An Image Creating System based on Kansei of Natural Language Sentences

Miki Ueno† Akihiko Yasui† Naoki Mori† Keinosuke Matsumoto†

†Graduate School of Engineering †College of Engineering
Osaka Prefecture University
1-1 Gakuencho, Nakaku, Sakai, Osaka 599-8531, Japan
ueno@ss.cs.osakafu-u.ac.jp

Abstract

Recently, lots of researchers in computer science field pay attention to the personal process called “Kansei” that is human’s mapping rule between an external stimulus to an emotion. However, it is difficult to represent the “Kansei” process as a computer algorithm. In this paper, we propose the novel method of utilizing “Kansei” in computer applications that produces fractal images based on numerical vector of user’s “Kansei”. We assign emotional labels to images according to questionnaire in order to map user’s “Kansei” to certain images. The computer experiments are carried out to confirm the effectiveness of proposed method.

Keywords: Automatic Drawing Picture by Natural Language, Fractal Figures, Kansei Engineering, Sentiment Analysis

1 Introduction

The automatic picture drawing based on human emotions/thoughts by computer is one of the most important and difficult topic in information science field. Several researchers have focused on the relation between human emotion/thoughts and representation. There are lots of variation of representation, we focus on picture representation in this study.

There is well known interesting result of relation between specific sound and specific object in psychological field called buba/kiki phenomenon[1]. The result of this phenomenon shows that 95% of English-speaking adults matched "bouba" with around, amoeoboid shape and "kiki" with an angular figure. Furthermore, the game is designed to assess 4 kinds of known mapping and to check differences between adults and children[2].

From the viewpoint of the way of communication, the method that translates a picture made of pictograms into a sentence and a sentence into symbolic pictures has been proposed[3]. In this method, the most important requirement is that the produced pictures must have only one meaning because target users of this method are people who cannot read, but can understand significant information such as disaster prevention through the picture.

In order to introduce human emotions to computer applications, rule-based affect analysis model has been proposed[4; 5]. To simplify the model and make analyzing emotions easy, sentiment analysis has also been proposed. Sentiment analysis is a method which is classifying the polarity of a given text based on the document or sentence into positive or negative by means of utilizing sentiment dictionary[6]. However, there are few research about quantitative analyses to translate human mind into picture image [7]. To solve this problem, we propose a novel method which translates a user input sentence into a picture by evaluating user’s “Kansei”. We also made standalone application of the proposed method to show the experimental results.

In this paper, we propose the method of producing images by user’s inputs in section 2. We show how to assign emotional labels in section 3. Computer experiments are described in section 4, while section 5 shows the method of expanding dictionary. Section 6 introduces application example using the proposed method. Finally, in section 7, we present the conclusions of this study.

2 Proposed Method

2.1 Input

Though humans have several expression to represent their kansei, we assumed that user’s input is natural language sentence because natural lan-
guage is one of the most common communication method in the internet environment.

2.2 Emotional Output

In this paper, we define *kansei* which varies from person to person as rules of mapping an external stimulus to an emotion. We define *kansei* of input $s$ as numerical vectors. In this paper, we define 2 types of numerical vectors $\mathbf{e}_1$ and $\mathbf{e}_2$ in order to represent *kansei*. Though it requires vast amount of elements to represent human affects, to make problem simple, elements of $\mathbf{e}_1$ represent the only intensity of positive and negative. On the other hand, elements of $\mathbf{e}_2$ represent the intensity of 6 basic emotions, *joy*, *anger*, *sadness*, *hate*, *fear*, *surprise*.

The emotional intensity is calculated by search engine result. We select the 6 typical words which have strong relation to 6 emotion categories above. These words are defined as *sentiment words*. Utilizing result of web search based on these *sentiment words*, we obtained $\mathbf{e}_2$. An emotional intensity is calculated by following method:

**Emotional Intensity**: Utilizing the number of result pages in Yahoo! JAPAN keyword search, the emotional intensity $e_i^w$ is determined by morpheme $w$ and *sentiment words* $\chi_i$ in $\chi$ by following equation:

$$ e_i^w = \frac{[\text{search}(w \text{AND} \chi_i)]}{[\text{search}(w)]} \tag{1} $$

where $i = 1 \cdots 6$ represents one of 6 emotions {1: joy, 2: anger, 3: sadness, 4: hate, 5: fear, 6: surprise}. The emotional intensity $e_i$ of $s$ is obtained by $\sum_w e_i^w$.

Search($x$) is the number of result pages of query $x$. "AND" means AND search by several queries.

2.3 Emotional Output Mapping

Proposed algorithm of mapping input $s$ to $\mathbf{e}_1$ and $\mathbf{e}_2$ is as follows.

2.3.1 *Kansei* Vector $\mathbf{e}_1$

1. Obtain user's input $s$.
2. The number of morphemes $N$ without particles, auxiliary verbs, unknown words and symbols is obtained after applying morphological analysis by Sen. Each morpheme is decided $w_i (i = 1, 2, \ldots, N)$.
3. Let $j = 1$. $W_p = W_n = 0$.
4. Let $t = 0$. If $j > N$ then go step 7, if $j \leq N$ then set $w_{tmp} = w_j$.
5. If $w_{tmp}$ is contained in the sentiment polarity dictionary and the sentiment polarity is positive then $W_p = W_p + 1$. If the sentiment polarity is negative then $W_n = W_n + 1$. After that, set $j = j + t + 1$ and go step 4. If $w_{tmp}$ is not contained in the dictionary then set $t = t + 1$ and go step 6.
6. If $j + t = N + 1$ then go step 7. If $t < 3$ then word combined $w_{tmp}$ and $w_{j+t}$ is defined as $w_{tmp}$ and go step 5. If $t = 3$ then set $j = j + 1$ and go step 4.
7. Finally, *kansei* vector $\mathbf{e}_1$ is set to $\left( \frac{W_p}{N}, \frac{W_n}{N} \right)$.

2.3.2 *Kansei* Vector $\mathbf{e}_2$

By equation (1), *kansei* vector $\mathbf{e}_2$ is calculated by input $s$ as follows.

1. Obtain user's input $s$.
2. The number of morphemes $N$ without particles, auxiliary verbs, unknown words and symbols is obtained after applying morphological analysis by Sen.
3. Set *sentiment words* vector $\chi = (\chi_1, \chi_2, \ldots, \chi_6)$ and *kansei* vector $\mathbf{e}_2 = (e_1, e_2, \ldots, e_6)$, where $i = 1 \cdots 6$ represents $\{1: \text{joy}, 2: \text{anger}, 3: \text{sadness}, 4: \text{hate}, 5: \text{fear}, 6: \text{surprise}\}$. Initial value of $e_j$ is set to 0. $\chi_i$ is decided manually based on emotion $i$.
4. Let $p = 1$.
5. Search in Yahoo! Japan Keyword by $w_p$ as query and define the number of result pages as $W$.
6. Define $X_j (j = 1, 2, \ldots, 6)$ as each number of page results obtained by AND search with $w_p$ and $\chi_j$.
7. Let $q = 1$.
8. Let $e_q = e_q + \frac{X_q}{W}$.
9. If $q \leq 6$, let $q = q + 1$ and go step 8.
10. If $p < N$, let $p = p + 1$ and go step 5.
11. Finally, let *kansei* vector $\mathbf{e}_2 = (e_1, e_2, \ldots, e_6)$.
2.4 Representation of Emotional Output

In the proposed method, colors of image are determined by \( \epsilon_1 \) and the pattern of image is determined by \( \epsilon_2 \). The pattern of image is created by the Julia set\(^8\) which is one of the popular fractal patterns.

2.4.1 Settings

In this paper, we use an image for an example of emotional output. A certain situation reminds human of some images such as symbolic images, scenery images and abstract paintings. Computers recognize the image as only matrix of color value on each pixels. We set the width and height of images to 500 pixels and color model to RGB.

2.4.2 The Julia Set

We use the Julia set to represent emotional output. Following equation of complex function of complex number \( z \) is defined first.

\[
f(z) = z^2 + C
\]  

Constant \( C \) is defined as follows.

\[C = p + qi\]

Define following recurrence formula:

\[Z_{n+1} = f(Z_n)\]

To obtain Julia set, judge whether the \(\lim_{n \to \infty} Z_n\) is convergent or not. The border between convergent and divergent is the Julia set.

2.4.3 Condition of Convergent

In theory, to judge whether a complex number is convergent or not, we check the limiting value of \( Z \) is required. However, in the proposed system, if absolute value of \( Z_n \) is over 2, we regard the value as divergent.

1. Let \( n = 0 \)
2. Let \( z_0 = 0 \)
3. Let \( n = n + 1, Z_{n+1} = f(Z_n) \)
4. If \( n \leq 2000 \) then \( n_{\text{div}} = 0 \) and go to step 3.
5. If \( |Z_n| > 2 \) then \( n_{\text{div}} = n \).

2.4.4 Drawing Algorithm

Each \( C_i (i = 1, 2, ..., 6) \) relates to a label of 6 emotions.

1. Obtain \( \epsilon_1, \epsilon_2 \).
2. We set the initial value of Julia set for 6 different emotions as follows:

\[
\begin{align*}
C_1 &= 0.3 \\
C_2 &= -1 - 0.27i \\
C_3 &= 0.4 + 0.1i \\
C_4 &= -0.74 - 0.18i \\
C_5 &= -0.06 - 0.68i \\
C_6 &= -1.37
\end{align*}
\]

3. In Julia set,

\[
C = \frac{\sum_{i=1}^{6} \epsilon_2 C_i}{\sum_{i=1}^{6} \epsilon_2 C_i} \quad (11)
\]

Because the elements of \( \epsilon_2 \) is represented the emotion as mentioned in Section 2.3.2, \( C \) is calculated by considering the emotional intensity \( \epsilon_2 \).

4. All pixel color of \((x, y)\) is calculated as following RGB\((r, g, b)\):

\[
\begin{align*}
r &= n_{\text{div}}(x, y) \frac{v_1 + w_1}{256} \quad (12) \\
g &= n_{\text{div}}(x, y) \frac{v_2 + w_2}{256} \quad (13) \\
b &= n_{\text{div}}(x, y) \frac{v_3 + w_3}{256} \quad (14)
\end{align*}
\]

In case of \( \epsilon_1 = (0, 0) \),

\[
r = g = b = n_{\text{div}}(x, y) \% 256 \quad (15)
\]

where \( \% \) is a modular operator. \( v = (v_1, v_2, v_3) \) and \( w = (w_1, w_2, w_3) \) are defined as follows utilizing emotional vector \( \epsilon_1 = (\epsilon_1, \epsilon_2) \). If \( v_1 = 0 \), we reset \( v_1 = 1 \).

\[
\begin{align*}
v &= \left( \left[ \frac{x_0}{x_1 + x_2} \right], \left[ \frac{x_1 - x_0}{x_1 + x_2} \right], \left[ \frac{x_1}{x_1 + x_2} \right] \right) \\
w &= \left( \left[ 255 - v_1, 255 - v_2, 255 - v_3 \right] \right)
\end{align*}
\]

where \([x]\) represents the maximum integer value which is not over \( x \). In general, if the first element of \( \epsilon_1 \) which represents positive is large, the main color of image becomes red. If the second element of \( \epsilon_1 \) which represents negative is large, the main color of image becomes blue. If positive value and negative value are almost equivalent, the main color of image becomes green. If positive value and negative value are 0, the color of image becomes gray-scale.


3 Assign Emotional Labels to Images

In this section, we show how to assign each emotion to images created by each \( C_i(i = 1, 2, \ldots, 6) \) in 2.4.4. First of all, we have to describe that finding the strict relation between emotions and images is one of the most difficult problem, so we utilized imperfect rule to assign emotion labels in this paper. We focus on only 6 Julia set images by initial values \( C_1, C_2, \ldots, C_6 \), and assign only one emotion to each image by result of human questionnaire. Figures 1 \(-\) 6 show Julia set images of \( C_1, \ldots, C_6 \), and table 1 shows the result of questionnaire. The algorithm of assignment is as follows:

1. Set the total number of emotional labels \( j \) assigned to image \( i \) to \( x_{ij} \). The number of emotion labels \( j \) is set to \( \{1 \rightarrow \text{joy}, 2 \rightarrow \text{anger}, 3 \rightarrow \text{sadness}, 4 \rightarrow \text{hate}, 5 \rightarrow \text{fear}, 6 \rightarrow \text{surprise}\} \). Where \( \sum_{j=1}^{6} x_{ij} = 12 \) and \( \sum_{i=1}^{6} x_{ij} = 12 \).

2. Denote \( i, j \) of \( \arg \max_{i, j} x_{ij} \) to \( i^*, j^* \).

3. Assign emotional label \( j^* \) to image \( i^* \), then remove all elements of \( x_{i^*, j} \) and \( x_{ij^*} \) from \( \{x_{ij}\} \).

4. If \( \{x_{ij}\} \neq \emptyset \), go to step 3.

In this result, the reliability of label of the figure 5 (label : joy) and the figure 3 (label : surprise) is very high because the number of votes is large compare to other emotional labels.

We give one sheet printed gray-scale images to participants, so the order does not affect result. We obtained participants comment as follows:

- Because adding emotion labels to a certain gray-scale images is very difficult, lots of annotators adopted elimination way to annotate.
- The shape features of the image such as roundness and ruggedness has strong affect to results.

Table 1. The Result of Questionnaire

<table>
<thead>
<tr>
<th>label/fig.</th>
<th>fig.1</th>
<th>fig.2</th>
<th>fig.3</th>
<th>fig.4</th>
<th>fig.5</th>
<th>fig.6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1: joy</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td>2: anger</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>3: sadness</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>4: fear</td>
<td>5</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>5: surprise</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 2. Elements of \textit{Kansei} Vector \( \epsilon_2 \) of Each Input

<table>
<thead>
<tr>
<th>emotion</th>
<th>input1</th>
<th>input2</th>
</tr>
</thead>
<tbody>
<tr>
<td>joy</td>
<td>0.1669</td>
<td>0.1424</td>
</tr>
<tr>
<td>anger</td>
<td>0.0960</td>
<td>0.1186</td>
</tr>
<tr>
<td>sadness</td>
<td>0.3693</td>
<td>0.2568</td>
</tr>
<tr>
<td>hate</td>
<td>0.0440</td>
<td>0.0674</td>
</tr>
<tr>
<td>fear</td>
<td>0.0574</td>
<td>0.0827</td>
</tr>
<tr>
<td>surprise</td>
<td>0.1187</td>
<td>0.1673</td>
</tr>
</tbody>
</table>

4 Computer Experiment

To show the results of the proposed method mentioned in Section 2, we created 2 images based on 2 example inputs by the proposed method.

\textbf{Input1} “I have a date with a pretty girl”.

\textbf{Input2} “I was really disappointed with a broken heart”.

Figures 7 and 8 show the images created by \textbf{Input1} and \textbf{Input2} respectively. In figure 7, the value of element \( r \) in RGB is large because positive input. On the other hand, in figure 8, the value of element \( b \) in RGB is large because of negative input.

The number of morphemes\(^1\) of \textbf{Input1} is 4, where the words of part of speech, auxiliary verbs, unknown words and symbols are removed. There exists 2 words (\textit{pretty}, \textit{date}) which have the positive sentiment in sentiment dictionary. On the other hand, there are no negative words in \textbf{Input1}. Therefore, \( \epsilon_1 \) of \textbf{Input1} is calculated as follows.

\[
\epsilon_1 = \left( \frac{2}{4}, 0 \right) = (0.5, 0)
\]  

The number of morphemes of \textbf{Input2} is 5, where the words of part of speech, auxiliary verbs, unknown words and symbols are removed. There are no positive words in input2. On the other hand, there exists 2 words (\textit{broken heart}, \textit{disappointed}) which have the negative sentiment in the sentiment dictionary. Therefore, \( \epsilon_1 \) of \textbf{Input2} is calculated as follows.

\[
\epsilon_1 = \left( \frac{2}{5}, 0 \right) = (0, 0.4)
\]  

Table 2 shows \( \epsilon_2 \) calculated by the algorithm in 2.3.2.

\(^1\) We count the number of morphemes in Japanese
Figure 1. Image of the Julia Set : 1

Figure 2. Image of the Julia Set : 2

Figure 3. Image of the Julia Set : 3

Figure 4. Image of the Julia Set : 4

Figure 5. Image of the Julia Set : 5

Figure 6. Image of the Julia Set : 6
that the appropriate $\chi$ with best elements can represents the emotion of sentences correctly.

Table 3 shows the number of words in each emotional candidate words dictionary. We try to add new words to emotional candidate words dictionary by machine learning with twitter results.

<table>
<thead>
<tr>
<th>emotion</th>
<th>words</th>
</tr>
</thead>
<tbody>
<tr>
<td>joy</td>
<td>458</td>
</tr>
<tr>
<td>anger</td>
<td>177</td>
</tr>
<tr>
<td>sadness</td>
<td>243</td>
</tr>
<tr>
<td>hate</td>
<td>118</td>
</tr>
<tr>
<td>fear</td>
<td>181</td>
</tr>
<tr>
<td>surprise</td>
<td>107</td>
</tr>
</tbody>
</table>

6 Application

In this section, we show the example of applying proposed method to an application.

We have researched picture drawing system by computer and have already proposed Twitter Bot called Cotopaint which can draw pictures semi-automatically according to user’s tweets. Cotopaint generates a picture by combining prepared parts and simple figures such as circles and squares. We try to introduce the proposed method into the drawing algorithm of Cotopaint.

6.1 Outline of Cotopaint

The main procedures of drawing pictures in Cotopaint are checking user’s tweet, executing selected drawing commands and uploading the picture with some comments to Twitter server. Generating pictures have 2 different layers called the base layer and the drawing layer. Base layer represents background of the picture such as blank or scenery pictures. Drawing layer represents variable parts of pictures and is changed every-time Cotopaint receives user’s tweets.

Outline of Cotopaint is as follows:

1. Cotopaint tweets in order to inform users about starting to draw. Let $t = 0$. Set minimum waiting time $m$ randomly in range [3,5]. Set variable waiting time $c$ randomly in range [0,15].

2. Select base layer picture from the background picture database.
3. Wait \( m + c \) minutes to receive user’s tweets.

4. After \( m + c \) minutes, get all tweets mentioned to Cotopaint to decide drawing commands. Execute commands to re-draw drawing layer.

5. Reply to each user in order to inform using drawing commands.

6. Output picture is generated by combining the base layer and drawing layer. This output picture of \( t \) is saved as new picture \( P_t \) into local disk.

    Then, upload new output picture to Twitter server. Set output picture of \( t \) to base layer of next time step \( t + 1 \). Clear drawing layer.

7. Set \( c \) randomly in range \([0,15]\). Set \( t = t + 1 \). Go to step 3.

    Figure 9 shows example of Cotopaint’s Twitter timeline.

6.2 Drawing Command of Cotopaint
    A drawing command \( d \) has 5 drawing parameters. \( p, c, n, r, s \) are defined as follows:

    \( p \) : \((x, y)\) coordinates of the drawing layer.

    \( c \) : RGB color represented by \((r, g, b)\).

    \( n \) : Times of executing a drawing command.

    \( r \) : The decided size for geometric pattern represented by pixel (px).

    \( s \) : Type of drawing pattern.

    Those parameters are decided based on user’s inputs. Cotopaint recognizes specific keywords in inputs and translates those keywords into numerical drawing parameters. Cotopaint draws type \( s \) pattern, \( n \) times with the radius \( r \), the color \( c \), the center location \( p \).

    Table 4 shows relation between drawing parameters and keywords in user’s input tweet. Parameter \( r \) has no keywords because \( r \) is decided randomly.

    Cotopaint can execute following procedures by receiving user’s mention tweets.

    1. Draw figures by using the drawing command \( d \).

    2. Set drawing parameters.

<table>
<thead>
<tr>
<th>parameter</th>
<th>keyword</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p )</td>
<td>&quot;above&quot;</td>
<td>( p = (0, a) )</td>
</tr>
<tr>
<td></td>
<td>&quot;below&quot;</td>
<td>( p = (0, S/2) + (0, a) )</td>
</tr>
<tr>
<td></td>
<td>&quot;right&quot;</td>
<td>( p = (S/2, 0) + (a, 0) )</td>
</tr>
<tr>
<td></td>
<td>&quot;left&quot;</td>
<td>( p = (a, 0) )</td>
</tr>
<tr>
<td>( c )</td>
<td>prepared color</td>
<td>set new color by the certain keyword</td>
</tr>
<tr>
<td></td>
<td>others</td>
<td>set new color randomly</td>
</tr>
<tr>
<td>( n )</td>
<td>1 - 9 times</td>
<td>set ( n ) by the certain number</td>
</tr>
<tr>
<td>( s )</td>
<td>circle</td>
<td>set square</td>
</tr>
</tbody>
</table>

\( n \) : The maximum range of \( n \). \( L \) is decided by Cotopaint subject to \( 0 \leq L \leq S/2 \).

6.3 Problem of Cotopaint

Cotopaint needs user’s instructions to draw a picture. Hence there are no surprise to see the generated picture. We would like to modify this feature of Cotopaint by introducing an unpredictability into generated pictures. However, we want to avoid showing the random pictures
and keep some relationship between user’s inputs and generated pictures.

Proposed method in this paper may be useful to achieve this because generated fractal figures are hard to be predicted and those figures have some relationship such as polarity of positive/negative of user’s inputs shown in 2.4.4. To analyze relationship between the generated picture and user’s Kansei, we have to continue to research about quantitative representation of Kansei.

7 Conclusion

In this paper, we propose the novel method which produces images based on user’s inputs and show experimental results.

Followings are subjects for further study.

Considering emotional vector In this paper, we try to represent humans emotions by the intensity of positive and negative, and the intensity of joy, anger, sadness, hate, fear and surprise. However, human emotions are more complex and consists lots of aspects. In order to represent human emotion in detail, we have to consider the definition of elements of emotional vector.

Creating various images In this paper, we utilize the Julia set as emotional image. However, there are lots of another types of images such as the Mandelbrot set, symbolic, graphic and cartoony images. Find an adequate relationship between emotions and images are important topic of this study.

Creating images with multi-labeled emotion Real emotion contains several basic emotions. For example, love is composed of joy, fear and another complex emotion. In order to translate all emotions to images, we have to show the reasonable algorithm of creating images which have relation to several emotions. However, it is difficult to generate figures which represent several emotions together. Our proposed method, a figure of mixing 2 emotions like joy and fear does not always have relation to fear and joy. Creating multi-labeled emotions figures is important further study.

Acknowledgment

This work was supported by Grant-in-Aid for JSPS Fellows (Grant Number: 25-10978).

References


