A collaborative platform supporting graphic pattern design and reuse of design knowledge

Wu Yingfei, Tang Mingxi, John Hamilton Frazer,
Design Technology Research Centre
The Hong Kong Polytechnic University, HKSAR, PRC
e-mail:sdwyf@polyu.edu.hk

Sun Shouqian, PanYunhe
Modern Industry Design Institute, Zhejiang University, Hangzhou, PRC

Abstract

A pattern generation system based on an object-oriented pattern knowledge representation method developed previously in Zhejiang University is introduced. We evaluate the effectiveness of this system from authors' current new research perspective in the Design Technology Research Centre of the Hong Kong Polytechnic University. We then put forward a new platform model to support graphical pattern design emphasising ethnic minority culture. The purpose of building this new platform is to increase the ability of the system in supporting collaborative design. New methods are developed for representing patterns as well as the knowledge about how they can be reused in design applications. In this paper the method for element dynamic classification, knowledge representation of pattern design and the system architecture are introduced.

1. Introduction

In this paper, we will first discuss a pattern generating system using pattern knowledge representation and synthesis reasoning and then evaluate the effectiveness of this system. Then based on the evaluation we will put forward a new platform model called JICPGS (Java-based Integrated Collaborative Pattern Generating System) based on a graphics and image database using a multi-level and dynamic classification method. This new model will improve the original method of pattern of knowledge representation used previously by the authors in Zhejiang University. Through this new platform model, different users at different sites can input patterns into the database, so as to allow designers to use this database later on without having to worry about the management of diversified knowledge and information. We intend to apply this model to pattern design with minority cultural style in China's Yunnan province.

Yunnan has 26 ethnic minority groups. There are a lot of typical patterns with minority styles. These artistic resources have now gaining wide interests nationally and internationally, for they provide inspirations for novel and creative design of textiles and products. But the usage of these materials is still very limited because these resources are still not managed scientifically and systematically. This situation will be greatly improved if a distributed and collaborative platform can be built.

In the second section of this paper, we discuss the mechanism of pattern generating system in our previous research work. In the third section we will analyse the shortcomings of the old system and put forward a new model to solve these problems. In this part, a dynamic interactive multi-level method to manage images and graphical elements according the real needs of designers is introduced. In the fourth section, we discuss the interactive and collaborative design model including collaborative working method and collaborative management. In the fifth section, we discuss an agent based program, which allows the graphical and image database to be extended.

2. Knowledge representation and pattern generation system

Designing a pattern is a process with scarce constraints in pattern layout, colour and element selection. Design rule and experience of designers are often difficult to represent structurally in a computer system. In previous research on computer based intelligent pattern generating system we investigated the issue of how to represent knowledge of pattern and utilise it to generate patterns using synthesis reasoning. The prototype was introduced [1]. In that system, we considered that the there are three basic elements within a pattern:

1. Graphic and image elements adopted in a pattern,
2. Pattern layout, and
3. Pattern background colour style.

We define the knowledge representation of a pattern in the following way:

Layout := <Group> |…| <Abstract Element>
Group :=<Abstract Element>|…| <Abstract Element>
Pattern := <Element> | …| <Element >

An abstract element is an element which has the properties such as position, angle, stretching parameters in the layout definition. A group is a definition that shows the mutual relationship of different abstract elements in the layout definition. A group consists of certain number of abstract elements and all the abstract elements in the same group will select same pattern elements during the stage of pattern generating.

Based on the above definition, here we put forward the definition of a new template. This template is the structural definition of the selection range of an abstract element in the layout definition. A template defines a design problem space within which a layout can be gradually developed into the final design result. A designer can use it to synthetically describe pattern design knowledge. The abstractness it beholds decides that we can use it to generate different patterns with similar styles using synthesis reasoning. The layout definition is more abstract than the template definition. So we have two levels of pattern design knowledge to support different designers who may have different roles in the whole design process. Figure 1 shows the whole process of pattern generation.

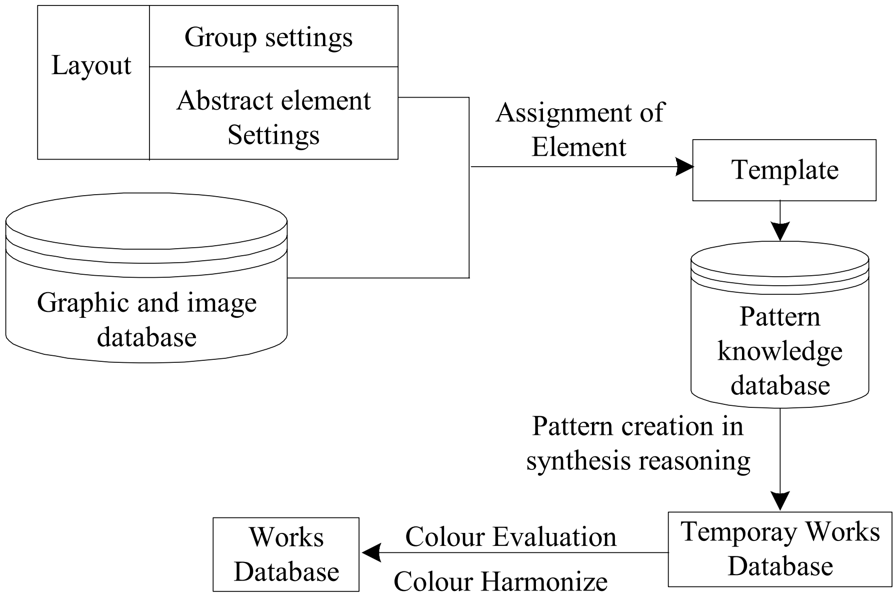

Figure 1: The process of pattern generation

We have used this system to generate images of Dunhuang style, which represents a significant collection of artistic patterns and painting along the ancient silk road.

3. Evaluation of the system developed

Through the long time real use of the system we found that there are three main shortcomings in our system as described in section two. First it is inadequate for complex searching method that designers really need. Second the system cannot handle polymorphism. Third it is difficult to let users share the same resources of the system concurrently. We will discuss the issue of how to solve first problem in section 3.1, the second problem in section 3.2, and the third problem in section 4.

3.1 The Need for a multi-classification method for pattern design knowledge

A pattern has many properties in our database. For example pattern with typical minority style has properties such as dynasty, minority group, usage, colour style, representation method, image quality and patent content and so on. Designers need different kinds of element classification methods to form a classification tree to design his template knowledge when they are facing with different design problems. We need to develop a new method for graphic element classification. This new method uses multi-classification to categorise elements in the database. Thus a designer can perform complex search and select the priority of each classification element to build the classification tree, which is suitable to his/her design project at hands. Here are the details of our classification and searching method:

<Classification set C > =
{(Classification element a),
(Classification element b), … ,
(Classification element n)}

<Searching method> =
<Classification element a> [And |Or] <Classification element b> …
<Classification method> =
<Classification element m> | <Classification element n> | <Classification element o> …

According the classification method discussed above, the sequence of classification element can be selected according the corresponding design task. Thus we can generate different pattern classification trees to suit the needs in different design projects.

All pattern- Classification element m
|- Classification element n
|- Classification element o
|- Classification element p

OR

All pattern- Classification element o
|- Classification element m
|- Classification element p
|- Classification element n

3.2 Polymorphism of pattern and dynamic classification

We also found that designers have different feelings facing with same pattern or pattern element because of their differences in professional knowledge, background and understanding. Here we call this phenomenon as the polymorphism of image and graphics. Our model must have the function for dynamic classification in order to address this issue. The same pattern can be linked to different categories in our database. For example, if we have a pattern, one designer's feeling is that the pattern is a lively pattern of a particular minority group and the other designer's feeling is that the pattern is a decorative pattern of a different minority group, then when we design the database we must anticipate such perception and satisfy the contextual needs of designers. The figure below is a method used in our database design for solving this problem.

Pattern attribute table

Record 1	Attribute 1	Attribute 2	Attribute n	File No
Record 2	…	…	…	

File record table

Record 1	File No	File Content
Record 2	…	…

Figure 2 Realisation of polymorphism

Through this method we can realise the polymorphism of graphics and images in the whole database.

4. Interactive collaborating model for pattern design

A design concept can be well represented with the use of layout and template. But it is still difficult for a designer to finish the whole design process by himself/herself and the mutual communication between designers is not convenient. A more advanced method based on a collaborative environment to allow designers to finish their design tasks collaboratively and effectively is needed [2].

There are two modes of collaborative design. One is asynchronous mode and the other is synchronous mode. In asynchronous mode a participant begins to work after the previous one passed the task to him/her, and only after he/she finishes the work can the later participant begin to work on the task. In synchronous mode all collaborative members work on the same project at the same time, and the modifications edited by each member will be broadcasted to others in real time so that everyone knows what is being modified. Furthermore this mode has two types. One is loosen coupling and the other is close coupling. In loosen coupling mode all the members work independently most time and they share some resources temporarily. In close coupling mode all members share the same views, resources, cursors and files most time [3, 4].

In the process of pattern design, the designers probably have multiple working modes so generally designers will need two modes working together. Here we will mainly discuss synchronous mode and how we realise the interactive design in a close coupling mode [5, 6].

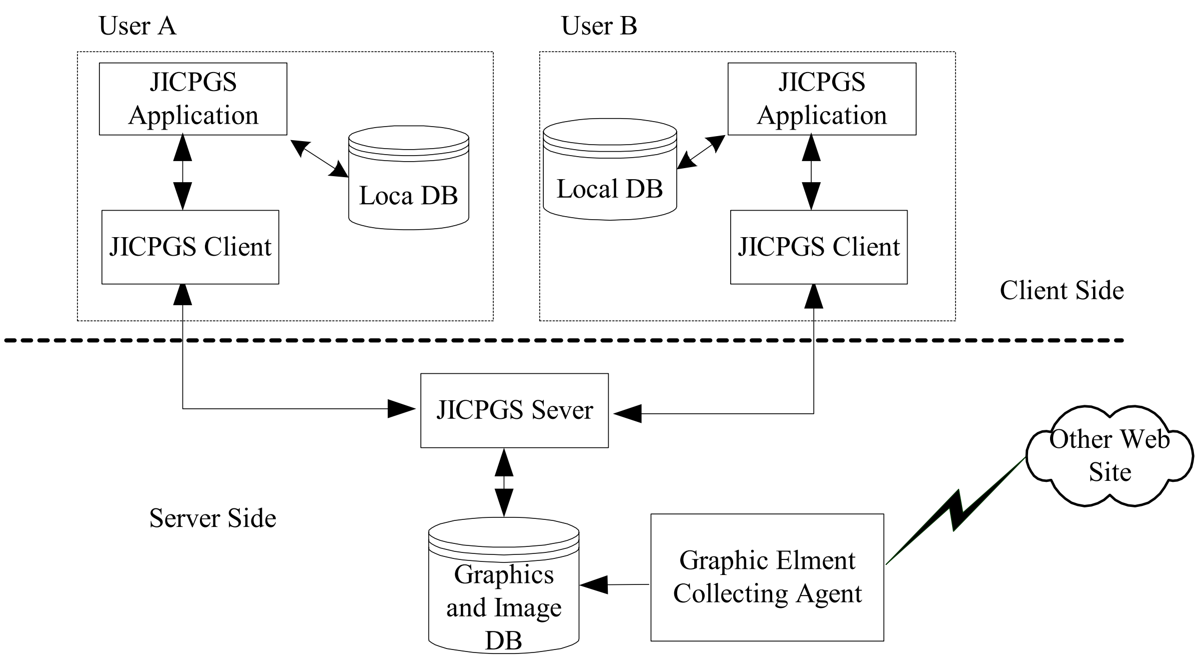

Figure 3: New system architecture

Our platform (JICPGS) includes five main parts: JICPGS applications, JICPGS client, JICPGS sever, database server, and image and graphics collecting agent. Figure 3 shows the system architecture.

4.1 Selection of collaborating mode

Short system response time is an important factor of an interactive system. In our system users can finish the whole design process just using actions such as drag and drop. The architecture of our collaborative system has two main types. One is the central architecture, and the other is the distributed architecture. We adopt the mixed mode of central architecture and distributed architecture to improve the system efficiency.

The realisation of central architecture is easy. But all actions of the members need to be transported to the server and results have to be calculated by the server before they can be displayed on users' screens. As a result, the response time is very long.

In the distributed architecture each member has a copy of the document, and can keep it identical so the response time is short. But it has the disadvantage for data management. When a new member joins the team the new comer has to inform everyone in the team and calculates the current state according the response from other members.

Our system model will adopt mixed modes as we discussed above so we can manage the files and data in both central and distributed manner. Through central and distributed management of files the response time can be cut down to the minimum. It is also easier to maintain the consistency of the files kept by individuals. The system can still work well if one of the member's computers crashes so the stability of the whole environment can be maintained.

In the design process if a user wants to drag or drop a graphic or image element from one place to another the transportation process often costs a lot of time. This kind of transportation occurs throughout the whole design process so the real time performance will be greatly affected by the bandwidth of the network. It is a key problem that how we can reduce the transportation to the minimal extent to improve the system performance. Through the investigation of designers' habits we find that designers often use only a part of large-scale database according to their design tasks. So at certain times only this part of the database is useful to the designer. We can save certain information during the design process to build personal information so the frequently used graphic and image elements can be stored in the user's client side. If the designer needs some more new elements then the system will transport them to the client side. Thus the efficiency can be greatly improved.

4.2 Collaborative management of system

In the collaborative design process the sharing and privacy of information coexist. In the absolute sharing collaborative mode a user can modify nodes without limitations. All members in the team can modify any node (such as group node and abstract element node). But this mode has disadvantage when members are working independently in different extents. A member of a team should have the way to set the permission for visiting, editing, deleting and so on to manage the nodes created by him/her to prevent from any wrong editing [7].

4.3 Concurrent control of collaborative design knowledge tree

We can use the figure below to represent pattern knowledge according our definition.

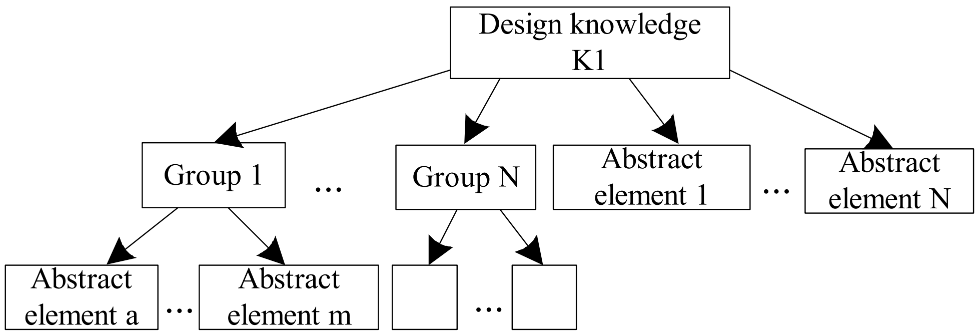

Figure 4: The architecture of design knowledge tree

4.3.1 Locking of directory path

If multi members modify the same node at the same time in our knowledge representation model corruption will occur thus the identity of design knowledge tree is destroyed. We must use concurrent control to avoid this situation.

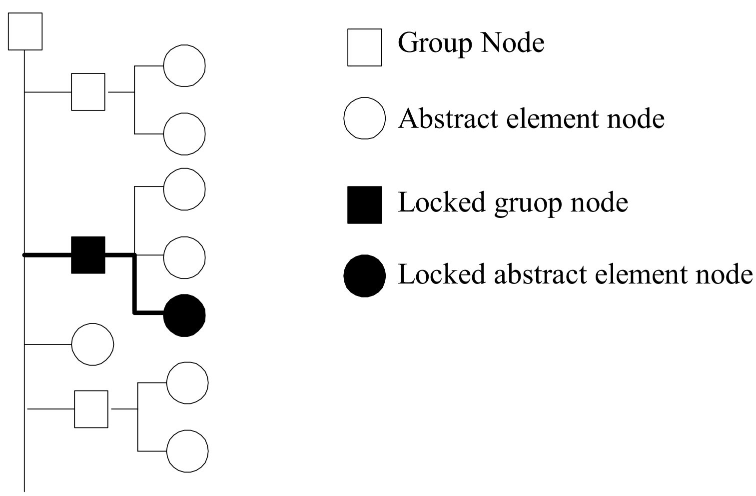

Figure 5: The design knowledge tree with locked directory path

When a member in the team wants to edit an abstract element node he/she has to first get the right that he/she can lock the node to prevent the other members from applying for this node. The other members can only browse this node. To avoid this node becoming an invalid one when other members delete the parent node of this node, the member should trace back to its root node and lock its parent node after getting the editing right of this node. The architecture is shown in Figure 5.

To realise this method effectively, we can set a counter for every group node and abstract element node to show the current state of the nodes. The counter of the abstract element and parent group node is set to false after the application for modifying one node is approved. When the editing procedure is finished these counters are set back to true.

4.3.2 Concurrent control of abstract element nodes

Users will often drag and drop an abstract element to the right position with the right angle, and stretch proportion when he/she wants to add or modify an abstract node. In this period no other user is permitted to modify the node and its parent group node to avoid confusion. The system will lock the node when a user edits the node and unlocks it after the user finishes the editing process.

4.3.3 Concurrent control of group node

There are some basic modification operations for group nodes, such as renaming of a group node, adding an abstract element node into a group node, deleting an abstract element node of a group node, renaming an abstract element node of a group node. Before modifying a node the system will check whether it is locked by other members.

5 Adaptability of element

The aim to research JICPGS is to develop a collaborative design environment within which designers can work together with the support of a large-scale graphic and image database.

To build such an environment a database alone cannot meet designers' needs. The system also needs the function for extending the database automatically so it can learn from the current popular design thinking and the user behaviour.

There are two ways that the system can learn from and improve itself. One way is through the interaction with designers. The other way is through an agent [8]. It is very important to get as many image and graphic elements as possible. With the development of Internet there are many useful resources across the Internet that we don't know. How to use these resources to help designers and integrate them into our platform is a meaningful challenge. Here we will simply introduce an image and graphic element collecting agent. This type of agent can collect elements automatically across Internet. After the collection these elements are not classified and they are simply input into our database without any order. We need another agent that can classify these elements. We will have the evidence when designers use some of the elements to build his design knowledge according certain understanding and usage. So the agent can classify these elements according this information.

6 Conclusions

In this paper we discussed the representation of pattern design knowledge and a collaborative interactive model based on a large-scale image and graphic database. Through this model, multi-level and dynamic classification of pattern elements is supported. The next step of our research is to actually develop ethnic minority pattern design environment through our collaboration with Yunnan Development Centre on the mainland China. In this collaboration, we expect to test our system in close collaboration with designers and artists.

Acknowledgements

This project is supported under a joint PHD supervision scheme between Zhejiang University and The Hong Kong Polytechnic University. We acknowledge the support from Professor Lu Dongming of Zhejiang University in Hangzhou, PRC.

References

[1] Model of image composition and pattern generation based on synthetic reasoning, Zhang Lingli, Lu Dongming, Pan Yunhe, Jian Zhengsan, Journal of Computer Aided Design and Computer Graphics, Vol. 12, No.5,May, 2000

[2] OpenDesign: an open user-configurable project environment for collaborative design and execution on the Internet, Lavana, H., Brglez, F., Reese, R., Konduri, G., Chandrakasan, A., Computer Design, 2000. Proceedings. 2000 International Conference on, 2000, Page(s): 567 –570

[3] Egret: A Framework for Advanced CSCW Applications, Philip M. Johnson, ACM Software Engineering Notes, Volume 21, Number 2, May, 1996.

[4] Theory and applications of computer collaborative working, Shi Meilin, Xiang Yong, Electronic Industrial Press.

[5] JMS: a flexible collaborative environment, Sun Wei, Lee Bu Sung, Yeo Chai Kiat, Internet Workshop, 1999. IWS 99, 1999, Page(s): 195 –202

[6] Implementation of a graphics design framework on the Web, Xiaoyan Xie, Hao Wang, Zhiyong Huang, Computer Graphics and Applications, 2000. Proceedings. The Eighth Pacific Conference on, 2000, Page(s): 421 –422

[7] Floor control for multimedia conferencing and collaboration, Hans-Peter Dommel, J.J. Gacia-Luna-Aceves, Multimedia Systems (1997) 5:23-28

[8] CoDesign - a collaborative pattern design system based on agent, Bo Jiang, Chun Chen, Jiajun Bu, Computer Supported Cooperative Work in Design, The Sixth International Conference on, 2001, Page(s): 319 -323