Online Image Retrieval System (OIRS)

Khalil Shihab
Department of Computer Science, Sultan Qaboos University, PO Box 36 Muscat 123, Oman,

Abstract: - In this paper, we presented an online image retrieval system that allows high-quality museum images and associated information to be made available over networks. Case-based reasoning is considered more efficient and of great benefit in this area. This is mainly because users both in indexing and retrieval processes, tend to use old cases by associating images that reveal similar features. In this work, we present an application of our integrated approach to image indexing and retrieval. The underlying technique uses high-level features to find the most possible assignment to the presented image description. It applies fuzzy reasoning to convert the quantitative attributes into qualitative terms for indexing and retrieval. To facilitate the storage and retrieval, we adopted XML technology.

Key Words: - Image Retrieval, Case-based Reasoning, Fuzzy Logic, XML.

1. Introduction
In 2003, during the second Iraq war, more than 15,000 items were stolen from Baghdad's Iraq Museum. A large number of the famous artifacts in history and treasurers like the beautiful carved-ivory Mona Lisa of Nimrod survived ten centuries, only to fall victim to chaos and looters, some sent by international art dealers.

This paper describes the design and prototype implementation of a novel architecture for integrated metadata and concept based indexing and retrieval of museum information [1]. The system constitutes a virtual museum preserving some works that are lost and providing more versatile access to the images and information of lost treasures from Baghdad museum, in particular.

In order to maintain a close match between the user queries and the retrieved images, we therefore used an integrated technique based on similarity matching and fuzzy reasoning for indexing and retrieval of images. We also adopted XML case-representation to facilitate the image storage and retrieval process [3].

2. Case-based Reasoning
Case-based reasoning makes use of past experiences to derive the solution for a new problem. It has been widely implemented in practical applications [4, 5]. To process past experiences (cases) efficiently, a common case-based reasoning technique is to select some characteristics that are representative of the cases and use them as indexes to store the cases. Later, to solve new problems, the system uses these characteristics as probe to retrieve the set of similar cases that are then adapted and modified to arrive at a targeted solution. Often, it is a common practice to narrow the set of retrieved cases by means of a similarity metric. We used XML as case representation for making up structured knowledge-rich data. XML is capable of representing sophisticated structures of a variety of types, well beyond the simple tables of delimited text commonly used to exchange information, and comes with tools for describing those structures.

In this work, we present an integrated approach that can deal with both qualitative and quantitative attributes. The approach converts the quantitative attributes into qualitative terms for indexing and retrieval. It applies fuzzy sets concepts to case indexing and retrieval to achieve that [6, 7].

Using fuzzy indexing and retrieval allows attributes that are characterized by numerical values to be converted into fuzzy sets to simplify comparison. For example, the height of the artifact can be converted into categorical scale (e.g. tall/large, medium, and short/small). Also, fuzzy sets allow multiple indexing of a case on a single value with different degrees of membership. For example, if the size is 60cm, this can be classified as tall with 0.4 and medium with 0.7, where 0.4 and 0.7 are the degrees that the height is classified as tall or medium.
respectively. This treatment increases the flexibility of case matching by allowing the case to be considered as a candidate when we are looking for an artifact with either large or medium size.

3. Database Construction

The system database was designed to emphasize simplicity and portability. These criteria can be achieved by using XML file structure that also enables a smooth navigation and editing of the document [8].

Every image added to the database is copied into the appropriate subfolder in the main directory of images and a resized small version of the file is copied into the thumbs directory. The XML directory contains the index files required to maintain the integrity of the directory structure and to manage the data extracted from the images. When an image is added to the database, features are extracted from the image and stored in an index file in the XML directory of the database.

4. Fuzzy Sets

In fuzzy sets an object may partially belong to a set, so the set must be represented by a continuous membership function that maps the domain of the set to an interval of [0, 1]. For example, the following functions (1-3) and Figure 1 show the membership functions of high, moderate and low [7].

1. \( \mu_{\text{high}}(x) = \frac{x - x_l}{d} \) if \( x_l < x < x_u \), 0 if \( x < x_l \), and 1 if \( x > x_u \)
2. \( \mu_{\text{moderate}}(x) = \frac{(x - x_o)}{0.5d} \) if \( x = x_{\text{midpoint}} \), 0 otherwise
3. \( \mu_{\text{low}}(x) = (x - x_m)/d \) if \( x_m < x < x_u \), and 1 if \( x < x_m \)

Where \( x_l, x_u, d_o, \) and \( \text{midpoint} \) are as follows:

\[
\begin{align*}
X_1 & = \begin{cases} 30 & X_{\text{CPU}} \\ 20 & X_{1/0} \end{cases} \\
X_2 & = \begin{cases} 70 & X_{\text{CPU}} \\ 40 & X_{1/0} \end{cases}
\end{align*}
\]

Since fuzzy sets use possibilities rather than binary membership values, a threshold value is often used to differentiate those considered highly likely to be a member of a set from those considered relatively unlikely. For example, when we are seeking for artifacts that have large size or tall, we may want to consider only those with membership grades of tall are above 5. This value is generally called a-cut. For example, if the membership function of tall, as defined in Figure 2, is given and if the a-cut is set at 5 for tall, then artifacts with height greater than 55cm are considered tall, whereas artifacts that have their heights greater than 61 cm are considered very tall.

An additional set of extensions to the index structure allows queries to specify numeric ranges. Information retrieval systems normally treat numbers as keywords; a user searching for "2000 B.C." can find the exact value "2000 B.C.", but it is not possible to search for "between 2000 and 2200".

To enable numeric queries, when the system encounters a number it stores it as several keywords representing different time periods which contain the target number. For example, "1999 B.C." might be indexed under the keywords "Sumerians" and "Elamites".

When a user searches for a range, the keywords necessary to exactly cover the range are identified. Query results from each of these keywords are combined disjunctively. For example, a user specifying a numeric range of 1000-2200 might cause a query for the keywords "Sumerians" and "Elamites".

5. Indexing and Retrieval of Cases

Case attributes can be either quantitative or qualitative. Qualitative attributes accept nominal values. For example, the artifact type is a qualitative attribute whose value may be stone, bronze/copper, clay, gold, ivory, or shell. Quantitative attributes, on the other hand, allow values to be measured on a numerical scale.

For cases with qualitative attributes only, indexing can be performed on attributes directly. For example, artifacts can be classified as large, medium, or small (three classes according to their size); or can be classified according to their materials into six classes: stone, bronze/copper, clay, gold, ivory, or shell. We can easily index systems by their materials. If we also want to include the height or size, indexing becomes more complicated since the value of this
attribute can be any positive real number. However, with a proper transformation into a few discrete classes based on practical requirements, indexing becomes easier to handle.

The process of fuzzy indexing is, therefore, of two stages. Quantitative attributes are first processed by the fuzzifier (called fuzzification) and then indexed on the resulting classes (indexing) before being stored in the CB.

5.1. Fuzzification Process

To illustrate the fuzzification process, a running example is used. The context is a lost treasure domain that contains Artifacts, Figurines, Inlays, Jewelry, Metal Vessels, Musical Instruments, Pottery, Relief, Seals, Sculpture, Vessels, and Terracotta. They are categorized into six different types: stone, bronze/copper, clay, gold, ivory, or shell. Figure 2 shows some of these objects in the XML database.

```xml
<?xml version="1.0"?>
<IMAGES>
  <IMAGE>
    <SERNO>1</SERNO>
    <MuseumNumber>IM19755</MuseumNumber>
    <CATEGORY>Limestone, Female</CATEGORY>
    <MATERIAL>Limestone</MATERIAL>
    <KEYWORDS>Female, Standing</KEYWORDS>
    <DESCRIPTION>Standing Female, Eyeballs of Shell</DESCRIPTION>
    <DIMENSION/HEIGHT/LENGTH>54cm, tall/0.62, medium/0.25, small/0.13</HEIGHT>
    <LOCATION>Tell Asmar</LOCATION>
    <PERIOD>Sumerian, Early Dynastic II 2600 B.C.</PERIOD>
    <STATUS>Stolen</STATUS>
    <URL>http://MySite/ImageGallery/Images/standing_pic1.jpg</URL>
  </IMAGE>
  <IMAGE>
    <SERNO>2</SERNO>
    <MuseumNumber>IM9659</MuseumNumber>
    <CATEGORY>Female, Stone</CATEGORY>
    <MATERIAL>Stone</MATERIAL>
    <KEYWORDS>Female, Standing</KEYWORDS>
    <DESCRIPTION>Statue of female wearing flounced garment leaving one shoulder bare, hands folded, standing on flat base</DESCRIPTION>
    <DIMENSION/HEIGHT/LENGTH>36cm, tall/0.059, medium/0.8, small/0.4</HEIGHT>
    <LOCATION>Khafaji</LOCATION>
    <PERIOD>Sumerian, Early Dynastic II 2600 B.C.</PERIOD>
    <STATUS>Stolen</STATUS>
    <URL>http://MySite/ImageGallery/Images/standing_pic2.jpg</URL>
  </IMAGE>
  <IMAGE>
    <SERNO>3</SERNO>
    <MuseumNumber>IM19759</MuseumNumber>
    <CATEGORY>Male, Stone, Standing</CATEGORY>
    <MATERIAL>Stone, Limestone</MATERIAL>
    <KEYWORDS>Male, Standing, Stone</KEYWORDS>
    <DESCRIPTION>Statue of male bearded, long hair, bare-chested wearing flounced skirt, hands folded, standing on flat base</DESCRIPTION>
    <DIMENSION/HEIGHT/LENGTH>54cm, tall/0.57, medium/0.46, small/0.42</DIMENSION>
    <LOCATION>Tell Asmar</LOCATION>
    <PERIOD>Sumerian, Early Dynastic II 2600 B.C.</PERIOD>
    <STATUS>Stolen</STATUS>
  </IMAGE>
  <IMAGE>
    <SERNO>4</SERNO>
    <MuseumNumber>IM9659</MuseumNumber>
    <CATEGORY>Female, Stone, Standing</CATEGORY>
    <MATERIAL>Stone, Limestone</MATERIAL>
    <KEYWORDS>Female, Standing</KEYWORDS>
    <DESCRIPTION>Statue of female wearing flounced garment leaving one shoulder bare, hands folded, standing on flat base</DESCRIPTION>
    <DIMENSION/HEIGHT/LENGTH>36cm, tall/0.059, medium/0.8, small/0.4</HEIGHT>
    <LOCATION>Khafaji</LOCATION>
    <PERIOD>Sumerian, Early Dynastic II 2600 B.C.</PERIOD>
    <STATUS>Stolen</STATUS>
  </IMAGE>
</IMAGES>

Fig. 2. Some cases in XML cases

In the transformation of the measurement data in Table 1, the fuzzifier handles the quantitative values that need to be converted into qualitative data. Usually, we classify the artifact height into three classes: tall, medium, and small. Using the membership functions, given above, the fuzzifier converts the height value 0.65 into membership grades of the respective classes: 0.88 for tall, 0.25 for medium and 0.13 for small. However, if the α-cut is set to 0.5, then the height, in this case, is classified as tall/0.88 only.
Table 1: An image instance

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>MuseumNumber</td>
<td>Multicasts</td>
</tr>
<tr>
<td>CATEGORY</td>
<td>Limestone</td>
</tr>
<tr>
<td>MATERIAL</td>
<td>Female Standing</td>
</tr>
<tr>
<td>KEYWORDS</td>
<td>Standing female, eyeballs of shell</td>
</tr>
<tr>
<td>DESCRIPTION</td>
<td>65</td>
</tr>
<tr>
<td>HEIGHT</td>
<td>Tell Asmar</td>
</tr>
<tr>
<td>LOCATION</td>
<td>2600</td>
</tr>
<tr>
<td>PERIOD/YEAR</td>
<td>Stolen</td>
</tr>
<tr>
<td>STATUS</td>
<td>Standing_Pic1.jpg</td>
</tr>
<tr>
<td>URL</td>
<td></td>
</tr>
</tbody>
</table>

5.2. Case Acquisition

Once a problem instance is indexed, four additional attributes are added before it becomes a case to be stored in the case base (CB). These additional attributes are: the case number, the unusual or the interesting property.

5.3 Case Retrieval

Faced with a problem instance, the case based reasoning first ranks cases based on their degree of similarity with the problem instance. A similarity score that is computed by comparing each case with the problem instance quantifies this. Next, CBR retrieves the most similar cases.

For improving retrieval we used a fuzzy method that combines the fuzzy terms with known qualitative attributes and uses them as keys for retrieval of similar cases. The selection of past cases that best match the present problem depends on being able to identify and evaluate relevant attributes and being able to perform simple matching between cases. Given the cases in Table 2, suppose the goal is to retrieve an image similar to that described by Table 3.

Table 2. Cases in a sample case base (CB)

<table>
<thead>
<tr>
<th>Case No.</th>
<th>Key Words</th>
<th>Description</th>
<th>Fuzzy Height</th>
<th>Fuzzy Price</th>
<th>Period/Y</th>
<th>Unusual Property</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Female</td>
<td>Standing</td>
<td>tall/0.62 medium/0.2 small/0.13</td>
<td>*</td>
<td>2600</td>
<td>*</td>
<td>Stolen</td>
</tr>
<tr>
<td>2</td>
<td>Female</td>
<td>Wearing garment, bare shoulder</td>
<td>tall/0.54 medium/0.5 small/0.45</td>
<td>*</td>
<td>2800</td>
<td>*</td>
<td>Unknown</td>
</tr>
<tr>
<td>3</td>
<td>Male</td>
<td>Bearded, long hair, Wearing skirt</td>
<td>tall/0.3 medium/0.6 small/0.7</td>
<td>*</td>
<td>2600</td>
<td>*</td>
<td>Stolen</td>
</tr>
<tr>
<td>4</td>
<td>Female</td>
<td>Standing, wearing garment</td>
<td>medium/0.4</td>
<td>*</td>
<td>2600</td>
<td>*</td>
<td>Stolen</td>
</tr>
<tr>
<td>5</td>
<td>Male</td>
<td>Standing, wearing skirt, beardless</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>Unknown</td>
</tr>
<tr>
<td>6</td>
<td>Female</td>
<td>Wearing garment, one shoulder bare</td>
<td>heigh/0.57, medium/0.46 small/0.42</td>
<td>*</td>
<td>2600</td>
<td>*</td>
<td>Unknown</td>
</tr>
<tr>
<td>7</td>
<td>Female</td>
<td>Standing</td>
<td>heigh/0.92 medium/0.1</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>Stolen</td>
</tr>
<tr>
<td>8</td>
<td>Male</td>
<td>Wearing short skirt</td>
<td>tall/0.45 medium/0.5 small/0.42</td>
<td>*</td>
<td>1800</td>
<td>*</td>
<td>Stolen</td>
</tr>
<tr>
<td>9</td>
<td>*</td>
<td>Standing figures</td>
<td>heigh/0.3 medium/0.4 small/0.7</td>
<td>*</td>
<td>1800</td>
<td>*</td>
<td>Stolen</td>
</tr>
<tr>
<td>10</td>
<td>Male</td>
<td>Standing, bearded</td>
<td>tall/0.9 medium/0.4</td>
<td>*</td>
<td>2800</td>
<td>*</td>
<td>Unknown</td>
</tr>
</tbody>
</table>

Table 3: An image instance

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>MuseumNumber</td>
<td>Limestone</td>
</tr>
<tr>
<td>CATEGORY</td>
<td>Female</td>
</tr>
<tr>
<td>MATERIAL</td>
<td>Standing female, eyeballs of shell</td>
</tr>
<tr>
<td>KEYWORDS</td>
<td>54</td>
</tr>
<tr>
<td>DESCRIPTION</td>
<td></td>
</tr>
<tr>
<td>HEIGHT</td>
<td>2600</td>
</tr>
<tr>
<td>LOCATION</td>
<td>Stolen</td>
</tr>
<tr>
<td>PERIOD/YEAR</td>
<td>URL</td>
</tr>
<tr>
<td>STATUS</td>
<td>Standing_Pic1.jpg</td>
</tr>
</tbody>
</table>
After transformation of data in Table 3, the following problem instance is produced and added to be a new entry in the XML database.

```xml
<IMAGE>
<SERNO>4</SERNO>
<MuseumNumber>IM9659</MuseumNumber>
<CATEGORY>Female, Stone, Standing</CATEGORY>
<MATERIAL>Stone, Limestone</MATERIAL>
<KEYWORDS>Female, Standing, Stone</KEYWORDS>
<DESCRIPTION>Statue of female wearing flounced garment leaving one shoulder bare, hands folded, standing on flat base</DESCRIPTION>
<DIMENSION> tall/0.059, medium/0.8, small/0.4 36cm </DIMENSION>
<LOCATION>Khafaji</LOCATION>
<PERIOD>Sumerian, Early Dynastic, 2600 B.C.</PERIOD>
<STATUS>Stolen</STATUS>
<URL>http://MySite/ImageGallery/standing_pic4.jpg</URL>
</IMAGE>
```

Fig. 3: A problem instance

Based on the matching attributes of the problem instance, the case retrieval can easily select the cases 1, 2, 6 and 7 from the CB to be used as bases for performance evaluation of this new problem instance. Fuzzy retrieval often results in a set of candidate cases for reasoning. The issue following fuzzy retrieval is to find the most similar case among candidates. There are several ways of finding the most similar case. In this work, we use the following algorithm (similarity measure).

1. The similarity measure, \( d_q \), is calculated as follows:

\[
\sum_{i=1}^{n} d_i
\]

2. \( d_q = \sum_{i=1}^{n} d_i \); where \( n \) is the number of the attributes.

3. The parameter, \( a_i \), is set to –1 if the unusual-property for both the problem instance and the case has the same value; \( a_i \) is set to 0 if the attribute’s value for the case is equal to the attribute of the problem instance; \( a_i \) is set to 0.5 if the attribute’s value for the case is a wildcard (i.e. "*").

4. The similarity measure for fuzzy attributes is calculated as follows:

\[
d_i = \sum_{j} \text{abs} \left( x_{ijk} - x_{ijn} \right) \text{ where } x_k \text{ and } x_n \text{ are the grades of attribute } i, \text{ class } j, \text{ for cases } k \text{ and } n \text{ respectively.}
\]

5. The similarity measure for the case is the sum of the results obtained from (1) and (2).

\[
d_c = d_q + d_i
\]

Table 4 displays the results of applying this algorithm to the problem instance in Figure 3 and the cases in Table 2. Case 6 is, therefore, the most similar case to the problem instance.

Table 4: Distances between the problem instance of Fig. 3 and the candidate cases

<table>
<thead>
<tr>
<th>Case</th>
<th>Fuzzy Height</th>
<th>Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Problem Instance</td>
<td>tall/0.59, medium/0.8, small/0.4</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>tall/0.62, medium/0.25, small/0.13</td>
<td>0.85</td>
</tr>
<tr>
<td>2</td>
<td>tall/0.54, medium/0.9, small/0.45</td>
<td>0.2</td>
</tr>
<tr>
<td>6</td>
<td>tall/0.57, medium/0.85, small/0.42</td>
<td>0.09</td>
</tr>
<tr>
<td>7</td>
<td>tall/0.92, medium/0.15, small/0</td>
<td>1.38</td>
</tr>
</tbody>
</table>

6. Experiment

Figure 4 provides the main menu of the image gallery. If the option “Search the Database” is selected, then the system allows the user to enter a text to start searching the database for images that reveal similar features. Now, if the data presented in Table 3 are entered, then the system will display the results shown in Figure 5. Clicking on any of the retrieved images, more information on that image along with its large size will be displayed. Figure 6 shows this action.

Cases may contain quantitative and qualitative attributes. In this work, we presented and used an integrated approach that uses fuzzy set concepts for indexing and retrieval of similar cases. The approach converts the quantitative attributes into qualitative terms. It applies fuzzy sets concepts to case indexing and retrieval in order to overcome the problem.
6. Conclusion

The results given above have been produced using XML file structure. This work aims to demonstrate the use of case-based reasoning for image storage and retrieval. This and similar types of automation have been considered essential for solving these problems in particular.

Cases may contain quantitative and qualitative attributes that are hard to index and manage in the case base; hence it is important to develop an effective method for handling them. We, therefore, used an integrated approach that uses fuzzy set concepts for indexing and retrieval of similar cases.

7. References