.

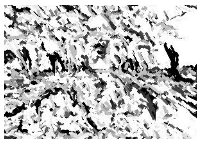
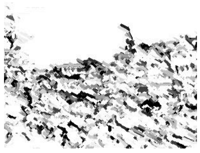
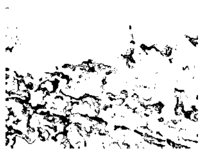
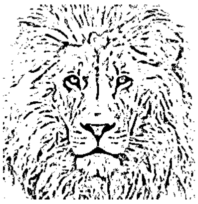
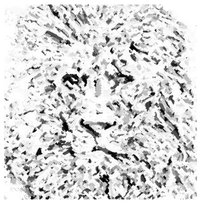
To distinguish “edge-enhanced NST” and “colourized-edge-enhanced NST” in our experiments, we would indicate the former as “normal edge-enhanced NST”. Since both methods overlay a line drawing result on the NST source image, they fall into the category of “edge-enhanced NST”, which would be used to refer to both algorithms in the rest of the report.

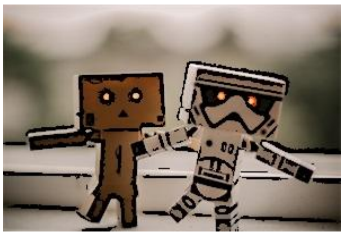
After obtaining results from each combination, we conduct a user study to verify our findings.

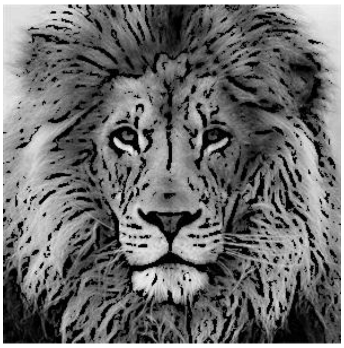
5. Experimental result

5.1. Non-NST experimental result

1					
2					
3					
4					

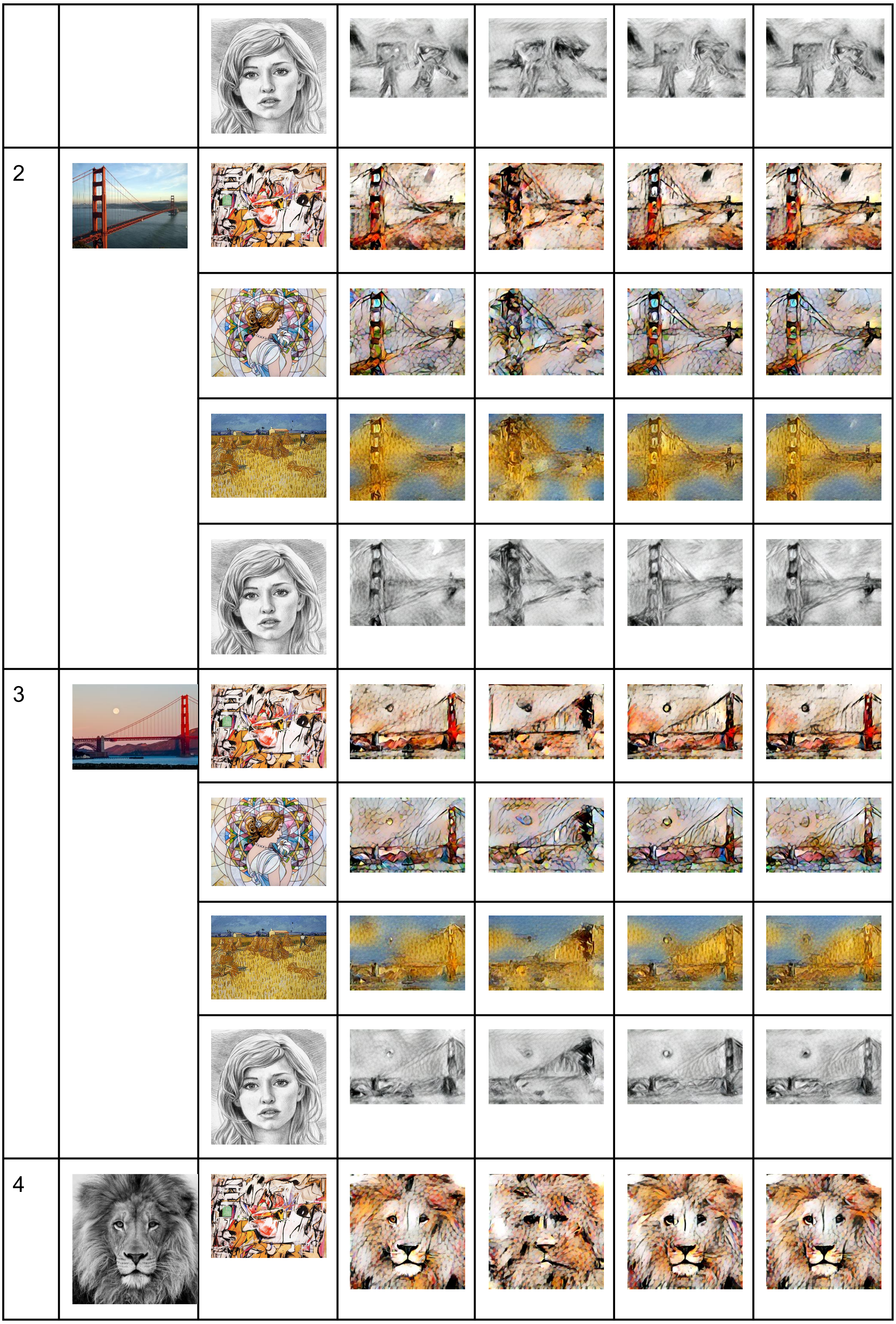
5					
6					
7					
	Source	Line drawing	SBR	brush painting	SBR to Line drawing

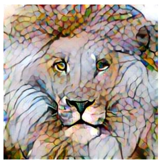
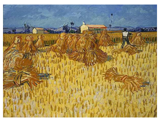
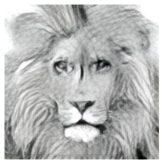
1			
2			
3			
4			

5			
6			
7			
	Source	Normal Edge-enhanced	Colourized edge-enhanced (c=2/3)

5.2. NST experimental result

1						
						
						


2
3
4

						
						
						
	Source	Style	Original AAMS NST	brush painting NST	Normal edge-enhanced NST	Colourized edge-enhanced NST (c=2/3)

6. User study

6.1. Design

Our team has conducted a user study to evaluate our experimental results quantitatively(see Appendix A for more information). It collects responses from 56 non-professional individuals.

The first part of the user study evaluates user preference over different experimental painterly rendering methods. Candidates are requested to rate their liking over 2 NST experimental results in a 5-Point Likert scale rating system, where one is the least preferred, and five is the most preferred. Then, it requests respondents to evaluate their favour on three non-NST experimental results in a 3-Point Likert scale rating system, namely, Brush painting and the method feeding SBR to line drawing(aka SBR to Line drawing). The order of experimental results is randomised to prevent ordering effect.

In the second half of the user study, users identify the style images of two NST results and evaluate the difficulty of identifying source images from different experimental painterly rendering techniques in a 5-scale rating system. The user study also requests users to assess the similarity between a given set of marker strokes and the stroke effects of three productions of brush paintings.

6.2. Result

A. Preference study

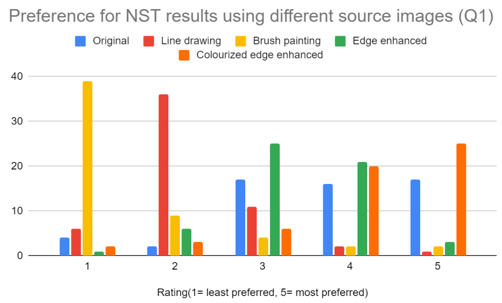


Chart 1: Preference for NST results using different source images (Q1)

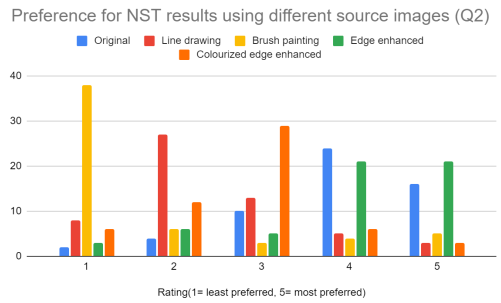


Chart 2: Preference for NST results using different source images (Q2)

The preference for experimental results can be observed from Chart 1 to Chart 5. From Chart 1 and Chart 2, it can be concluded that feeding line drawings or brush paintings into NST results in much lower scores compared to other inputs that preserve colour information. This is not unexpected since NST methods are mostly colour-oriented. The difference among the original, Edge-enhanced, and Colourized-edge-enhanced NST is less significant.

To look closer into this difference, we conduct a paired t-test for our two methods against the original AAMS NST. Results indicate that in Question 1, both Normal Edge-enhanced and Colourized-edge-enhanced NST outperform the original method to some degree (with p=0.069 and p=0.020 respectively). However, in question two, there is not sufficient evidence (p=0.325) to conclude that edge-enhanced NST is preferred over the original NST although its average score is slightly higher. Moreover, colourized-edge-enhanced NST receives much lower ratings (p=1.24*10^-9). This is due to unknown factors and more details are explained in Comparison 4 in section 7.3.

It is clear that candidates favour the outcome of edge-enhanced NST, whether colourized or not, over a normal NST at least in some settings like Question 1 (see the lion in Group 4 of section 5.2, with the first style image). It might be because both methods strengthen the edges and maintain the image features of the source image, which nourish the aesthetic effect of the NST result.

However, it is worth noticing that these new methods may sometimes produce results that are less preferable. For example, the colourized edge-enhanced for Question 2 (see the bridge in Group 2 of section 5.2, with the second style image) fails to preserve the shape of the object. We find this very difficult to explain since both edge-enhanced and original NST do not cause this problem. This might be a coincidence where the method just happens to fail.

The scale of our study is too small for discovering the underlying factors, and we leave this to future work.

In addition, there are not enough statistics to show the preference tendency of a normal edge-enhanced NST and colourized edge-enhanced NST. The effect of the colourization coefficient c remains unclear.

Both charts indicate that the combination of brush painting and NST is least favourable among all the NST experiments.

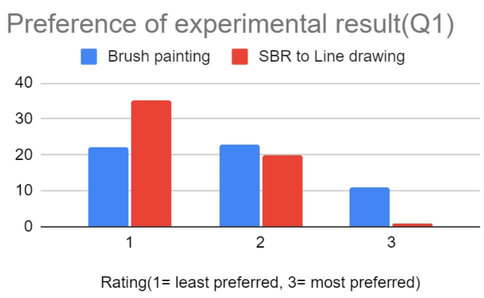


Chart 3: Preference for non-NST experimental results (Q1)

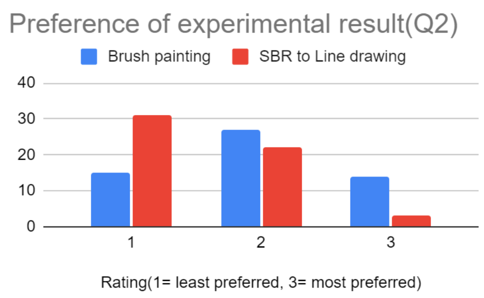


Chart 4: Preference for non-NST experimental results (Q2)

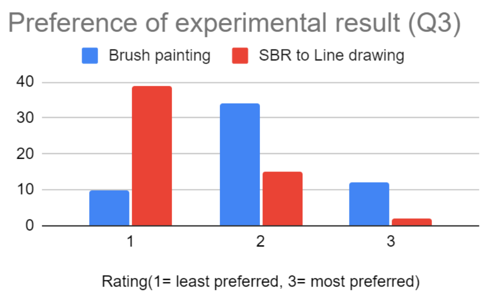


Chart 5: Preference for non-NST experimental results (Q3)

According to the statistics of participants’ preference for non-NST experimental results (Chart 3 to Chart 5), the brush painting method is more preferred than the SBR to Line drawing method.

B. Difficulty to recognize source image from experimental result

Difficulty to recognize source image Q1					
	Non-NST experiment		NST experiment		
Difficulty	brush painting	SBR to Line drawing	Brush painting	Edge enhanced	Colourized-edge enhanced
1	10	6	11	3	24
2	10	18	7	23	2
3	21	16	18	4	1
4	9	11	5	22	8
5	6	5	15	4	21
<=2	20	24	18	26	26
3	36	32	38	30	30

Table 1: Difficulty to recognize source image Q1

Difficulty to recognize source image Q2					
	Non-NST experiment		NST experiment		
Difficulty	brush painting	SBR to Line drawing	Brush painting	Edge enhanced	Colourized-edge enhanced
1	7	8	8	20	8
2	9	14	8	12	20
3	16	15	15	5	4
4	14	16	11	5	20
5	10	3	14	14	4
<=2	16	22	16	32	28
>=3	40	34	40	24	28

Table 2: Difficulty to recognize source image Q2

Table 1 and 2 shows the difficulty level of identifying source images from experimental results. After categorising the difficulty into less(2 or lower) and more(3 or higher), our team deduced the following investigation.

First, compared with brush painting, in Q1(lion) users find it equally difficult ($p=0.45$) to recognize the source image from the result of Line drawing to the

SBR method, but in Q2 (bridge), users tend to think the brush painting is harder to recognize (p=0.03).

Second, in Q1 all NST algorithms receive similar responses (p=0.412, p=0.466, p=0.192), and in Q2 the combination of brush painting and NST has the highest difficulty level among all the NST experiments (p=0.039, p=0.081) while there is some but not significant evidence that normal edge-enhanced NST is easier to recognize than colourized-edge-enhanced NST (p=0.104).

However, we notice that there are many extreme ratings (1 and 5) and fewer moderate ones for NST, which suggests users' opinions may be very different from each other and it may be necessary to split participants into groups using a pretest for a more reliable conclusion.

C. Identify style image from NST experimental result

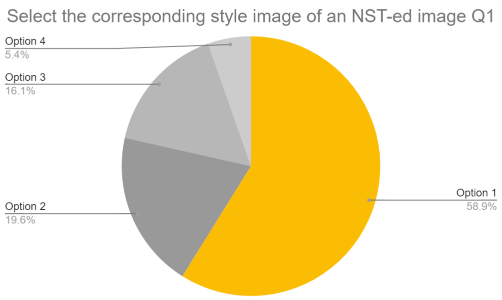


Chart 6: Correct rate of style identification of an NST-ed image Q1

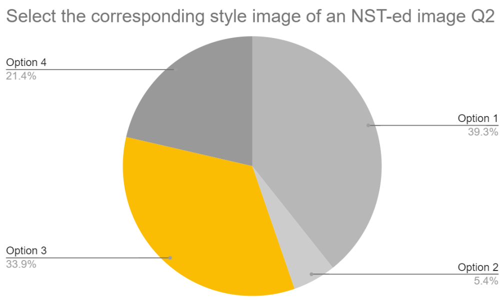


Chart 7: Correct rate of style identification of an NST-ed image Q1

Note: Yellow refers to the correct option in Chart 6 and Chart 7

Chart 6 and 7 show the correct rate in identifying style images from NST results. According to the result, there are insufficient statistics to prove individuals can recognise the style of an NST result in our study.

D. Evaluation of similarity between marker strokes and brush painting effect

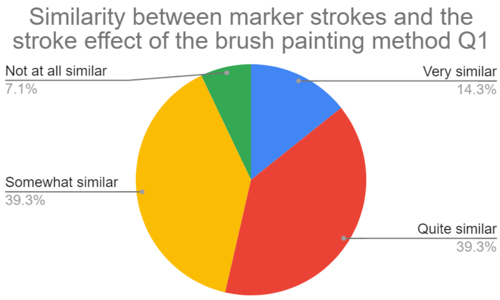


Chart 8: Similarity between marker strokes and the stroke effect of brush painting production Q1

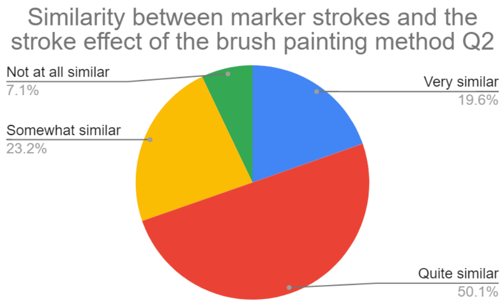


Chart 9: Similarity between marker strokes and the stroke effect of brush painting production Q2

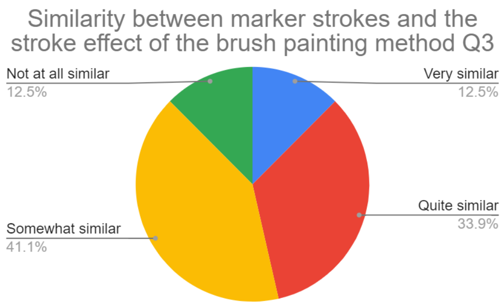


Chart 10: Similarity between marker strokes and the stroke effect of brush painting production Q3

Charts 8, 9, and 10 illustrate users' evaluation of the similarity between marker strokes and the stroke effects produced by brush painting. As most of the rating falls in the option "Quite similar" or higher, our team inferred that the brush painting method successfully mimicked the stroke of marker texture.

7. Strength and Limitation Analysis

7.1. Brush painting

The brush painting method successfully simulates the aesthetic effect of marker drawing. Referring to the user study, most participants agree that it creates a similar brush effect to the marker.

However, this technique cannot avoid line distortion and losing detailed information about the source image. The abovementioned drawback makes it hard to recognize the image object (see section 5.1 Group 4, Brush painting). It also faces the problem of artefacts that reduce the realism of the result.

7.2. Edge-enhanced NST

Our proposed edge-enhanced style transfer technique improves the clarity of the edges and emphasises the main image object. According to the user study, both normal and colourized-edge-enhanced style transfer methods provide an overall better artistic effect than the original methods. It also helps the audience to identify the image object as well.

For example, the original AAMS on the left fails to depict the landscape in the back, which has high contrast in the source image. In comparison, our edge-enhanced NST on the right draws the landscape very clearly.

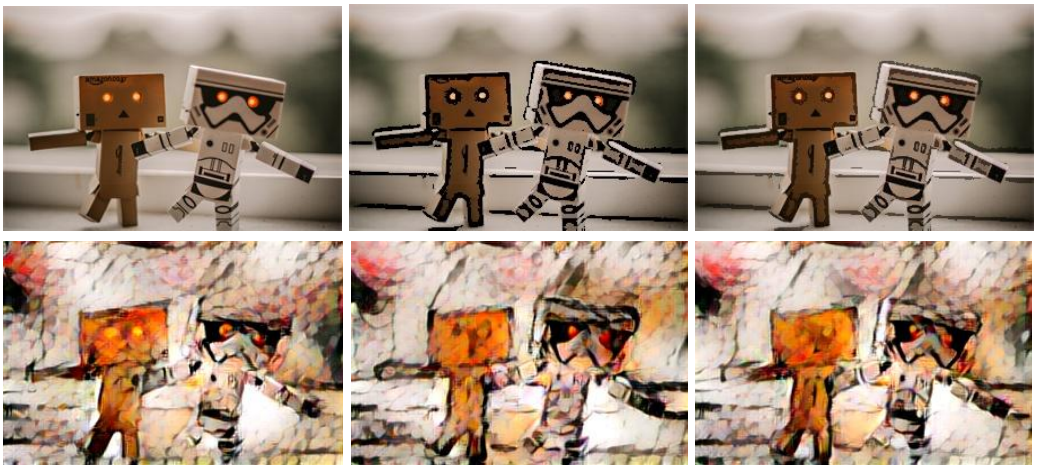
Comparison 1: AAMS vs. edge-enhanced AAMS (bridge)

Also see Comparison 2, where the lion's fur and eyes are painted more clearly and receive higher user ratings.

Comparison 2: AAMS vs. edge-enhanced AAMS (lion)

In many cases, the edge-enhanced NST provides better artistic effects than NST. However, its major limitation is that clusters of black pixels as part of the line drawing may overwrite too much information, or may sink into the background when there is not enough contrast. See 7.3 for more details.

7.3. Colourized-edge-enhanced NST



Comparison 3: NST vs. edge-enhanced vs. colourized-edge-enhanced (c=2/3), with inputs

Our colourized-edge-enhanced method aims to alleviate the drawback we just mentioned. As we can see from the above comparison 3, compared to the original NST, the edge-enhanced method successfully preserves important edges that form the head of the robot on the right. However, it lays too many black pixels on the eyes of the robot on the left due to a small high-contrast region. This is transferred to undesired patterns that do not exist in the source image. In contrast, colourized-edge-enhanced NST integrates the advantages of both methods and overcomes their limitations.

However, we notice that this algorithm may occasionally produce odd patterns not seen in normal edge-enhanced NST due to unknown reasons. See comparison 4 below where colourized-edge-enhanced NST misses an important edge.

Comparison 4: NST vs. edge-enhanced vs. colourized-edge-enhanced (c=2/3)

Although there is uncertainty regarding the stability of performance, we should note that the parameter c in colourized-edge-enhanced NST is adjustable by the user, which even allows the user to use original NST and black edge-enhanced NST (using c=1 and c=0 respectively). This means the user may be able to find a satisfactory result after attempting different values of c.

7.4. Others

Due to the loss of colour information from Brush painting, the combination of Brush painting and NST fails to present the shadow and colour features of the original picture. Feeding a Brush painting result to NST result has a poor artistic effect (see Section 5.2). It leads to the lowest preference in the user study and increases the difficulty in identifying the source image. The same rationale applies to feeding line drawings into NST.

Our team also observed that the SBR to Line drawing methods failed to generate satisfactory artistic images. It may be because the production of stroke-based rendering suffers from a degree of line distortion. Hence, feeding the stylized result to a line-drawing algorithm would lead to an over-abstraction of the image object, which harms the aesthetic effect and the similarity between the outcome and the original picture.

8. Future work

We have proposed two novel painterly rendering methods that can simulate the aesthetic effect of marker drawing and help emphasise the main object respectively. However, both methods suffer from problems like line distortion which fails to present the details from the original image.

One solution is to replace the line drawing algorithm with a smoother and clearer one. It helps to alleviate the problem of line distortion. The addition of different brush radii and the variety of brushes might also help generate more realistic, artistic images as it helps the diversity in stroke-based rendering, preventing generating monotonous images.

In addition to enhancing the algorithms, there is a lot of space for improvement in terms of a more extensive user study. User preference is related to a large number of factors, which is difficult to control in NST. Therefore, to understand the effectiveness of NST algorithms, it is necessary to identify different groups of users and the reasons behind their preference for specific features. For example, we may provide more options for the colourized-edge-enhanced NST results with different colourization coefficients c in the user study.

9. Conclusion

Our team reviewed three existing painterly rendering technologies and explored the feasibility of such combinations. We proposed two novel painterly rendering methods namely brush painting and edge-enhanced style transfer. We showed the details of our research and studies, the implementation details of the combination of technologies, and examined our experimental results with a user study. Brush painting successfully imitated the visual appearance of marker drawing. The edge-enhanced style transfer process enhances the edges' clarity and draws attention to the main feature of the image. Despite the benefits provided by the two approaches, the brush painting technique is unable to prevent line distortion and the loss of certain details from the source image. Additionally, the combination of Brush painting and NST failed to display the shadow and colour aspects of the original image due to the loss of colour information from Brush painting.

In the future, more work can be done on user studies to discover underlying factors that affect user preference over our methods. In addition, more advanced methods for SBR, Line drawing and NST can be used as components of combinations for better results.

Reference

[1] J. Lansdown, and S. Schofield, “Expressive rendering: a review of nonphotorealistic techniques,” IEEE computer graphics and applications, vol. 15, no. 3, pp. 29–37, 1995, doi: 10.1109/38.376610.

[2] D. Vanderhaeghe. and J. Collomosse, “Stroke Based Painterly Rendering,” in Image and Video-Based Artistic Stylisation, vol. 42, London: Springer London, 2012, pp. 3–21.

[3] J. Lee, J. Choi, and S. Seo, “Emotion-Inspired Painterly Rendering,” IEEE Access, vol. 8, pp. 104565–104578, 2020, DOI: 10.1109/ACCESS.2020.2999460.

[4] M. Son, H. Kang, Y. Lee, and S. Lee, “Abstract Line Drawings from 2D Images,” in 15th Pacific Conference on Computer Graphics and Applications (PG’07), 2007, pp. 333–342, doi: 10.1109/PG.2007.63.

[5] A. Hertzmann, “A survey of stroke-based rendering,” IEEE computer graphics and applications, vol. 23, no. 4, pp. 70–81, 2003, doi: 10.1109/MCG.2003.1210867.

[6] S. Kim, D. Kang, and K. Yoon, “GPGPU-Based Painterly Rendering for Mobile Environment,” E-Learning and Games, pp. 224–227,2017, DOI: 10.1007/978-3-319-65849-0_24.

[7] Hertzmann A., “Painterly Rendering with Curved Brush Strokes of Multiple Sizes”, Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques(SIGGRAPH '98), pp. 453-460, 1998. [Online]. Available: https://doi.org/10.1145/280814.280951

[8] C. Lu, L. Xu, and J. Jia, “Combining sketch and tone for pencil drawing production,” Proceedings of the Symposium on non-photorealistic animation and rendering, 2012, pp. 65–73.

[9] H. Kang, S. Lee, and C. K. Chui, “Coherent line drawing,” In Proceedings of the 5th international symposium on Non-photorealistic animation and rendering (NPAR '07). Association for Computing Machinery, New York, NY, USA, 43–50, 2007, doi: 10.1145/1274871.1274878

[10] S. Wang, X. Liu, Y. Chen, and E. Wu, “Abstract Line Drawings from Photographs Using Flow-Based Filter,” Chinese Journal of Computers, vol. 37, no. 3, pp. 611–620, 2014, doi: 10.3724/SP.J.1016.2014.00611.

[11] S. LIU and Z. LIU, “Line drawing via saliency map and ETF,” Frontiers of Computer Science, vol. 16, no. 5, p. 165707–, 2022, doi: 10.1007/s11704-021-1027-z.

[12] L. A. Gatys, A. S. Ecker, and M. Bethge. Image style transfer using convolutional neural networks. In CVPR, 2016.

[13] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In ECCV, 2016.

[14] C. Li and M. Wand. Precomputed real-time texture synthesis with markovian generative adversarial networks. In ECCV, 2016. 1, 2

[15] X. Huang and S. Belongie, “Arbitrary Style Transfer in Real-time with Adaptive Instance Normalization,” 2017, doi: 10.48550/arxiv.1703.06868

[16] S. Liu et al., “AdaAttN: Revisit Attention Mechanism in Arbitrary Neural Style Transfer,” 2021, doi: 10.48550/arxiv.2108.03647.

[17] Y. Zhang et al., “Domain Enhanced Arbitrary Image Style Transfer via Contrastive Learning,” arXiv.org, 2022, doi: 10.1145/3528233.3530736.

[18] Y. Yao, J. Ren, X. Xie, W. Liu, Y. -J. Liu and J. Wang, "Attention-Aware Multi-Stroke Style Transfer," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 1467-1475, doi: 10.1109/CVPR.2019.00156.

[19] Y. Liu, X. Yang, S. Wang and E. Wu, "An improved method of colorized line drawing," 2011 International Conference on Multimedia Technology, 2011, pp. 189-192, doi: 10.1109/ICMT.2011.6003048.

Appendix

A. User study survey

Painterly Rendering User Study

There are 12 questions in this questionnaire, please refer to the provided pictures and answer the questions..

*Required

NST comparison 1

Please rate the following images.

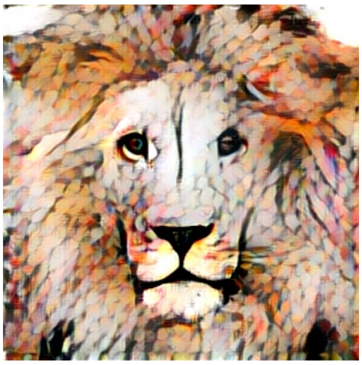
Image 1

Image 2

Image 3

Image 4

Image 5



1. Please rate the above images according to preference *
(5=most preferred, 1=least preferred)

Mark only one oval per row.

	1	2	3	4	5
Image 1					
Image 2					
Image 3					
Image 4					
Image 5					

NST comparison 2

Please rate the following images.

Image 1

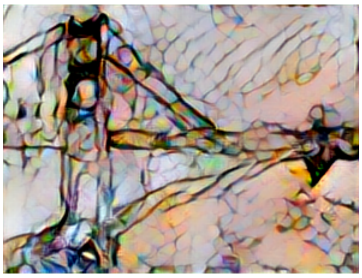


Image 2

Image 3

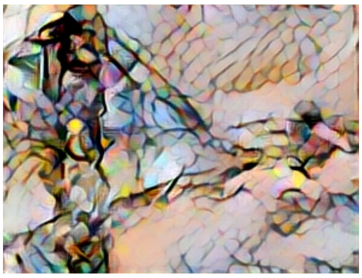


Image 4

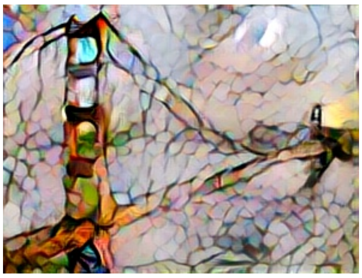


Image 5

2. Please rate the above images according to preference *
(5=most preferred, 1=least preferred)

Mark only one oval per row.

	1	2	3	4	5
Image 1	◯	◯	◯	◯	◯
Image 2	◯	◯	◯	◯	◯
Image 3	◯	◯	◯	◯	◯
Image 4	◯	◯	◯	◯	◯
Image 5	◯	◯	◯	◯	◯

Painterly rendering method comparison

Please rate the following images

Image 1

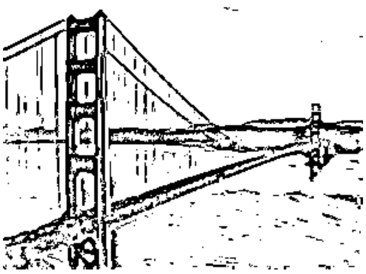
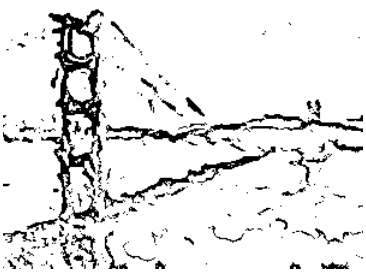


Image 2

Image 3



3. Please rate the above images according to preference *
(3=most preferred, 1=least preferred)

Mark only one oval per row.

	1	2	3
Image 1	◯	◯	◯
Image 2	◯	◯	◯
Image 3	◯	◯	◯

Painterly rendering method comparison 2

Please rate the following images

Image 1

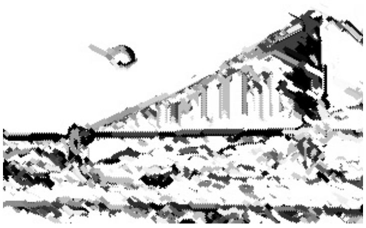


Image 2

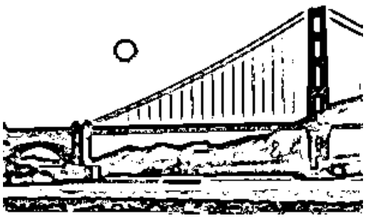


Image 3

4. Please rate the above images according to preference *
(3=most preferred, 1=least preferred)

Mark only one oval per row.

	1	2	3
Image 1	◯	◯	◯
Image 2	◯	◯	◯
Image 3	◯	◯	◯

Painterly rendering method comparison 3

Please rate the following images

Image 1

Image 2

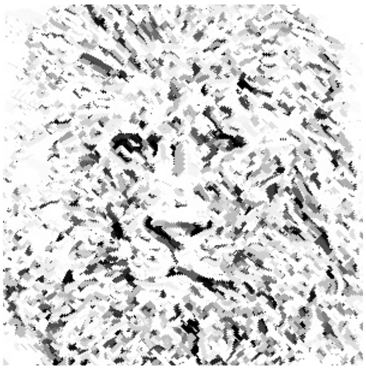
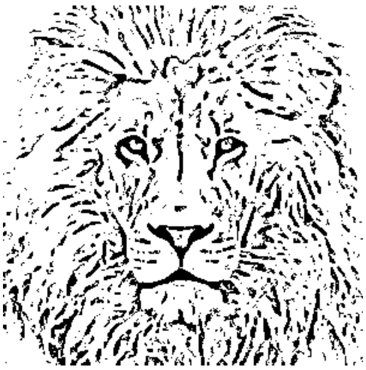


Image 3



5. Please rate the above images according to preference *
(3=most preferred, 1=least preferred)

Mark only one oval per row.

	1	2	3
Image 1	◯	◯	◯
Image 2	◯	◯	◯
Image 3	◯	◯	◯

Rating of the difficulty level to recognize the source image

Please rate the following images

Image 1

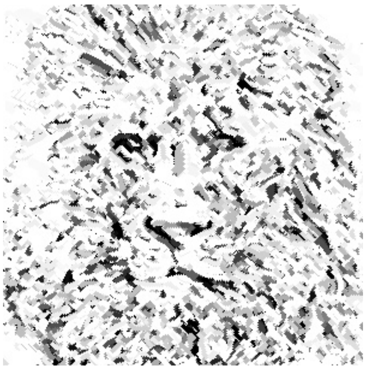
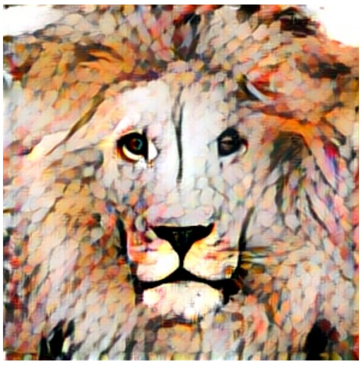


Image 2

Image 3

Image 4

Image 5



6. Please rate the difficulty to recognize the image object *
(5= most difficult, 1=easiest)

Source image

Mark only one oval per row.

	1	2	3	4	5
Image 1	⬭	⬭	⬭	⬭	⬭
Image 2	⬭	⬭	⬭	⬭	⬭
Image 3	⬭	⬭	⬭	⬭	⬭
Image 4	⬭	⬭	⬭	⬭	⬭
Image 5	⬭	⬭	⬭	⬭	⬭

Rating of the difficulty level to recognize the source image 2

Please rate the following images

Image 1

Image 2

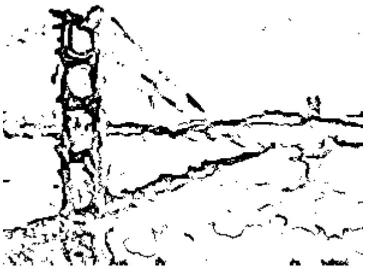


Image 3

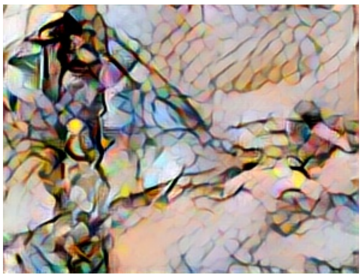


Image 4

Image 5

7. Please rate the difficulty to recognize the image object *
(5=most difficult, 1=easiest)

Mark only one oval per row.

	1	2	3	4	5
Image 1	⬭	⬭	⬭	⬭	⬭
Image 2	⬭	⬭	⬭	⬭	⬭
Image 3	⬭	⬭	⬭	⬭	⬭
Image 4	⬭	⬭	⬭	⬭	⬭
Image 5	⬭	⬭	⬭	⬭	⬭

Please pick the corresponding style image

8. Please pick the corresponding style image for the following picture *

Mark only one oval.

⬭ Option 1

⬭ Option 2

⬭ Option 3

⬭ Option 4

9. Please pick the corresponding style image for the following picture *

Mark only one oval.

Option 1

Option 2

Option 3

Option 4

Brush comparison

Reference strokes

10. How similar is the strokes in following image to the reference strokes? *

Mark only one oval.

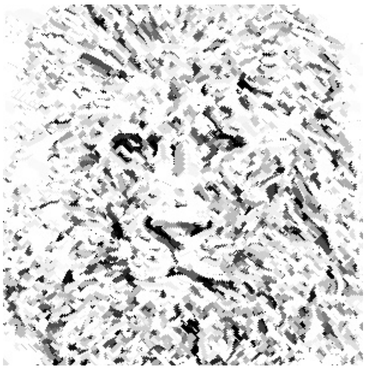
- Very similar
- Quite similar
- Somewhat similar
- Not at all similar

11. How similar is the strokes in following image to the reference strokes? *

Mark only one oval.

- Very similar
- Quite similar
- Somewhat similar
- Not at all similar

12. How similar is the strokes in following image to the reference strokes? *



Mark only one oval.

- Very similar
- Quite similar
- Somewhat similar
- Not at all similar

Thank you so much for your participation