PAPER

Associating Colors with Mental States for Computer-Aided Drawing Therapy

Satoshi MAEDA[†], ***Student Member and*** **Tadahiko KIMOTO**[††a)], ***Nonmember***

SUMMARY The aim of a computer-aided drawing therapy system in this work is to associate drawings which a client makes with the client's mental state in quantitative terms. A case study is conducted on experimental data which contain both pastel drawings and mental state scores obtained from the same client in a psychotherapy program. To perform such association through colors, we translate a drawing to a color feature by measuring its representative colors as primary color rates. A primary color rate of a color is defined from a psychological primary color in a way such that it shows a rate of emotional properties of the psychological primary color which is supposed to affect the color. To obtain several informative colors as representative ones of a drawing, we define two kinds of color: approximate colors extracted by color reduction, and area-averaged colors calculated from the approximate colors. A color analysis method for extracting representative colors from each drawing in a drawing sequence under the same conditions is presented. To estimate how closely a color feature is associated with a concurrent mental state, we propose a method of utilizing machine-learning classification. A practical way of building a classification model through training and validation on a very small dataset is presented. The classification accuracy reached by the model is considered as the degree of association of the color feature with the mental state scores given in the dataset. Experiments were carried out on given clinical data. Several kinds of color feature were compared in terms of the association with the same mental state. As a result, we found out a good color feature with the highest degree of association. Also, primary color rates proved more effective in representing colors in psychological terms than RGB components. The experimentals provide evidence that colors can be associated quantitatively with states of human mind.

key words: *drawing therapy, psychological primary colors, color analysis, machine-learning classification*

1. Introduction

Drawing therapy is a kind of clinical practice of art therapy [1], where psychological treatment is provided by a psychotherapist to a client via drawings. Here, let us refer to a psychotherapist as a therapist for short. The client is encouraged to make any drawings there. Such creative activities are expected to have therapeutic effects on the client. Thus, drawing therapy is recently considered a simple and effective way of psychological treatment, so that it is often used in psychotherapy practices.

The therapist looks into drawings to deduce the client's mental state by interpreting them in psychological terms. The procedure of interpreting drawings is usually composed of three steps:

1. Looking for those outward features in the whole drawing, such as coloring, the size of drawing contents, and the location of drawing contents, which have psychologically important aspects.
2. Looking for those objects in the drawing contents, such as humans, the moon, and flames, which symbolize psychologically important matters.
3. Reading the client's mental state in the whole drawing [2].

Here, drawing contents mean figures which were intentionally made on the drawing sheet, and the whole drawing means the whole sheet including blanks. In step 3, the therapist makes an interpretation of the drawing by using the results of steps 1 and 2, and often refers to the client's actual situation so as to describe the client's mental state more appropriately.

As for coloring in step 1, there are various colors which are related to mental states in color psychology. For instance, green is related to calmness, blue to sadness, black to a feeling of death, bright color to positive feeling, dark color to loneliness or anxiety, and colorful arrangements to the existence of various feelings [3]. The therapist looks for these coloring features in the drawing contents. The features observed in this step represent properties common to human mind, but they provide psychologically primitive terms about an individual mind in step 3. Hence, features in step 1 are very important in interpreting drawings.

Different therapists are likely to make their respective interpretations of the same drawing. One reason for this is that results of the above steps 1 and 2 may depend on expert knowledge of therapists. Moreover, a more probable reason is that in step 3, therapists can see a drawing from their respective viewpoints based on clinical experiences. Accordingly, drawing therapy is subject to therapist's subjective judgments.

Features in the above step 1 are easy to quantify by computer image analysis [4]. Such objective features are expected to reduce the subjectivity of interpretation in step 3. Also, numerical data can be used in further computer processing. The effectiveness of such quantitative analysis has been also reported in the field of painting arts [5]. However, no ongoing studies on computer-aided psychotherapy systems, to our knowledge, except our work have ever been reported.

The aim of our computer-aided drawing therapy sys-

Manuscript received February 15, 2023.
Manuscript revised July 29, 2023.
Manuscript publicized September 14, 2023.

[†]The author is with the Graduate School of Science and Engineering, Toyo University, Kawagoe-shi, 350–8585 Japan.
[††]The author is with the Faculty of Science and Engineering, Toyo University, Kawagoe-shi, 350–8585 Japan.
a) E-mail: kimoto@toyo.jp

DOI: 10.1587/transinf.2023EDP7022

tem is to analyze drawings by computer and make the results useful for psychotherapeutic diagnosis. Focusing on colors based on color psychology, in our earlier work [6], we proposed a practical method of analyzing pastel drawings which a client made in a clinical setting so that a therapist can interpret the drawings in terms of simple colors. In addition, we defined a measure of colors against psychological primary colors [7], which we refer to as primary color rates here. A psychological primary color is associated with both positive and negative properties of human emotions [8]. Hence, it is difficult to describe a mental state simply in terms of psychological primary colors. In contrast, a primary color rate measures a color as a rate of properties of the corresponding psychological primary color which are supposed to affect the color. Then, we proposed making a color feature of a drawing from primary color rates of the representative colors extracted from the drawing [7]. However, the effectiveness of the color feature in representing a concurrent mental state remains unproven. Also, the advantages of using primary color rates instead of original color values should be verified.

Following the previous studies, the present paper deals with the association between colors and mental states in detail. To find out which color feature can be most closely associated with a mental state, it is necessary to evaluate a color feature from the viewpoint of the effectiveness in representing a mental state. As an evaluation method, we build a classification model by training it on a given dataset to learn the relationship between a color feature and mental state classes, and consider the classification accuracy reached by the model as the degree of association of the color feature with the mental state. Hence, our computer-aided drawing therapy system requires another psychotherapy which should be conducted in parallel with the drawing therapy on the same client, and also where the client's mental state is evaluated as numerical scores. Using experimental data which were obtained in the clinical setting with the requirements satisfied, we carry out the same case study in this work as before. For a given dataset of samples generated from the experimental data, we improve a classification model gradually from scratch by iteratively training new models on the dataset with extending the configurations. Thus, we find out the best classification accuracy for the dataset.

The experimental data we actually obtained for the case study were only 47 pairs of a pastel drawing and a mental state score; this data shortage may be a common situation in psychotherapy practice. Training a machine-learning model on such a small dataset usually causes overfitting. To make the resulting classification-accuracy reliable as a measure of association, we will specify the making of a color-classification model on a very small dataset.

The rest of this paper is organized as follows. In Sect. 2, the scheme of computer-aided drawing therapy in this study is explained. The clinical setting of the case study and clinical data for experiments are also described. Section 3 explains a method of color analysis for extracting representative colors from each drawing under the same conditions. Also, the definition of primary color rates is described. Section 4 deals with color classification for evaluating the association with mental states. A practical way of building a neural network through training and validation on small datasets is explained in detail. In Sect. 5, the results of experiments using the clinical data are shown. The color features are compared in terms of the degree of association with the mental state. Lastly, Sect. 6 concludes the paper.

2. A Scheme of Computer-Aided Drawing Therapy

2.1 An Approach to Computer-Aided Drawing Therapy

Figure 1 illustrates an approach to computer-aided drawing therapy by extending a conventional (non-computer-aided) process of drawing therapy that a psychotherapist usually gives a client. In the drawing therapy, a client makes drawings at will. Then, a psychotherapist observes the drawings trying to read the client's emotional cues according to the therapist's knowledge and experience. Although the observations may be rather descriptive than quantitative, and also rather subjective than objective, they can help the therapist to diagnose the client's mental state.

Our system supposes another psychological therapy as Fig. 1 indicates. This therapy must meet two requirements: One is that the client receives the therapy treatment in parallel with the drawing therapy. The other one is that the therapist evaluates the client's mental state as numerical values, which are depicted as mental state scores in Fig. 1. Thus, both drawings and mental state scores are to be obtained from the same client in the same period of therapy.

Now, a computer-aided path can be added parallel with the data path from the drawing therapy as shown in Fig. 1. Drawings are analyzed to extract image features such as colors by using image processing techniques by computer. These features are based only on image signals and hence, expressed in objective quantity. Thus, the features can be processed together with the numerical mental state scores

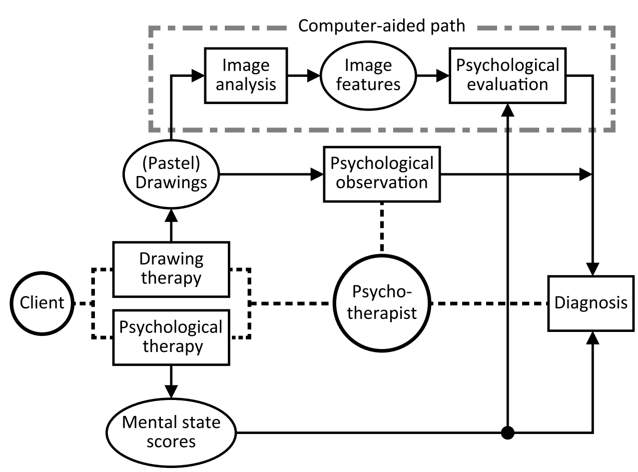


Fig. 1 A system concept of computer-aided drawing therapy.

for psychological evaluation. The results are to be provided for diagnosis.

2.2 A Case Study

To develop the above system, we conducted a case study using experimental data which had been obtained at a clinical site. In the clinical setting, cognitive behavioral therapy [9]–[11] was carried out to obtain mental state scores. A client had received the two kinds of therapy concurrently for almost four years, while seeing a therapist at a mental health clinic once a month.

2.2.1 Conducting Two Therapies on One Client

At home, the client made drawings of any contents he/she liked. All the drawings were made on a sketchbook with color pastels in this case. On the other hand, the client did homework of mental tasks assigned in the cognitive behavioral therapy. Also, the client wrote down his/her feelings in a diary at any time, for instance, when he/she felt stress or emotional ups and downs.

At the clinic, starting with drawing therapy, the client showed the therapist a drawing made within the last one month. Then, moving on to cognitive behavioral therapy, the client showed the therapist the diary written in over the last one month. The therapist evaluated the client's mental state from the diary, and estimated a degree of how much the previous psychological problem was mitigated in the period by an oral psychological evaluation using a 100-point scale: a score of 100 meant that the problem was completely got rid of, and a score of 0 meant that no change was found. Translated into the range [0, 1], the resultant value was used as a mental state score at the time of therapy. Next, the therapist talked with the client about his/her psychological problems to assign him/her mental tasks and homework for the next month.

2.2.2 Client's Data

A drawing is considered an expression of the client's unconscious emotions at around the time when it was made. Although an above mental state score is estimated as a relative change in the client's feelings in one month, we assume that it also indicates a degree of unconscious composure of the client to accept a change of mind. From this point of view, we use the score as a quality which represents the client's mental state varying through the therapy period. On this assumption, we consider it reasonable to connect a drawing with a mental state score of a monthly pair on the basis that both data can express the client's unconscious mental state in the month. Thus, we collected 47 monthly pairs over the therapy period.

On the other hand, a practical way of drawing therapy specifies a kind of data for the succeeding image analysis. In this case study, accordingly, the analysis must involve the processing of digital images with characteristics of pastel drawings.

3. Extracting and Measuring Representative Colors

3.1 Source Images

3.1.1 Preprocessing

Drawings in this case study were made with pastels, which are solid materials of powdered colors. Drawing with a pastel directly on a sheet makes lumps of the material scattered there. Hence, a pastel drawing often looks like a halftone picture of colors distributed on a white background coarsely by density modulation. Also, the drawings were made by hand, and they could hardly include very fine patterns. Accordingly, high resolution is unnecessary to their digitized images. From these viewpoints, the source images were reduced in size so that the amount of data processing could decrease in the following image analysis. The spatial reduction must involve low-pass filtering.

3.1.2 Segmentation

The color analysis is supposed to be applied only to the contents drawn on a drawing sheet. We consider that the contents are composed of not only colored areas, which have a color other than a paper color, but also uncolored areas surrounded by colored areas. Also, we refer to the whole of the content pixels in a source image as the drawing region, and the whole of the other pixels as the background region. An image segmentation algorithm we have previously proposed divides the source image of a pastel drawing into its drawing region and background region. The algorithm uses morphological methods to deal with geometric features peculiar to pastel materials (see [6] for details).

For each drawing, a result of the algorithm is compared with the source image with the eye to judge whether the extracted picture region seems like what the client intended to make. The main purpose of the segmentation is to determine those parts which apparently seem not included in the picture contents. Hence, we include those parts for which it is difficult to judge whether they should belong to the picture or the background in the former region. On the other hand, those colored parts which apparently seem unintentional, such as sorts of stains attached on the sheet, are likely to be extracted in the picture region by the algorithm, but should be included in the background. The algorithm alters its result with some parameters such as a sheet color. Adjusting the parameters enables us to look for a desirable picture region. If the further change is required, we can modify the segmentation result manually.

3.2 Determining Representative Colors

To describe colors used in a source image, we define two kinds of representative color.

3.2.1 Approximate Colors

Dividing a set of colors used on a drawing region into a small number of clusters of similar colors, we consider those colors which represent the respective clusters approximately as representative ones of the region. To carry out such clustering and approximation of colors, we use k-means, which is one of commonly used clustering algorithms and performs vector quantization [12]. k-means divides a dataset into a specified number of clusters, which is usually represented by the value of k, according to the similarity between data. Thus, a value of k means the number of approximate colors we want. As described below, we figured out a way of applying k-means to all the drawings in the same way to find out the respective values of appropriate k.

The k-means algorithm alternates between two steps: assigning each data point to the closest cluster centroid, and then updating each cluster centroid from the data points that are assigned to it. The algorithm is finished when the assignment of data points to clusters no longer changes. The resulting clusters are subject to the initial cluster-centroids to start with.

To represent a drawing region in N colors, the k-means algorithm with $k = N$ deals with the set of pixels in a 3-d color space and divides it into k clusters. Each cluster has two properties: its color and pixels. For a cluster C, its pixels, denoted by C_{PELS}, represent the set of pixels belonging to C, and its color, denoted by C_{COLOR}, is the color of the centroid of C_{PELS}:

$$\begin{aligned} C_{COLOR} &= centroidColor(C) \\ &= \sum_{p \in C_{PELS}} pelColor(p) \Big/ |C_{PELS}| \end{aligned} \tag{1}$$

where the function $pelColor(p)$ gives the color of a pixel p, which is a 3-d vector in the color space, and $|C_{PELS}|$ denotes the area of C_{PELS}, that is, the number of pixels. C_{COLOR} is representative of all the colors within C_{PELS}. To evaluate how closely C_{COLOR} approximates the whole color of C_{PELS}, we use a color error of a cluster C, denoted by C_{ERROR}, defined by the mean of the color variance within the cluster:

$$\begin{aligned} C_{ERROR} &= meanVariance(C) \\ &= \sum_{p \in C_{PELS}} \|pelColor(p) - C_{COLOR}\|^2 \Big/ |C_{PELS}| \end{aligned} \tag{2}$$

where for two points c_1 and c_2 in the 3-d color space, $\|c_1 - c_2\|$ denotes the Euclidean distance.

k-means clustering is applied to a drawing region repeatedly with increasing k one by one until appropriate color reduction achieves. We refer to this way of implementing k-means as the increasing k-means in this paper. Let R_0 be the set of all pixels of the drawing region. The first cluster, $C^{(1)}$, is given by setting $C^{(1)}_{PELS}$ to R_0. Then, $C^{(1)}_{COLOR}$ is evaluated by Eq. (1). Let the number of existing clusters, denoted by n, be one at the start. The procedure we propose for the increasing k-means is as follows:

1. Evaluate $C^{(i)}_{ERROR}$ by Eq. (2) for $i = 1, 2, \ldots, n$. Also, evaluate a mean color error of the whole region by
$$E_0 = \sum_{i=1}^{n} C^{(i)}_{ERROR} \cdot \left|C^{(i)}_{PELS}\right| \Big/ |R_0|. \tag{3}$$
2. If E_0 is small enough for approximating the source drawing, then the procedure is completed.
3. If n reaches the upper limit of the number of approximate colors, then the procedure is terminated.
4. Find out the largest color error in the n clusters. Let j indicate the corresponding cluster, expressed as
$$j = \arg\max_{1 \le i \le n} \left\{C^{(i)}_{ERROR}\right\}. \tag{4}$$
5. For the above j, apply k-means with $k = 2$ to $C^{(j)}_{PELS}$ by using the k-means++ algorithm [13] to determine two initial points. Let the resulting clusters be denoted by $C^{(j)\prime}$ and $C^{(j)\prime\prime}$.
6. Apply k-means with $k = n + 1$ to R_0 by beginning with the $n+1$ colors $C^{(1)}_{COLOR}, \ldots, C^{(j-1)}_{COLOR}, C^{(j)\prime}_{COLOR}, C^{(j)\prime\prime}_{COLOR}, C^{(j+1)}_{COLOR}, \ldots, C^{(n)}_{COLOR}$ as the initial points.
7. Now let the result obtained in the preceding step replace the current set of clusters. Increment n by one; express the new set as $\{C^{(1)}, C^{(2)}, \ldots, C^{(n)}\}$; then, proceed to step 1.

In step 6, using the initial points including the existing cluster colors in the next k-means prevents the resultant cluster colors from changing drastically. The conditions to finish the procedure in steps 2 and 3 will be demonstrated later in the experiment.

After the increasing k-means finishes, the last set of clusters is determined for the drawing region, and the clusters are to give the respective approximate colors. Then, letting N_COLORS be the number of clusters, we sort the clusters in descending order of area, and rewrite the set as the ordered set $\left\{C^{(i)}\right\}_{i=1}^{N\_COLORS}$ such that $\left|C^{(1)}_{PELS}\right| \ge \left|C^{(2)}_{PELS}\right| \ge \ldots \ge \left|C^{(N\_COLORS)}_{PELS}\right|$. We refer to the approximate colors in this order: the first approximate color is $C^{(1)}_{COLOR}$, for instance.

3.2.2 Area-Averaged Colors

As another kind of representative color of a drawing region, we estimate an average color of a portion, for example, 50% of the whole drawing region. Pixels that compose such a drawing portion are determined as follows: First, from the above ordered set of N_COLORS clusters $\left\{C^{(i)}\right\}_{i=1}^{N\_COLORS}$, we obtain an accumulated cluster $U^{(i)}$ by defining its pixels $U^{(i)}_{PELS}$ and color $U^{(i)}_{COLOR}$ by

$$U^{(i)}_{PELS} = \bigcup_{j=1}^{i} C^{(j)}_{PELS} \quad \text{and} \tag{5}$$

$$U^{(i)}_{COLOR} = \sum_{j=1}^{i} \left|C^{(j)}_{PELS}\right| \cdot C^{(j)}_{COLOR} \Big/ \left|U^{(i)}_{PELS}\right|, \tag{6}$$

respectively, for $i = 1, 2, \ldots, N\_COLORS$, so that $\left|U^{(1)}_{PELS}\right| < \left|U^{(2)}_{PELS}\right| < \ldots < \left|U^{(N\_COLORS)}_{PELS}\right|$. Thus, $U^{(i)}_{COLOR}$ yields the area-averaged color of the first i clusters in the ordered set and accordingly, the average color over the area of the ratio to the whole region, $\left|U^{(i)}_{PELS}\right|/|R_0|$. Note that the area ratio of $U^{(i)}$ increases stepwise with i and depends on the result of clustering.

Next, we define the following function for the accumulated clusters, which we call an area-color function, so as to estimate an average color at any area ratio r to the whole region $(0 < r \leq 1)$ regardless of the discrete area ratios of the clusters: Letting A_i represent $\left|U^{(i)}_{PELS}\right|/|R_0|$ for convenience,

$$areaColor(r) = U^{(j)}_{COLOR} \tag{7}$$

where j satisfies $A_{j-1} < r \leq A_j$ with A_0 defined to be 0 $(1 \leq j \leq N\_COLORS)$. Also, the pixels that correspond to $areaColor(r)$ are supposed to be $U^{(j)}_{PELS}$ of the same j as that of Eq. (7). Thus, averaging partly within the drawing region is carried out on an area such that as many similar colors as possible are included. We call the color value of $areaColor(r)$ an r-area color, for example, a 50%-area color here. Note that area-averaged colors in themselves are invisible on a drawing. Also, only a 100%-area color is determined regardless of the clustering result for each drawing.

3.3 Psychological Color Measurement

To evaluate colors from a psychological viewpoint, we define a measurement of the relationship between a color c and a psychological primary color p [7]: Letting d_c and d_0 denote the Euclidean distances $\|p - c\|$ and $\|p - p_0\|$, respectively, in the 3-d color space,

$$pcr(c, p) = \begin{cases} 1 - \dfrac{d_c}{d_0} & \text{if } d_c < d_0, \\ 0 & \text{if } d_c \geq d_0 \end{cases} \tag{8}$$

where c, p and p_0 are all represented in the L*a*b* uniform color space, and p_0 is a constant color mentioned below. We call $pcr(c, p)$ a primary color rate, for instance, a primary *red* rate for p of the psychological primary red.

As seen from the relationship of locations among c, p, and p_0 in Eq. (8), if c is located inside a sphere which has its center at p and a radius of d_0 in the 3-d color space, $pcr(c, p)$ is larger as c is closer to p, and $pcr(c, p) = 1$ only if c is coincident with p. In contrast, if c looks different so much from p that c is outside the sphere, $pcr(c, p) = 0$. We refer to such a sphere as a primary color sphere, for instance, a primary *yellow* sphere. Thus, $0 \leq pcr(c, p) \leq 1$. It means that c has a rate of $pcr(c, p)$ of the psychological properties associated with p.

The point p_0 performs the origin of psychological primary colors. It should be of no emotion and accordingly be an achromatic color. Hence, we have located it at the 3-d coordinate origin. As for psychological primary *black*, we can place it at a distance from the origin, for instance, at the color "kuro" based on Japanese Industrial Standards (JIS).

4. Making Psychological Color-Evaluation Models

4.1 Classification of Colors into Mental Categories

4.1.1 Associating of Colors with Mental States

We suppose that drawings made in a drawing therapy have close relationship with mental state scores given in the concurrent psychological therapy, as mentioned in Sect. 2.2.2. Then, by examining the relationship between representative colors extracted from the drawings and the mental state scores, we can estimate how closely the colors are associated with the mental state. Our approach to measuring the relationship is to build a machine-learning model for classifying colors into mental categories by training on datasets of colors and mental state scores. The classification performance reached by the resulting model is considered as the degree of association of the colors with the mental state scores.

4.1.2 Datasets for Color Classification

(1) Mental state classes

Mental state scores given in psychotherapy, for instance, cognitive behavioral therapy originally have a value of a quantized real number. The quantization step is usually too fine, for example, one hundredth, to use the scores as labels in psychological terms. Hence, mental state scores should be classified, or coasely re-quantized into a few, $N\_CLASSES$, classes.

(2) Timeseries samples

Both drawings and mental state scores are obtained as timeseries data in a therapy period. Mental state scores obviously have properties of time-varying variables depending on a human mental state, which will be demonstrated later in the experiment. Accordingly, it is proper to deal with colors of the drawings as timeseries data to classify.

(3) Making datasets

Making timeseries samples is as follows: Each drawing is translated into its representative colors by the color analysis. The number of the colors may be different for each drawing. Then, we decide on one or more kinds of representative color to use through all the drawings, and measure the colors as primary color rates. As a result, each drawing is represented in a vector of a fixed number, $N\_FEATURES$, of primary color rates, which is referred to as a color feature vector, or simply a color vector.

Samples to classify are made from the timeseries of color vectors. A sample consists of consecutive vectors of a certain length, denoted by N_STEPS. Each sample is extracted from the timeseries at every timestep from the first, as illustrated in Fig. 2. Thus, given N color feature vectors, we obtain $(N - N\_STEPS + 1)$ samples.

Labeling samples for machine learning is as follows:

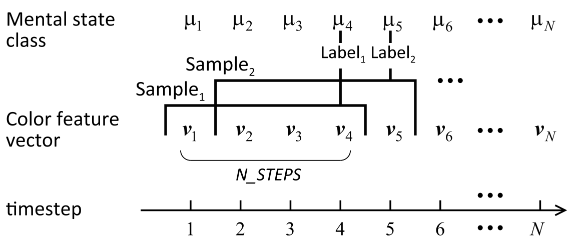


Fig. 2 Relationship between samples and labels on timesteps for example of $N\_STEPS = 4$.

A mental state class can be evaluated from the corresponding mental state score every timestep. From a timeseries of the mental state classes, a label associated with a timeseries sample can be determined. In this paper, we simply use the mental state class at the last timestep of each sample, as also illustrated in Fig. 2.

4.2 A Neural Network Model for Color Classification

To deal with timeseries classification, we use a recurrent neural network model. The input to the model is a timeseries sample of color vectors. The output from the model is a vector of $N\_CLASSES$ probability scores. Each score is the estimated probability that the input belongs to one of $N\_CLASSES$ mental state classes. Then, the class with the highest probability is considered the classification result of the model.

4.2.1 Requirements on Neural Networks

A classification network is to learn the relationship between inputs and corresponding outputs using datasets for training. However, datasets which can be obtained in a clinical site, especially from the same client during the period of his/her treatment program, are very limited in most cases. Training a network on such a small dataset is usually prone to cause overfitting the model only to the existing samples [14]. For coping with overfitting, we use the following strategies in the present study:

- Making a model of a small network such that it contains a small number of layers with a small number of units
- Using dropout regularization to mitigate overfitting
- Evaluating the performance of a model by K-fold cross-validation

We exclude other techniques commonly used in training a model such as batch normalization and gradient clipping to simplify the model implementation in this paper.

4.2.2 A Neural Network Architecture

We use the Long Short-Term Memory (LSTM) algorithm in a recurrent layer. An LSTM layer takes a timeseries of input vectors as its layer input, and generates an output vector at each timestep as its layer output. The network internally loops by using the preceding output as its recurrent input. Dropout regularization can be applied to the layer input and/or the recurrent input. Let L_{u_in,u_out} denote an LSTM layer with u_in input units and u_out output units.

In addition, we use a Dense layer which has fully connected parameters between its layer-input and layer-output vectors. Dropout is to be applied to the layer input. Let D_{u_in,u_out} denote a Dense layer with u_in input units and u_out output units.

We implement a color-classification model with a sequential network by stacking layers on top of each other. According to the dataset described in Sect. 4.1, the network architecture is as follows: The first, or the bottom layer is an LSTM layer expressed as $L^{(1)}_{u_0,u_1}$ where $u_0 = N\_FEATURES$. The next LSTM layer must be $L^{(2)}_{u_1,u_2}$. When a total of N_LM LSTM layers are stacked sequentially ($N\_LM \geq 1$), the last one is expressed as $L^{(N\_LM)}_{u_{N\_LM-1},u_{N\_LM}}$. Both input dropout and recurrent dropout are carried out on the LSTM layers except $L^{(1)}$ where only recurrent dropout is used. Then, the output of the last LSTM layer at the last timestep is connected to the input of the first Dense layer, accordingly expressed as $D^{(1)}_{v_0,v_1}$ where $v_0 = u_{N\_LM}$. Additional Dense layers can be connected sequentially such that $D^{(j+1)}_{v_j,v_{j+1}}$ is on $D^{(j)}_{v_{j-1},v_j}$ for $j = 1, 2, \cdots$. When a total of N_DS Dense layers are stacked ($N\_DS \geq 1$), the last one is expressed as $D^{(N\_DS)}_{v_{N\_DS-1},v_{N\_DS}}$ where $v_{N\_DS} = N\_CLASSES$. All the Dense layers are to use input dropout.

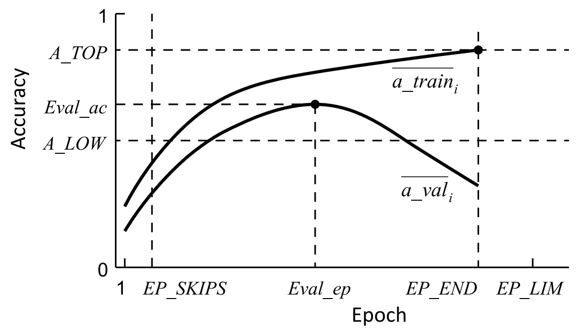


Fig. 3 Evaluating the result of training.

4.3 Training and Validating a Neural Network

Considering that only a few dozen samples are available for machine learning, training and validating a neural network is conducted as below (also see Fig. 3).

4.3.1 Training a Network

Suppose that a dataset for training, referred to as Z_train, and that for validation, referred to as Z_val, are given. Each dataset contains pairs of a timeseries sample and a corresponding target class. We use all the data of Z_train in the mini-batch stochastic gradient descent method in one training epoch. At the end of each epoch, accuracy of the network, which is a ratio of correct predictions, is measured both on the whole Z_train and on the whole Z_val. Let a_train_i and a_val_i denote the training accuracy and the validation accuracy, respectively, at the ith epoch ($i = 1, 2, \ldots$).

As the epochs proceed, the sequences $\{a\_train_i\}_i$ and $\{a\_val_i\}_i$ are being collected. Accuracy has a value in units of the inverse of the number of samples used. In the case there are a small number of samples, for instance, ten or so for validation like in the present study, the unit becomes so large, and accordingly, the proper evaluation of these quantities requires smoothing their variations. Thus, an exponential moving average (EMA) of the training accuracies, denoted by $\overline{a\_train_i}$, is also calculated at the end of ith epoch.

The training epoch iterates until the averaged training accuracy first reaches a predetermined threshold value, A_TOP. This stopping criterion is applied after a few first epochs, denoted by EP_SKIPS epochs, because an early state of the network is likely to show uncertainties owing to random initialization. The upper limit of times the epoch can iterate, denoted by EP_LIM, is also specified in case the stopping criterion is still hard to satisfy after plenty of epochs.

Supposing that training the network finishes in EP_END epochs, we have the sequences of accuracies, $\{a\_train_i\}_{i=1}^{EP\_END}$ and $\{a\_val_i\}_{i=1}^{EP\_END}$. We evaluate the result in the following way: First, an EMA of a_val_i, denoted by $\overline{a\_val_i}$, is calculated for $i = 1, 2, \ldots, EP\_END$. Next, given a predetermined threshold value of accuracy A_LOW ($< A\_TOP$), we find out the largest value of $\overline{a\_val_i}$ under the condition that $\overline{a\_train_i} \geq A\_LOW$ in the range of i from $EP\_SKIPS + 1$ to EP_END. Let j indicate the corresponding epoch index, that is,

$$j = \arg \max_{EP\_SKIPS < i \leq EP\_END} \left\{ \overline{a\_val_i} \,\middle|\, \overline{a\_train_i} \geq A\_LOW \right\}. \tag{9}$$

Then, the accuracy, $Eval\_ac$, defined with the above j by

$$Eval\_ac = \overline{a\_val_j} \tag{10}$$

is considered as a validation score for the network in this time of training. If there is no $\overline{a\_train_i}$ that satisfies the condition in Eq. (9), which can happen in the case the training is terminated by the upper limit number of epochs, the result is to be excluded from the evaluation of the network.

4.3.2 Evaluation Protocol

To evaluate a network trained on a small dataset, we use K-fold cross-validation [14]. K being considered as an integer value, first, the available data are split into K partitions. Then, using $K - 1$ partitions as Z_train and the remaining partition as Z_val, the model is trained and its validation score is evaluated. This process is carried out repeatedly with the partition for Z_val changed. Thus, K validation scores for the network are obtained from the dataset. The average of them is considered as the result of one time of K-fold cross-validation.

Furthermore, to make the evaluation more reliable, we iterate K-fold cross-validation on the same network and the same dataset multiple times. The whole dataset is randomly shuffled every time before it is split to partitions. By iterating N_KFOLD times, N_KFOLD results of K-fold cross-validation are obtained. Then, we consider the average of them as the final validation score for the network.

4.4 Building Neural Networks

4.4.1 Setting a Network Configuration

In a sequential neural network which is constructed in a manner described in Sect. 4.2.2, the dimensionality of each layer is given by its output units. Let us write the number of output units of each layer in sequential order as u_1, u_2, $\cdots$, u_n where n is the total number of layers, namely, $n = N\_LM + N\_DS$, u_1 is the number of output units of the first LSTM layer, and u_n is the number of output units of the last Dense layer, namely, $u_n = N\_CLASSES$. The network is to perform classification, and accordingly, we make a condition between the numbers of units as $u_{i-1} \geq u_i$ for $i = 2, 3, \ldots, n$. It is very important to determine the number of units of each layer while making the whole configuration small so that the model can show good performance. In the experiments described later, to adjust the layer configuration in a given sequence of layers, we will examine the model by changing u_i within the three sizes u_{i-1}, $3/4u_{i-1}$, and $1/2u_{i-1}$ for $i = 2, 3, \ldots, n-1$ under the condition that $u_{n-1} \geq u_n$. Then, we can figure out a network of the best performance in a way of grid search [12].

4.4.2 Criterion for Extending a Sequential Network

To figure out a classification model that has as good performance as possible while avoiding overfitting, we consider extending layers of a sequential network one by one, starting from a two-layer network. Suppose that a network, M, has the highest validation accuracy, α, for the present. Let us express the layer configuration of M as

$$M : L^{(1)} \cdots L^{(i)} L^{(i+1)} \cdots L^{(n)}$$

where n is the number of layers and $L^{(i)}$ denotes its ith layer for $1 \leq i \leq n$. Consider a new network, M', by adding a layer, L', to M such that

$$M' : L^{(1)} \cdots L^{(i)} L' L^{(i+1)} \cdots L^{(n)}.$$

In our sequential architecture, L' is actually either the last LSTM layer or the first Dense layer. Then, we choose a few unit-configurations of M that have relatively high validation scores. Using each of the configurations, the layers of M' are made up in a manner such that the units from $L^{(1)}$ to $L^{(i)}$ are the same as those of M, and the units from L' to $L^{(n)}$ are set variously. Thus, a variety of the extended model M's with various configurations are trained on a given dataset, and the respective validation scores are evaluated. Let α'

denote the best one of the resulting validation accuracies. If $\alpha' > \alpha$, the network extension is considered to be a success, and M' is adopted; otherwise, being considered ineffective in improving a classification performance, M' is rejected, and the network extension is completed.

4.4.3 Protocol for Achieving a Color-Evaluation Model

Given a dataset of timeseries of color-feature vectors and corresponding class labels, we build a color-classification model with as high a validation score as possible according to the following steps:

1. Make a baseline model which consists of one LSTM layer and one Dense layer without any dropout.
2. Add LSTM layers to the model obtained by step 1, and determine the sequence of LSTM layers.
3. Add Dense layers to the model obtained by step 2, and determine the whole sequential network.
4. For the model obtained by step 3, adjust the parameters of the optimizer used.

As a result, the best classification model for the given dataset is obtained. It is used as the model for color evaluation; we consider the validation accuracy which the model achieves as the measurement of the association between the color feature and the mental state scores.

5. Experiments and Discussion

5.1 Processing of Image Data

Original drawings were made on separate sheets of letter size. In the experiment, first, each drawing was digitized with a scanner at resolution of 300 dots per inch (dpi) and stored in 24-bit BMP-format without any distortion caused by image compression. In the preprocessing process, each digital image was low-pass filtered by a 12×12 uniform averaging operator, and then reduced to one sixth of its original size by spatially subsampling. Thus, we got source images of spatial resolution corresponding to 50 dpi.

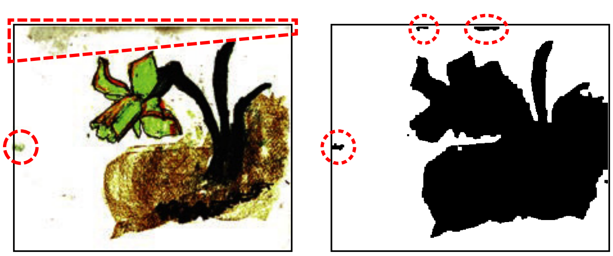
Figure 4 demonstrates a case where a segmentation result requires manual modification: Fig. 4 (a) shows a source image with contrast enhanced for clear visibility. There are some stains at the upper and left sides of the image, as enclosed with red long-broken lines. A segmentation result for the image is shown in Fig. 4 (b) where a picture region is depicted in black, and a background region in white. We observe wrong parts of the picture region caused by the stains, enclosed with red short-broken lines, which are to be moved into the background region by manual operation, for instance, by changing their color from black to white on Fig. 4 (b) with an image editor.

The color space of source images was supposed to be sRGB. Then, in the evaluation of primary color rates, color values of the source images were converted into CIE 1976 ($L*, a*, b*$) (CIELAB) color space via the XYZ color space by using the coordinate relationship [15] with the inverse Gamma correction to convert nonlinear sRGB values to linear ones [16] and Bradford chromatic adaptation transform to convert white points from D65 to D50 [17].

5.2 Color Analysis

Figure 5 shows an example of the process of extracting representative colors from a pastel drawing by the color analysis. The reduced source image of Fig. 5 (a) is 549 by 445 pixels of 24-bit RGB colors. First, the whole image was segmented into a drawing region and a background region which Fig. 5 (b) depicts in black and white, respectively.

Next, the increasing k-means algorithm was applied to the drawing region. Figure 5 (c) shows the results of color



(a) Source image (enhanced) (b) Segmentation result

Fig. 4 An example of segmentation result requiring manual modification.

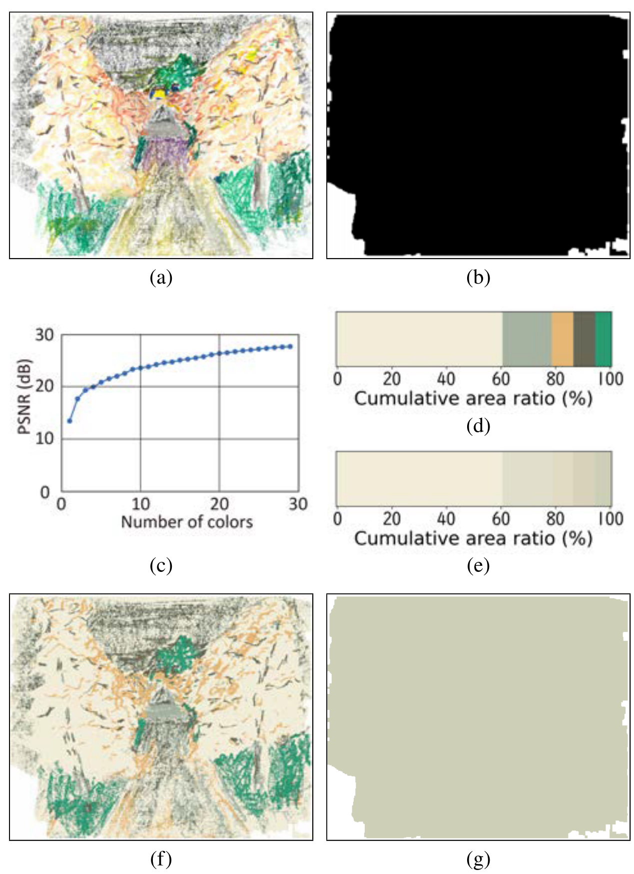


Fig. 5 An example of the process of color analysis of a pastel drawing.

clustering by plotting the approximation qualities in terms of peak signal-to-noise ratios (PSNRs) against the number of colors. It demonstrates that the method of initializing cluster-centroids described in Sect. 3.2.1 results in a sequence of cluster sets such that the approximation quality increases smoothly with the number of clusters. From the results for various drawings including this one, taking into account that our purpose was to reduce the number of approximate colors while achieving good approximation quality, we determined a criterion for deciding on the number of colors, N_COLORS, as follows:

$$N\_COLORS = \min_{N_{inf} \leq n \leq N_{sup}} \{n \mid psnr(n) > Q_{inf}\} \tag{11}$$

where $psnr(n)$ represents a PSNR value (dB) for the number of colors n. If there is no n satisfying the condition in Eq. (11), N_COLORS is set to be N_{sup}. Here we set $Q_{inf} = 20$, $N_{inf} = 5$, and $N_{sup} = 10$. In the case of Fig. 5 (c), we determined the number of approximate colors to five. The five colors are arranged in descending order of cluster area, as illustrated in the area chart of Fig. 5 (d). In addition, the approximate image with these five colors is depicted in Fig. 5 (f).

From the above ordered five approximate colors, area-averaged colors were evaluated using Eq. (6). The area chart of Fig. 5 (e) shows the five evaluated colors in order, and it also illustrates the definition of the area-color function $areaColor(r)$ of Eq. (7) for this drawing. Also, Fig. 5 (g) depicts the drawing region by painting it in the 100%-area color so that we can see a kind of subliminal color which the image presents as a whole impression. In Fig. 5 (e), no prominent color is found. If a large variation occurs in the $areaColor(r)$ with changing r, it suggests that each approximate color or a partially area-averaged color can be more significant in psychological diagnosis than a 100%-area color.

Thus, we determined a set of ordered approximate colors and an area-color function for each of the 47 experimental drawings. Then, we obtained some kinds of sequence of 47 representative colors, for instance, 100%-area colors.

5.3 Psychological Color Measurements

Each of the representative colors obtained from a drawing was measured as primary color rates as follows: We used four psychological primary colors for the primary color rate function: red denoted by PP_R, green PP_G, blue PP_B, and yellow PP_Y, which are considered most effective in representing human emotions in color psychology [18]. Numerical values of these colors remain undefined in the psychology, so that we used the following values: (1, 0, 0) as PP_R, (0, 1, 0) as PP_G, (0, 0, 1) as PP_B, and (1, 1, 0) as PP_Y in (R, G, B) coordinates. Figure 6 depicts four primary color spheres corresponding to the respective center colors in the L*a*b* space. The coordinates ($L*$, $a*$, $b*$) of the centers of the sphere are (54.3, 80.8, 69.9), (87.8, −79.3, 81.0), (29.6, 68.3, −112.0), and (97.6, −15.7, 93.4), and their radii are 119.9, 143.4, 134.5, and 136.0, respectively. The primary color rates with these four psychological primary colors are denoted by pcr_r, pcr_g, pcr_b, and pcr_y, respectively, for simplicity here.

Figure 7 shows examples of the measurements of primary color rates for the sequence of 47 drawings. Figure 7 (a) shows four primary color rates of the 100%-area color of each drawing. The measurements are considered to be a representation of each color in terms of psychological primary colors. As another example, Fig. 7 (b) shows the measurements for the first approximate colors. The difference between these two figures seems to imply a difference between the two colors in psychological terms. In particular, the difference between the sequences of primary red rates is considerably large.

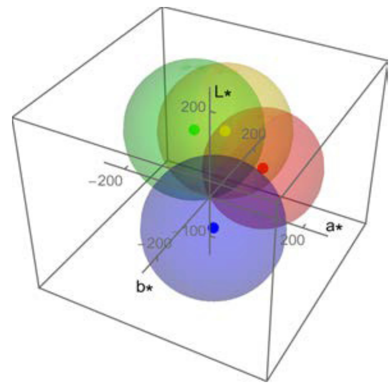
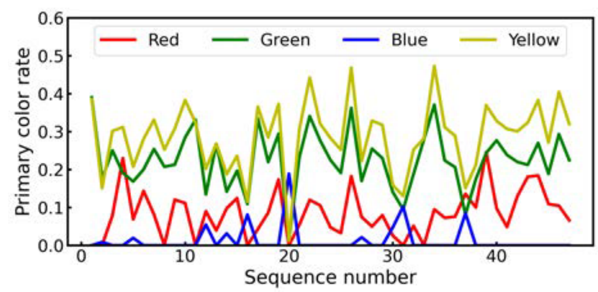
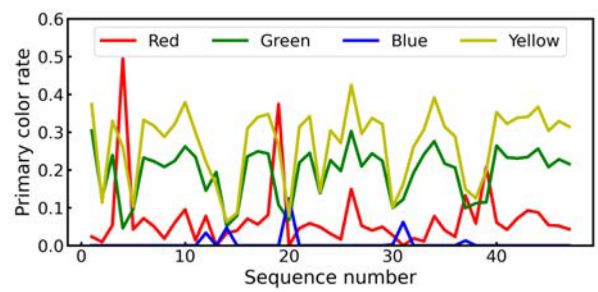


Fig. 6 Primary red, green, blue and yellow spheres.



(a) 100%-area colors



(b) First approximate colors

Fig. 7 Measurements of primary color rates.

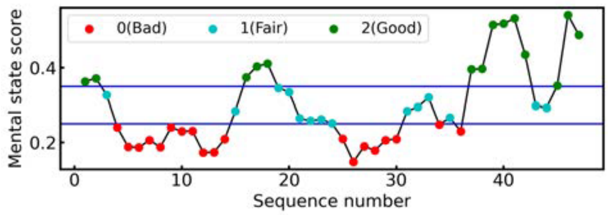


Fig. 8 Categorization of Mental States.

Table 1 Experimental color-feature vectors

Vector	Dim.	Description of elements
$CV1$	4	*pcr_r*, *pcr_g*, *pcr_b* and *pcr_y* of a 100%-area color
$CV2$	4	*pcr_r*, *pcr_g*, *pcr_b* and *pcr_y* of a 50%-area color
$CV3$	4	*pcr_r*, *pcr_g*, *pcr_b* and *pcr_y* of the first approximate color
$CV4$	5	*pcr_r*, *pcr_g*, *pcr_b* and *pcr_y* of a 100%-area color, and *pcr_r* of the first approximate color
$CV5$	3	*pcr_r*, *pcr_g* and *pcr_b* of a 100%-area color
$CV6$	3	*r*, *g* and *b* of a 100%-area color in RGB

5.4 Color Evaluation Through Classification

5.4.1 Datasets for Training a Classification Model

(1) Categorization of mental states

We determined mental state classes from mental state scores for experiment in the following way: Fig. 8 shows a timeseries of 47 mental state scores used in the case study, which range around between 0.15 and 0.55. As an example of the categorization of these scores, we classified them into three categories ($N\_CLASSES = 3$) by using two thresholds 0.25 and 0.35. The thresholding and resultant classes are also shown in Fig. 8, which indicates that a mental state class happens to change large and discontinuously. Thus, we obtained 19 class 0s, 14 class 1s, and 14 class 2s in the timeseries.

(2) Color-feature vectors and timeseries samples

We used three kinds of representative color to make up color feature vectors: a 100%-area color, a 50%-area color, and the first approximate color. All the colors were measured as primary color rates. Using the measurements, we defined several color-feature vectors for comparative evaluation, which are listed in Table 1.

Datasets for making classification models were generated from the sequences of color-feature vectors with a value of N_STEPS changed in the way described in Sect. 4.1.2 (see Table 2). The number of timeseries samples generated from 47 drawings for $N\_STEPS = 2$, 4, and 6 was 46, 44, and 42, respectively.

5.4.2 Building Color-Evaluation Models

Building color-evaluation models was carried out on the datasets by the method described in Sect. 4.4. K-fold cross validation was implemented with $K = 4$ and iterated over 20 times ($N\_KFOLD \geq 20$) to evaluate each configuration of a neural network. Thus, a model of the best validation score was found out for each dataset. The resulting models and scores are included in Table 2.

Table 2 Results of color evaluation

Exp.	Dataset		Color-evaluation model	
	Vector	N_STEPS	Configuration†	Score
A	$CV1$	2	$L_4^{20}L_{20}^{20}D_{20}^{20}D_{20}^{20}D_{20}^{3}$	0.500
B		6	$L_4^{16}L_{16}^{8}D_8^8D_8^8D_8^3$	0.721
C		4	$L_4^{20}L_{20}^{10}D_{10}^{8}D_8^3$	0.728
D	$CV2$			0.633
E	$CV3$			0.552
F	$CV4$		$L_5^{16}L_{16}^{16}D_{16}^{16}D_{16}^{3}$	0.748
G	$CV5$		$L_3^{20}L_{20}^{15}D_{15}^{8}D_8^3$	0.723
H	$CV6$		$L_3^{12}L_{12}^{6}D_6^6D_6^3$	0.665
I	$CV1$		$D_{16}^{n}D_n^3$	0.466‡
J			$L_4^nD_n^3$	0.569‡

†L_m^n denotes an LSTM $L_{m,\,n}$, and D_m^n denotes a Dense $D_{m,\,n}$.
‡An averaged score over various n's of configuration.

5.4.3 Discussion of Experimental Results

We now discuss the experimental results according to Table 2.

(1) Timeseries of color-feature vectors

We discuss the appropriateness of dealing with color-feature vectors as timeseries for classification. A timeseries of four vectors of $CV1$ was flattened to a 1-d vector and used to train a two-Dense-layer network in Experiment I, while the same timeseries was used to train a baseline recurrent network in Experiment J. Comparing these two resulting scores indicates that it is more appropriate to associate colors as timeseries data with mental states than as set data.

The length of a timeseries sample, N_STEPS, was examined in Experiments A, B, and C where the datasets were made from the same vector set with different numbers of N_STEPS. The result implies that color-feature vectors need somewhat long timeseries to represent the association with a mental state. As for $CV1$ and the mental state scores, timeseries of four or more vectors are required.

(2) Effectiveness of primary color rates

We conducted Experiments G and H to examine the classification performance of the measurements of *pcr_r*, *pcr_g* and *pcr_b* by comparison with that of the source values *r*, *g* and *b* of the same color. The result shows that primary color rates are obviously more effective in representing the relationship to the mental state scores than RGB components.

(3) Association of color features with mental state

Experiments C, D and E examined the respective kinds of representative color, by using the same four kinds of *pcr* and the same model. These three colors yielded much different validation scores; $CV1$ of 100%-area colors showed the best. In addition, the color-feature vectors $CV4$ that were composed of *pcr*'s obtained from two kinds of representative color were examined in Experiment F. The resulting validation score showed that $CV4$ outperformed $CV1$ in learning the relationship between color features and mental state scores.

From these results, we found out that a degree of association with the given mental state scores varies substantially by a color feature. Also, making up a color feature from more than one representavie colors can improve the degree of association. Within the experimental results, $CV4$ was found most closely associated.

6. Conclusion

In the experiments, we extracted representative colors from the given drawing sequence, and made some kinds of color feature from them using primary color rates based on four main psychological primary colors. As for a color feature made from one color, the 100%-area color showed closer association with the mental state scores than the first approximate color. On the other hand, the color feature composed of two colors showed even closer association. The fact that a degree of association with the same mental state scores varies by a color feature just provides evidence that appropriate colors can represent human mental states.

Also, the experimentals showed that the measurements by primary color rates were more closely associated with the mental state scores than the RGB color-components, just as expected. Primary color rates can specify in quantitative terms emotional properties which psychological primary colors describe in psychological terms. Using primary color rates is expected to make drawing therapy fine-tuned beyond color psychology.

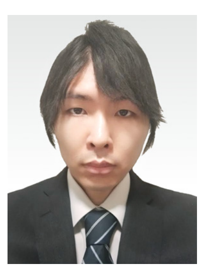
Considering the nature of psychotherapy, the experimental results must have been specialized to the given clinical data. The association between colors and mental states probably depends on various personal factors: a client's psychological problem and/or personality, his/her way of making drawings colored, a kind of mental state score to use and a categorization method for the scores, and so on. Accordingly, it is necessary to find out effective color features by examining several representative colors of drawings in each therapy situation. Primary color rates based on other psychological primary colors such as brown may be also preferable. Given good color features, advanced numerical processing suitable for an individual can be expected.

Acknowledgments

We would like to thank Prof. Chieko Kato at Toyo University for providing clinical data in the case study.

References

[1] C.A. Malchiodi, ed., Handbook of art therapy, 2nd ed., Guilford Press, USA, 2011.
[2] S.I. Buchalter, Art therapy techniques and applications, Chap. 3, Jessica Kingsley Publishers, London UK, 2009.
[3] N.A. Nijdam, "Mapping emotion to color," 2005, https://www.semanticscholar.org/paper/Mapping-emotion-to-color-Nijdam/5f0de6e7bc1d5443243f9f42f2379db9639a933d
[4] K.E. Thyme, B. Wiberg, B. Lundman, and U.H. Graneheim, "Qualitative content analysis in art psychotherapy research: Concepts, procedures, and measures to reveal the latent meaning in pictures and the words attached to the pictures," The Arts in Psychotherapy, vol.40, no.1, pp.101–107, 2013.
[5] D. Kim, S.-W. Son, H. Jeong, "Large-scale quantitative analysis of painting arts," Scientific Reports, DOI: 10.1038/srep07370, pp.1–7, 2014.
[6] C. Kato and T. Kimoto, "Psychological interpretation of drawings by color analysis for mental therapy," Advances in Image and Video Processing, vol.3, no.4, 2015.
[7] T. Kimoto, "Color feature extraction for computer-aided art therapy," Intl. Conf. Modeling and Simulation, UKSim2020, 2020.
[8] Color psychology, http://micco.se/wp-content/uploads/2010/05/Micco-Groenholm-on-Color-Affects-System.pdf
[9] K. Roushani and M.M. Honarmand, "The effectiveness of cognitive-behavioral therapy on anger in female students with misophonia: a single-case study," Iran J. Med. Sci., vol.46, no.1, pp.61–67, 2021.
[10] L.D. Cromer, B.A.F. Pangelinan, and T.R. Buck, "Case study of cognitive behavioral therapy for nightmares in children with and without trauma history," Clinical Case Studies, vol.21, no.5, pp.1–19, 2022.
[11] P.C. Buijs, A.S. Bassett, D.A. Gold, and E. Boot, "Cognitive behavioral therapy in 22q11.2 deletion syndrome: a case study of two young adults with an anxiety disorder," Journal of Intellectual Disabilities, vol.25, no.4, pp.695–704, 2021.
[12] A.C. Müller and S. Guido, Introduction to Machine Learning with Python, O'Reilly Media, Inc., CA, 2017.
[13] D. Arthur and S. Vassilvitskii, "k-means++: The Advantages of Careful Seeding," Technical Report, Stanford, 2006 (http://ilpubs.stanford.edu:8090/778/?ref=https://githubhelp.com).
[14] F. Chollet, Deep learning with Python, Manning Publications Co., NY, 2018.
[15] W.K. Pratt, Digital image processing, 4th ed., Sec. 3.5, John Wiley & Sons Inc., Hoboken NJ, USA, 2007.
[16] M.S. Tooms, Colour reproduction in electronic imaging systems, App. H, John Wiley & Sons Ltd., Chichester West Sussex, UK, 2016.
[17] ibid, App. C.
[18] A. Wright, The beginner's guide to colour psychology, Ch. 1, Colour Affects Ltd. London UK, 1998.



Satoshi Maeda received his B.S. and M.S. degrees from Toyo University, Japan in 2013 and 2016, respectively. He is currently a doctoral student at the graduate school of Toyo University.

Tadahiko Kimoto received his Dr. Eng from Tokyo University, Japan. He is currently a professor at Toyo University. He is a member of the Japanese Society of Psychopathology of Expression & Arts Therapy.