Picture models
for 2-scene comics creating system

Miki Ueno[a], Naoki Mori[a], Keinosuke Matsumoto[a]

[a] Department of Computer Science and Intelligent Systems, Osaka Prefecture University
1-1 Gakuencho, Nakaku, Sakai, Osaka 599-8531, Japan;
e-mail: ueno@ss.cs.osakafu-u.ac.jp

KEYWORD

Automatic Drawing Picture
2-scene Comic Creating System
Picture State Transition
Picture Model
Introducing Semantics to Picture
Comic Engineering

ABSTRACT

Recently, computer understanding pictures and stories becomes one of the most important research topics in computer science. However, there are few researches about human like understanding by computers because pictures have not certain format and contain more lyric aspect than that of natural language. For picture understanding, a comic is the suitable target because it is consisted by clear and simple plot of stories and separated scenes.
In this paper, we propose 2 different types of picture models for 2-scene comics creating system. We also show the method of the application of 2-scene comics creating system by means of proposed picture model.

1 Introduction

In artificial intelligence, finding the algorithm of generating intellectual products such as novels, comics and musics is ultimate purpose.

Human represents such products using characters, certain symbols and pictures including lines and shapes. Picture is a strong way to communicate because pictures represent the position and postures more clearly than natural language[Kurlander, D. *et al.* 1996] and include nonverbal information, so they are understandable for various people regardless of age and national origin.

In the field of picture research, despite current advances of image recognition[Herot, C. F. 1976, Eitz, M. *et al.* 2012, Fu, H. *et al.* 2011] and analyzing brain activities for images[Walther, D. B. *et al.* 2011], there are few research about semantics of picture[Ito, K. *et al.* 2007, Gross, M. D. *et al.* 2000].

To solve this problem, we have proposed a novel dialogue system called *Picture Information Shared Conversation Agent*(*Pictgent*) [UENO, M. *et al.* 2012] that shares conversational background knowledge by showing prepared pictures with model to users. Though we have proposed the picture model to represent semantics of picture in Pictgent, we have not obtained complete one because of difficulty of defining what understanding pictures is[Arai, N. *et al.* 2012]. In order to solve the problem, we focus on the continuous pictures with transition of object such as 4-scene comics because they have meaningful plot of stories.

In this paper, we propose 2 types of picture models for creating 2-scene comics and analyzing information from existing comics and also show the system which can create comics like pictures flexibly based on objects in the prepared dataset. We focused 2-scene comics first because this is the minimum set of pictures which contains some kinds of object transition.

To achieve computer understanding pictures, 3 issues are required as follows:

1. Estimate situations in pictures.
2. Represent feel or emotion of picture numerically.
3. Expect next scene after current pictures.

We propose the method of picture models for creating 2-scene comics in section 2. We show the operator based picture model and the constitution of application in section 3. We also show the text based picture model in section 4. Finally, in section 5, we present the conclusions of this study.

2 Proposed Method

2.1 Proposed picture models

To analyze 2-scene comics by computer, there exists 2 following different types of approaches.

- **Generate 2-scene comics**

Generating 2-scene comics by computers.

- **Analyzing constitution of existing comics**

Making databases of existing comics for computers.

We propose 2 types of picture models for both approaches called "operator based picture model" and "text based picture model". Operator based picture model is utilized to generate 2-scene comics by computer. On the other hand, text picture model represents information of existing comics.

2.2 Picture model for 2-scene comics

We have to consider following 3 issues to make 2 types of proposed picture models for 2-scene comics. In this paper, we focus on the 1st step.

1. Plot of stories

In this study, we prepare the database using existed comics and common sense. At first, we omit the details of size and position from existing comics but focus on meaningful plot of stories. We should define how to represent information of the plot in order to use it easily by computers.

2. Effective representation

In comics, the way of draw such as an exaggeration of expressions and a deformation has strong relation with an impact of pictures and a clearness of the meanings of stories. After defining the representation of plot, we will try to find the efficiency of representation.

3. Character's words

Existing comics are able to classify into 3 types from the view point of the dependency of stories on character's words as follows:

- Can understand only pictures
- Can understand by pictures and character's words
- Can understand only character's words

There are few stories to be able to be understood without character's words because character's words are very important information. In this research, we focused on stories to be able to be understood by pictures and character's words. On the other hand, stories which are understandable without pictures are not suitable for this study, e.g. slip of the tongue.

We should consider how to represent character's words in picture model and generate new character's words while keeping the same meaning. However, paraphrasing is one of the difficult tasks in natural language processing, so we represent original first of all and try to translate them into some templates in future work.

3 Operator based Picture Model

In this section, we proposed operator based picture model for 2-scene comic creating system.

3.1 Representation of picture for operator based picture model

In order to represent semantics of pictures for computer, we propose the "operator based picture model" in which we define each of elements in pictures including background images and balloons as certain object.

It is difficult to create reasonable operator based picture model for all pictures. Therefore, as previously mentioned, we focused on 2-scene comics because 2 sequential pictures have enough information rather than a single picture. In this method, we should prepare dataset including images of objects, information of

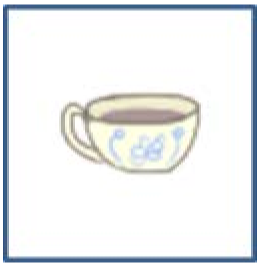


Fig. 1. Cup (original image)

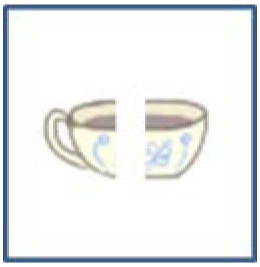


Fig.2. Cup (apply $\hat{\phi}_{\text{divide}}$)

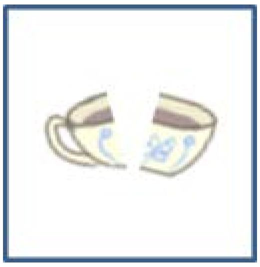


Fig. 3. Cup (apply $\hat{\phi}_{\text{rotate}}\hat{\phi}_{\text{divide}}$)

objects, types of areas, drawing operators, and transition. Each picture has above information and history, and those are representation of picture for operator based picture model.

3.2 Operator based picture model

In this study, initial picture p_0 is generated by system. p_0 is a blank picture or a background picture and has several areas determined by system. User can operate picture as user input. User operation set in time t is represented by $\boldsymbol{u_t}$, and $\boldsymbol{u_t}$ contains all kinds of user's operation for picture p_t . p_{t+1} is created applying $\boldsymbol{u_t}$ to p_t . We describe this as follows:

$$p_{t+1} = \prod_{u_{i,t} \in \boldsymbol{u_t}} u_{i,t} p_t \tag{1}$$

where $u_{i,t}$ represents each user's operations in time t. The relation among p_t, o_t, $\boldsymbol{u_t}$, $\{O_t\}$ and $\{U_t\}$ is as follows:

$$\{O_t\} = \{o_t \in p_t\} \tag{2}$$

$$\{U_t\} = \{\boldsymbol{u_1}, \boldsymbol{u_2}, \ldots, \boldsymbol{u_t}\} \tag{3}$$

$$p_{t+1} = \prod_{u_{i,t} \in \boldsymbol{u_t}} u_{i,t} p_t = \prod_{u_{i,t} \in \boldsymbol{u_t}} u_{i,t} \prod_{u_{i,t-1} \in \boldsymbol{u_{t-1}}} u_{i,t-1} p_{t-1}$$

$$\ldots \tag{4}$$

$$= \prod_{j=0}^{t} \prod_{u_{i,j} \in \boldsymbol{u_j}} u_{i,j} p_0$$

Definite operation of $u_{i,t} \in \boldsymbol{u_t}$ is represented by drawing operator ϕ and operand object in p_t. Each ϕ also has specific augments such as color, angle or other objects. Object o_t is able to represent object transition. Object o_n in n-th scene has Markov model as bellow:

$$o_n = \prod_{i=0}^{N_n^{\max}} \phi_i o_0 \tag{5}$$

where $N^{\max}$ is total number of ϕ to applied to o . $N_n^{\max} \geq n-1$ because each u_i contains at least one ϕ.

3.3 Base drawing operator $\hat{\phi}$

Drawing operator ϕ has the function of transform for pictures. Product of several drawing operators is also drawing operator and the set of base operator is defined as $\{\hat{\phi}\}$. Increasing the number of base operators, the ability of representation of the system also increases more and more because system can create lots of new images utilizing current objects.

3.4 Implementation of drawing operator ϕ

In this paper, the system is written by Java and the operator is implemented by command pattern in design pattern[Gamma, E. *et al.* 1995]. Base drawing operator corresponds to command and product of drawing operator corresponds to macro-command. In Java implementation, product of operators is represented by using queue.

For example, ϕ_{crack} which represents cracked transition is composed of base drawing operator $\hat{\phi}_{\text{rotate}}\ \hat{\phi}_{\text{divide}}$.

2 base drawing operators are applied to o as follows:

$$\phi_{\text{crack}}o = \hat{\phi}_{\text{rotate}}\hat{\phi}_{\text{divide}}o \qquad (6)$$

Figure 1-3 show the variation of image by applying base drawing operator of ϕ_{crack}.

3.5 Flow of proposed system

The basic flow of proposed system is described as follows[1]:

1. Select the base image as background for initial states. We define this picture as p_0.

2. System waits for user inputs. User can decide several actions to p_0. In this paper, user action is only setting an object in objects set shown by system GUI. User can set their interested object on one area of p_0 with drawing operator and emotion information. Any number of objects are added to p_0 under condition of existing only one object in each area. When user put an object to p_0', new areas decided by set object are added to picture.

3. After applying all user inputs $\boldsymbol{u}$ to p_0, p_0' is obtained. This p_0' becomes 1st scene.

4. System tries to change objects in p_0' based on user decided drawing operator and emotion information to target object. New image of changed objects is obtained by transition DB.

5. Applying all changes to p_0', p_1 is obtained. p_1 represents 2nd scene.

   2-scene comic :

   p_1 is final output.

   n **-scene comic :**

   p_1' becomes p_2. System waits for user inputs again and repeats above flow until p_{n-1}' becomes p_n.

3.6 User input $\boldsymbol{u}$

As mentioned above, user action is putting some objects on one area of the picture. User decides the object and the area to set shown by system. Set objects in $\{O_{\text{set}}\}$ and picture objects in p_t are different as though set objects become picture objects after user setting, where $\{O_{\text{set}}\}$ represents a group which is applicable to picture. When user sets the object to picture, user also sets the "drawing operator" and "emotion" information to decide transition pattern of setting object. Finally, user can decide following information by one user input.

Object o **in** $\{O_{\text{set}}\}$:

[1] p_t' represents the latest picture after user decides several actions for p_t in eq.1. The operator applied to p_t' is decided uniquely based on user action and p_{t+1} is obtained after it is executed.

Table 1. Object DB

ID	Object Name	Class Name	Transition:Frequency	Image File Path	Included Area ID
1	cup	container	drink:4,crack:2,drop:1	/fig/cup.png	null
2	cat	animal	bite:5,attach oneself to:3,run away:2	/fig/cat.png	null
3	girl	human	stand:3,sit:1, ⋯	/fig/girl.png	2, 3, 4

Some objects have several areas with IDs where user can set a new object.

Class name is important information to use external resources such as Wordnet.

Table 2. Transition DB

ID	Object ID	Area ID	Operator Name	Emotion Name	Drawing Operator
1	*	*	divide	sad	crack(Object ID)
2	*	2	*	*	drop(Object ID)

*: Wildcard

In database, we describe $\phi_{\text{crack}} \rightarrow$ crack(x).

Table 3. Area Name DB

ID	Area Name
1	whole
2	ground
3	hand

Table 4. Area DB

ID	Area ID	Object ID	Start point of area	End point of area
1	1	*	(*, 170)	(*, 190)
2	2	3	(20, 40)	(40, 60)

*: Wildcard

The object information of user setting objects.

Area a :

Set area of o.

Operator m :

Operator type of setting object o.

Emotion e:

User can set the emotion e of setting object. This information effects transition of o .

Using those information, system decide the transition of each objects in the current picture. All transitions are represented by change of image. Therefore, system have to prepare new images of all objects in picture after transition. Creating new images has the difficulty of huge range of variation. To solve this problem, we prepared only basic image sets and created various images to apply drawing operators to an images. For example, we can create various image pattern of cup such as cracked cup, rotated cup or different colored cup from only one cup image.

3.7 Databases in the proposed system

Our final purpose is to establish a database which can represent the transition of objects in 3.2. However, to make this, we have to decide the complete distribution model of each object. Since this is a difficult task, we simplified the object transition that all objects have only 1 transition result. The transition pattern of each object is decided by type of object, area of object, operator for objects and emotion of objects uniquely. We created those patterns based on existing pattern of comics and common sense, e.g. cup is cracked or we use the umbrella in the rain. Although we set only 1 typical pattern in this study, transition patterns will be extended based on distribution of existing data in future work. We created 4 databases (DBs) for managing those information as follows.

Object DB

Object DB manages the information of all objects. In object DB, each object has the attributes of object name, class name, what kinds of transition and the rarity in existed comics and common sense, image file path of image, area IDs. Table 1 shows the example of object DB.

Transition DB

Transition DB manages the distribution of object transition. This database contains user input $\boldsymbol{u}$ as key and drawing operator ϕ as value. User input $\boldsymbol{u}$ is represented by following equation:

$$\boldsymbol{u}=(o, a, m, e)$$

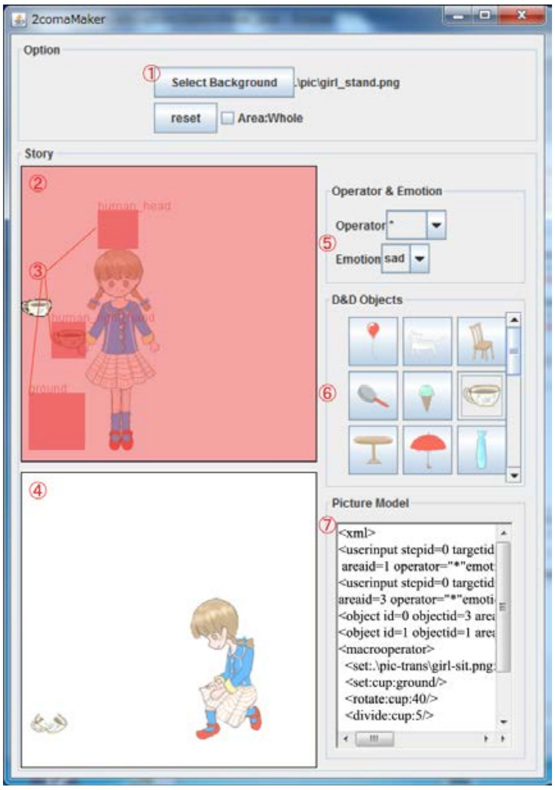
Operator name m is name string corresponding to each base drawing operator name. Table 2 shows the example of transition DB.
In current transition DB, only 1 operator is considered for each $\boldsymbol{u}$ although $\boldsymbol{u}$ has the distribution of several operators.

Area name DB & Area DB

Area name DB and area DB manage the area information in picture. Though area is not appeared visually in the picture, this is important to control the location of set objects. When user sets the object by mouse operation, user can recognize the area by specific color.
Table 3 shows area name DB which has the relation between area ID and area name. Table 4 shows area DB which has the start point and the end point of area. The start and end point of same area name are different among objects.

3.8 Application

We developed 2-scene comic creating system based on the proposed method. Figure 4 shows the outline of system. Figure 5 shows XML of operator based picture model of 2nd scene.
Followings are explanations of each part of system corresponding to the number in figure 4.



.\pic\girl_stand.png

☐ Area:Whole

Operator *

Emotion sad

```
<xml>
<userinput stepid=0 targetid
 areaid=1 operator="*"emot
<userinput stepid=0 targetid
areaid=3 operator="*"emoti
<object id=0 objectid=3 area
<object id=1 objectid=1 area
<macrooperator>
  <set:.\pic-trans\girl-sit.png
  <set:cup:ground/>
  <rotate:cup:40/>
  <divide:cup:5/>
```

Fig. 4. Outline of 2-scene comic creating system

1. Selectable background image

Select button of the base image for 1st scene. The blank picture or prepared picture is able to be selected.

2. 1st scene image

The display area of 1st scene image. After applying user input, p_0 becomes $p_0{}'$.

3. Selectable areas

Areas where user can set object in 1st scene.

4. 2nd scene image

The display area of 2nd scene image which is generated by applying user input of 1st scene.

Table 5. XML tag of operator based picture model

Tag Name	Explanation
<object>	Information of object. The attributes of this tag are object ID in picture, ID of object DB and area ID.
<macrooperator>	Information of applied drawing operator to the object.
<userinput>	Information of user’s inputs. The attributes of this tag are picture step, target object ID, area ID, operator name and emotion name.

```
<xml>
<userinput stepid=0 targetid=3 areaid=1
operator=“*”emotion=“*”/>
<userinput stepid=0 targetid=1 areaid=3
operator=“*”emotion=“sad”/>
<object id=0 objectid=3 areaid=1 />
<object id=1 objectid=1 areaid=2 />
<macrooperator>
<set:.\pic-trans\girl-sit.png:whole/>
<set:cup:ground/>
<rotate:cup:40/>
<divide:cup:5/>
<zoom:cup:50@40/>
</macrooperator>
</object>
</xml>
```

Fig. 5. XML of operator based picture model of 2nd scene

5. Operator & emotion

The display area of operator and emotion for set objects.

6. Selectable object

The display area of $\{O_{\text{set}}\}$ which is set of user selectable objects.

7. Operator based picture model

Operator based picture model XML of 2nd scene.

4 Text Based Picture Model

In this section, we proposed text based picture model to represent information of existing comics.

4.1 Representation of picture for text based picture model

To define text based picture model focused on the plot as mentioned in 2.2, we show the way of creating text based picture model from existing comics in this section.

1. Simplify stories of comics
First, we simplify information of comics and represent its plot by natural language. We show 2 examples as follows:

Story 1: Character (o_1) with a tough-look carry stuffed toy (o_3) in concealment from character (o_2).
Story 2: Character (o_1) with a bandage over the eyes try to chop water-melon (o_2) but chop another one (o_3).

2. Reconstruct information
To make stories clear, we reconstruct arranged object based on existing comics. In this study, all of 2-scene comics were created by using existing 4-scene comics. Figure 6, 8 show the examples of reconstructed information based on existing 4-scene comics.

3. Represent plot by text based picture model
We represent reconstructed information by text based picture model.

4.2 Text based picture model

We focus on the 2 important approaches in order to create text based picture model using existing comics.

- Compress the information and represent it by simple way.
- Reduce the dependency on natural language.

The format of text based picture model is as follows:

o:
function1()
function2()
…
function*N*()

where o represents each object in picture and function represents features of o. There are several functions in text based picture model. Each function has specific argument values and some functions have the special argument named "req".

Figure 7, 9 show the examples of text based picture model. Table 6 shows each elements of text based picture model.

Table 6. The special argument and function of text based picture model

req	The special argument of the element of class and property. "req" is optional. The format showed as "req = value". It means the intensity of exchangeability of element. The range of value is [0, 1]. The element of req = 0 is not important and can be exchanged to other elements. The element of value = 1 is important and cannot be exchanged to other elements.
max_position	max_position(x, y) represents of the number of separated parts for x-axis and y-axis. Only "max_position" is the information per scene.
class	class(*type*, req=n) represents of kinds of object. Basic type has 4 patterns; character, non-character, balloon, background. More detailed information is represented by nest of information. Character means not only human but also human like objects such as talking animals and thinking inanimate objects.
position	position((x_1, y_1),…,(x_N, y_M)) represents all lattices of covered by object. At least, 1 lattice is decided for each object. The lattice structure of pictures is decided by max_position. Fig. 8 represents the example of max_position(2, 2). The position is important if "max_position" is large.
property	It means the information of attached state. There are lots of patterns so we represent by natural language at first and will classify them in future.
angle	The argument of angle are only 4 types, front, back, left, right in the picture facing. We should consider the case that the direction of face is different from that of body in future work.
posture	posture(x, y, z) represents character's posture. x, y z represent the degree of standing, sitting and lying respectively. The range of x, y, z is [-1, 1] and $\lvert x + y + z \rvert = 1$. E.g. sit down on one's heels : (0.5, 0.5, 0), a handstand : (-1, 0, 0), push-up with knees : (0, 0.5, 0.5), push-up : (0, 0.2, 0.8)
hand	hand(*state*) represents the state of the hand. Now argument state is represented by natural language. The state of hand has important information often because the part of human body appears and the others are concealed especially the lower half of the body in scene of existing comics. We classify the state of hand into the kinds of typical one in future.

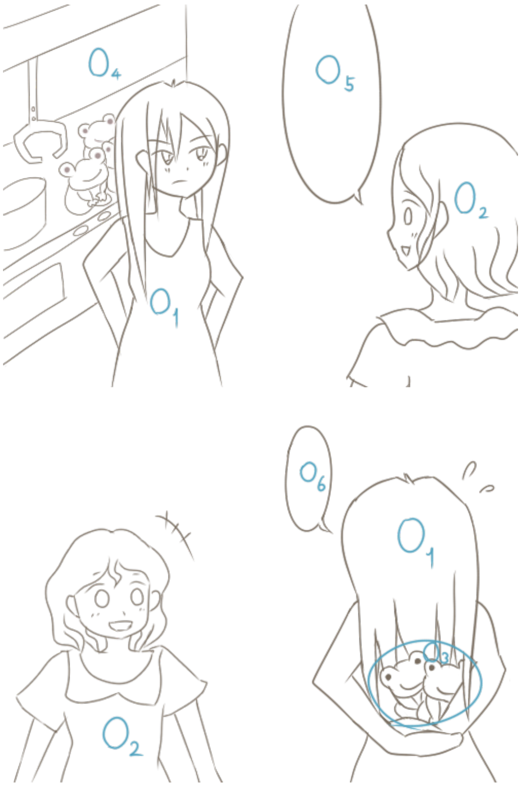


Fig. 6. The reconstucted 2-scene comic by story 1

5 Conclusion

In this paper, we proposed the 2 types of picture models for 2-scene comics creating system and developed creating the system utilizing the proposed operator based picture model. We also show the text based picture model taking sample pictures as examples. Followings are important future work:

```
>1-scene
position-max(2, 1)

o1:
class(character, req = 1)
position(1, 1)
property("tough-look", req = 1)
angle(front)
posture(1, 0, 0)

o2:
class(character, req = 1)
position(1, 1)
angle(back)
posture(1, 0, 0)

o4:
class(background(inner-building(shop))
property("amusement arcade")

o5:
class(balloon(O_2))
property(“You play racing game or fighting game, do you?")

>2-scene
position-max(2, 2)

o1:
class(character, req = 1)
position((2, 1), (2, 2))
property("tough-look", req = 1)
angle(back)
posture(1, 0, 0)
hand(carry in concealment(O_3))

o2:
class(character, req = 1)
position((1, 1), (1, 2))
angle(front)
posture(1, 0, 0)

o3:
class(non-character(inanimate object(toy("stuffed toy"))))
property("cute", req = 1)

o4:
class(background(inner-building(shop))
property("amusement arcade")

o6:
class(balloon(O_1))
property("Yes...")
```

Fig. 7. Text based picture model of story 1

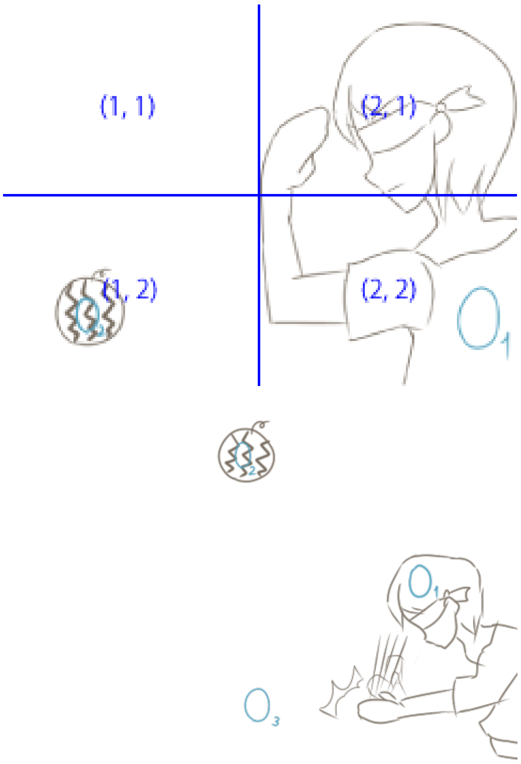


Fig . 8 The reconstucted 2-scene comic by story 2

- **Consider the relation between 2 types of picture models**

In order to generate new 2-scene comics directly based on analyzed information of existing comics, we should consider the relation between operator based picture model and text based picture model.

- **Define appropriate types and representation of drawing operator**

We try to represent pictures information by drawing operator. However, picture information contains lots of aspects and more complex information, so we have to extend the number of drawing operator and modify its elements.

- **The dependency of the locality**

The funny stories vary according to the locality. We used Japanese comics but we should check whether foreign people understand the plot of Japanese comics and feel funny or not.

```
>1-scene
max_position(2, 2)

o1:
class(character, req = 1)
position((2, 1), (2, 2))
property("with bandage over the eyes")
angle(left)
posture(1, 0, 0)
hand(up)

o2:
position(1, 2)

>2-scene
max_position(3, 3)

o1:
class(character, req = 1)
position((3, 2), (3, 3))
property("with bandage over eyes")
posture(0.3, 0.5, 0.2)

o2:
class(non-character(inanimate          object(fruit("water-
melon"))))
position(1, 2)

o3:
class(none, req = 0)
position(2, 3)
```

Fig. 9. The text based picture model of story 2

- **Deep analysis of exchangeability of objects in existing comics**

Exchangeability of objects is essential part of understanding comic stories. To analyze the exchangeability of objects, we have to investigate the reaction of users who watch the generated stories by replacing some objects to other objects in existing comics.

If we solve those problems completely, we will create conversational system between humans and computers, interactive picture book and drawing application which displays comments about user's drawing.

6 Acknowledgment

This work was supported by Grant-in-Aid for JSPS Fellows (Grant Number: 25.10978).

References

[Kurlander, D. *et al.* 1996] D. Kurlander, T. Skelly, D. Salesin : *Comic Chat*, SIGGRAPH 96 Conf. Proc., pp.225-236, (1996)

[Herot, C. F. 1976] C. F. Herot : *GRAPHICAL INPUT THROUGH MACHINE RECOGNITION OF SKETCHES*, Computer Graphics (Proc. SIGGRAPH) Vol.10, Issue.2, pp.97-102, (1976)

[Eitz, M. *et al.* 2012] M. Eitz, J. Hays and M. Alexa : *How Do Humans Sketch Objects?*, ACM Trans. Graph. (Proc. SIGGRAPH), vol.31, pp.44:1-44:10, (2012)

[Fu, H. *et al.* 2011] H. Fu, S. Zhou, L. Liu, N. J. Mitra : *Animated Construction of Line Drawings*, ACM Trans. Graph. (Proc. SIGGRAPH ASIA) vol.30 No.6, (2011)

[Walther, D. B. *et al.* 2011] D. B. Walther, B. Chai, E. Caddigan, D. M. Beck and L. Fei-Fei : *Simple line drawings suffice for functional MRI decoding of natural scene categories*, Proc. National Academy of Sciences, Vol.108, No.23, pp.9661-9666, (2011)

[Ito, K. *et al.* 2007] K. Ito, M. Matsuda, M. J. Durst and K. Hasida : *SVG Pictograms with Natural Language Based and Semantic Information*, SVG Open, (2007)

[Gross, M. D. *et al.* 2000] M. D. Gross, E. Y. Do : *Drawing on the Back of an Envelope : A Framework for Interacting with Application Programs by Freehand Drawing*, Computers & Graphics, Volume 24, Issue 6, pp. 835-849, (2000)

[UENO, M. *et al.* 2012] M. Ueno, N. Mori, K. Matsumoto : *Picture Information Shared Conversation Agent: Pictgent*, Distributed Computing and Artificial Intelligence Advances in Intelligent and Soft Computing, Springer, Volume 151, pp.91-94, (2012)

[Arai, N. *et al.* 2012] N. Arai, T. Matsuzaki : *Can a Robot Get into the University of Tokyo? : The Artificial Intelligence Project at the National Institute of Informatics* ,Journal of the Japanese Society for Artificial Intelligence, Vol.27, No.5, pp.463-469, (2012)

[Gamma, E. *et al.* 1995] E. Gamma, R. Helm, R. Johnson and J. Vlissides : *Design Patterns: Elements of Reusable Object- Oriented Software*, Addison-Wesley, (1995)