Pakistan Journal of Life and Social Sciences
www.pjlss.edu.pk

https://doi.org/10.57239/PJLSS-2024-22.2.001453

RESEARCH ARTICLE

Combining Bisecting K-Means and Conditional Generative Adversarial Network for Crafting Harmonious Color Palettes

Louis Aldorio Efendi[1*], Gede Putra Kusuma[2]

[1,2] Computer Science Department, BINUS Graduate Program – Master of Computer Science, Bina Nusantara University, Jakarta, Indonesia, 11480

ARTICLE INFO

Received: Sep 18, 2024

Accepted: Nov 13, 2024

Keywords

Color Palette

Color Clustering

Bisecting K-Means

Conditional Generative Adversarial Network

Harmonious Color Palette

***Corresponding Author:**

louis.efendi@binus.ac.id

ABSTRACT

Crafting harmonious color palettes is vital in fields like design, art, and marketing. This paper introduces a novel approach by combining Bisecting K-Means clustering with a Conditional Generative Adversarial Network (CGAN) to generate harmonious color palettes. The Bisecting K-Means algorithm clusters an art image dataset, representing the ground truth palettes. A CGAN is then trained on these clusters to generate new, harmonious color combinations. The research utilizes two clustering methods—K-Means and Bisecting K-Means—as feature extractors on the WikiArt dataset, which comprises artworks from renowned painters across history. The effectiveness of the approach is evaluated using the CIEDE2000 metric, where lower values indicate better performance. The proposed method achieves a score of 23.5836, improving upon the baseline of 24.8320 by 1.2484. The results show that this method generates color palettes that are both diverse and visually appealing while adhering to principles of color harmony. This research contributes by providing an automated, robust framework for color palette generation, enhancing creativity and design processes.

INTRODUCTION

A color palette is a collection of colors used in art, design, or creative projects. It is not merely decorative but significantly influences the observer's perception of a work. The complex interplay of colors within a palette creates an overall atmosphere and serves as a key element in the creative process. Color harmony refers to aesthetically pleasing arrangements of colors in a composition (Burchett, 2001). Despite the development of intuitive skills over time, defining why certain colors blend well remains challenging. Creating algorithmic rules to explain this phenomenon is even more complex (Moussa & Watanabe, 2022). According to Itten's color theory (Itten, 1974), colors have a strong psychological impact on human emotions and perceptions. For example, red is often associated with energy and courage, while blue is linked to calmness and stability. Choosing a color palette involves considering color psychology to achieve the desired communication or message. Color harmony encompasses various combinations such as monochromatic, analogous, complementary, and triadic. The right harmony enhances the visual appeal of a design and aids in its comprehension. Cultural and social contexts also play a crucial role; for instance, white symbolizes purity in some cultures but signifies mourning in others. Understanding these contexts is essential to avoid misinterpretations. Modern tools like Adobe Color CC generate color palettes based on user preferences and imported images (Pan & Westland, 2018). A harmonious color palette, as shown in Figure 1, may contain varying numbers of colors depending on the intended use.

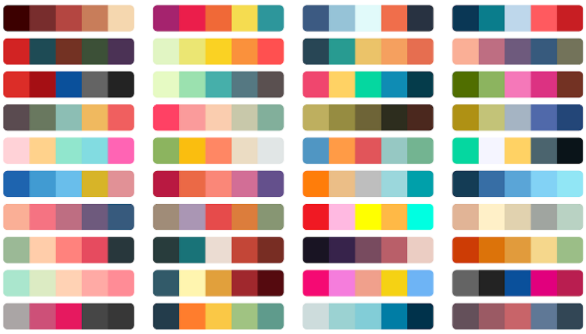

Figure 1: Color palettes

Creating harmonious color palettes has grown more complex over time, surpassing the basic use of color wheels. Color wheels suggest harmony without assessing it, often leading to ambiguous and laborious combinations (T.Wei, et al., 2019). Figure 2 illustrates the visual discomfort of non-harmonious palettes versus the aesthetic appeal of harmonious ones.

Figure 2: Harmonious and non-harmonious color palette

A harmonious color palette can be quantitatively measured through color brightness and chromaticity relationships. Colors are often considered aesthetically pleasing when they exhibit similar brightness and chromaticity levels, promoting visual balance and unity (Yang, Chen, Xiao, & Westland, 2020). Common methods for measuring color palette harmony include CIELab and CIEDE2000 (Gomez-Polo, et al., 2016). This paper introduces a novel approach combining Bisecting K-Means and Conditional Generative Adversarial Networks (CGAN) to automate the creation of harmonious color palettes. This method leverages clustering and machine learning techniques to generate visually appealing and contextually appropriate color combinations, aiming to simplify and enhance the design process.

Hue Sort, Brightness Sort, and Binary Palette Sort for sorting colors in extracted palettes from paintings using the k-means algorithm was proposed (Phan, Fu, & Chan, 2018). They introduced the GPLVM (Gaussian Process Latent Variable Models) for predicting new colors to be added to the palettes. Their evaluation showed that Binary Palette Sort outperformed the other methods, with the smallest average color distance of 0.65 evaluated with Eucleadian Distance of CIELab, achieved when using the combination of GPLVM and BPS. This method relies heavily on the initial palette extraction, which may not capture all relevant colors from the paintings. Various modified ArtGAN models were compared (Tan, Chan, Aguirre, & Tanaka, 2018). They introduced an Image Quality Strategy where the generator produces images at twice the resolution, which are then downsampled before being passed to the discriminator. Their evaluations using Inception Score and Objectness Score revealed inconsistencies between these metrics, highlighting the importance of visual inspection for generative model evaluation. Inception Score (IS) and Objectness Score (OS) revealed inconsistencies. IS: 7.9, OS: 0.58. There were significant inconsistencies between the Inception Score and Objectness Score, emphasizing the need for visual inspection as a complementary evaluation method. Jeong, Yang, & Shin (2019) combined statistical computations, superpixel segmentation, DBSCAN, and Hierarchical Agglomerative Clustering to generate color palettes from images. Their method focused on identifying pure pixels and grouping similar pixels to reduce variation. However, their evaluation was purely observational with no quantitative metrics provided. Sensitivity to sensor noise and difficulty handling fine textures were significant challenges, limiting the robustness of their method.

Reyes & Lara-Alvarez (2019) proposed a geometric approach to formulating color palettes. They evaluated three chromatic circle patterns (analog, complementary, and triad) and assessed spatial color relationships to identify harmonious colors. Their experiments revealed significant influences of hue patterns and linear patterns on color harmony perception, but they acknowledged the subjective nature of color perception and the limitations of their geometric approach. Significant influences of hue patterns and linear patterns on color harmony perception were observed. The approach is subjective due to the nature of color perception, and the geometric method has inherent limitations in addressing complex color relationships. Peng & Chou (2019) used NLP to generate color palettes from text descriptions, mapping phrases to predefined color palettes. Their work did not include quantitative evaluation, focusing instead on visual inspection. They noted the need for more comprehensive evaluation and the limitations imposed by fixed mappings of phrases to palettes. The method is limited by the fixed mappings of phrases to palettes, which may not capture the full nuance of text descriptions. Lertrusdachakul, Ruxpaitoon, & Thiptarajan (2019) generated color palettes from images using K-Means clustering, followed by hue and saturation analysis. They proposed two methods for selecting representative colors: one based on hue differences and clustering, and the other on the brightest colors in Munsell color groups. Their qualitative evaluation using a Likert scale showed satisfactory results across different image types (3.67 for photos, 3.68 for scanned images, and 3.69 for computer graphic images). The method relies on qualitative assessment, which can be subjective and may not provide a comprehensive evaluation of the color palettes.

Lu, et al (2020) proposed a method for generating color palettes optimized for data visualization. They formulated the task as an optimization problem balancing point distinctness, name difference, and color discrimination. Their method used simulated annealing to select optimal colors, ensuring clarity and distinction in visual data representations. Their evaluation showed that their method produced color palettes that improved visual distinction in data visualizations compared to traditional methods. The optimization process can be computationally intensive, and the method's effectiveness may vary depending on the specific data visualization context. No quantitative or qualitative evaluations were conducted in the research. Westland & Lai (2020) used standard K-Means clustering to create color palettes from fashion images, focusing on preprocessing by cropping image areas before color extraction. The research used 48 Burberry fashion show images and gathered color palettes from 22 respondents as ground truth. The evaluation showed that the ΔE value, indicating color difference in LAB color space, was 6.6 for the ground truth, 14.9 for the model without preprocessing, and 7.7 for the model with preprocessing, making the latter more accurate. However, the segmentation method included skin color, potentially reducing accuracy when clothing colors overlapped with skin tones. Further exploration is needed to overcome these limitations and to investigate segmentation in other color spaces beyond the simple k-means approach. Kim & Kang (2021) developed a GAN-based Color Palette Extraction System that uses Chroma Fine-tuning with Reinforcement Learning. The method involves three steps: extracting RGBY image features using a Super Resolution Convolutional Neural Network (SRCNN), training a GAN with SRCNN as part of its Generator, and using Reinforcement Learning to fine-tune the chromatic values of the generated color palettes. This approach achieved a high accuracy of 0.9140, evaluated based on the difference between predicted and actual RGB values within an error margin of ±15 units per channel. The dataset consisted of 100,000 images of paintings and 100,000 movie posters created by the researchers.

Huang (2021) explored efficient palette generation for color image quantization. The method involves a unique algorithm derived from the researcher's original idea, dividing the RGB color space into 4096 non-overlapping cubes of 16x16x16 each. Initially, to create the color palette, cubes are selected based on having a number of points greater than or equal to a certain threshold. The centroid of the selected cubes is then calculated, and if not yet chosen, they are included in the color palette as initial colors. For subsequent colors, the Euclidean distance between each unselected cube's centroid and the already selected color in the palette is calculated. The smallest value from all Euclidean distance calculations is then biased based on the initial number of points. The largest biased value determines the next color in the palette. This process repeats until the desired palette size is reached. This method successfully quantized high-quality images into 256 colors while maintaining low computational costs, particularly with a sample rate of 0.125 taking only 0.3 seconds. The dataset used was the USC-SIPI Image Database. Liu, Tao, Huang, Wang, & Li (2022) proposed a method for generating color palettes from images using BASNet for saliency score prediction and a Super Pixel

Fully Convolutional Network for segmentation. They mapped hue values to a saliency-hue circle and clustered them to create palette candidates. An algorithm ensured hue diversity and avoided similar colors, adding more colors if the desired palette size wasn't reached. Evaluations included qualitative user experiments with a 72.22% preference rate, quantitative reproduction of designer palettes scoring 186.60 in the Minimum Color Difference Model, and tests on low-rated AVA dataset images, showing good palette quality. The method's limitations include recommended palette sizes of 4 to 7 colors; too many ($n \geq 10$) results in similar colors, while too few may overlook less visually significant areas. Yuan, et al (2022) introduced a VAEAC-based method to recommend color palettes customized for infographic layouts. Using the InfoVIF dataset of 13,245 infographic images, they trained the VAEAC model on extracted features. The evaluation involved four steps: qualitative case studies, controlled user studies, surveys, and interviews. Artist-designed palettes scored highest in color harmony (mean = 5.45; 95% CI = [5.31; 5.59]) and readability (mean = 5.72; 95% CI = [5.59; 5.85]). Among other methods, InfoColorizer-recommended palettes scored the highest in harmony (mean = 4.60; 95% CI = [4.44; 4.75]) and readability (mean = 5.20; 95% CI = [5.06; 5.34]). Significant differences between methods were found using the Friedman test, with InfoColorizer performing better than random and ColorBrewer palettes in both harmony and readability. Moussa & Watanabe (2022) introduced two neural network variations for generating color palettes from images: a Variational Autoencoder (VAE) and a VAE GAN (Variational Autoencoder Generative Adversarial Network). The VAE uses a Convolutional Encoder, an intermediate sampling layer, and a Bidirectional LSTM decoder to extract features and generate color palettes. The VAE GAN adds adversarial training with a generator and two discriminators to produce high-quality palettes. Using datasets from DesignSeeds and ColorPalettes, the models were evaluated by 12 participants, rating accuracy and quality with Mean Opinion Score. Human-crafted palettes scored highest (4.514 and 4.008), followed by the VAE (3.168 and 3.514) and the VAE GAN (1.82 and 3.34). Limitations include the dependence on training data quality and the occasional mismatch with input images, indicating a need for further refinement.

Sharma, Tandukar, & Bista (2023) proposed a method for creating color palettes using combination k-means and n-grams, commonly used in natural language processing. This method computes all possible permutations to predict the next color in a palette. The principle is similar to Markov chain modeling, computing the highest probability to generate the next color in a palette. The method uses bi-grams and tri-grams, computed separately and then combined to produce a 5-color palette. The evaluation in (Sharma, Tandukar, & Bista, 2023)'s research was conducted with human feedback on the color palettes generated by the proposed method, comparing it to human-created palettes. The n-grams method achieved a 92.8% likelihood compared to color palettes from Colourlovers.com (which the researchers consider human-created palettes). The dataset for this research was scraped from Colourlovers.com.

Sharma, Tandukar, & Bista's method (2023) demonstrates potential for improvement, as it relies solely on K-Means clustering and n-grams for palette generation. By incorporating generative models like CGAN, there is an opportunity to achieve more diverse and flexible outputs. This research advances their approach by replacing K-Means with Bisecting K-Means to avoid empty clusters and achieve more balanced clustering. Additionally, n-grams are replaced with CGAN to better capture complex color relationships and generate more varied palettes, conditioned on a given input color. This research uses the public WikiArt dataset, which includes 81,444 paintings representing various artistic styles. Unlike previous studies that used qualitative evaluations, this research will employ the CIEDE2000 metric for quantitative evaluation of color palette generation, as it offers better color difference measurement compared to the CIELab metric (Gomez-Polo, et al., 2016).

METHOD

Previous research using combinations of K-Means and n-grams (Sharma, Tandukar, & Bista, 2023) to generate color palettes has not been sufficiently effective and lacks standardized quantitative evaluation. Harmonious color interactions can form patterns that are difficult to learn, necessitating a more complex approach than K-Means and n-grams alone. This research aims to find the best combination of methods from Clustering (K-Means and Bisecting K-Means as Feature Extractors) and CGAN (Conditional Generative Adversarial Network) to produce harmonious color palettes. This research will compare the evaluation outputs against a baseline model (Sharma, Tandukar, & Bista,

2023) to identify the best method combination for generating harmonious color palettes. Evaluation will be conducted using the CIEDE2000 metric to measure the performance of the proposed method against the baseline model.

Dataset

The dataset used in this research is the WikiArt dataset, which was obtained from previous research (Tan, Chan, Aguirre, & Tanaka, 2018). This dataset is a modified version of the original WikiArt dataset (WikiArt.org - Visual Art Encyclopedia, 2010), with additional data incorporated. Portion of data for each training, validation and test stage can be seen in Table 1. Meanwhile, the overview of the sample images in the dataset can be seen in Figure 3.

Table 1: The division of the dataset for the training, validation and testing processes.

Dataset	Total
Training (80%)	65.156
Validation (10%)	8.144
Test (10%)	8.144
Total	81.444

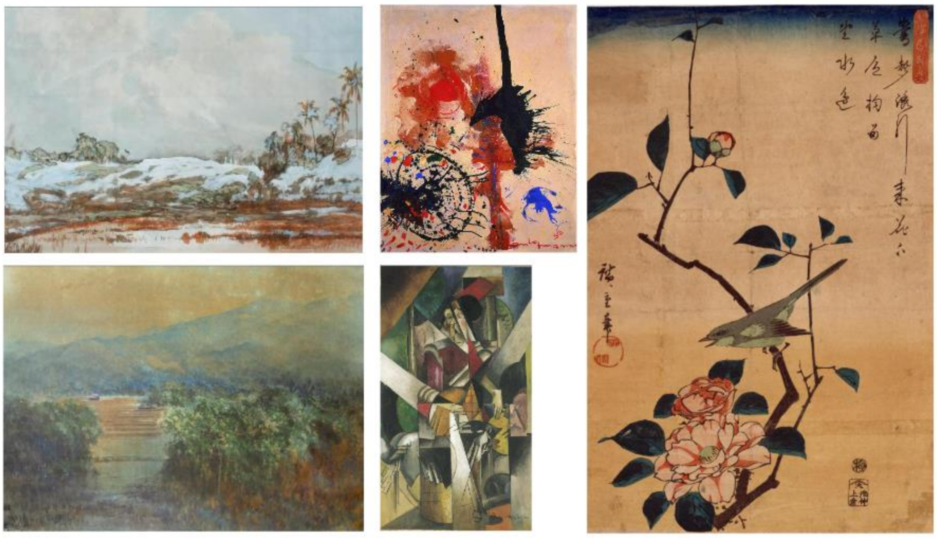

Figure 3: Sample paintings in the dataset

To address the constraints of computational resources and time, each image in the dataset was downsampled by selecting a random subset of 3,000 pixels prior to initiating the clustering process. This approach ensures that the analysis remains feasible without compromising the integrity of the overall clustering results. By sampling a consistent number of pixels (n = 3,000) from each image, we were able to maintain a manageable dataset size that allowed for efficient processing while still capturing sufficient visual information for meaningful clustering.

Overview of the proposed method

This research proposes a novel approach to generating harmonious color palettes. It combines two unsupervised learning methods, where one algorithm acts as a feature extractor and the other as a generator to produce the output. The learning process begins with the feature extraction process referred to as Auxillary Task in Figure 9 which is a clustering algorithm generating outputs with cluster amount of 5 or 10, k = 5 and k = 10, representing the number of colors in the palette that will be generated in this research. These outputs are then used to train the generator in the Conditional Generative Adversarial Network (CGAN) model in the Palette Generation Process with CGAN in the next stage as shown in Figure 4. The overall workflow visualization of the proposed method can be seen in Figure 4.

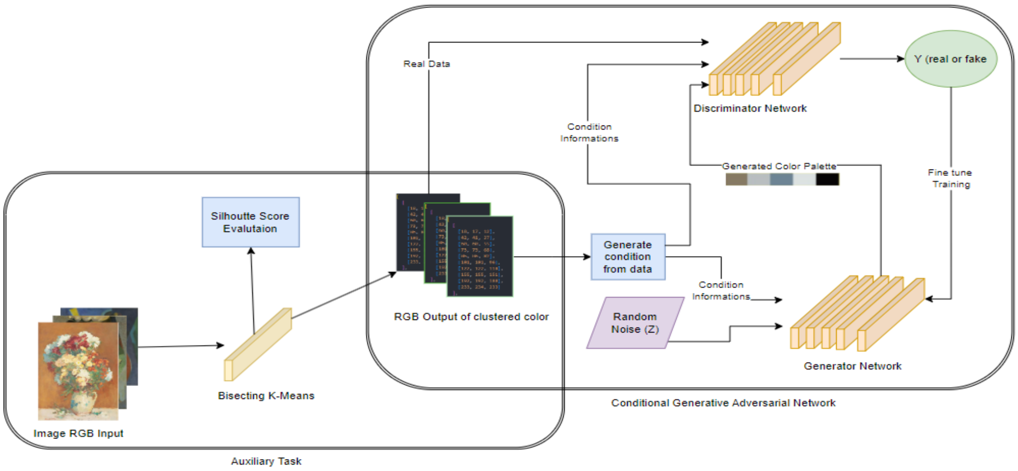

Figure 4: Workflow visualization of the proposed method

The CGAN model used in this research refers to the model developed by (Mirza & Osinderi, 2014) originally trained on the MNIST dataset. However, for the architecture of the Generator and Discriminator, the CGAN model architecture from Mindspore Huawei (Huawei Technologies Co., Ltd, 2022) is utilized. This model was selected based on the evaluation results using the Kernel Density estimate with Parzen window (Gramacki, 2017) (Tan, Yin, & Zhao, 2018), where it achieved a score of 283.03 ± 2.15. In comparison, the original model from (Mirza & Osinderi, 2014), only reached a score of 132 ± 1.8, with higher values indicating better performance, the result can be seen on Table 2. Evaluation will be done quantitatively to measure the harmony of generated color palettes and comparing the result with the baseline model with the CIEDE2000 metric.

Table 2: Evaluation Index for various CGAN

Model	**Kernel Density estimate with Parzen window for the MNIST dataset**
GAN (original)	225 ± 2
CGAN (original)	132 ± 1.8
CGAN (MindSpore)	283.03 ± 2.15

Feature extraction using clustering methods

Bisecting K-Means is chosen as the Feature Extractor for its ability to achieve global optimum, unlike K-Means, which can result in suboptimal local optima due to initial cluster center sensitivity (Chen, et al., 2020) as shown in Figure 8. The extraction process converts 81,444 paintings into palettes with k = 5 and k = 10 in RGB color space. For comparison, K-Means is also implemented as a Feature Extractor.

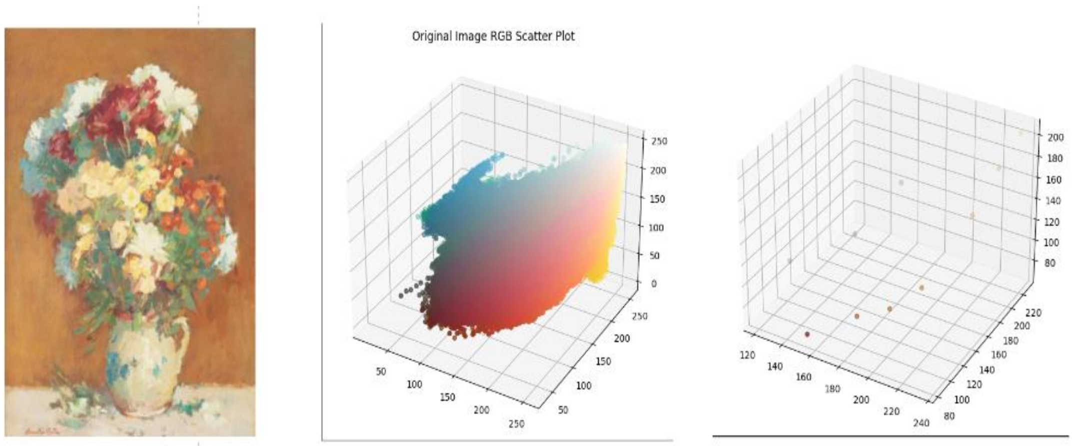

Figure 5: 3D Visualization of clustering result in RGB color space

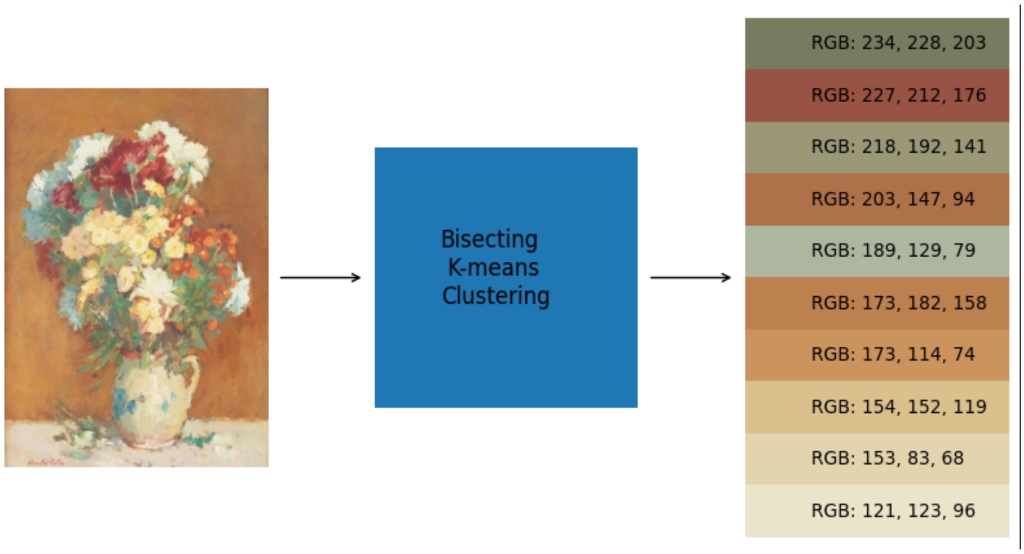

Figure 6: Sample output of the clustering or feature extraction

Figures 6 shows a sample of the clustering result using Bisecting K-Means, which serves as input for the Generative Adversarial Network model. Each pixel in an image is represented numerically in the RGB color space. Figure 5 illustrates the RGB 3D visualization of pixels, cluster centers, and the final cluster values used as input for the second model (CGAN). The extraction process iterates over 81,444 images, creating a new JSON dataset representing harmonious color combinations. After constructing a new dataset using the Feature Extractor, labels are assigned by calculating the average RGB values of each extracted palette as shown visually in Figure 7.

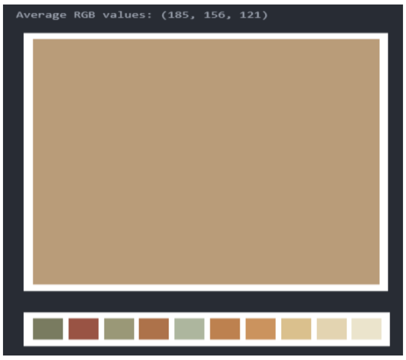

Figure 7: Calculated Label from the extracted ground truth palette.

$$\bar{R} = \frac{\sum_{i=1}^{n} R_i}{n} \quad (1)$$

$$\bar{G} = \frac{\sum_{i=1}^{n} G_i}{n} \quad (2)$$

$$\bar{B} = \frac{\sum_{i=1}^{n} B_i}{n} \quad (3)$$

Where $\bar{R}$,$\bar{G}$,$\bar{B}$ is the average of red, green and blue amount from the n colors extracted from the image, n can be 5 or 10.

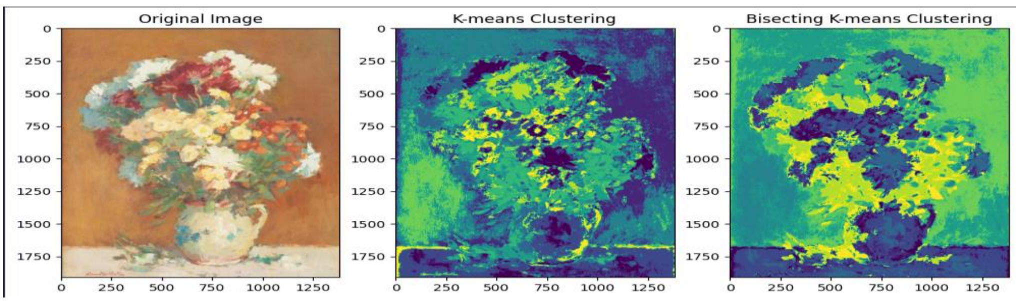

Figure 8: Visual Proof of K-Means experiences local optimum

The K-Means method is one of the most used algorithms in cluster analysis for grouping data into distinct clusters based on feature or attribute similarity. K-Means is an iterative algorithm that aims to partition data into K distinct clusters. Each cluster is represented by a central point called a centroid. The algorithm seeks to minimize the sum of squared distances between each data point and the centroid of its cluster (Mahmud, Mamun, Hossain, & Uddin, 2018).

Bisecting K-Means is a variant of the K-Means algorithm that adopts a hierarchical approach to clustering data. Bisecting K-Means employs a top-down strategy to recursively split the data into smaller groups. The algorithm starts with a single large cluster that encompasses the entire dataset. It begins by dividing this large cluster into two smaller clusters and then repeats this process on one of the resulting clusters until the desired number of clusters is achieved (Abirani & Mayilvahanan, 2016).

Color palette generation using conditional generative adversarial network

A Conditional Generative Adversarial Network (CGAN) is a type of GAN model that allows for better control over the generated data by incorporating additional information into the generator. CGAN is an evolution of the standard GAN (Goodfellow, et al., 2014) model, enabling the creation of more directed synthetic data by including additional information, known as conditions, into the generator. These conditional inputs can be class labels, attributes, or other features that influence the formation of the generated data. With this conditional information, CGAN can produce data that is more varied and relevant to specific application needs (Mirza & Osinderi, 2014).

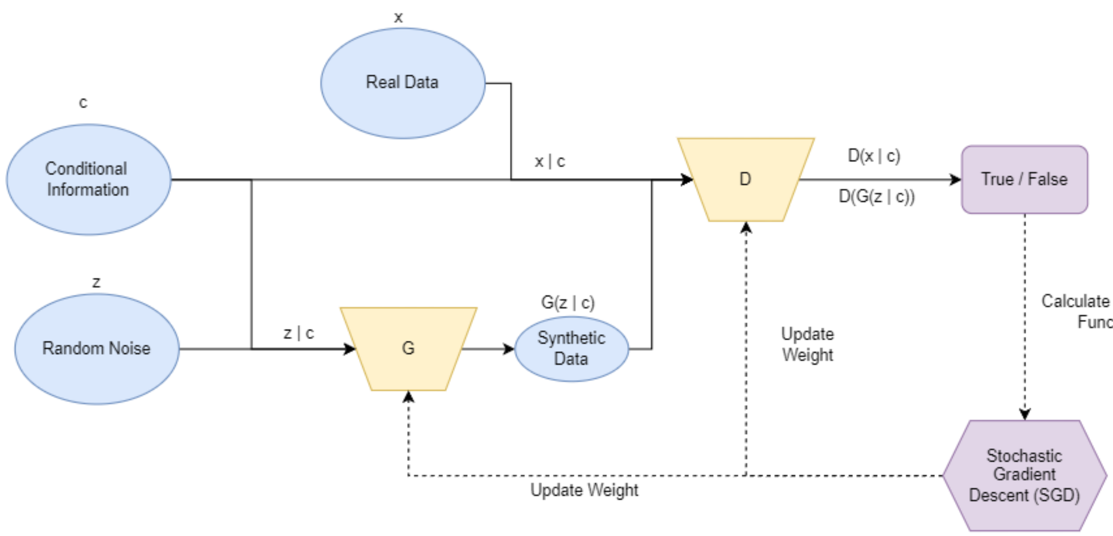

Figure 9: CGAN architecture

Figure 9 illustrates the general architecture of a Conditional Generative Adversarial Network (CGAN). The key difference from the standard GAN architecture lies in the input received by the Generator (G) and Discriminator (D) models. The input to the Generator, consisting of the latent space Z, is combined with additional information that conditions the synthetic data output. This additional information, denoted as c, is also provided to the Discriminator to ensure it correctly differentiates between conditioned synthetic data and real data. After calculating the loss values for both models, the weights of the Generator and Discriminator are updated. The model reaches convergence when the Generator achieves minimum loss, and the Discriminator achieves maximum loss.

$$\min_G \max_D V(D, G) \ldots\ldots\ldots\ldots\ldots\ldots (4)$$

$$V(D, G) = E_{x\sim p_{dt}(x)}[log\, D\,(x, c)] + E_{z\sim p_z(z)}\left[log\left(1 - D\big(G(z, c)\big)\right)\right] \ldots\ldots (5)$$

Where $E_{x\sim p_{dt}(x)}[log\, D\,(x, c)]$ represents the expectation of the logarithm of the probability that the discriminator correctly predicts the real data x given the condition c. This means that the discriminator is expected to assign a high probability (close to 1) to real data that matches the given condition. $E_{z\sim p_z(z)}\left[log\left(1 - D\big(G(z, c)\big)\right)\right]$ represents the expectation of the logarithm of the probability that the discriminator predicts the synthetic data generated by the generator G with condition c as fake. This means that the discriminator is expected to assign a low probability (close to 0) to synthetic data that does not match the given condition.

Table 3: Generator architecture for 5 Colors

Layer	Input Shape	Output Shape
Input	100	100
Dense	100	256
LeakyReLU	256	256
BatchNormalization	256	256
Dense	256	512
LeakyReLU	512	512
BatchNormalization	512	512
Dense	512	1024
LeakyReLU	1024	1024
BatchNormalization	1024	1024
Dense	1024	15
Reshape	15	(5,3)

Table 4: Generator architecture for 10 colors

Layer	Input Shape	Output Shape
Input	100	100
Dense	100	256
LeakyReLU	256	256
BatchNormalization	256	256
Dense	256	512
LeakyReLU	512	512
BatchNormalization	512	512
Dense	512	1024
LeakyReLU	1024	1024
BatchNormalization	1024	1024
Dense	1024	30
Reshape	30	(10,3)

Table 5: Discriminator architecture for 5 colors

Layer	Input Shape	Output Shape
Input	(5,3)	(5,3)
Flatten	(5,3)	15
Dense	15	512
LeakyReLU	512	512
Dense	512	512
LeakyReLU	512	512
Dropout	512	512
Dense	512	512
LeakyReLU	512	512
Dropout	512	15
Dense	512	1

Table 6: Discriminator architecture for 10 colors

Layer	Input Shape	Output Shape
Input	(10,3)	(10,3)
Flatten	(10,3)	30
Dense	30	512
LeakyReLU	512	512
Dense	512	512

LeakyReLU	512	512
Dropout	512	512
Dense	512	512
LeakyReLU	512	512
Dropout	512	15
Dense	512	1

In Table 3, Table 4, Table 5 and Table 6 are the architecture for the Generator and Discriminator models for 5 colors and 10 colors palette, the modification is on the input layer and output layer to match the use case of this research which are 5 colors with 3 channels and 10 colors with 3 channels of RGB.

Evaluation process

The hyperparameter tuning will be implemented on 2 sets of hyperparameters. This research will utilize two sets of hyperparameters derived from the studies by Mirza and Osinderi (Mirza & Osinderi, 2014), and the MindSpore model implemented by Huawei (Huawei Technologies Co., Ltd, 2022).

The hyperparameters from Mirza and Osinderi (Mirza & Osinderi, 2014) are as follows:

- Optimizer: SGD (Stochastic Gradient Descent)
- Learning Rate: 0.1
- Momentum: 0.5
- Decay Factor: 1.00004

The hyperparameters for the MindSpore model by Huawei (Huawei Technologies Co., Ltd, 2022) are as follows:

- Optimizer: Adam
- Learning Rate: 0.0002
- Momentum: 0.5

The best model from the hyperparameter tuning process on the validation set will be selected to represent the best model and will be used to compare against the baseline model on test set. Finally, the best combination and lowest of CIEDE2000 score will be proposed as Color Palette Generator model, since the lower the score indicates better harmonious pattern in the palette.

The proposed method will be evaluated through three phases: training, validation, and testing. Dataset will be divided into three parts, as detailed in Table 1. These data subsets will be utilized in two experimental stages: the Clustering stage and the Color Palette Generation stage. In the clustering stage, the training, validation, and testing datasets will undergo clustering using two methods: K-Means and Bisecting K-Means. Four new datasets will be generated as follows:

- K-Means; k = 5
- K-Means; k = 10
- Bisecting K-Means; k = 5
- Bisecting K-Means; k = 10

Clustering will be applied to the training, validation, and testing data for each method and k-value, resulting in four new JSON-formatted datasets, each containing 81,444 clustered samples. The clustering quality will be evaluated using the Silhouette Score (Shahapure & Nicholas, 2020) for each dataset, with the average Silhouette Score calculated to represent the clustering quality. Additionally, the average Silhouette Score across k-values (k = 5 and k = 10) will be computed. Silhouette Score will be computed using the formula below for each clustering result:

$$s(i) = \frac{b(i)-a(i)}{max\{a(i),b(i)\}} \text{................} (6)$$

$$S = \frac{1}{N}\sum_{i=1}^{N} s(i) \ldots\ldots\ldots\ldots\ldots\ldots\ (7)$$

The results of this stage will include the four JSON datasets segmented by training, validation, and testing, which will be used in the subsequent Conditional Generative Adversarial Network (CGAN) model training phase. Furthermore, the clustering quality will also be assessed using the average CIEDE2000 score to evaluate palette quality. These evaluations will be analyzed and used to compare it against the palette generated by the CGAN model with same label or condition.

In the Palette Generation Stage, Several CGAN models will be trained and validated with the hyperparameters mentioned above. The performance measurement for the hyperparameter tuning process will use the CIEDE2000 metric.

Table 7: Details of the CGAN model to be trained.

K (Colors)	Feature Extractor	Hyperparameters
5	K-Means	Optimizer = SGD Learning Rate = 0.1, Momentum = 0.5, Decay Factor = 1.00004
5	K-Means	Optimizer = Adam Learning Rate = 0.0002 Momentum = 0.5
5	Bisecting K-Means	Optimizer = SGD Learning Rate = 0.1, Momentum = 0.5, Decay Factor = 1.00004
5	Bisecting K-Means	Optimizer = Adam Learning Rate = 0.0002 Momentum = 0.5
10	K-Means	Optimizer = SGD Learning Rate = 0.1, Momentum = 0.5, Decay Factor = 1.00004
10	K-Means	Optimizer = Adam Learning Rate = 0.0002 Momentum = 0.5
10	Bisecting K-Means	Optimizer = SGD Learning Rate = 0.1, Momentum = 0.5, Decay Factor = 1.00004
10	Bisecting K-Means	Optimizer = Adam Learning Rate = 0.0002 Momentum = 0.5

Table 7 outlines the eight CGAN models to be trained, each with different hyperparameters, feature extractors, and color quantities for palette generation. The models are categorized based on the number of colors (K) and the clustering method (K-Means or Bisecting K-Means). Each model will use one of two optimizers (SGD or Adam) and various hyperparameter settings. Each model will be trained using clustered data from the respective feature extractors, aiming to generate palettes with the specified number of colors.

Performance metric using CIEDE2000

CIEDE2000 is a color difference metric developed by the International Commission on Illumination (CIE) and published in 2000. Its primary goal is to provide a more accurate measure of color differences based on human perception. Unlike simpler color distance metrics, such as the Euclidean distance in the CIELab color space, CIEDE2000 considers various factors that influence how humans perceive color differences (Gomez-Polo, et al., 2016). CIEDE2000 was first formulated in the research by Luo, Cui, & Rigg (2001).

CIEDE2000 uses a more complex calculation than CIELab, considering differences in Lightness (L*), Chroma (C*), and Hue (H*) between two colors, and modeling the non-linear effects of human color perception. This design allows CIEDE2000 to more accurately estimate color differences in line with human perception, making it more consistent with how humans see color differences compared to simple metrics like Euclidean distance. CIEDE2000 also accounts for metamerism, where two colors with different spectral properties appear the same under different lighting conditions, making it more adaptive to various lighting conditions. Although CIELab is still widely used in many applications, CIEDE2000 has become more popular in industries like printing, product design, and color quality testing due to its higher accuracy and consistency in measuring color differences (Gomez-Polo, et al., 2016).

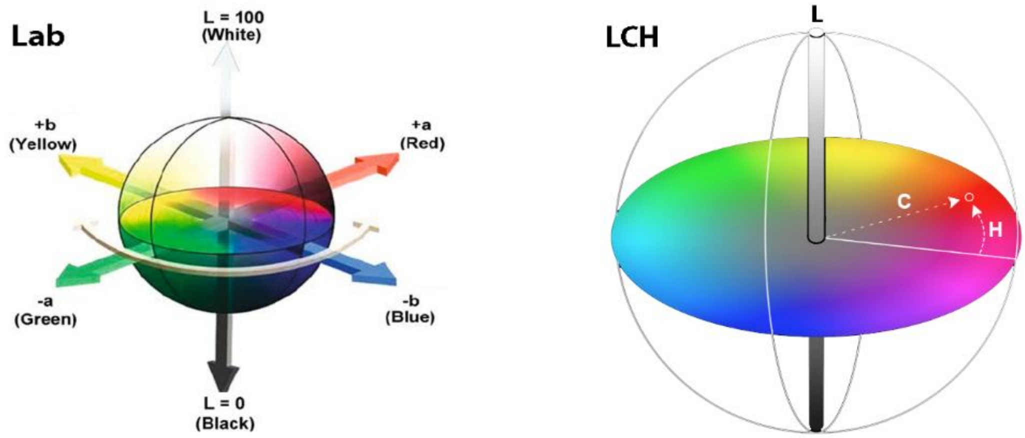

Figure 10: Illustration of the differences (L*), (a*), (b*), the color space used in CIELab Metric and (L*), (C*), (H*) color space which is used in CIEDE2000

Figure 10 illustrates the Lab and LCH color spaces, which are perceptually uniform models widely used in color science. The Lab color space defines colors in three dimensions: L for lightness, a for the green-red axis, and b for the blue-yellow axis, making it device-independent and suitable for accurate color comparison. The LCH space, derived from Lab, represents colors cylindrically, with L for lightness, C for chroma (saturation), and H for hue, providing a more intuitive approach for manipulating colors based on their perceptual attributes. This distinction is crucial when calculating color differences, as metrics like CIEDE2000—an advanced color-difference formula—operate in the LCH space to quantify perceptual differences between colors more accurately.

FINDINGS AND DISCUSSION

The research to produce harmonious color palettes follows the methods outlined in the early section. This process involves training, validation, and testing phases using a combination of K-Means, Bisecting K-Means, and CGAN models. Additionally, the results will be compared with a baseline model, which is a combination of K-Means and n-grams. The best CIEDE2000 score is achieved by the combination of Bisecting K-Means with CGAN with Adam Optimizer in 5 colors palette which is 23.5836, followed by the combination of K-Means and CGAN with Adam Optimizer with score of 23.7852, and the baseline model with score of 24.8320. For the 10 colors palette also by the same combination of Bisecting K-Means and CGAN with Adam Optimizer with 30.5865 CIEDE2000 score.

Findings

Figure 11 shows visual form of the clustering result from image in dataset, along with the evaluated Silhouette Score and CIEDE2000 Score, can be seen that higher Silhouette Score does not imply better visual inspection, as CIEDE2000 metric value represents the human visual judgement quantitatively.

The training and validation results quantitatively demonstrate that the proposed method performs slightly better than the previous methods. It is also observed that a higher number of clusters, which translate to more colors in the palette, positively influences the palette's harmony as measured by the CIEDE2000 metric. The result of validation can be seen in Table 10, on the other hand, Table 11 shows the chosen models with best result during the validation process with hyperparameters and evaluated with the CIEDE2000 metric. Table 12 shows the performance of the models in the test set. As seen in Table 12, the proposed method in this research performs slightly better than the baseline method, which combines K-Means and n-grams, both in validation and test set.

Table 8: Silhouette evaluation on clustering result

	K-Means 5	K-Means 10	Bisecting K-Means 5	Bisecting K-Means 10
Average Train	0.4945	0.4155	0.4734	0.3925
Average Validation	0.4965	0.4171	0.4758	0.3944
Average Test	0.4942	0.4146	0.4726	0.3919

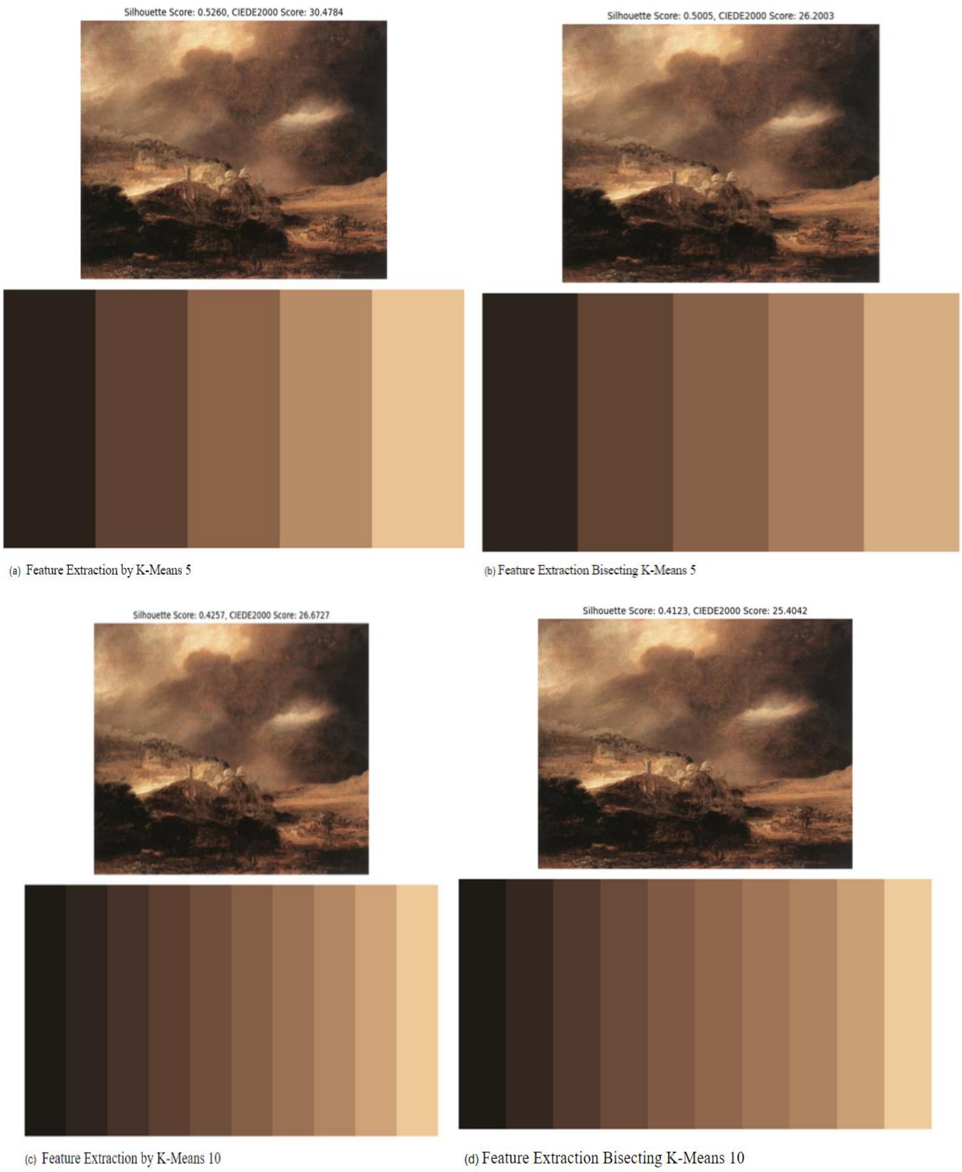

Figure 11: Visual example of feature extraction result with clustering

Table 9: CIEDE2000 evaluation on clustering results

	K-Means 5	K-Means 10	Bisecting K-Means 5	Bisecting K-Means 10
Average Train	29.6178	27.6319	28.6210	26.9720
Average Validation	29.6833	27.6752	28.6590	27.0172
Average Test	29.7126	27.7055	28.6790	27.0725

Table 8 shows that the Silhouette Score for K-Means is closer to 1 compared to Bisecting K-Means, indicating better clustering quality. However, when evaluated within the domain context using the CIEDE2000 metric as shown in Table 9, Bisecting K-Means outperforms K-Means. In CIEDE2000, a score closer to 0 indicates better color harmony within a palette. This contrast suggests that while traditional clustering metrics like the Silhouette Score may favor K-Means, domain-specific metrics such as CIEDE2000 provide a more nuanced evaluation of clustering effectiveness, particularly in tasks focused on color harmony. Hence, Bisecting K-Means, though seemingly less optimal in general clustering metrics, is more effective for color palette generation within the context of perceptual color differences.

Table 10: Result on validation and Hyperparameter tuning

Methods	**Optimizer**	**n-clusters**	**Average Quality (CIEDE2000)**
Bisecting K-Means + CGAN	Adam	5	23.6215
Bisecting K-Means + CGAN	SGD	5	33.8083
Bisecting K-Means + CGAN	Adam	10	30.5865
Bisecting K-Means + CGAN	SGD	10	31.7504
K-Means + CGAN	Adam	5	23.7610
K-Means + CGAN	SGD	5	25.5777
K-Means + CGAN	Adam	10	34.0472
K-Means + CGAN	SGD	10	33.3893

Figure 12 and Figure 13 visualize the training progress of 8 combinations of models shown in Table 7 in this research. It can be observed that higher Discriminator loss led to better generator in generating color palettes. This observation highlights the critical role of the Discriminator in guiding the Generator toward producing more refined and harmonious color palettes, emphasizing that the adversarial interplay between the two models is key to the overall success of the generative process.

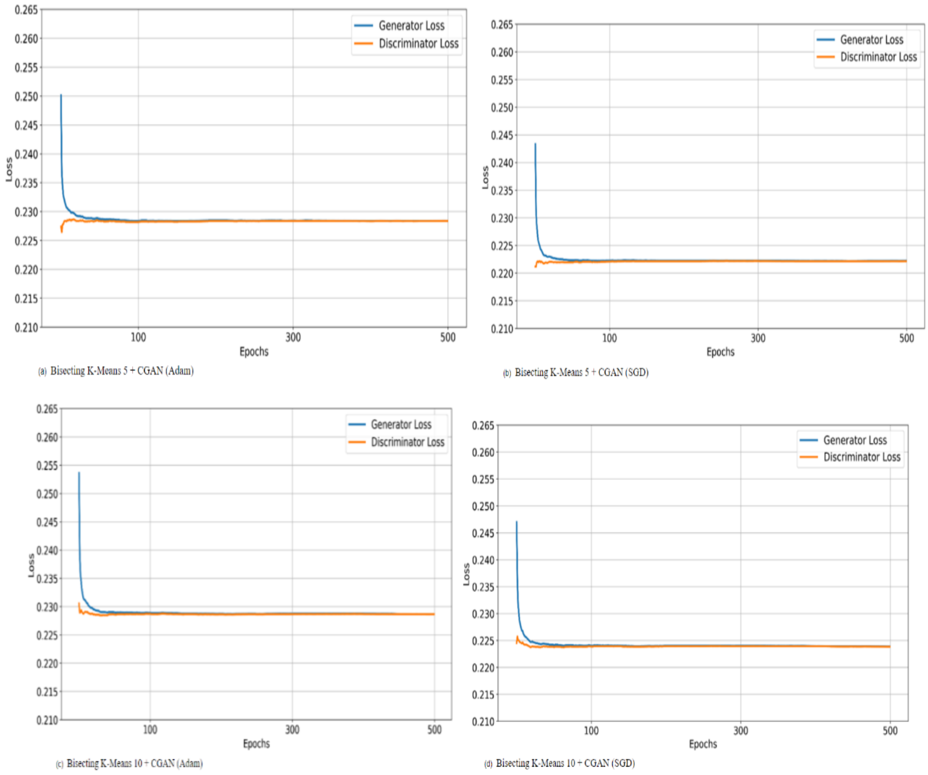

Figure 12: Training progress visualization on bisecting K-means + CGAN

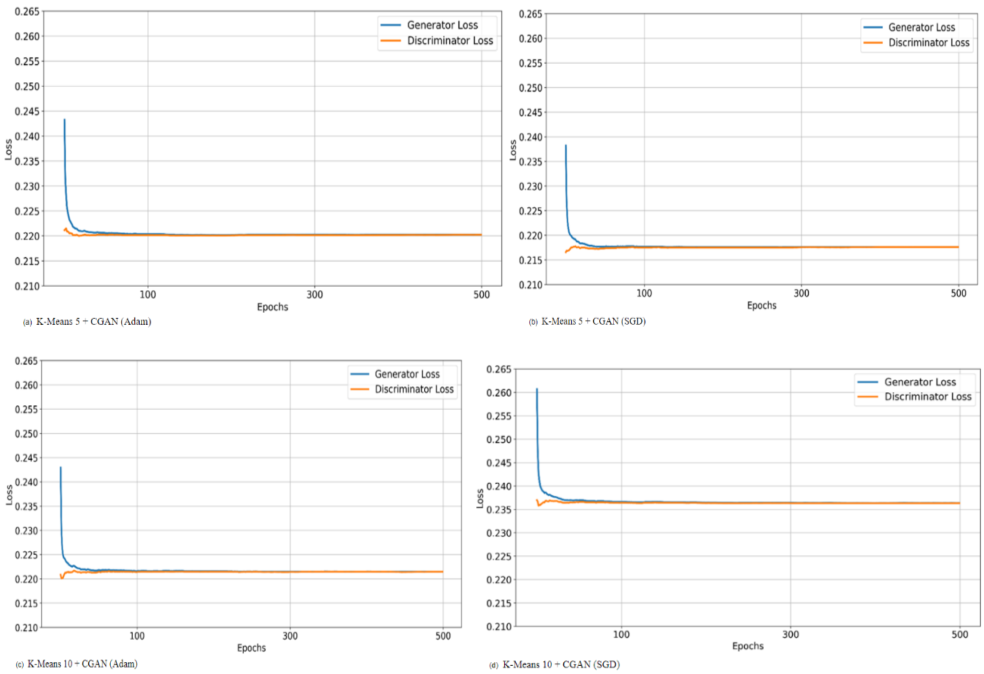

Figure 13: Training progress visualization on K-Means + CGAN

Table 11: Summary of the best models from validation and hyperparameter tuning

Methods	**n-clusters**	**Optimizer**	**Average Quality (CIEDE2000)**
Bisecting K-Means + CGAN	5	Adam	23.6592
Bisecting K-Means + CGAN	10	Adam	30.5865
K-Means + CGAN	5	Adam	23.7610
K-Means + CGAN	10	SGD	33.3893
K-Means + n-grams (Sharma, Tandukar, & Bista, 2023)	5	-	25.0198

Table 12: Summary of testing results from the best model from validation stage

Methods	**n-clusters**	**Optimizer**	**Average Quality (CIEDE2000)**
Bisecting K-Means + CGAN	**5**	**Adam**	**23.5836**
Bisecting K-Means + CGAN	**10**	**Adam**	**30.5628**
K-Means + CGAN	5	Adam	23.7852
K-Means + CGAN	10	SGD	33.3768
K-Means + n-grams (Sharma, Tandukar, & Bista, 2023)	5	-	24.8320

DISCUSSION

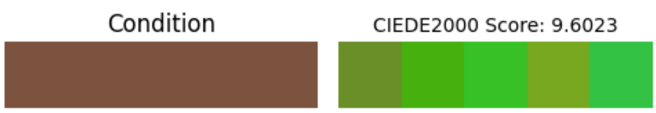

Figure 14: Sample of generated color Palette of 5 colors with low CIEDE2000 score

Figure 15: Sample of generated color palette of 5 colors with high CIEDE2000 score

Figure 16: Sample of generated color palette of 10 colors with low CIEDE2000 score

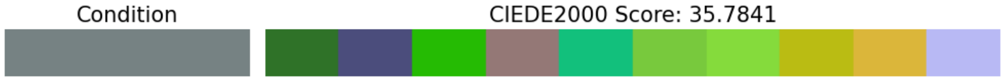

Figure 17: Sample of generated color palette of 10 colors with high CIEDE2000 score

The figures presented showcase the effectiveness of the proposed model in generating harmonious color palettes. The performance of the model is evaluated using the CIEDE2000 score, a metric known for its precision in measuring color differences by considering human visual perception factors such as Lightness (L), Chromaticity (C), and Hue (H).

Figure 14 displays a generated color palette of 5 colors with a low CIEDE2000 score of 9.6023. This low score indicates a high degree of color harmony within the palette. The colors are predominantly green, suggesting a natural and cohesive blend, which can be ideal for themes inspired by nature, such as forests or gardens. The palette is generated based on the input label or in the context of CGAN is called the condition. As we can see visually the label is brown-like color which may represent woods, the model generates a palette that matches the condition which might be interpreted contextually with nature theme.

Figure 15 shows a palette with a high CIEDE2000 score of 37.7835. The higher score signifies greater color variation within the palette. Despite this, the palette includes a range of colors from blue to pink, demonstrating the model's capability to produce diverse color combinations. This can be particularly useful for artistic or design purposes where a broad spectrum of colors is desired.

Figure 16 and Figure 17 present palettes with 10 colors, both having relatively high CIEDE2000 scores of 16.8802 and 35.7841, respectively. These palettes exhibit more complex color schemes, with a mixture of hues and tones. The model's ability to generate such palettes shows its flexibility and understanding of color theory. For example, the inclusion of green and blue hues in a single palette mimics natural elements like water and foliage, enhancing the visual appeal and thematic relevance.

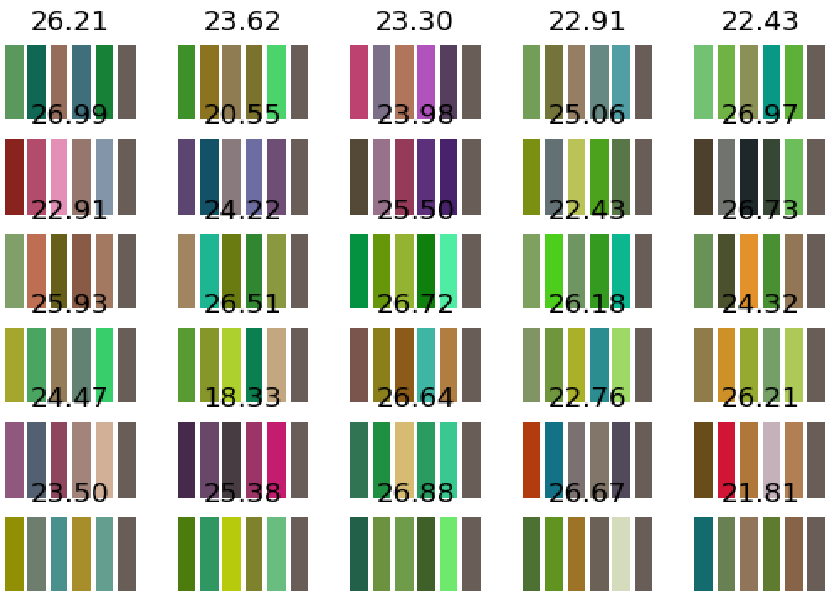

Figure 18: 5 colors palettes generated from the same conditional color

Figure 18 illustrates the ability of the model to generate various 5-color palettes from the same conditional color, which is the last color in each palette. The numbers above each palette represent the CIEDE2000 scores, reflecting the degree of color harmony within each palette. The figure demonstrates that the model can produce a diverse array of color palettes, all anchored by the same conditional color. This highlights the model's capability to maintain thematic coherence while introducing variations in the accompanying colors. Despite the same conditional color, each palette exhibits unique combinations, suggesting the model's proficiency in exploring the color space effectively. The CIEDE2000 scores range from approximately 18.33 to 26.99. Lower scores indicate better harmony, meaning the colors in the palette are perceived to be more similar and visually cohesive. Palettes with lower CIEDE2000 scores, such as those around 18.33, show more tightly grouped color schemes, which are likely to be perceived as more harmonious by human observers. The generated palettes display a wide range of hues, from greens and yellows to reds and purples, showing the model's versatility. For instance, one can observe how the model generates palettes with predominantly green tones (e.g., the palette with a score of 25.50) as well as more varied palettes incorporating contrasting colors like purple and green (e.g., the palette with a score of 24.22). The conditional color, often a prominent hue within the palette, acts as an anchor, ensuring that the generated palettes are contextually relevant and suitable for the intended theme. This relevance is critical for applications in design where the conditional color represents a brand color, theme, or specific context. The model's ability to generate harmonious palettes is evident from the CIEDE2000 scores and visual inspection. Palettes with lower scores appear more harmonious, while those with higher scores exhibit more diversity, suitable for different design needs. In practical scenarios, the ability to generate varied yet harmonious palettes from a single conditional color is invaluable. For instance, in branding, maintaining a consistent primary color while varying secondary colors can keep designs fresh yet consistent with the brand's identity. The analysis of Figure 18 shows that the proposed model is adept at generating diverse color palettes from the same conditional color while maintaining harmony, as indicated by the CIEDE2000 scores. This capability is crucial for design and artistic applications where consistency and variety are both desired. The use of CIEDE2000 as a metric ensures that the generated palettes are evaluated in a manner that aligns with human visual perception, making the results both scientifically robust and practically relevant.

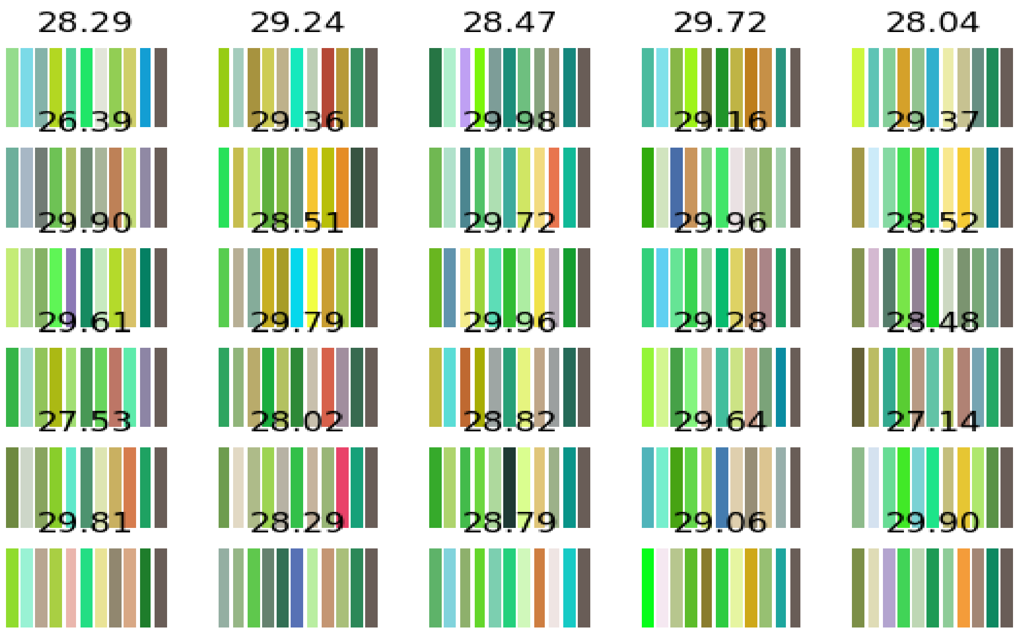

Figure 19: 10 colors palettes generated from the same conditional color

Figure 19 showcases the model's ability to generate various 10-color palettes from the same conditional color, which is the last color in each palette. The model demonstrates its capability to produce a diverse set of 10-color palettes, all anchored by the same conditional color. This emphasizes the model's strength in maintaining thematic consistency while introducing different variations in the complementary colors. Each palette, although sharing the same conditional color, displays unique combinations, highlighting the model's effectiveness in exploring the extensive color space. The model generates palettes that include both analogous colors (e.g., various shades of green and blue) and contrasting colors (e.g., greens paired with reds or purples), which can cater to different design needs. The ability to generate varied yet harmonious 10-color palettes from a single conditional color is invaluable for practical applications in design and art. For instance, in UI/UX design, maintaining a consistent primary color while varying secondary colors can enhance the user experience without losing the brand's identity. The analysis of Figure 19 demonstrates that the proposed model excels in generating diverse 10-color palettes from the same conditional color while maintaining overall harmony, as reflected by the CIEDE2000 scores. This capability is essential for design and artistic applications where both consistency and variety are required.

The analysis of these figures underlines the model's proficiency in creating harmonious and contextually appropriate color palettes. The use of CIEDE2000 as an evaluation metric is crucial, as it aligns with human visual assessment by factoring in Lightness, Chromaticity, and Hue. This similarity to human perception ensures that the generated palettes are not only technically sound but also aesthetically pleasing.

The proposed method demonstrates an improvement over the baseline model combining K-Means and n-grams, as evidenced by the quantitative results. A higher number of clusters, corresponding to more colors in a palette, positively impacts the perceived harmony, as measured by CIEDE2000. This indicates that the proposed model can effectively balance color diversity and harmony, making it suitable for practical applications in design and art.

CONCLUSION

This research involves two main stages: Feature Extraction and Palette Generation. In the Feature Extraction process, clustering methods were used to extract color palettes as ground truth based on the RGB color space. During the Feature Labeling stage, the conditional color is calculated by averaging the colors within the palette to be used as a condition for training the CGAN model. In the palette generation stage, this research utilizes a generative model in the form of CGAN. There is potential for further development by replacing CGAN with other GAN models that are better suited to the research use case. However, this depends on the feature extraction methods employed in this research.

This research successfully combines K-Means with CGAN, meeting the first objective and outperforming the baseline model of K-Means and n-grams. Quantitative evaluation using the CIEDE2000 metric showed a score of 23.7852 for the K-Means and CGAN combination, compared to 24.8320 for the K-Means and n-grams model, reflecting an improvement of 1.0468 in the 5-color palette. The baseline method (Sharma, Tandukar, & Bista, 2023) did not generate a 10-color palette. The Bisecting K-Means and CGAN combination surpassed the K-Means and CGAN model, with a CIEDE2000 score of 23.5836 compared to 23.7852 in the 5-color palette, indicating a 0.2016 improvement. In the 10-color palette, Bisecting K-Means with CGAN achieved a score of 30.5628, compared to 33.3768 for K-Means with CGAN, marking an improvement of 2.814. This advancement fulfills the second objective, with Bisecting K-Means and CGAN also outperforming the baseline model by 1.2484 in the 5-color palette.

Acknowledgement

The author expresses deep gratitude to WikiArt (WikiArt.org - Visual Art Encyclopedia, 2010)and Tan, Chan, Aguirre, & Tanaka (2018) for providing the dataset, which was vital to this research. Special thanks to Mrs. Giovanny Halimko for her support and encouragement. The author also appreciates the contributions of everyone who directly or indirectly supported this work. Finally, sincere thanks to Bina Nusantara University for the resources and facilities that made this research possible.

REFERENCES

Abirani, K., & Mayilvahanan, D. (2016). Performance Analysis of K-Means and Bisecting. International Journal of Emerging Technologies in Engineering Research (IJETER), 4(8).

Burchett, K. E. (2001). Color harmony. Color Research And Application, 27(1), 28 - 31. doi:10.1002/col.10004

Chen, Y. X., Liu, W., Zhao, H. J., Cao, S. L., Fu, S. S., & Jiang, D. D. (2020). Bisecting k-means based fingerprint indoor localization. Wireless Networks, 27(5), 3497 - 3506. doi:10.1007/s11276-019-02222-0

Gomez-Polo, C., Munoz, M. P., Luengo, M. C., Vicente, P., Galindo, P., & Casado, A. M. (2016). Comparison of the CIELab and CIEDE2000 color difference formulas. The Journal of Prosthetic Dentistry, 115(1), 65 - 70. doi:10.1016/j.prosdent.2015.07.001

Goodfellow, I. J., Pouget-Abadie, J., Mehdi, M., Xu, B., Warde-Farley, D., Ozair, S., . . . Bengio, Y. (2014). Generative Adversarial Nets. Advances in neural information processing systems, 27.

Gramacki, A. (2017). Nonparametric Density Estimation and Its Computational Aspects. Springer, Cham. doi:10.1007/978-3-319-71688-6_2

Huang, S. C. (2021). An Efficient Palette Generation Method for Color Image Quantization. Applied Sciences, 1043. doi:10.3390/app11031043

Huawei Technologies Co., Ltd. (2022). Huawei MindSpore AI Development Framework. In: Artifical Intelligence Technology. Singapore: Springer. doi:10.1007/978-981-19-2879-6_5

Itten, J. (1974). The Art of Color: The Subjective Experience and Objective Rationale of Color (Revised ed.). John Wiley & Sons. doi:0471289280

Jeong, T., Yang, M., & Shin, H. J. (2019). Succinct Palette and Color Model Generation and Manipulation Using Hierarchical Representation. Computer Graphics Forum, 38(7), 1 - 10. doi:10.1111/cgf.13811

Kim, S. H., & Kang, S. J. (2021). GAN-based Color Palette Extraction System by Chroma Fine-tuning with Reinforcement Learning. Journal of Semiconductor Engineering, 125 - 129. doi:10.22895/jse.2020.0101

Lertrusdachakul, T., Ruxpaitoon, K., & Thiptarajan, K. (2019). Color Palette Extraction by Using Modified K-means Clustering. 2019 7th International Electrical Engineering Congress (iEECON). Hua Hin, Thailand: IEEE. doi:10.1109/iEECON45304.2019.8938867

Liu, S. q., Tao, M. T., Huang, Y. F., Wang, C. B., & Li, C. H. (2022). Image-Driven Harmonious Color Palette Generation for Diverse Information Visualization. IEEE Transactions on Visualization and Computer Graphics, 1 - 16. doi:10.1109/TVCG.2022.3226218

Lu, K., Feng, M., Chen, X., Sedlmair, M., Deussen, O., Lischinski, D., . . . Wang, Y. (2020). Palettailor: Discriminable Colorization for Categorical Data. IEEE Transactions on Visualization and Computer Graphics, 27(2), 475 - 484. doi:10.1109/TVCG.2020.3030406

Luo, M. R., Cui, G., & Rigg, B. (2001). The development of the CIE 2000 colour-difference formula: CIEDE2000. Color Research & Application, 26(5), 340 - 350. doi:10.1002/col.1049

Mahmud, M. R., Mamun, M. A., Hossain, M. A., & Uddin, M. P. (2018). Comparative Analysis of K-Means and Bisecting K-Means Algorithms for Brain Tumor Detection. 2018 International Conference on Computer, Communication, Chemical, Material and Electronic Engineering (IC4ME2) (p. 5). Rajshahi, Bangladesh: IEEE. doi:10.1109/IC4ME2.2018.8465607

Mirza, M., & Osinderi, S. (2014, 11 6). Conditional Generative Adversarial Nets. arxiv.org. doi:10.48550/arXiv.1411.1784

Moussa, A., & Watanabe, H. (2022). Generation and Extraction of Color Palettes with Adversarial Variational Auto Encoders. Proceedings of 6th International Congress on Information and Communication Technology, ICICT 2021 (pp. 889 - 897). Virtual Online: Springer Science and Business Media Deutschland GmbH. doi:10.1007/978-981-16-2380-6_78

Pan, Q., & Westland, S. (2018). Comparative Evaluation of Color Differences between Color Palettes. Color and Imaging Conference, 1, pp. 110 - 115. doi:10.2352/ISSN.2169-2629.2018.26.110

Peng, Y. F., & Chou, T. R. (2019). Automatic Color Palette Design Using Color Image and Sentiment Analysis. 2019 IEEE 4th International Conference on Cloud Computing and Big Data Analysis (ICCCBDA). Chengdu, China: IEEE. doi:10.1109/ICCCBDA.2019.8725717

Phan, H. Q., Fu, H., & Chan, A. B. (2018). Color Orchestra: Ordering Color Palettes for Interpolation and Prediction. IEEE Transactions on Visualization and Computer Graphics, 24, 1942-1955. doi:10.1109/TVCG.2017.2697948

Reyes, T., & Lara-Alvarez, C. (2019). A Geometric Approach to Harmonic Color Palette Design. Color Research & Application, 44(1), 106 - 114. doi:10.1002/col.22292

Shahapure, K. R., & Nicholas, C. (2020). Cluster Quality Analysis Using Silhouette Score. 2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA). Sydney, NSW, Australia: IEEE. doi:10.1109/DSAA49011.2020.00096

Sharma, S., Tandukar, J., & Bista, R. (2023). Generating Harmonious Color through the Combination of n-Grams and K-means. Journal of Computing Theories and Applications, 140 - 150. doi:10.33633/jcta.v1i2.9470

T.Wei, Y., Fang, X., Deng, Z., Li, F., Ling, Y., & Wang, X. (2019). A Color-Pair Based Approach for Accurate Color Harmony. Pacific Graphics, 38(7). doi:10.1111/cgf.13854

Tan, J., Yin, S., & Zhao, S. (2018). Generative Model with Kernel Density Estimation. 2018 International Conference on Network Infrastructure and Digital Content (IC-NIDC) (pp. 304-308). Guiyang, China: IEEE. doi:10.1109/ICNIDC.2018.8525628

Tan, W. R., Chan, C. S., Aguirre, H. E., & Tanaka, K. (2018). Improved ArtGAN for Conditional Synthesis of Natural Image and Artwork. IEEE Transactions on Image Processing, 28, 394 - 409. doi:10.1109/TIP.2018.2866698

Westland, S., & Lai, P. (2020). Machine learning for colour Palette extraction from fashion runway images. INTERNATIONAL JOURNAL OF FASHION DESIGN, TECHNOLOGY AND EDUCATION, 13(3), 334 - 340. doi:10.1080/17543266.2020.1799080

WikiArt.org - Visual Art Encyclopedia. (2010). Retrieved from WikiArt: https://www.wikiart.org/

Yang, J., Chen, Y., Xiao, K., & Westland, S. (2020). Predicting Visual Similarity Between Colour Palettes. Color Research & Application, 45(6), 401 - 408. doi:10.1002/col.22492

Yuan, L. P., Zhou, Z. Q., Zhao, J., Guo, Y. Q., Du, F., & Qu, H. M. (2022). InfoColorizer: Interactive Recommendation of Color Palettes for Infographics. IEEE Transactions on Visualization and Computer Graphics, 28(12), 4252 – 4266. doi:10.1109/TVCG.2021.3085327