Synthesizing Method for Subjective Similarity Retrieval Service Adapting for Given Multimedia Contents and Subjective Criteria for them

Akihiro OGINO
Graduate School of Science and Engineering
CHUO University
Tokyo, Japan
ogino@indsys.chuo-u.ac.jp

Toshikazu KATO
Faculty of Science and Engineering
CHUO University
Tokyo, Japan
kato@indsys.chuo-u.ac.jp

Abstract
This study addresses synthesizing of subjective similarity retrieval services, which provide contents based on individual subjective criteria about similarity among contents. This paper proposes a design methodology of a software platform for synthesizing subjective similarity retrieval services based on it. Specifically, this study defines commands and its syntax of platform for easily and correctly designing and developing the service.

1 Introduction
Recently, subjective similarity retrieval services, which have subjective evaluation criteria models of individual to contents, have been developed. These services provide similarity contents based on model, which is made by supervised learning, to every user.

Existing subjective similarity retrieval services are developed ad-hoc and there is not an infrastructure service, in a survey paper. Therefore, we are difficult to design and develop services without an in-depth knowledge and skill. For solution to these problems, we have proposed a framework for subjective evaluation criteria modelling and for retrieval contents based on individual models [1]. In this framework, we have thought that people hierarchically perceive similarity and interpret the meaning of multimedia contents.

In this paper, we propose a software platform for subjective similarity retrieval services based on our framework.

2 Framework for Modelling and Retrieval
For a design and development of subjective similarity retrieval services, we have proposed a framework for subjective evaluation criteria modelling and for retrieval contents based on individual models.

2.1 Framework for Modelling Subjective Evaluation Criteria
2.1.1 Physical level process
The physical level is the pre-perception process. In this level, we think that people perceive physical feature of contents, such as photographic images, graphic symbols and 3D-objects.We think that people evaluate similarity degree of contents based on their physical graphic features.

2.1.2 Physiological level process
We consider that people perceive local contrast features as well as global features on brightness and colours [2, 3] in early stage of content's perception process. Such a mechanism originates in the lateral inhibition and it is commonly accepted various kinds of contents.

By using features of this level, in which we call physiological features, as evaluation criteria to retrieve contents, we can find out similarly contents.

2.1.3 Psychological level process

The psychological level corresponds to the subjective similarities among contents for individual. There are differences in each user's dominant factors of graphical features or local contrast features in hierarchical classification according to his experience and knowledge. Therefore, we calculate a correlation between weights of subjective attention features, which are made by supervised learning in order to simulate subjectively perception process.

2.1.4 Cognitive level process

We often give unique impressions by words, even when viewing the same contents. The cognitive level process corresponds to the correlation between the concepts and the contents features on physical, physiological or psychological level.

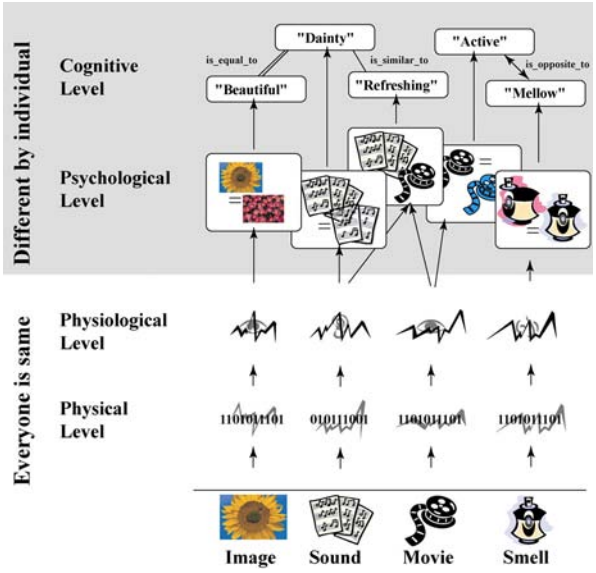

Fig.1 Hierarchical Modelling Process of *KANSEI*

We can simulate each user's interpretation process using individual models, which are found by canonical correlation among them through statistical analysis and learning, on retrieval services.

2.2 Framework for retrieval using subjective models

We organize a subjective similarity retrieval process as shown Fig 2:

(1) Measurement Process: This process calculates various features from contents or quantifies subjective evaluations about contents for learning.

(2) Leaning Process: This process makes individual models, which simulate individual's evaluation criteria of contents, using features of contents and quantified subjective evaluation.

(3) Determination Process: This process determines contents suitable for individual evaluation criteria about similarity or impression using individual models.

3 Software Platform for Subjective Similarity Retrieval Process

We design and develop a software platform for subjective similarity retrieval services based on our framework. We can design services through our platform, which has three mechanisms:

(1) Management of components: We design a common data model to manage components based on OEM (Object Exchange Model)[4], which is one of semi-structured data model. We extend a data type of OEM for a management of contents and program, such a Java and C, C++.

(2) Design of service: We defined commands and it's syntax for service design on our platform.

(3) Execution of designed service: Our platform synthesizes PROGRAM based on a design specification and makes a service process. Our platform automatically applies DATA and MODEL to the service process if user assigned them in a platform command.

3.1 Management of Components

3.1.1 Components

An aim of this study is to define a methodology to design and develop subjective similarity retrieval services based on our framework easily and correctly. For the purpose, we need to define a requested and design specification for them and to clarify a structure of them.

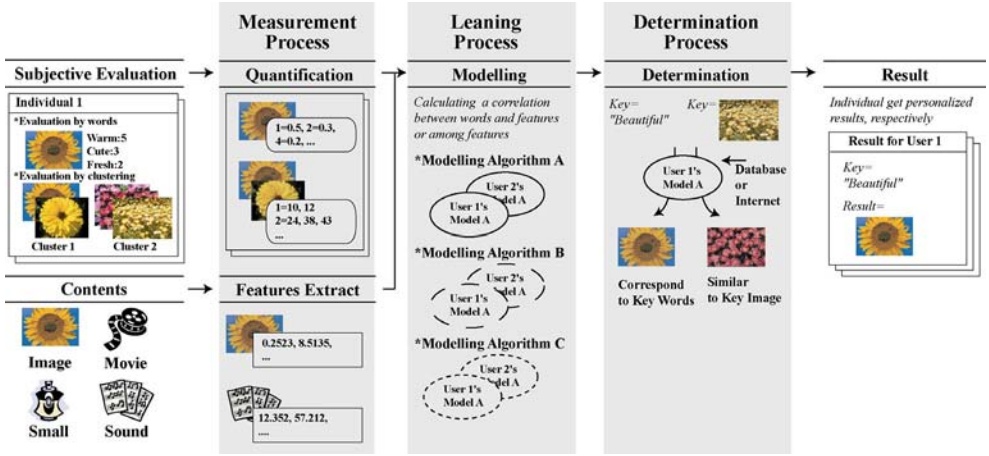

Fig.2 KANSEI Retrieval Process

Table.1 Expiation of PROGRAM

Kind	Explanation	Example
Measurement	To calculates various features from contents or quantifies subjective evaluations about contents	For Feature Calculation -Colour Histogram -Autocorrelation For Quantification *Subjective Evaluation
Learning	To model individual's evaluation criteria using features and quantified evaluation	-Principle Component Analysis -Discriminant analysis
Determination	To determine contents suitable for individual criteria about similarity or impression , by calculating distance between key and objects	-Euclidean Distance -City Block Distance

We tried to organize components of the service based on our framework in this study. We define data structure of components in Appendix A.

(1) DATA: We define contents, features and evaluation data, which we use for a modelling and retrieval, as DATA.

(2) MODEL: We define data, which shows simulated individual evaluation criteria by linear or nonlinear methods, as MODEL. We can think eight models in our framework.

(3) PROGRAM: We define a program for modelling and retrieval processing as PROGRAM. We use statistical methods, such as discrimination analysis and canonical correlation analysis, to make a model. We divide these programs to three categories based on process of a retrieval framework as shown Table 1: Measurement, Learning and Determination.

If we can describe a relation of DATA, MODEL and PROGRM based on our framework easily, we can design and develop a subjective similarity retrieval services. We have to manage various kinds of components for design and develop of services. However, numbers of attributes for components are not a static, especially PROGRAM has many arguments to process, respectively.

Therefore, we used semi-structured data model, which is has Id, Label, Type and Vale as attribute of component, to manage components because it is a self-descriptive and nested structure and it can manage different type of structured data.

```
CREATE SYSTEM "Psychological(Model 3) System"
  /* Program of module 1 = ID:65 on database */
  {Physical-MODELING=[53, SET, %][%, image, CSV:ppm][#1, file, CSV:P-1.stringP-other.double]}
  /*Program of module 2 = ID:76 on database*/
  {Psychological-MODELING=[76,SET,%][%, file, CSV:string][#2, file,CSV:P-1.stringP-other.double]}
  /*Program of module 3 = ID:102 on database*/
  +[102,SET,%][%, file, #1][%, file, #2][#2, file,CSV:P-1.stringP-other.double]}
  /*Program of module 3 = ID:35 on database*/
  {Psychological-RETREIVAL=[35,SET,%][%,file,#2][%,file,CSV:P-1.stringP-other.double][#3,file,CSV:P-1.stringP-other.double]}
```

Fig. 3 Platform Command

3.2 Design of services

We defined five commands and it's syntax for designing services on our platform in Appendix B.

We design and develop a service using these commands like follows:

(1) We insert data or program into a platform for design. INSERT is a command for insert components into a platform. We can insert PROGRAM, which is made by Java, C and C++, and DATA though this command. If we had mistake on an insert, we can delete it by DELETE.

(2) We create a service based on a design specification. CREATE is a command for design of service by synthesizing PROGRAM. We can design a service by describing commands based on syntax. We use SELECT to create a CREATE command. We retrieve PROGRAM from database of our platform using the command.

(3) Finally, we execute a designed service on our platform. EXECUTE is a command to process designed service by a CREATE command.

4 Case Study

In this section, we describe a method of requirement and design specifications and evaluate a developed service based on it using our platform. We assume that users of platform are researchers, which have knowledge about a modelling and retrieval of subjective evaluation criteria.

We use a scenario for explanation platform as follows:

"I want to make a subjective similarity retrieval service about art images, which retrieve contents similar to key's contents based on individual criteria."

We convert this design specification to a platform command to develop a service corresponding to Physiological Level.

1. We divide a service process to modules, which are same PROGRAM or a combination of PROGRAM. In this example, we can divide four modules from this service process.
2. We create a platform command by synthesizing based on a relation of module programs. We can write CREATE command based on a process of modules and framework.
3. We can execute this command on our platform.

5 Conclusion

In this paper, we proposed a conceptual modelling methodology and a software platform for subjective similarity retrieval services.

We organised a framework of modelling and retrieval of subjective evaluation criteria. In this methodology, we showed hierarchical process of subjective evaluation about perception and interpretation of contents. We clarified needed components, which are DATA, MODEL and PROGRM, in processes of modelling and retrieval of subjective evaluation criteria.

We developed a software platform for subjective similarity retrieval services based on our framework. Our platform has five commands for design and development of services. We insert components into platform easily and crate a service by synthesizing programs, which are selected

from database. In the last result, we execute designed service process as a command on our platform.

6 Reference

1 T. Kato. Computational Modeling of *Kansei* Processes for Human-centered Information Systems. Proc. Of *KANSEI* 2001.
2 L. Spillmann and **J. S. Werner**. Visual Perception. Academic Press, 1990.
3 M. Tada, **S. Soda** and **T. Kato**. Automatic Classification and Analysis Facility for Similarity Retrieval of Design Objects. Proc. of 6th Asia Design Conference, Oct. 2003.
4 D. Quass, A. Rajaraman, Y. Sagiv, J. Ullman, and J. Widom. Querying Semistructured Heterogeneous Information. In International Conference on Deductive and Object-Oriented Databases, 1995

Appendix A (Data structure of Components)

<Components> ::=< Symbol > < Kind D | Kind M | Kind P> < = (equal)>
< Name | Model Name >< Information >
< Symbol > ::= D(Data) | M(Model) | P(Program)
< Kind D >::= Subjective Evaluation | Quantification | Contents | Features for modelling |
Features for retrieval |Results
< Kind M >::= Physical | Physiological | Psychological | Cognitive
< Kind P >::= Measurement | Learning | Determination
< Name > ::= Variable-length string for an index of the components
<Model Num>::= <Model 1[n]8>
< Information >::=<Attribute> <Value>
<Attribute>::= Variable-length string for an expiation of the components
<Value>::= Variable-length string for values of the attributes

Appendix B (Syntax of platform commands)

<Command> :: = INSERT <Object> INTO <Formula> |
CREATE / EXECUTE <Object> <Name> <Formula> |
SELECT / DELETE <Object> WHERE <Formula>
<Object>::= DATA | MODEL | PROGEAM | SYSTEM
<Name>::= Variable-length string for an expiation of the object.
<Formula>::=< { (braces)> <Process> < = (equal)> < [(brace)> <Property A> [<Property B>] <]
(brace)> < } (braces)>
<Process>::= <Level> < - (minus) > <Processing>
<Level>::= Physical |Physiological |Psychological | Cognitive
<Processing::= Modelling | Retrieval
<Property A>::= < [(brace)> < Id > <Type> <Value> <] (brace)>
<Property B>::= < [(brace)> < "%" > <Type> <Value> <] (brace)>
<Id>::= Unique variable-length identifier for the object.
<Type::= The data type of the object's value.
Each type is any one of an atom type (such as integer, etc), images type (such as jpg, ppm, etc), real-world data type (such as time, date, etc), program type (such as Java, C, etc), or the type set.
<Value>:: Variable-length string for an expiation of the object.