International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE)
ISSN: 0976-1353 Volume 26 Issue 5 – AUGUST 2019.

FABRIC PATTERN PRODUCTION SYSTEM USING CASE-BASED REASONING

Khin Aye Mar[#1]

[#]*Lecturer, Faculty of Computing, University of Computer Studies (Mandalay), Mandalay, Myanmar*

***Abstract*— Nowadays, people have become very demanding in attractive life of wearing styles. These changes have forced designers to explore diverse range of previous designs. To perform such operation, good knowledge and experiences are needed. Therefore, the effective concept that is used to replace the traditional methods in intelligent system, Case-Based Reasoning (CBR) is presented. This paper is proposed to demonstrate that CBR could be integrated in textile design generation to recommend the best applicant suitable design cases. Database of product designs are stored in the case library. The system starts with inputting details of product descriptions. It is then matched with previous stored cases and retrieved the most similar cases. The nearest-neighbor concept is employed to match cases and the similarity percentages are shown as a result. One of the shown product design is selected to modify or produce new design. Finally, the new textile design is retained in the design database for future references.**

***Index Terms*—CBR Inference and CBR cycle, Similarity measure, the local-global principle**

I. INTRODUCTION

Case-Based Reasoning (CBR) solves new problems by adapting previously successful solutions to similar problems. CBR supports design by reminding designers of previous experiences that can help with new situation [1].

Before a new style of textiles comes into the market, it has to go through the process of market research, design, sample making, try on testing, price negotiation and production. Every stage requires much work and long time. Similar designs exist in one place, so some work will be redesigned once more in the design of new products [2]. This makes improve specific design speed and quality. Hence, storing and retrieving information on previous product designs are the most important tasks to provide both designers of new products and (re)designers of existing products.

In this system, the features of a specific textile design are represented in attribute-value pairs. A similarity table is constructed for local similarity measure of each attribute. The values of the system user entered are considered as a new problem case. It is then matched with all the stored cases in design database by using the local-global similarity measure. As a result, the system outputs the ranked similar cases. Based on the retrieved cases, the system user can modify or reuse it as the design case solution. Whenever the retrieved design case is reused as the current design problem solution, the system counts the reuse time. For case base maintenance, the system user needs to discard old design cases which are not suitable for future use. Finally, the updated new design case is maintained in the design database for later use.

The paper aims at presenting a case-based reasoning system for production of textile designs to support for customer demanded designs. The remainder of the paper is organized as follows. Section 2 presents the background theory, CBR cycle and the similarity measure utilized in this system. Section 3 illustrates the system design. Section 4 depicts the implementation of the system and conclusion is given in section 5.

II. BACKGROUND THEORY

Case-based reasoning (CBR) is a problem-solving paradigm that remembers previous similar situations (or cases) and reuses the information and knowledge about the stored cases for dealing with new problems. CBR systems have intuitive appeal because much of human problem solving capability is experience based, that is, humans draw on past experience when solving problems and can readily solve problems that are similar to ones encountered in the past [3].

A. CBR Cycle

CBR module contains five parts: description of the issues, case retrieval, case reuse, case revise, and case adaption.

(1) Problem Description: User describes the problem to solve in functions, parameters and other elements.

(2) Case Retrieval: Given a description of a problem, a retrieval algorithm, using the indices in the case-memory, should retrieve the most similar cases to the current problem or situation. Adaptation looks for prominent differences between the retrieved case and the current case and then applies formulae or rules that take those differences into account when suggesting a solution.

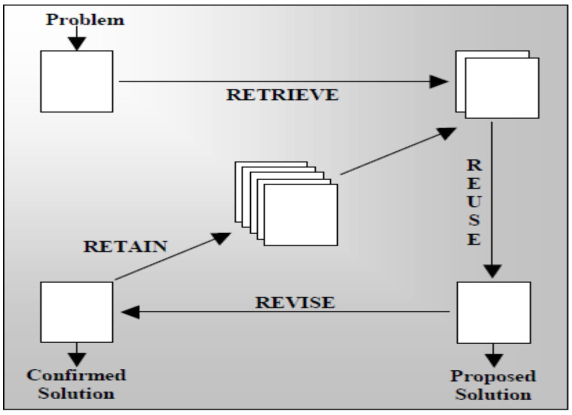

Figure 1. CBR Inference Cycle

(3) Case Reuse: After selecting one or several similar cases, the reuse step tries to apply the contained solution information to solve the new problem. Often a direct reuse of a retrieved solution is impossible due to differences between the current and the old problem situation. Then the retrieved solution(s) have to be modified in order to fit the new situation.

(4) Case Revise: Depending on the employed adaptation procedure, the correctness of the suggested solution often cannot be guaranteed immediately. Then it becomes necessary to revise the solved case. If the revise step fails, the case has to be repaired or a new attempt to generate a valid solution has to be carried out. This new attempt can be realised in different ways.

(5) Case Retain: If the solved case has passed the revise step successfully, a *tested/repaired case* will be available representing a new experience that might be used to solve similar problems in the future. The task of the CBR cycle's last step is to retain this new case knowledge for future usage. Therefore, the new case may be added to the case base.

CBR systems make use of many types of knowledge about the problem domain for which they are designed. Similarity measures include the knowledge about the similarity measure itself and the knowledge used to choose the most efficient organisation of the employed case base and the most appropriate case-retrieval method.

B. *Similarity measure*

A similarity measure is a function Sim:$D_D \times D_D \rightarrow [0,1]$. By computing the similarity between the query, *q* and the case characterizations of the cases contained in CB, the retrieval mechanism has to identify a list of cases, called retrieval result, ordered by the computed similarity values. The number of cases to be retrieved may be specified by an integer value which denotes the maximal number of cases to be retrieved. [5].

C. *The local-global principle*

A similarity measure that can be adapted on a particular attribute-value based case representation is called the local-global principle. According to this principle it is possible to decompose the entire similarity computation in a local part only considering local similarities between single attribute, and a global part computing the global similarity for whole cases based on the local similarity assessments.

A local similarity measure for an attribute *A* is a function Sim_A:$A_{range} \times A_{range} \rightarrow [0,1]$, where A_{range} is the value range of *A*. For unordered symbol types, the only feasible way to represent local similarities is an explicit enumeration in form of a lookup table, called similarity tables. [5]

Table 1. Example: Similarity table for color

	red	green	blue
red	1.0	0.0	0.0
green	0.0	1.0	0.0
blue	0.0	0.0	1.0

The second important part of similarity measures defined according to the local-global principle is attribute weights. They are used to express the different importance of individual attributes for the entire utility approximation.

Global similarity measure is represented by an aggregation function computing the final similarity based on the local similarity values computed previously and the attribute weights defined:

A global similarity measure for *D* is a function

Sim: $D_D \times D_D \rightarrow [0, 1]$, of the following form:

Case Sim (query, case) = $\frac{\sum_{i=1}^{n} w_i * sim_i}{\sum_{i=1}^{n} w_i}$

where, sim_i is similarity of i^{th} feature, w_i is weight of i^{th} feature [5].

III. SYSTEM DESIGN

The CBR works in cyclic form as the following procedures. When the user inputs new problems to be solved, the system first of all searches the case base for the same case or similar cases. If the retrieved case meets with the user needs, reuse it as the current problem solution. Otherwise, modify the necessity parts to produce an optimal solution and output this case. At the same time, store the solution in case base to be capitalized and eventually reused as reference for a new problem.

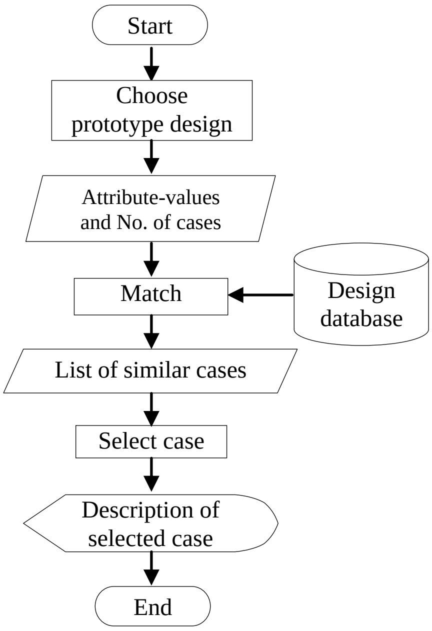

Figure 2. System flow for case retrieval

The proposed system is designed to enhance the reusability of designs and generation of new designs by using the CBR concept. The system allows the user to reuse the retrieved design as solution or make adaptation on this design for production of new design [4]. Therefore, the system generally consists of three main steps named case retrieval, case adaptation and case maintenance. For retrieval of useful cases for customer requirements, the matching of query case and the stored cases in the case base is performed by the similarity measure discussed in section 2.2.

The computed similarity result is ranked and shown in percentage. The user makes selection on one of the retrieved cases and views the detail description of the selected design case. If the user meets with his/her needs, reuse the selected case as the current design solution and increase the reuse times. Otherwise, modify the necessity parts and re-input new attribute values. The new prototype design is made under Photoshop program.

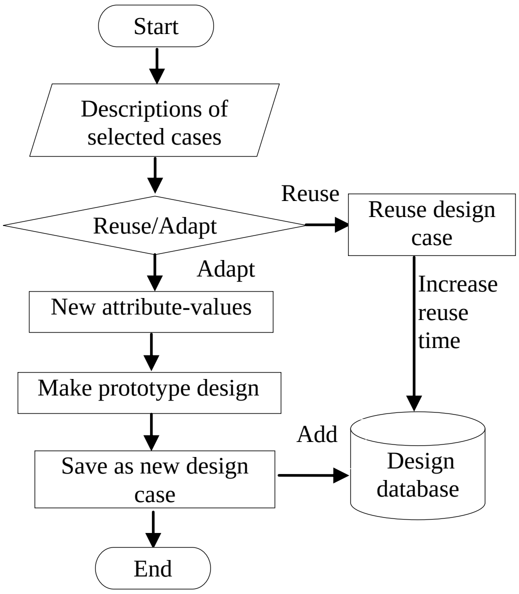

Figure 3. System flow for case adaptation

And then, detail descriptions, the sketch and the sample photo of this new design are stored in the design database as a new design case. The system flow diagrams of retrieval and adaptation processes are illustrated in Figure 2 and 3 respectively.

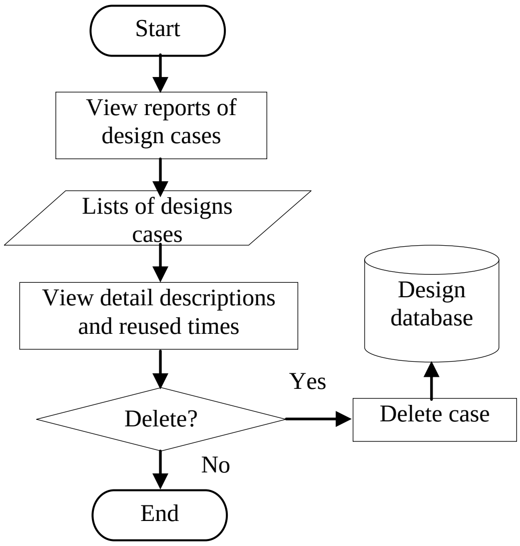

Figure 4. System flow for case maintenance

As the case-base becomes grow, the searching time of the relevant cases gets slow. Therefore, we need to maintain the case-base. This system will use the case addition and deletion policy. Case-addition option of case retain step is described in Figure 3, in which the new design case is added to the design database for future use.

A case base comprises of a large number of design cases with different features. Some are old design cases which have renewable ability for harmonizing with the customer requirements. This type of design cases will be reused according to the demand of aesthetic designs. However, if some design cases are not convenient for later use and the reuse time is low, we need to decide whether to delete it or not. The process of case-deletion is shown in Figure 4.

Table 1 Attribute Dataset

Types	Subtypes
Seasons	Summer Raining Winter Spring
Person	Woman Man Baby Kids Girl
Costume	Shirt Maxi Scrub Coat Blouse Jacket Skirt Ruffle Pant
Cloth	Sweater Cotton Raglan Lace Silk
Color	Red Green Yellow Blue Pink Orange Purple White Black Gray Brown

IV. SYSTEM IMPLEMENTATION

Products and fashion trend knowledge are very important to the new product design. Therefore, the knowledge library must be dynamic in order to update the new fashion trend. In this system, the prototype designs both in digital and in the physical forms are created for descriptions of a design case. The standard textile library includes 10 sample textile prototype designs which are arranged by their distinct patterns. And each has its own attributes of fabric type, fabric color, Costume style, and Sewing type and decoration pattern. The number of cases and the brief information such as the wearing type and styles for each sample prototypes is also provided.

The proposed system is tested by the customer requirements. The first design case is to find an occasional wear for baby textile design which is made of cotton for fabric; costume is short-maxi and decoration pattern is digital. The fabric colors are red. For attribute weights, the user can mark on check box for more important attributes. The system user can fill partial descriptions for design problem case. All the user information is matched with the stored design cases. The ranked similarity percentage of the matching is shown as the result.

The user then browses the ranked similar cases and makes selection on the top-level one. The retrieved design case is decorated with sequence, the common popular design, especially for celebrities and grooms.

For adding new design case in the design database, the system user can substitute the attribute values or input new values. And then, the new prototype design is made. Not only new design case, the system enables the user to modify the retrieved design. The new successful design case is stored in the design database.

V. CONCLUSION

Case-based reasoning concept is reviewed and applied in textile design production. A specific design case has better ability to improve the design efficiency than clear design rules. The reusable and adaptable ability of CBR is mainly utilized in this system. The work flow of the system is discussed with case study. The system is linked to Photoshop program for sample making of new prototype design so as to perform design modification by the system user according to the feedback from the customer demand. As a result, the system user can get an exposure to the past remarkable designs to produce new innovative textile designs.

REFERENCES

[1] Mary Lou Maher and Andres Gomez de Silva Garza, "Case-Based Reasoning in Design", AI in Design, IEEE, 1997, pp. 34-41.

[2] Tan Juanjuan, Peng Wenli, Zhang Wenni and Wan Pengbo, "Reusability of Shoes' Design Information Based on KBE and CBR", Proceedings of the 2009 International Symposium on Information Processing (ISIP'09), Huangshan, P.R. China, August 21-23, 2009, pp. 302-305.

[3] Cunningham P, del Pobil AP, Mira J, Ali M. CBR: Strengths and weakness. In: Proc 11th Int Conf on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, Lecture Notes in Artificial Intelligence 1416, vol2; 1993, pp 517-523.

[4] Suthep BUTDEE, "Casexpert System for Leather Good Fashion Design", Global Congress on Manufacturing and Management, Santos, BRAZIL, November 19-22, 2006.

[5] Armin Stahl, "Learning of Knowledge-Intensive Similarity Measures in Case-Based Reasoning", PhD Dissertation, University of Kaiserslaitern, August, 2004.