ingegneria sismica
International Journal of Earthquake Engineering
Anno XLIII - Num. 1 - 2026

Exploring the Application of Artificial Intelligence-Assisted Creative Tools in Digital Painting

Yuan Tian[1,*]

[1] Nanhang Jincheng College, Nanjing, Jiangsu, 211156 China

SUMMARY: *This study incorporates the smart learning features with intelligent image processing technology in developing an artificial intelligence stylized painting system. By employing a color transformation model and a smart learning algorithm, the system is able to provide intelligent color matching and continuous stylistic progression. In color matching experiments, the system has achieved PSNR scores between 31.22 dB and 33.48 dB, while its SSIM values are 0.9852 and 0.9925. The results show a clear superiority compared to various conventional classification algorithms. In terms of the quality of generated images, the system's PSNR scores for pictures produced under various themes range between 75.83 dB and 78.72 dB, while their SSIM scores exceed 0.97. This shows significant superiority over other comparison models utilizing style translation and texture synthesis methods. In addition, the system has shown excellent performance in line processing by consuming less than 4.97 ms in processing highly intricate lines. In terms of resource utilization, the system still uses only 45.45% GPU resources and 19.5% CPU usage for generating 500 images – much lower than those of the comparison approaches.*

KEYWORDS: *Artificial Intelligence; Digital Painting; Style Transfer; Color Matching*

1 Introduction

Since the information era and the advent of digital applications, there have been significant developments in digital paintingt [1, 2]. Digital painting entails the use of computers to create paintings. This involves the use of computer software to analyze information [3, 4].Digital painting entails encoding every single pixel into many bits to create digital visual imagery. Digital painting is one of the most progressive forms of paintings in this new era of art compared to traditional paintings. Digital painting is different in form and concept from traditional paintings.

With the arrival of the age of artificial intelligence, we need to reconsider the fundamental principles of digital painting creation [5]. Not only does AI reshape the digital painting creation process, but it also poses a challenge to traditional concepts of arts [6, 7]. With technological advancement, artistic creation experiences substantial changes—an evolution that goes beyond technological changes and impacts art creation and appreciation at the core level [8-10]. Based on deep learning algorithms, AI can explore extensive databases of paintings and even create novel artworks whose techniques and styles match those created by artists [11, 12]. The adoption of AI in the field of art has led to philosophical and ethical questions related to "creativity." On one hand, AI has broadened the scope of artistic creation and democratized the process, making it possible for everyone to create art with AI tools without being constrained

*QQDDqqdd778855@163.com
https://doi.org/10.65102/is2026440
Published: 30 April, 2026

by traditional artistic skills [13-15]. However, AI does not possess human emotions and individualistic creativity. Instead, it merely serves as a "production" tool independent of any social interaction [16, 17]. In any case, the fundamental principle of art is still "the expression of human emotions and thoughts." Although the integration of AI in art creation mainly contributes to expanding artistic genres, it does not negate the nature of art. Cooperation and collaboration models between AI and human artists point towards the future development trend of art.[18-21].

Reference [22] discusses an AI-based Support Vector Machine algorithm and studies AI-based computer-aided digital painting design from an experimental point of view, proving that computer-aided AI-based technologies could be used in digital painting design. Reference [23] reviews the connection between development of AI technology and interactive art forms, and analyzes this connection in terms of macro-historical evolution of both technologies and arts. Reference [24] highlights the extensive usage of AI technologies in such areas as painting and discusses the way to look at the integration of these two from a dialectical perspective. Reference [25] describes the application of AI technology to management systems, developing an intelligent design system for digital painting through analysis of the process and results of digital painting design. The purpose of such analysis is to understand the value of digital painting design and analyze the demand of Chinese digital painting. Reference [26] shows how AI changed the illustrators' work process, making it an indispensable component of digital illustration while creating some problems for illustrators. Literature [27] discusses the effect of artificial intelligence (AI) on digital painting by noting the benefits that artificial intelligence can bring to artists regarding improving their artistic abilities and efficiency while addressing issues such as ethics and technical constraints. Literature [28] revisits the development trends of AI in the arts to analyze the effects of artificial intelligence on the art world and possible future trends. Literature [29] proposes developing a painting learning system based on artificial intelligence and augmented reality. Experiments conducted in classrooms with learners reveal that using artificial intelligence technology during painting lessons boosts learners' creativity and artistic skills. Literature [30] describes the unique nature of digital painting as an artistic form by stressing the use of advanced digital technologies, including computer-aided design techniques, in the creation process. The author proposes using generative adversarial networks in digital art creation to improve expressive abilities. Literature [31] proposes developing a painting learning system based on deep learning algorithms to enhance learners' artistic creation capabilities.

The following paper is going to explore the use value and application methods for the implementation of artificial intelligence in digital paintings. The paper starts with an analysis of the software development for digital painting systems, focusing on creating an environment which allows for artificial intelligence function implementation. The initial stage comprises developing painting color conversion models and the key element of AI assistance—smart learning which possesses the property of evolution. Having laid down the fundamental groundwork, the paper goes further and considers the practical application of image processing techniques in painting. With the help of artificial intelligence algorithms, the computer can effectively perform such monotonous operations as grayscale transformation, improvement of images, and precise segmentation of pictures, especially portraits. In result, the fundamental development and key technologies are integrated into an AI-based stylization painting system. Being the ultimate combination of technologies, the system utilizes artificial intelligence algorithms in order to implement a certain style to the user's pictures.

2 Architecture and Image Processing Techniques for an AI-Based Digital Painting System

2.1 Digital Painting System Software Design

The successful application of AI-assisted creation techniques depends on an efficient software development platform. The development platform is designed based on two basic processes: firstly, the process of painting colors to convert into an intelligent color system; secondly, the learning process that makes the system capable of evolving continually. This paper discusses these two processes in detail.

2.1.1 Painting Color Conversion Methods

Conversion of colors in the painting process depends on the amount of energy that different control commands have. Let's assume that $s(e)$ is the signal released by the command for the painting process; then, its energy can be estimated using the formula:

$$E = \frac{s(e)}{\mathrm{TR}(x, y)} \tag{1}$$

In the formula: E represents the energy contained in the signal; $\mathrm{TR}(x, y)$ denotes the position requiring conversion to the color mapping region; (x, y) indicates the coordinates of the corresponding pixel. The energy conversion process establishes a linear relationship that translates into different color levels. Using the rainbow coding method, these levels are mapped to the three primary elements: red, green, and blue. The mapping functions for the three primary colors can be expressed as:

$$\begin{cases} R(x, y) = \dfrac{f(x, y) - 96}{32} \\ G(x, y) = \dfrac{f(x, y) - 193}{63} \\ B(x, y) = \dfrac{39 - f(x, y)}{27} \end{cases} \tag{2}$$

In the formula: $R(x, y), G(x, y), B(x, y)$ represent the red, green, and blue color element functions, respectively; $f(x, y)$ represents the color point function on the palette. The color element variations obtained from the above formula are divided into 50 equal segments. The color values of the basic elements within the color space are calculated and recorded. A 3×3 array chromosome encoding method is used to encode the functional units of the basic colors obtained. The encoding results are shown in Table 1.

Table 1: The encoding method of color elements

Bit-Red	Bit-Green	Bit-Blue	Configuration functional
1	1	1	Lshiftn
1	1	1	Maxn
1	0	0	Addern
1	0	0	Aan
0	1	1	Averagen
0	1	1	Suhn
0	0	0	Rshiftn
0	0	0	Minn

As shown in Table 1, the functional unit coding scheme defines the value 0 as disabling the function and the value 1 as enabling the configuration unit. After initializing the color unit, the code suppresses delays generated by the drawing system during task execution. Following the implementation of color variation functionality in the drawing system, the intelligent learning capability of the drawing system must be realized.

2.1.2 Implementation of Smart Learning Features

When implementing smart learning functionality, evaluate the adaptability of the configuration during the painting process based on hardware and color feature settings. Set the pixels of the generated image to $P(i, j)$, using the pixel values as input for adaptability. The resulting adaptability calculation formula can be expressed as:

$$F(i, j) = \frac{I(i, j) - m}{P(i, j)} \tag{3}$$

In the formula: $I(i, j)$ denotes the functional output value; m denotes the fitness evaluation parameter; i, j denotes the coordinates of the pixel point corresponding to the configuration function. The hardware configuration with the smallest fitness value is adopted as the learning constraint value. Once this value is passed, a placement parameter is set, which can be expressed as:

$$M_d = \frac{\sum_{i=0}^{R-1} \sum_{j=0}^{C-1} I(i, j) - F(i, j)}{CR} \tag{4}$$

In the formula: C denotes the column number where the pixel is located; R denotes the row number where the pixel is located. Using the above placement parameters as the reference standard, the intelligent learning process is divided into static and dynamic modules, ultimately forming the intelligent learning process shown in Figure 1.

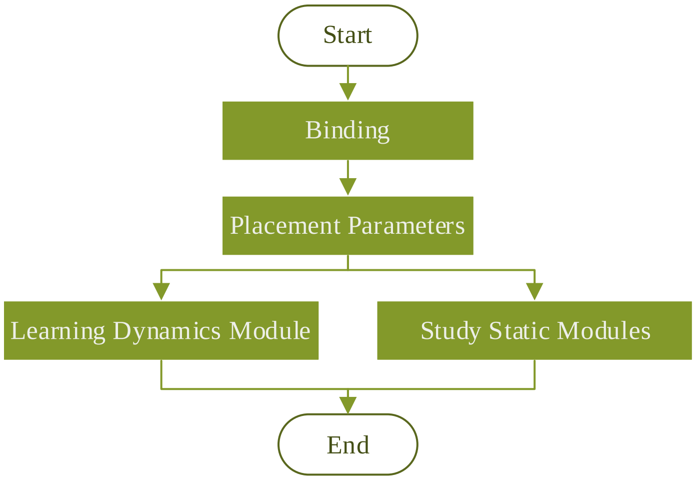

Figure 1: The process of achieving intelligent learning

In the intelligent learning process depicted in Figure 1, a recommendation function is implemented within the painting system to perceive the user's intent. The constructed recommendation function is as follows:

$$T(c) = \frac{M_d}{\varepsilon} c_{ij} \tag{5}$$

In the formula: $T(c)$ denotes the recommendation function; c_{ij} denotes the painting operation function; ε denotes the recommendation coefficient. In practical application, the digital painting system aggregates data generated by the aforementioned operations to form a painting behavior function. The recommendation function uses this painting behavior function as a benchmark to automatically match system operations that correspond to the painting behavior, thereby implementing intelligent learning capabilities within the digital painting system. Based on the aforementioned hardware and software research, the design of an intelligent learning-based digital painting system was ultimately completed.

2.2 Applications of Intelligent Image Processing Technology

After exploring the foundational architecture for designing digital painting system software, this paper shifts its focus to the in-depth application of intelligent image processing technologies in specific painting tasks.

The three most distinctive characteristics of portrait sketches are: prominent image contour information; clear representation of texture features; and vivid color gradation with sharp contrast between light and dark areas. Only when all three characteristics are satisfied can a portrait sketch be considered qualified. Therefore, intelligent image processing technology is primarily designed to address these three aspects. Figure 2 illustrates the application process of intelligent image processing technology in the automatic generation of portrait sketches.

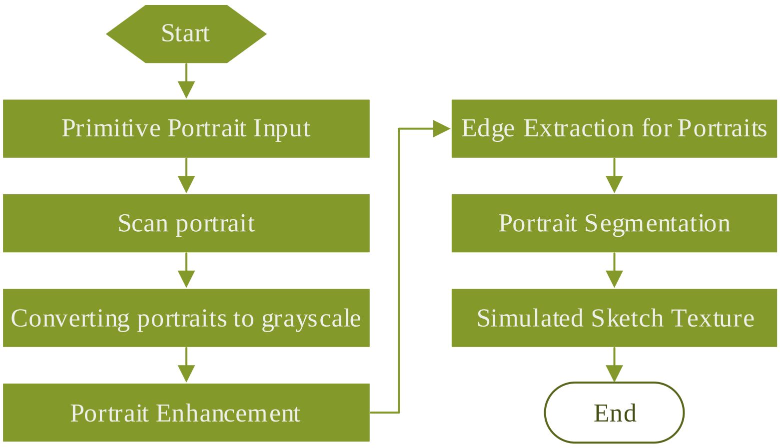

Figure 2: The Application in Automatic Generation of Human Portrait Drawings

As shown in Figure 2, the application of image processing technology in the automatic generation of portrait sketches primarily involves the following steps: First, the original portrait is input into the computer. It is then automatically scanned, followed by grayscaling, image enhancement, edge extraction, and segmentation. Finally, the system simulates sketch textures to generate a portrait sketch.

2.2.1 Grayscaling Portraits

The defining characteristic of pencil sketches is their relatively monochromatic palette, primarily composed of black, white, and gray tones. Therefore, the first step in automatically generating portrait sketches via computer involves converting the input color image into what we refer to as a black-and-white image. This is the layman's understanding of image grayscaling, while the technical term describes the process of making the R, G, and B color components of the original color image equal, i.e., $R = G = B$. Numerous methods exist for grayscaling images, with the most common being the component method, maximum value method, average value method, and weighted average method. The first three approaches are relatively straightforward, treating all three components equally without considering their relative importance. Consequently, this section employs the final method—weighted averaging—for grayscaling. The weighted averaging method assigns distinct weights to each of the three color components based on their relative importance. It then calculates the weighted average and applies this weighted mean to the grayscale conversion. This method can be described by the following mathematical formula:

$$f(i, j) = 0.30R(i, j) + 0.59G(i, j) + 0.11B(i, j) \tag{6}$$

As for changing pictures into grayscale images, the weighted averaging technique remains the most popular one due to its capability to generate more logical outcomes of grayscale conversion. Moreover, after performing this task, it becomes essential to invert the values of gray scale pixels for improving the automatic creation of portrait sketches.

2.2.2 Portrait Enhancement

Upon grayscaling an image, the object of interest becomes slightly less noticeable than before. Hence, from the perspective of human visual features, the eyes are keenly sensitive to bright colors but become insensitive after the image is grayscaled. In order to improve the quality of automatically generated sketch images, it is necessary to enhance the grayscaled image.

Enhancing an image serves as a means to compensate for the weaknesses that exist in the image by sharpening its details and contrasting it to make it more visible. This makes the image more distinguishable from other objects in the image making it easier for the computer to understand and recognize the image. There are basically two types of image enhancements. These include frequency domain enhancements and spatial domain enhancements. The types of enhancement include low pass filter, high pass filter, histogram equalization, and image smoothing. Although these methods help to enhance parts of interest in the image, they tend to introduce noise to the image.The procedure is described as follows: Firstly, the grayscale image is split into image elements at various frequency bands through wavelet decomposition, which includes low frequency and high frequency elements. They are handled separately. The low frequency element usually stands for the interesting area in the image, which is the target individual, and it can be smoothed by a single filter. The procedure is elaborated as follows: The built filter is used to filter the grayscale image. The grayscale value of every pixel is recorded, arranged from the lowest to the highest one, and then the median value is found. It is set for the center pixel of the image. The high frequency elements are usually associated with the uninteresting regions in the image, which do not include the target person. The handling procedure is based on threshold denoising, which includes the following procedures: Firstly, the high frequency elements are further divided. Then, the divided high frequency elements are quantized with a threshold for soft thresholding at each divided level.

2.2.3 Portrait Segmentation

Following the step of extracting the edges of the portraits, comes segmentation. This can be defined as the process of segmenting an image into several parts. There are four types of segmentation techniques used at present: threshold segmentation, region segmentation, edge segmentation, and theory segmentation. The method used in this study is edge segmentation. First, the portrait that has been extracted is binarized by setting its pixels either to 0 or to 255. This separates the target subject from the background, mathematically expressed as follows:

$$f'(x,y)=\begin{cases}0 & f(x,y)\geqslant R\\ 255 & f(x,y)<R\end{cases} \tag{7}$$

In the formula, $f'(x,y)$ represents the binary-converted portrait; $f(x,y)$ denotes the mathematical expression of the image; and R is a value within the grayscale range $[e,e']$.

The most critical aspect of image binarization is determining an appropriate threshold R. If R is set too high, it may cause misidentification of objects in the image; if set too low, recognition becomes impossible. The most commonly used methods for determining R include: the bimodal method, p-parameter method, de Bary method, maximum entropy threshold method, and iterative method. Here, the bimodal method is selected to determine threshold R.

The next step is to eliminate the non-featured connected regions in the binary image, and the process is as follows: (1) mark all pixels in the binary processed portrait with a gray value of 0 as unprocessed; (2) randomly select a pixel point from the pixels with a gray value of 0, and record it as O; calculate the 8 connected regions around O, and record it as $G(O)$, and then mark all the pixels in the 8 connected regions as processed; (3) Judge whether an edge point exists in $G(O)$, if it exists, the region is eliminated, if it does not exist, the region is retained, and the formula is as follows:

$$G(O)=\begin{cases}0, O\in[e,e']\\ 1, O\notin[e,e']\end{cases} \tag{8}$$

(4) Repeat steps 2 to 4 above until the status of all pixel points in the binary portrait is marked as processed and image segmentation is finished.

2.3 Artificial Intelligence Stylized Drawing System

Through the analysis of intelligent image processing techniques such as portrait graying, enhancement and segmentation, it can be seen that artificial intelligence is powerful in the underlying image processing. On this basis, the article further explores how to integrate these techniques into a more advanced AI stylized painting system to achieve the sublimation from a technical module to a complete creative tool.

2.3.1 System components

The artificial intelligence stylized drawing system consists of four parts, namely, text encoder, style module, image information generator and image decoder. Taking the generation of a 512×512 pixel image as an example, the overall flow of the system is shown in Fig. 3, which is divided into the following steps: first, the text encoder converts the text description input by the user into a vector feature, which is usually denoted as c. Second, the style module converts the vector feature based on the style parameter input by the user into a vector feature adapted to the style input by the user. Again, the image information generator receives this vector feature and converts it into an information array with a dimension of 4 × 64 × 64 dimensions. This array contains key information about the content and style of the image. Finally, the image decoder decodes and renders this information array into a final image with dimensions of 3×512×512 dimensions, where 3 denotes the number of color channels (red, green, and blue) and 512×512 denotes the width and height of the image.

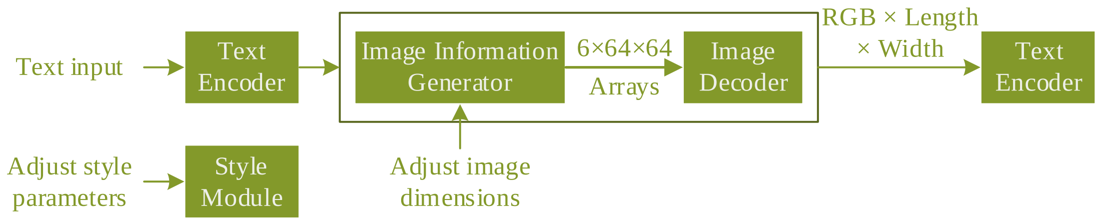

Figure 3: The overall process of the painting system

The various components of the system image of the system cooperate with each other to realize the complete generation process from the user's text input to the final image output. The user inputs a text description that determines the content. The user inputs parameters related to the five picture styles. These parameters are positive and negative integers that determine the style tendency; the size of the parameters determines the step size of the gradient descent in the aesthetic gradient method. After the user inputs the desired image aspect, the image information generator generates the size of the information array based on that aspect, which determines the pixel size of the final output image. The user input parameters and text embedding will determine the image result generated by the system.

2.3.2 System design and user experience

The system design and user experience is composed of 3 parts: perceptual vocabulary collection, style embedding model training and AI painting. In the first part, we collect and analyze network data to obtain user perceptual needs and summarize them, and get 5 groups of representative perceptual intention vocabularies: ancient-future, western-oriental, bright picture-dark picture, realistic-anime, and complex-simple. The image samples are collected and the image sample set is classified and labeled according to these 5 sets of vocabulary to obtain the training image set for each style, ensuring the diversity and comprehensiveness of the user's style choices. In part 2, the image training set is used to train the aesthetic embedding model, and the weights of the stylized text encoder are calculated according to the aesthetic embedding model and applied to the painting, which ultimately ensures that the styles of the system's output images have consistency with the user's needs. In part 3, the user inputs the text description and image style parameters through the operator interface, which are processed by the stylized text encoder and then passed through the image information generator and the image decoder to generate the final image.

3 Performance evaluation and experimental analysis of artificial intelligence-assisted drawing system

After completing the architecture design and key technology implementation of the AI-based digital painting system, in order to verify the performance of the described system in practical applications, this paper further carries out the experimental evaluation and analysis of the system. Chapter 3 will comprehensively evaluate the comprehensive performance of this system through quantitative experiments and comparative analysis around multiple dimensions such as automatic color matching, image generation effect, line processing performance and system resource consumption.

3.1 Painting color auto-matching effect

3.1.1 Experimental design

In order to verify the automatic matching effect of painting colors in the artificial intelligence stylized painting system in this paper, comparison experiments are conducted with the current popular classification models.

(1) Comparison Models

The baseline models are traditional support vector machine (SVM), K-nearest neighbor support vector (KNN-SVM), boundary point-based support vector machine (BP-SVM), and deep learning-based support vector machine (CNN-SVM).

(2) Evaluation metrics

Peak signal-to-noise ratio (PSNR) and structured similarity (SSIM) are used to evaluate the accuracy of color auto-matching between this system and the above models.

PSNR belongs to the quantitative calculation of the error between the completed image and the original image of the color auto-matching, the larger the value of PSNR, the smaller the distortion of the system's color auto-matching; SIMM indicates the similarity between the structure of the image after the color auto-matching and the structure of the original image, and the closer the value of SIMM is to 1, the better the effect of the system's color auto-matching on the image.

(3) Data set

The experiments use a publicly available, large-scale line coloring benchmark dataset CBD, which is widely used to evaluate the performance of automatic coloring algorithms and contains more than 100,000 high-quality image pairs. The samples cover a wide range of subjects such as natural landscapes, portraits, still lifes, and architecture.

3.1.2 Analysis of experimental results

In the dataset, 300 uncolored line drawings were randomly selected, and the painting system based on the above four different classification methods and this system were used to implement color auto-matching for these 300 uncolored pictures, and the PSNR and SIMM values of the color auto-matching of the five systems were tested, and the results of the tests are shown in Table 2 and Table 3, respectively. In order to show the effect gap between the model systems more clearly, the 3D comparisons of the five models under different numbers of pictures are also drawn as shown in Figures 4 and 5.

Table 2: The PSNR values of five system colors automatically matched

Number of pictures	PSNR/dB				
	SVM	KNN-SVM	BP-SVM	CNN-SVM	OURS
50	30.44	31.95	31.26	32.60	33.48
100	29.37	30.69	31.11	31.34	33.44
150	28.48	29.73	30.20	29.38	31.41
200	27.65	29.32	29.68	28.62	31.35
250	27.25	28.53	28.99	27.84	32.24
300	26.53	27.47	28.50	27.32	31.22

Table 3: The SIMM values of five system colors automatically matched

Number of pictures	SIMM				
	SVM	KNN-SVM	BP-SVM	CNN-SVM	OURS
50	0.8254	0.8554	0.8967	0.9430	0.9914
100	0.7490	0.7828	0.8128	0.9053	0.9925
150	0.6666	0.7078	0.7679	0.8666	0.9852
200	0.6177	0.6578	0.7041	0.6852	0.9869
250	0.5515	0.5853	0.6829	0.6466	0.9852
300	0.5190	0.5565	0.6491	0.5778	0.9861

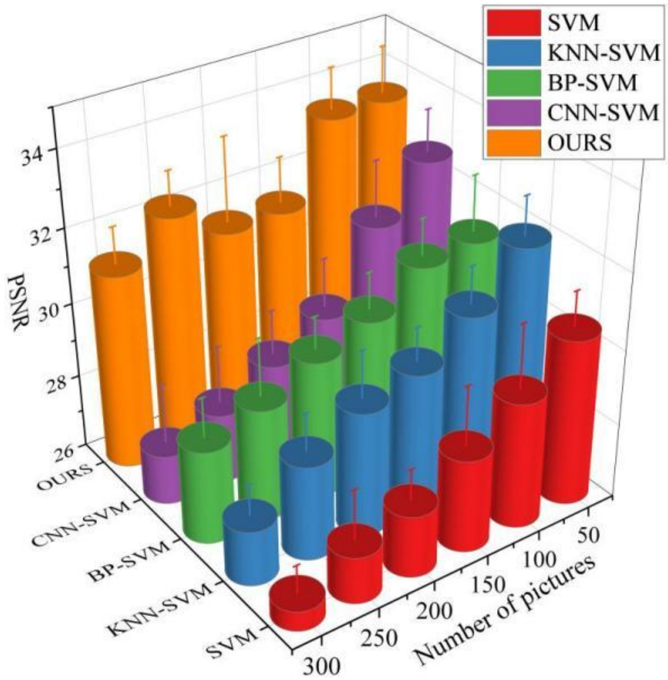

Figure 4: The PSNR values of five system colors automatically matched

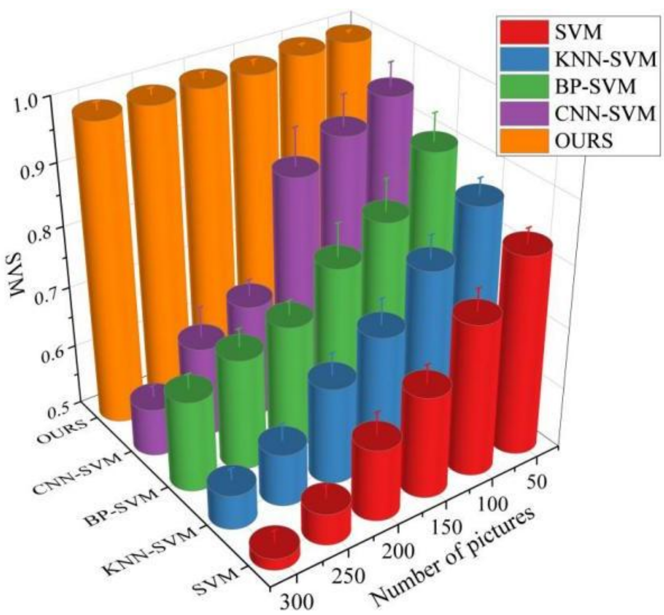

Figure 5: The SSIM values of five system colors automatically matched

Regarding color similarity, the PSNR of the proposed approach always performed better among different numbers of test images, always being in the range of 31.22dB to 33.48dB. However, for all compared models, there was a gradual reduction in PSNR as the number of images increased. For example, the SVM approach had its PSNR fall from 30.44dB using 50 images to 26.53dB using 300 images. The performance of these models shows that conventional approaches are highly sensitive to variations in the dataset as far as their ability to colorize is concerned.

In terms of structural similarity between images, this is where our system truly shines, as we can always rely on SSIM to be in the vicinity of 0.9852 and 0.9925—values extremely close to the optimum of 1. This clearly shows that the colorized images created by our system remain structurally very consistent with the source images. In turn, there is an observable trend of significant SSIM reduction as the number of tested images increases for alternative models. More precisely, after passing the number of 200 images, SSIM for the CNN-SVM model becomes around 0.65, whereas SSIM in the case of the traditional SVM becomes less than 0.70.

3.2 Image Effects Generated by Digital Painting Systems

Following the confirmation of the excellent results of automatic color matching provided by the system, it becomes important to conduct a further study of its performance for overall image generation tasks. This part of the paper is intended to examine the advantages of the system for image generation in a comparative experiment.

In order to verify the usefulness of the AI generated pictures from the AI style painting system in this paper, two different types of experiments were performed. First was interactive painting using style transfer (Method 1), while second was intelligent painting using image texture rendering (Method 2).

More experiments were conducted on the CBD data set, which was classified into seven categories based on theme, namely natural scenery, portraits, animals, flora, still life, architecture, and automobiles. Images in each of these styles were synthesized by AI using the three approaches, with a hundred images per category. The PSNR and SSIM were chosen as the main criteria for evaluating their performance.

The mean value of the Peak Signal to Noise Ratio (PSNR) and Structural Similarity (SSIM) values for the images created by using the three different methods are tabulated in Table 4. Moreover, bubble charts are plotted for the three methods based on the Peak Signal to Noise Ratio (PSNR) and Structural Similarity (SSIM) in Figures 6, 7, and 8.

Table 4: The PSNR and SSIM values of the AI images generated by the three methods

	PSNR/dB			SSIM		
	Method 1	Method 2	OURS	Method 1	Method 2	OURS
Nature	56.73	63.18	77.31	0.7477	0.8446	0.9814
Person	54.05	61.64	75.83	0.7336	0.847	0.9765
Animal	55.10	62.55	76.47	0.7366	0.8346	0.9842
Plant	57.18	63.60	78.06	0.7411	0.8505	0.9879
Object	57.27	64.01	78.72	0.7561	0.8543	0.9987
Architecture	56.58	63.63	77.94	0.7347	0.8389	0.9707
Transportation	56.47	63.74	76.82	0.7531	0.8354	0.9751

In terms of peak signal-to-noise ratio, our system significantly outperforms the comparison methods across all seven subject categories. These PSNR values lie within a range of 75.83 dB to 78.72 dB, reaching the maximum of 78.72 dB for the "still life" category. In comparison, Method 1, which uses an interactive painting scheme based on style transfer, gives PSNR values from 54.05 to 57.27 dB, and Method 2, which employs an intelligent painting scheme using image texture rendering, produces PSNR values in the range of 61.64 to 64.01 dB. Thus, it is clear that the images produced using our system have less errors than those produced by real images and better color accuracy and SNR performance.

Regarding structural similarity measurements, our algorithm also exhibits impressive performance, with SSIM scores always exceeding 0.97, especially achieving almost perfect scores of 0.9987 when dealing with the “still life” topic. On the contrary, Methods 1 and 2 have resulted in SSIM scores ranging from 0.73 to 0.85, much lower compared to our proposed algorithm. From this, we can conclude that our algorithm is not only efficient in accurately reproducing pixel values but also successfully maintains structural information and visual consistency, generating outputs similar to the actual structure of real-world images.

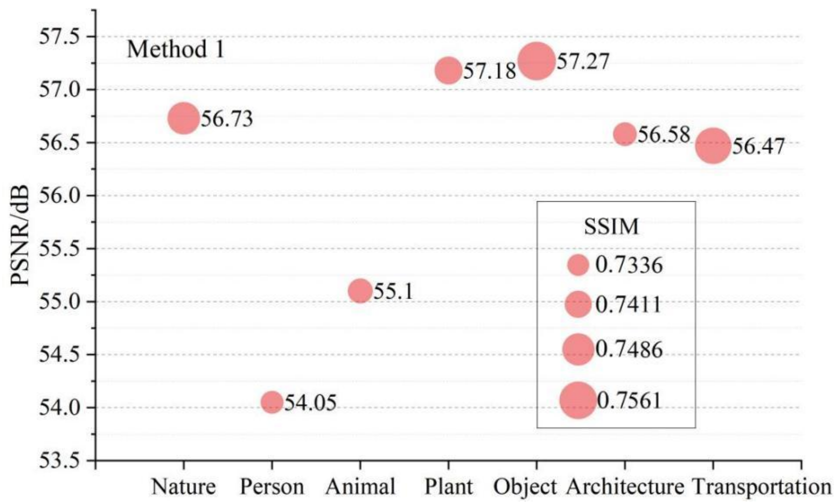

Figure 6: The PSNR and SSIM values of the AI images generated under Method 1

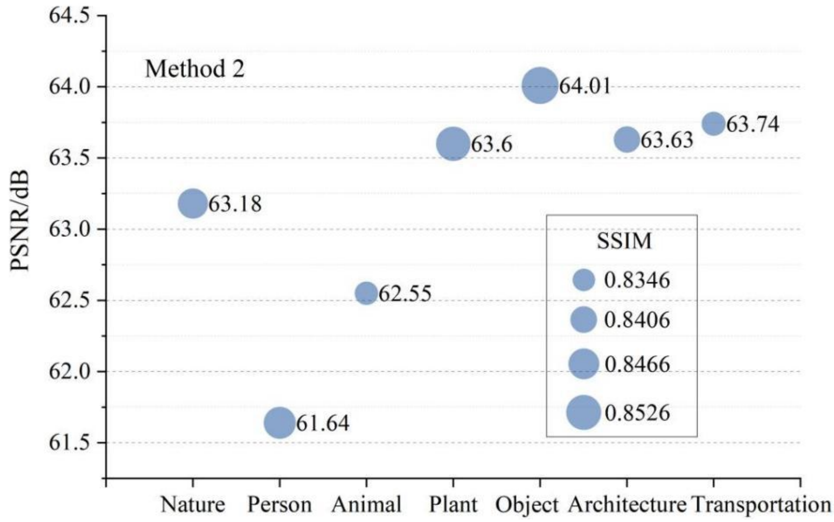

Figure 7: The PSNR and SSIM values of the AI images generated under Method 2

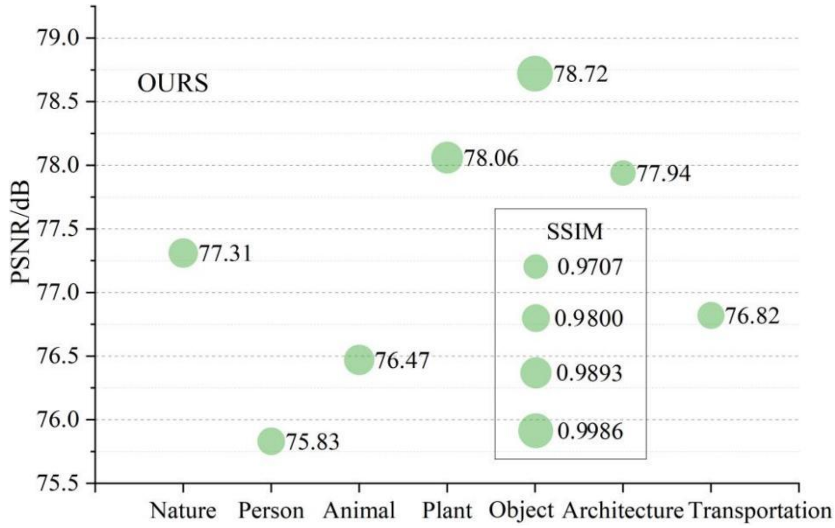

Figure 8: The PSNR and SSIM values of the AI images generated under OURS

From a combination of the two, it is evident that our AI system always creates high-quality images regardless of the situation, be it the natural environment, portrait painting, or complicated architecture. Furthermore, from the bubble graph, we observe that the AI system is capable of creating high quality images with respect to both PSNR and SSIM for several subjects, thereby proving their overall superiority when it comes to color preservation and image structures.

3.3 Performance Analysis of Drawing Line Processing

Apart from the quality of images generated by the system, the ability to process lines is an important measure of performance in digital painting applications. This section discusses the performance of the application in processing lines based on experimental results on different curve complexities.

Inputted into the computer interface were 500 curved lines of varying lengths where the more curved lines were those with longer lengths. The average time taken to process the curves using each technique is illustrated in Figure 9.

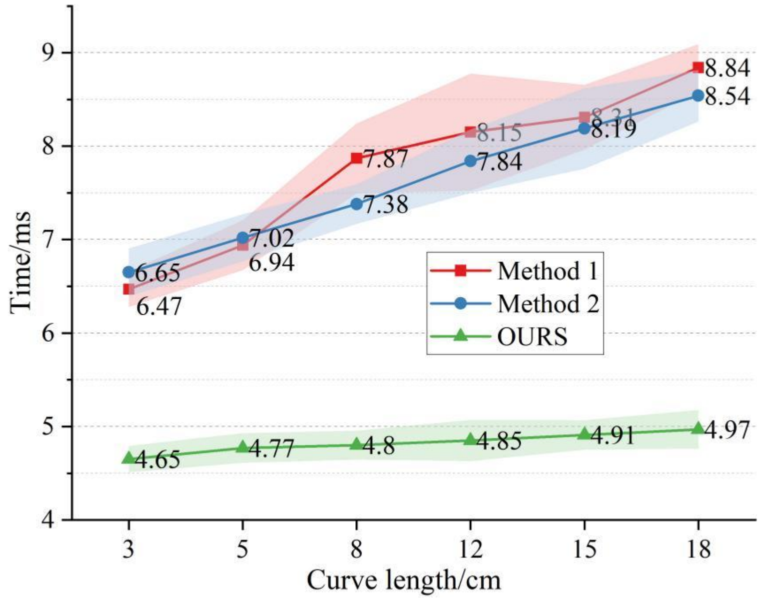

Figure 9: The average time cost for processing each method's curve

With the length of the curve going from 3cm to 18cm, the complexity of the curvature of the line became higher. For both the time taken by all the three approaches, there is a clear increase, however, this difference in time taken is not proportional to the processing efficiency of each approach. For Method 1, the processing time is seen to go up from 6.47ms to 8.84ms. Similarly, the processing time of Method 2 is seen to increase from 6.65ms to 8.54ms. There is a significant rise in the time required in processing when considering highly complex curves. On the other hand, the time taken by the proposed approach is quite low; gradually increasing from 4.65ms to 4.97ms. Even when dealing with the highly complex 18cm curve, its time taken is significantly lesser than the other two methods' times taken to process a relatively simpler 3-unit curve. This clearly illustrates the fact that the line processing algorithm used in this proposed method has an extremely efficient and stable characteristic.

3.4 Analysis of Resource Consumption in Graphics Rendering

Apart from analyzing the functionality of the system, it is also important to consider how many resources the system consumes when it is working. To determine the GPU (Graphics Processing Unit) and CPU (Central Processing Unit) usage of each digital painting method in 3D rendering, 500 randomly chosen digital paintings from the CBD database were rendered using the three methods. Lighting, texture, and scene rendering were used to evaluate the rendering power of the three digital painting methods. The usage of the GPU and CPU when the three digital painting methods render is presented in Figures 10 and 11, respectively.

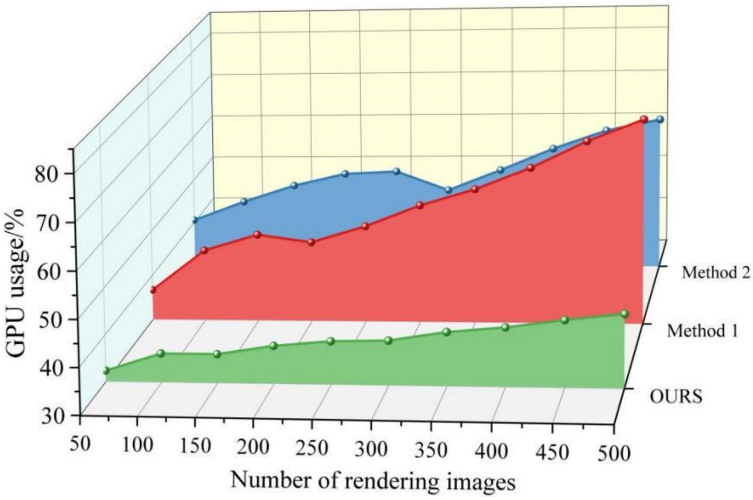

Figure 10: The usage of computer GPU during the rendering process

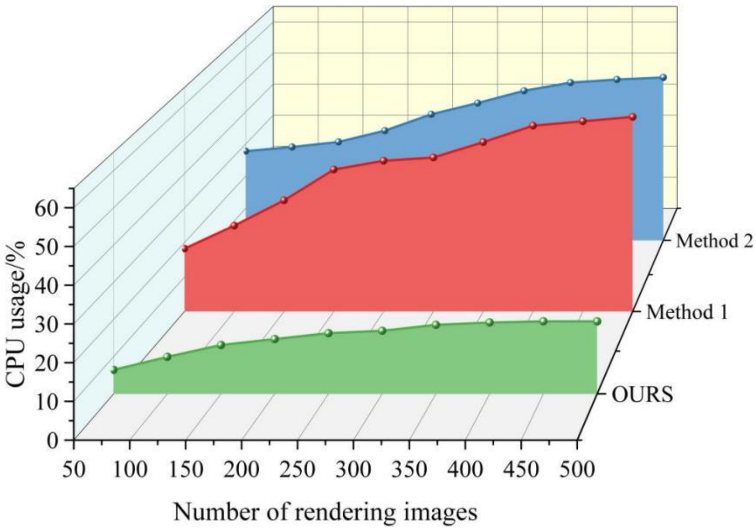

Figure 11: The usage of computer CPU during the rendering process

It can be easily seen from the above graph that there is a remarkable benefit of the proposed method in terms of efficient use of computing resources. With an increase in the number of renderings, the rise in the use of GPU and CPU is visible for all three methods. But the rate at which the rise takes place is relatively lower for the proposed method.

As far as the usage of the graphic card is concerned, initially, there was a GPU usage of 32.34% by the proposed method, which increased to only 45.45% later on. On the other hand, the GPU usage for Method 1 and Method 2 escalated rapidly to 74.38% and 63.74% from 36.71% and 40.53%, respectively. By rendering 500 frames, it was evident that there was 28.93% lower GPU usage of the proposed method than Method 1, and 18.29% lower than Method 2.

Even more clearly, the benefit of the proposed approach can be seen when it comes to CPU performance. The CPU usage in this technique rises steadily from 6.43% to 19.5%, while in Method 1, it spikes up from 18.23% to 56.44% and in Method 2, it grows from 27.77% to 50.72%. At full capacity rendering, the difference between the CPU performance of the proposed approach and Method 1 is about 36.94 percentage points and Method 2 is around 31.22 percentage points.

Generally, the low GPU and CPU usage by the proposed method is an indication of how efficient the rendering process is. This means that the use of less hardware resources while producing high-quality images is possible. In addition to enhancing the real-time rendering process of digital paintings, this will offer sustainable solutions to large or complex scenes. This goes to confirm the effectiveness and superior nature of the proposed AI digital paintings system.

4 Conclusion

The question of AI-supported creation of digital paintings has been resolved with the successful development of the AI stylization system, which combines smart learning abilities with intelligent image processing techniques.

(1) Speaking about automatic color matching, PSNR values for this system vary within 31.22 dB to 33.48 dB, whereas SSIM values are 0.9852 to 0.9925; thus, these results outperform those obtained by traditional classification algorithms, such as SVM and KNN-SVM.

(2) Speaking of the quality of stylized images, in comparison with two main approaches to intelligent painting, generated images on seven primary topics (natural landscapes, portraits, etc.) show significantly better PSNR values (75.83 dB ~ 78.72 dB) and SSIM values (>0.97).

(3) Concerning the performance of the line process, the system is capable of providing fast and effective results in terms of real-time operation; processing time of various complexity curves does not exceed 5ms (4.65ms to 4.97ms).

(4) As regards the usage of resources, although processing of 500 images is required, GPU and CPU loads do not go beyond 45.45% and 19.5%, respectively, meaning low resources consumption and smooth curve growth.

About the Author

Yuan Tian (1990-3) female, Han Chinese, born in Nanjing, Jiangsu Province, doctor, Nanhang Jincheng College, lecturer, research interests: digital media art graphics and images, computer art design.

References

[1] Sochorová, Š., & Jamriška, O. (2021). Practical pigment mixing for digital painting. ACM Transactions on Graphics (TOG), 40(6), 1-11.

[2] Blatner, A. M., Ferwerda, J. A., Darling, B. A., & Bailey, R. J. (2011, January). TangiPaint: a tangible digital painting system. In Color and Imaging Conference (Vol. 19, pp. 102-107). Society of Imaging Science and Technology.

[3] Milad Zreiba, H. (2024). Digital Art as a Medium to Enrich a Contemporary Pictorial Painting. Arts and Architecture Journal, 5(2), 11-20.

[4] Zhang, Y. (2023). An Analysis of the Commercial Prospect of Digital Painting. Frontiers in Art Research, 5(4).

[5] Aris, S., Aeini, B., & Nosrati, S. (2023). A digital aesthetics? artificial intelligence and the future of the art. Journal of Cyberspace Studies, 7(2), 219-236.

[6] Rani, S., Jining, D., Shah, D., Xaba, S., & Singh, P. R. (2023, April). The role of artificial intelligence in art: a comprehensive review of a generative adversarial network portrait painting. In International Conference on Intelligent Computing & Optimization (pp. 126-135). Cham: Springer Nature Switzerland.

[7] Blum, E. (2025). Painting with AI: Enhancing Creativity and Understanding in the Arts Classroom. International Journal of Emerging and Disruptive Innovation in Education: VISIONARIUM, 3(1), 5.

[8] Ali Elfa, M. A., & Dawood, M. E. T. (2023). Using artificial intelligence for enhancing human creativity. Journal of Art, Design and Music, 2(2), 3.

[9] Kurt, D. E. (2018). Artistic creativity in artificial intelligence. Master diss., Radbound University.

[10] Qiu, G., & Zhang, J. (2023). Application of digital technology in painting using new media and big data. Soft Computing-A Fusion of Foundations, Methodologies & Applications, 27(17).

[11] Stark, L., & Crawford, K. (2019). The work of art in the age of artificial intelligence: What artists can teach us about the ethics of data practice. Surveillance & Society, 17(3/4), 442-455.

[12] De Cock Buning, M. (2018). Artificial intelligence and the creative industry: new challenges for the EU paradigm for art and technology by autonomous creation. In Research handbook on the law of artificial intelligence (pp. 511-535). Edward Elgar Publishing.

[13] Trach, Y. (2021). Artificial intelligence as a tool for creating and analysing works of art. Culture and arts in the modern world, 22, 164-173.

[14] Mihăilă, C. O. (2023). ARTISTIC CREATIONS. ORIGINALITY, INSPIRATION, IMITATION AND ARTIFICIAL INTELLIGENCE. Challenges of the Knowledge Society, 541-558.

[15] Xu, L., & Cheng, P. (2024, November). The Application of Artificial Intelligence in the Creation of Four Grid Painting and Product Design. In Proceedings of the 2024 International Conference on Artificial Intelligence, Digital Media Technology and Interaction Design (pp. 536-543).

[16] Oksanen, A., Cvetkovic, A., Akin, N., Latikka, R., Bergdahl, J., Chen, Y., & Savela, N. (2023). Artificial intelligence in fine arts: A systematic review of empirical research. Computers in Human Behavior: Artificial Humans, 1(2), 100004.

[17] Ke, M. F. (2023). Applications and challenges of artificial intelligence in the future of art education. Pacific International Journal, 6(3), 61-65.

[18] Liu, Q. (2022, March). The construction of autonomous classification system of digital painting images based on artificial intelligence technology. In The International Conference on Cyber Security Intelligence and Analytics (pp. 493-501). Cham: Springer International Publishing.

[19] Giannini, T., Bowen, J., Michaels, C. A., & Smith, C. H. (2022, July). Digital art and identity merging human and artificial intelligence: Enter the metaverse. In Proceedings of EVA London 2022 (pp. 1-7). BCS Learning & Development.

[20] Sabetsarvestani, Z., Sober, B., Higgitt, C., Daubechies, I., & Rodrigues, M. R. (2019). Artificial intelligence for art investigation: Meeting the challenge of separating x-ray images of the Ghent Altarpiece. Science advances, 5(8), eaaw7416.

[21] Işık, V. (2024). Exploring artistic frontiers in the era of artificial intelligence. Sanat ve Tasarım Dergisi, 14(2), 577-603.

[22] Zhao, B., Zhan, D., Zhang, C., & Su, M. (2023). Computer-aided digital media art creation based on artificial intelligence. Neural Computing and Applications, 35(35), 24565-24574.

[23] Shen, Y., & Yu, F. (2021). The influence of artificial intelligence on art design in the digital age. Scientific programming, 2021(1), 4838957.

[24] Liu, X. (2020, October). Artistic reflection on artificial intelligence digital painting. In Journal of Physics: Conference Series (Vol. 1648, No. 3, p. 032125). IOP Publishing.

[25] Yu, Y. (2016). Research on digital art creation based on artificial intelligence. Revista Ibérica De Sistemas e Tecnologias De Informação, (18B), 116.

[26] Agista, Y. A., & Murtono, T. (2024). The Role and Challenges of Artificial Intelligence in Digital Illustration Work at DB 2.4 Studio Surakarta. In Proceeding of Internasional Seminar on Arts, Artificial Intelligence & Society (pp. 152-171).

[27] Castelán-Urquiza, D., & Monroy-Mondragón, C. (2024). Artificial Intelligence in Artistic Creation: Revolutionizing Digital Drawing. Journal Educational Theory/Revista de Teoría Educativa, 8(19).

[28] Rani, S., Jining, D., Shah, D., Xaba, S., & Shoukat, K. (2025). Examining the impacts of artificial intelligence technology and computing on digital art: A case study of Edmond de Belamy and its aesthetic values and techniques. AI & SOCIETY, 40(4), 2417-2435.

[29] Zhang, W., & Seong, D. (2024). Cultivating creativity and engagement in painting education through AI digital painting: An interdisciplinary approach to enhance student learning experience. Arts Educa, 41.

[30] Zhang, X., & Wu, L. (2024). Automated method for digital art creation and display based on computer aided design. Computer-Aided Design and Applications, 21, 140-153.

[31] Chiu, M. C., Hwang, G. J., Hsia, L. H., & Shyu, F. M. (2024). Artificial intelligence-supported art education: A deep learning-based system for promoting university students' artwork appreciation and painting outcomes. Interactive Learning Environments, 32(3), 824-842.