Design and Implementation of a Concept-based Image Retrieval System with Edge Description Templates

J. H. Choi a *, S. H. Park a, S. J. Park a

a Electronics and Telecommunications Research Institute, 161 Gajeong-dong, Yuseong-gu, Daejeon, 305-350 KOREA

ABSTRACT

In this paper, we design and implement a concept-based image retrieval system using feature information, more specifically, edge histogram description. The general edge histogram framework is a novel index mechanism which allows us to describe a content of images. However, there is a significant drawback in the framework that it can not accommodate a concept-based retrieval. Even if images are only conceptually related with user queries, it may be capable of proving them to be irrelevant since their features can be different each other. Our system adapts an edge histogram descriptor and includes a knowledge used for capturing concepts from images. In the knowledge base, a concept is expressed as some of templates, which can be described by common edge histograms for the images to represent the concept well. The templates can be generated by clustering the training images related with a concept. Consequently, since an image can also be matched with some of the templates, our system is able to support an automatic mechanism for indexing the image with the concept. The indexing mechanism enables users to retrieve the images related with a query which is formulated with their intended concepts. In addition, we also demonstrate that our concept-based approach makes a favorable comparison with an approach based on color or edge features.

Keywords: Image Retrieval, Edge Histogram Descriptor, Knowledge Base, Template

1. INTRODUCTION

Currently, with the construction of digital libraries containing a huge volume of images in Internet, many techniques for indexing and retrieving images from them is developed continuously. In general, the techniques may depend on the model describing images considerably [1][5]. There are two models for indexing images; annotation-based model and feature-based model [6][14]. The former is based on the annotations which can explain their contents with natural language description [10]. The latter is based on primitive features which are contained in images, such as color, texture and shape [11][12].

The annotation-based model has the advantage that it fully explores the detail of the information contained in an image. It provides a way to access the image by means of its semantic content rather than just by its visual content like color or texture [9]. Therefore, it is possible to support a concept-based image retrieval because user’s queries can be evaluated conceptually [21]. That is, it enables users to search images which are indexed by concepts described at his/her queries. Most of image retrieval services may employ this model due to its simplicities for realizing a concept-based retrieval, such as Yahoo, AltaVista and Lycos.

However, the limitation of current technologies makes it impossible to describe the image annotation automatically [4][14]. Image annotation may be usually performed by a manual process which has several drawbacks: The cost is high as it is time consuming. Unfortunately, the volume of images is so huge that it can not accommodate their manual annotation. In other hand, the annotation may be biased and limited by domain experts, because it may be differently described according to

* Corresponding Author: e-mail jhchoi@etri.re.kr; phone +82-42-860-1220
application domains. Thus, it is inevitable to cause the vocabulary mismatch problem between the indexing terms and the query terms [21].

The feature-based model can perform fully automatic indexing for a massive volume of images by extracting the features of their content. The main features of images used for indexing and retrieving are colors, textures and shape [1][13][22]. It may retrieve the relevant images matched with features of the image presented as user queries. Typically, PHOTOBOOK would retrieve the images indexed by similar features with user’s query, which is described by an example image [7][17][18]. In order to also retrieve the relevant images taking layouts of colors and shapes similar with the query, QBIC can exploit an image sketched by users as a query [9][15][20]. Therefore, it is the advantage that this model can automatically index a massive volume of images by their features [8][19].

However, it can not evaluate user’s query conceptually, since the indexing is limited only by the feature information [6][14]. For example, when being intended to retrieve the images related with a concept ‘motor vehicle’, users can not sketch nor present a query image to simultaneously stand for ‘bus’, ‘truck’, ‘car’, etc. In addition, since a car also has the various appearances according to view points, only an image query can be not enough to represent user’s intention for retrieving image conceptually related with it.

In this paper, we design and implement a concept-based image retrieval system, which can make a supplementation of the drawbacks for two retrieval methods respectively. Our system differs from other feature-based image retrieval systems in that it can make the index based on concepts and allows users to query by them. For such a concept-based image retrieval functionality, it adopts an edge histogram descriptor and includes a knowledge used for capturing concepts from images. The knowledge base expresses a concept as some of templates, which can be described by common edge histograms for images to represent the concept well. The adopted edge descriptor expressions are known as international standards of MPEG-7 [2]. The templates can be generated by clustering the training images related with a concept. Consequently, since an image can also be matched with some of the templates, our system is able to support an automatic mechanism for indexing the image with the concept. The indexing mechanism enables users to retrieve the images related with a query which is formulated with their intended concepts. In addition, our system also provides a retrieval approach based on color and edge features for easily comparing it with the concept-based approach.

The remainder of this paper is organized as follows. Section 2 reviews edge histogram descriptors to efficiently express the edge feature of images. In addition, a knowledge base is defined to specify a concept with them. Section 3 explains how to index and retrieve the images related with concepts, when employing our knowledge base. Section 4 demonstrates the implementation of our image retrieval system and compares our approach based on concepts with one based on edge features. Finally, conclusions and further researches follow in section 5.

2. KNOWLEDGE BASE

A knowledge base employed for our image retrieval system is consisted of concepts, templates and relationships among them. That is, a concept may take relationships with templates to stand for the common edge feature of images, which may discriminate the concept considerably. In this section, we introduce a technique, called as edge histogram description, to effectively describe the edge feature of images. It is also appropriated for describing templates for a concept. Additionally, we will also explain structures and constructions of the knowledge base in detail. In our system, the knowledge base is a core component to index and retrieve the images by concepts.

2.1 Edge Histogram Description

Edges in images are an essential feature to effectively represent their contents. In general, it is described by edge histograms, which enable the frequency and the directionality of their brightness changes to be represented uniquely. MPEG-7 employs descriptors as an international standard, which can specify edge distributions of images with the histograms. This Edge Histogram Descriptor (EHD) can express only the local edge distribution in an image. That is, since it is important to keep
the size of the descriptor as compact as possible for efficient storage or transmission, the normative MPEG-7 edge histogram is designed to contain only 80 histogram bins describing the local edge distribution. These bins are the only standardized semantics for MPEG-7 EHD. However, it may be insufficient to only use the local histogram bins for representing global features of the edge distribution. In order to improve its expressiveness, EHD has employed semi-global and global edge histogram bins which reflect global edge distribution of images. In our system, the EHD is also used to represent templates taking the relationships with a concept and to evaluate the similarity between images and templates.

For extracting edge features for an image, it is divided into sub-images, called as blocks. A block is characterized by generating a histogram of its edge distribution. There are 3 kinds of block: global, semi-global and local block. The histograms for each block could also represent the occurrence frequencies for 5 types of edge, called as bin: vertical, horizontal, 45-degree diagonal, 135-degree diagonal and non-directional edge bin. Five types of edge are depicted in Figure 1. The local blocks are generated by dividing an image into 4×4 non-overlapping blocks. We can also produce 13 semi-global blocks by combining 4 successive local blocks. Therefore, an image is represented by 30 blocks (1 global block, 13 semi-global blocks and 16 local blocks) containing 150 bins (30 blocks × 5 bins/block). The global and semi-global block bins may considerably reflect edge distributions for the whole image [2].

For extracting edge features for an image, it is divided into sub-images, called as blocks. A block is characterized by generating a histogram of its edge distribution. There are 3 kinds of block: global, semi-global and local block. The histograms for each block could also represent the occurrence frequencies for 5 types of edge, called as bin: vertical, horizontal, 45-degree diagonal, 135-degree diagonal and non-directional edge bin. Five types of edge are depicted in Figure 1. The local blocks are generated by dividing an image into 4×4 non-overlapping blocks. We can also produce 13 semi-global blocks by combining 4 successive local blocks. Therefore, an image is represented by 30 blocks (1 global block, 13 semi-global blocks and 16 local blocks) containing 150 bins (30 blocks × 5 bins/block). The global and semi-global block bins may considerably reflect edge distributions for the whole image [2].

2.2 Representation of Knowledge Base

Knowledge base is constituted of two components: concepts and templates. A concept takes the relationships with a set of templates, which may represent the common edge features of images to discriminate the concept well. The templates are also described by the same structure as edge histogram descriptor. We could formally define it as the following notation.

\[
\text{KB} = \langle \text{C}, \text{EDT} \rangle
\]
Here, C is set of concepts and EDT is set of edge description templates for all concepts. The set of templates related with a concept \( c \) is denoted as KB\( (c) \). That is, when a template \( t \in \text{EDT} \) takes a relationship with a concept \( c \in C \), we can be denoted as \( t \in \text{KB}(c) \). For example, Figure 3 depicts the representation of set of edge description templates KB\( (c) \) for a concept \( c='car' \). It is formalized as KB\( (c)=[t_1, t_2, t_3, \ldots] \) for \( t_1, t_2, t_3 \in \text{EDT} \), which may be considered as the common edge histogram descriptors for images related with a concept \( c \).

The knowledge base, specifying the relationships among concepts and edge description templates, has been constructed as following 4 steps. First, we should collect training images, which can stand for a concept as concretely as possible. It is possible to gather many of candidate images from Internet search engines such as Google, Yahoo and Altavista. A number of the images existed in their database may be annotated and classified by domain experts according to various concept categories well. For training concepts, we use only the images which are seldom background out of them, since the concept taking the relationship with EDTs should represent not background, but objects included in the images. In other hand, since some of current researches are advanced for eliminating the background of images, we are able to adopt one of them. However, they may unsatisfy us for not separating the boundary of objects included in the image accurately. Second, an edge detector extracts histogram descriptors from the training images. For representing their normalized EHD, we make their scale fixed by the size of 100×100 pixels. Third, our system performs the clustering of the training images for a concept by similarities among their descriptors. A cluster can be evaluated with the average value of descriptors for images contained in it. The images taking similar features may be included in the same cluster. Therefore, the clusters can be defined as edge description templates, briefly templates, which are identified by the concept. In the final step, relationships among the concept and the templates are generated and stored in knowledge base.

![Figure 3: Edge Description Templates for a Concept](image1)

![Figure 4: Construction Steps of Knowledge Base](image2)

Although the clustering may be performed by k-means method, we can also employ other methods according to application domains [3]. In the clustering process, distances and similarities between two edge histogram descriptors, \( h \) and \( h' \), are...
respectively evaluated with \( \text{dist}(h, h') \) and \( \text{sim}(h, h') \) functions. If \( h \) is approximately equal to \( h' \), the values for two functions should almost approach to \( \text{dist}(h, h') \approx 0 \) and \( \text{sim}(h, h') \approx 1 \). We define the functions as followings.

\[
\text{dist}(h, h') = \sum_{i=1}^{n} |\text{LB}(h[i]) - \text{LB}(h'[i])| + 5 \times \sum_{j=1}^{m} |\text{GB}(h[j]) - \text{GB}(h'[j])| + \sum_{k=1}^{q} |\text{SB}(h[k]) - \text{SB}(h'[k])| \\
\text{sim}(h, h') = e^{\text{val}(h, h')}
\]

Here, \( \text{LB}(h[i]) \), \( \text{GB}(h[j]) \) and \( \text{SB}(h[k]) \) might respectively represent \( i \)_th bin for local block, \( j \)_th bin for global block and \( k \)_th bin for semi-global block for a edge histogram descriptor \( h \). In specific, \( \text{sim} \) function employs an exponential equation for normalizing the similarities into values between 0 and 1.

In the other hand, it is very easy for our knowledge base to structurally accommodate other image feature descriptions as the same way, if needed. Specifically, the color feature may be also a good description according to application domains. However, since the paper is focused on a concept-based mechanism for indexing and retrieving images, not extracting the image features, we use the edge description as only an example for explaining our mechanism well. In addition, the edge description may be known in MPEG-7 as a good solution to represent the image feature comparatively.

### 3. IMAGE INDEXING AND RETRIEVAL

In this section, we will design a concept-based system model that is capable of indexing and retrieving images required by users. Because of employing a knowledge base described with the relationships among concepts and EDTs, it is able to index an image as concepts. The indexing enables a user to retrieve the images relevant to the query which may be formulated by concepts. In specific, the knowledge base is expanded for supporting conceptual query evaluations of higher level.
In Figure 5, the system architecture is depicted for our concept-based indexing and retrieval mechanism. The knowledge base takes two types of relationships. One is the inter-relations between the concepts and the templates, which are generated by clustering images according to the similarities of their EHD. The relationship is used for indexing an image as concepts. In section 3.1, the indexing process is explained in detail. The other is the relationship among the concepts, which may is used for conceptually evaluating user’s query. Its details are described in section 3.2. In the image database, an image could be indexed with concepts, whose templates may be similarly matched with its edge histogram descriptor. Our retrieval module makes the use of ‘Index DB’, which can store information for the images indexed by concepts. That is, since a user query described by concepts can be expanded with the relationships among the concepts to avoid the vocabulary mismatch problem, the retrieval enables users to search the relevant images indexed with concepts, which are contained in the expanded query. The retrieved images can be visualized by the order of similarities between the query and the images.

3.1 Image Indexing

In general, for the concept-based retrievals, it is necessary to index images with concepts which are able to stand for their contents reasonably. We have exploited the knowledge base which can represent the relationships among concepts and templates. Therefore, an image is indexed by the concepts taking a relationship with templates which are approximately matched with its edge histogram descriptor. The approximate matching evaluated by fuzzy degree also enables an image to be indexed by one or more concepts. Since EDT can be described by the same structure as EHD, the fuzzy degrees can be calculated by \( \text{sim} \) function which evaluates the similarity between two edge histogram descriptors.

Formally, for the set of templates taking a relationship with a concept \( c \), \( \text{KB}(c) = \{t_1, t_2, t_3, \ldots\} \), the edge histogram descriptor of an image \( p \), \( h_p \), could be matched with a template \( t \in \text{KB}(c) \) as a fuzzy degree \( \alpha \). If \( \alpha \) might be greater than a threshold value \( \beta \), it could be indexed by \( c \). In this case, it can be expressed as \( p \in \text{IDX}(c) \). The threshold value \( \beta \) might be adequately determined through tuning tests for images relevant to application domains. Therefore, the fuzzy degree \( \alpha \) between \( h_p \) and all \( t \in \text{KB}(c) \) is evaluated by following \( \max \) equation.

\[
\alpha = \max\{\alpha_1, \alpha_2, \ldots, \alpha_n\},
\]

for \( t_i \in \text{KB}(c) \) and \( \alpha_i = \text{sim}(h_p, t_i) \)

Figure 6 depicts an example for indexing a target image \( p \) as a concept, \( c = \text{‘car’} \). The edge histogram descriptor for \( p \), \( h_p \), can respectively be matched with the templates for \( c \), \( \text{KB}(c) = \{t_1, t_2, t_3, \ldots\} \). Since the fuzzy degrees for each matching are evaluated as \( \alpha_1 = \text{sim}(h_p, t_1) \), \( \alpha_2 = \text{sim}(h_p, t_2) \) and \( \alpha_3 = \text{sim}(h_p, t_3) \), the image \( p \) can be indexed by a concept \( c = \text{‘car’} \) as a fuzzy degree \( \alpha = \max(\alpha_1, \alpha_2, \alpha_3, \ldots) \), only if \( \alpha \geq \beta \). Consequently, it can be denote as \( p \in \text{IDX}(\text{‘car’}) \) for the image \( p \). On the occasions, \( p \) may be indexed by other concept \( c' \), since the templates of \( \text{KB}(c') \) can be matched with \( h_p \), as a fuzzy degree \( \alpha' \geq \beta \). In this case, users are able to simultaneously retrieve the image \( p \) by two concepts, \( c \) and \( c' \).

3.2 Image Retrieval

For supporting concept-based image retrieval of higher level, we can make an expansion of the knowledge base which only specifies the relationships between concepts and templates. In other hand, concepts involved in the expanded knowledge base can be classified into non-terminal concepts and terminal concepts. If a concept takes the relationships with its templates, it is classified as a terminal concept. If not, it is classified as a non-terminal concept which may take more specialized sub-concepts: non-terminal or terminal concepts. Since the direct or indirect sub-concepts of a concept \( c \) can, in turn, have terminal concepts, \( c \) may form a concept hierarchy together with the sub-concepts. We call \( c \) as a top-level concept deriving a concept hierarchy. The expansion of the knowledge base enables user’s query to be processed conceptually and interpreted variously without the vocabulary mismatch problem.

Figure 7 illustrates an example for the expanded knowledge base. A top-level concept ‘vehicle’ could be taking two sub-concepts, ‘motor vehicle’ and ‘craft’, which have more concrete semantics than ‘vehicle’. A sub-concept ‘motor vehicle’
could also be classified with three concepts, ‘truck’, ‘car’ and ‘bus’. They are terminal concepts because of directly taking the relationship with their templates. That is, a terminal concept ‘car’ is specified with three edge description templates, $t_{c1}$, $t_{c2}$ and $t_{c3}$. Ontology or thesaurus like WordNet, generally employed in the field of natural language process, helps to construct the relationship among the concepts [16] [24].

In our image retrieval, a user query is formulated with Boolean notation for concepts, and is controlled by concepts included in knowledge base. In this paper, we may standardize it as mono, disjunctive and conjunctive queries.

For a concept $c \in C$ of knowledge base, the mono-query is described as $Q=c$. In turn, for mono-queries $Q_i$ $i=1...n$, the disjunctive query is formulated as $Q = Q_1$ OR $Q_2$ OR ... OR $Q_n$. Continuously, when giving the disjunctive queries $Q_i$ $i=1,...,n$ as $Q_i=Q_{i,1}$ OR $Q_{i,2}$ OR ... OR $Q_{i,m_i}$ for mono-queries $Q_{ij}=Q_{1j}$ OR $Q_{2j}$ OR ... OR $Q_{nj}$, the conjunctive query is defined as $Q = Q_1$ AND $Q_2$ AND ... AND $Q_n = (Q_{1,1}$ OR $Q_{1,2}$ OR ... OR $Q_{1,m_1})$ AND $(Q_{2,1}$ OR $Q_{2,2}$ OR ... OR $Q_{2,m_2})$ AND ... AND $(Q_{n,1}$ OR $Q_{n,2}$ OR ... OR $Q_{n,m_n})$.

As applying the concept relationships in the knowledge base, an initial user query $Q$ may be expanded into $Q' = Q_1'$ AND $Q_2'$ AND ... AND $Q_n'$ = $(Q_{1,1}'$ OR $Q_{1,2}'$ OR ... OR $Q_{1,m_1}')$ AND $(Q_{2,1}'$ OR $Q_{2,2}'$ OR ... OR $Q_{2,m_2}')$ AND ... AND $(Q_{n,1}'$ OR $Q_{n,2}'$ OR ... OR $Q_{n,m_n}')$. When giving the terminal concepts for $c \in C$ as $c_k \in C$, $k=1,...,s$, a mono-query $Q_{ij} = c$ may be expanded as a disjunction query, $Q_{ij} = c_1$ OR $c_2$ OR ... OR $c_s$. For example, a query $Q='motor vehicle'$ is expanded as $Q'='truck' OR 'car' OR 'bus'$ in Figure 7. Therefore, all of the queries, including the expanded queries, are normalized by the conjunction of disjunctions for mono-queries.

In the other hand, we will explain query evaluations for retrieving the relevant images. We can define $||Q||$ as set of images satisfying a user query $Q$. For an image $p$, $p \in a[Q]$ can also denote that $p$ satisfies $Q$ as the fuzzy degree $a$. If $p$ is indexed by $c$ as a fuzzy degree $\alpha$, namely $p \in a[IDX(c)]$, a mono-query $Q = c$ is evaluated as $p \in a[Q]$. Continuously, when giving the mono-queries $Q_i$ and $p \in a[Q_i]$ $i=1,...,n$, a disjunctive query $Q=Q_1$ OR $Q_2$ OR ... OR $Q_n$ is evaluated as $p \in a[Q]$, $a = max(a_1, a_2, ..., a_n)$. In the same way, for the disjunctive query $Q_i$ and $p \in a[Q_i]$ $i=1,...,m$, a conjunctive query $Q=Q_1$ AND $Q_2$ AND ... AND $Q_n$ is evaluated as $p \in a[Q]$, $a = min(a_1, a_2, ..., a_n)$. The $min$ and $max$ are functions for approximating and/or between two fuzzy values.
Figure 8 explains the processes for concept-based image retrieval with the knowledge base. An initial query, Q='motor vehicle' can be expanded as Q’ = “truck OR car OR bus” using the terminal concepts of ‘motor vehicle’. The query expansion may be performed with the relationships between concepts in KB. Since an image $p_1$ satisfies $p_1 \in 0.9 \text{IDX}(\text{‘car’})$, $p_1 \in 0.0 \text{IDX}(\text{‘truck’})$ and $p_1 \in 0.3 \text{IDX}(\text{‘bus’})$ in the same time, we can evaluate $Q$ as $p_1 \in \alpha |Q|$, $\alpha = \max(0.9, 0.0, 0.3) = 0.9$. Therefore, the images are retrieved and visualized by the query according to their fuzzy degrees.

4. IMPLEMENTATIONS

This section is dedicated to describing the functionality of the concept-based image retrieval system in order to demonstrate the feasibility of our approach. Also, it is capable of retrieving the images based on color and edge features for an image query. Our system is developed on top of Windows XP with Microsoft Visual C++. An object-relation database management system, ORACLE, is employed to manage the image data, indexing information and knowledge base. For convenient evaluations, the test images are constituted with images retrieved by a concept ‘vehicle’ in a commercial search engine.

First of all, we explain a method for the feature-based image retrieval in our system. It may perform the feature matching between a given query image and database images. Only two features of images, edge and color, are used for the matching. As previously mentioned, it has a drawback that the image query can not represent user’s conceptual intent. That is, since the query is described by an example image, the concept can not be expressed on the whole, but partially. Therefore, the method can not retrieve images that may be different from features of the query image, although they may be related with the concept intended by users.

Figure 9 depicts an example for our edge feature-based retrieval by a query image. The query is described by a diagonal car image. Most of the retrieved images are also taking the similar features with it. However, if a user may request all of images related with a concept ‘car’, the result is not sufficient to satisfy him/her. The reason is that it may not contain the front and side images for a car. That is, since the image query can not reflect the user intents conceptually, they can not be retrieved. Moreover, the method can not evaluate more conceptual query such as ‘motor vehicle’, since it is almost impossible to express such conceptual semantics by an image query. The drawback makes our concept-based retrieval necessary.

Our concept-based retrieval may perform the matching between a search concept of user’s query and an index concept of images. During query processing and image indexing, the concepts are controlled by a knowledge base. In other hand, since
the user queries are described by concepts combined with Boolean operators, our system can retrieve the relevant images by performing the fuzzy set operations for images indexed by the search concepts. Its result may be able to contain all of the images that are related with the concept of the query, although their features may be different from each others.

Figure 10 depicts an example for our concept-based retrieval by a query concept ‘car’. The user query is described in a text box, ‘Concept Query’, of ‘Concept Query Dialog’. The dialog is opened by pushing ‘Concept Query’ button. The retrieval makes the query concept ‘car’ matched with index concepts for images. That is, if an image is retrieved by the query concept, it should have the index concept ‘car’. Therefore, we may overcome the drawback of feature-based retrieval, which can not retrieve images that may be only different from features of a query image, but be related with the concept intended by users. For example, although three images ordered by (3), (4) and (5) in the result frame have each other features, they are retrieved by a concept ‘car’. Figure 11 is other example for processing more conceptual query, ‘motor vehicle’. The processing result contains images for bus and truck as well as car, such as the images ordered by (5), (6) and (8) in the result frame.

Figure 10: Retrieval for a concept query ‘car’

Figure 11: Retrieval for a concept query ‘motor vehicle’

5. CONCLUSIONS AND FURTHER WORKS

Currently, most image retrieval systems may only exploit the primitive features to automatically index and retrieve the relevant images. However, they can not satisfy user’s requirements which intend to retrieve the images with conceptual queries. In this paper, we design and implement the concept-based image retrieval by employing the knowledge base. Since the knowledge base can specify the relationships among concepts and edge description templates for the related images, it is likely that images can be conceptually indexed and retrieved by them.
As further researches, complementary works for our system may be needed. First, the knowledge base would be developed in detail as exploiting the various features, since it is a core component for conceptually indexing the images. Second, we must be able to detect two or more objects contained in an image in order to be accurately indexed by their names.

For example, the image (a) in Figure 12 can be indexed by two concepts, ‘tree’ and ‘car’, if the further researches are resolved. That is, it is necessary to detect the boundaries of ‘car’ and ‘tree’ objects contained in the image. Since the indexing by the detected objects enables a retrieval system to exploit the relationships among them, it may process the complex queries like as “retrieve the images for the cars parked between palm trees at the seashore”. For indexing the image (b) by a concept ‘Desert’, the knowledge base may be specified more accurately. That is, a concept should be represented with more various feature description templates for not only edge but also color, shape and texture.

![Figure 12: Images Indexed by Two Concepts](image-url)

**REFERENCES**