A Fuzzy Image Matching Algorithm with Linguistic Spatial Queries

TZUNG-PEI HONG1, SZU-PO WANG2, TIEN-CHIN WANG3, BEEN-CHIAN CHIEN3
1 Department of Electrical Engineering, National University of Kaohsiung
2 Institute of Information Management, I-Shou University
3 Department of Information Engineering, I-Shou University

Kaohsiung, Taiwan, R.O.C.

tphong@nuk.edu.tw, m9022021@stmail.isu.edu.tw, king@isu.edu.tw, cbc@nuk.edu.tw

Abstract: - The function for spatial queries has been provided in some image retrieval systems for finding images satisfying given spatial relations. However, images themselves may contain ambiguity or the image processing technologies adopted may extract objects not accurately enough. Processing image queries in a flexible way is thus desired. In this paper, we thus propose a fuzzy image matching algorithm to calculate the fuzzy match degrees of images with linguistic spatial queries. It consists of two main stages. In the first stage, the objects in a linguistic spatial query are first used to filter the images in the image database, thus avoiding some unnecessary checking in the second stage. The fuzzy match degrees between the converted spatial queries and the promising images are then calculated in the second stage by using the fuzzy operations and the membership functions. The proposed algorithm is thus suitable for images with uncertain objects and satisfies users' linguistic queries in a flexible and efficient way.

Key-Words: - fuzzy set, fuzzy match, image database, linguistic query, spatial relation, membership functions

1 Introduction

Multimedia applications have grown very rapidly in recent years due to the dramatic increase in computing power and the concomitant decrease in computing cost. Examples include educational services, movie industry, travel industry, home shopping, medical care, and others [20]. Among the various types of multimedia data, image data are quite commonly seen in real applications since they provide a trade-off between visibility and data size.

Spatial queries are provided in some image retrieving systems for finding images satisfying spatial relations given. In the past, spatial queries were usually processed in a crisp way. That is, each image was judged to either match or non-match a given query. Images themselves may, however, contain ambiguity. For example, the boundary of a cloud in an image is usually not precise. The image processing technologies adopted may also extract objects not accurately enough, such as if only rectangles are used to represent objects, then an object with an irregular shape may have a certain degree of error. Processing image queries in a more flexible way is thus desired.

Recently, the fuzzy set theory has been more and more frequently used in real applications because of its simplicity and similarity to human reasoning. The theory has been applied to many fields such as manufacturing, engineering, diagnosis, economics, and others [9][11][23]. In this paper, we thus adopt the fuzzy concepts to increase the matching flexibility of image data. A fuzzy image matching algorithm is designed to calculate the fuzzy match degrees with linguistic spatial queries. It can evaluate the similarity degrees between queries and images in a fuzzy way and output the promising ones. It consists of two main stages in calculating the matching degrees. In the first stage, it collects the objects in a linguistic spatial query and uses them to filter the images in the image database. In the second stage, it then calculates the fuzzy matching degrees of the spatial relations between the linguistic query and the promising images by the fuzzy operations. The images satisfying user-defined criteria are then output in a descending matching order.

2 Review of Related Approaches

A great number of technologies based on image processing and information extraction have been implemented and applied [5]. Image indexing techniques have been largely used to support pictorial information retrieval from an image database [4][6][21]. An image can usually be associated with two kinds of descriptors: information about its content and information about the spatial relations of its pictorial elements [2][3][7][13][14][15]. Appropriate data structures must be adopted to store the related information and make the image query feasible.

In the past, several famous approaches for effectively retrieving spatial images were proposed.
Chang et al. proposed the 2D-String approach to represent the spatial relations of objects in an image [2][3]. Lee and Hsu then generalized 2D-String and proposed 2D-CString to represent the spatial relations of objects [13][14][15]. Some other variants based on 2D-String were also proposed in the literature [10]. Petraglia et al. then proposed the concept of virtual images based on 2D-CString for spatial image retrieval [18].

Since images themselves may contain ambiguity or the image processing technologies adopted may extract objects not accurately enough, applying fuzzy concepts to increase the retrieval flexibility of image data is thus a good attempt. Chieng et al. [7][8] proposed an algorithm for users to retrieve similar images by fuzzy match. The similarity values of query images with stored images in an image database are calculated from the relative distances of objects in these images. In this paper, we process the image query in a linguistic way. The proposed approach can process users’ linguistic spatial queries and find a set of promising images as output.

3 Review of Related Fuzzy Concepts
Fuzzy set theory was first proposed by Zadeh and Goguen in 1965 [22]. Fuzzy set theory is primarily concerned with quantifying and reasoning using natural language in which words can have ambiguous meanings. This can be thought of as an extension of traditional crisp sets, in which each element must either be in or not in a set. A fuzzy set can be defined by a membership function as:

\[ \mu_X : X \rightarrow [0, 1] \]

where [0, 1] denotes the interval of real numbers from 0 to 1, inclusive. The function can also be generalized to any real interval instead of [0,1].

There are a variety of fuzzy set operations. Among them, three basic and commonly used operations are complement, union and intersection. These operations will be used later for fuzzy match in image databases.

4 Related Definitions
An image table describing the information of objects in an image consists of seven fields: Image_Name, Obj_ID, Obj_Name, XLB, XUB, YLB and YUB. The field Image_Name describes the name of the image in which the object with Obj_ID and Obj_Name appears. Different objects must have different Obj_IDs, but may have the same Obj_Name. XLB and XUB respectively represent the X-axis values of the left edge and the right edge of the object. Similarly, YLB and YUB represent the Y-axis values of the top edge and the bottom edge of the object.

A linguistic spatial relation between two objects in an image consists of one or more constraints. Each constraint is a logical expression evaluated by the proposed fuzzy matching algorithm. For example, the spatial relation Left of (A, B) has only the following one constraint:

\[ \frac{XUB(A) - XLB(B)}{[(XUB(A) - XLB(A)) + (XUB(B) - XLB(B))] / 2} < 0 \]

However, the spatial relation Identical (A, B) have the following four constraints:

\[ \frac{XLB(A) - XLB(B)}{[(XUB(A) - XLB(A)) / 2 + (XUB(B) - XLB(B)) / 2]} = 0 \]
\[ \frac{XUB(A) - XUB(B)}{[(XUB(A) - XLB(A)) / 2 + (XUB(B) - XLB(B)) / 2]} = 0 \]
\[ \frac{YLB(A) - YLB(B)}{[(YUB(A) - YLB(A)) / 2 + (YUB(B) - YLB(B)) / 2]} = 0 \]
\[ \frac{YUB(A) - YUB(B)}{[(YUB(A) - YLB(A)) / 2 + (YUB(B) - YLB(B)) / 2]} = 0 \]

The denominators in the constraints are used to normalize the fuzzy evaluation values when the constraints are combined. A linguistic spatial query may consists of one or more spatial relations connected by the “and” or “or” logical connectors.

Membership functions are predefined and used by the fuzzy matching algorithm to evaluate the fuzzy matching degrees of images. Five membership functions are defined, respectively for the comparison operators “< 0”, “≤ 0”, “= 0”, “> 0” and “>>0”, as shown in Figure 1.

![Figure 1. The membership functions for the comparison operators.](image-url)

Fuzzy evaluation is useful for performing linguistic spatial queries for ambiguous and not-accurately processed images. For example,
Figure 2 shows three images with two objects $A$ and $B$. If we make a query of “$A$ is left of $B$”, then $I_1$ should best match the query among the three images in Figure 3. $I_2$ matches better than $I_3$ since $A$ is left of $B$ in $I_2$, but meets $B$ in $I_3$. In $I_2$ and $I_3$, $A$ is not certainly left of $B$ if $A$ and $B$ have some ambiguity or imprecision in the boundaries. It is thus reasonable that these three images have different fuzzy match values with the given query.

Example 1: Assume an image database includes six images $I_1$ to $I_6$, which are shown in Figure 3.

Figure 3: An image database used in Example 1.

Figure 2: The three images with different fuzzy match values for the query “$A$ is left of $B$”.

The proposed fuzzy image matching algorithm thus calculates the fuzzy match degrees of linguistic spatial queries with stored images and outputs the ones with high matching values. The proposed algorithm is described as follows.

**The fuzzy image matching algorithm:**

Input: A linguistic spatial query, an image database, an image table describing the information of objects in the images.

Output: A set of promising images matching the linguistic spatial query.

**STEP 1:** Collect the object names included in the query. Denote them as $O_i, O_2, ..., O_r$.

**STEP 2:** Find the definition $D^k_{ij}$ of each spatial relation $R^k_{ij}$ between any two objects $O_i$ and $O_j$ in the query.

**STEP 3:** Find the images that include all the objects in the query from the image table.

**STEP 4:** Do STEPs 5 to 9 for each image $I$ found in STEP 3.

**STEP 5:** Calculate the left-hand-side value $v_{ij}^{km}$ of the $m$-th constraint $D^m_{ij}$ in $D^k_{ij}$, $m=1$ to $|D^k_{ij}|$, for each possible combination $s$ of $O_i$ and $O_j$ in $I$, where $|D^k_{ij}|$ is the number of constraints for the spatial relation definition $D^k_{ij}$.

**STEP 6:** Transform $v_{ij}^{km}$ into a fuzzy value $f_{ij}^{km}$ using the given membership functions according to the comparison operator in the constraint $D^m_{ij}$.

**STEP 7:** Evaluate the fuzzy value $f_{ij}^{km}$ of $s$ satisfying the spatial relation definition $D^m_{ij}$. If two constraints $D^{km_1}$ and $D^{km_2}$ are connected by an “and” logical operation, set the resulting membership value as:

$$\min(f_{ij}^{k_{m_1}}, f_{ij}^{k_{m_2}}).$$

If they are connected by an “or” logical operator, set the resulting membership function as:

$$\max(f_{ij}^{k_{m_1}}, f_{ij}^{k_{m_2}}).$$

**STEP 8:** Set the fuzzy value $f_i^k$ of image $I$ satisfying the spatial relation definition $D^k_{ij}$ as:

$$f_{ij}^k = \max_{s=1}^{I} f_{ij}^k,$$

where $I$ is the number of possible combinations of $O_i$ and $O_j$ in image $I$.

**STEP 9:** Evaluate the fuzzy value $f_i^k$ of image $I$ in matching the given query. If two spatial relations $D^{k_1}_{ij}$ and $D^{k_2}_{ij}$ exist in the linguistic query and are connected by an “and” logical operation, set the resulting membership value as:

$$\min(f_{ij}^{k_1}, f_{ij}^{k_2}).$$

If they are connected by an “or” logical operator, set the resulting membership function as:

$$\max(f_{ij}^{k_1}, f_{ij}^{k_2}).$$

**STEP 10:** Output the images satisfying user-defined criteria in a descendant order of matching degrees.

Note that several kinds of user-defined criteria can be adopted in STEP 10. For example, users may require that only the best several matched images are output or that the images with their matching degrees larger than a threshold $\alpha$ are output.

5 Two Illustrating Examples

In this section, two examples are given to illustrate the proposed fuzzy image matching algorithm.

**Example 1:** Assume an image database includes six images $I_1$ to $I_6$, which are shown in Figure 3.
All the images have two objects A and B with different spatial relations. Assume each object in an image has been segmented out as a rectangle. An image table describing the information of objects is shown in Table 1 for effective image retrieval.

<table>
<thead>
<tr>
<th>Image_Name</th>
<th>ObjID</th>
<th>Obj_Name</th>
<th>XLB</th>
<th>XUB</th>
<th>YLB</th>
<th>YUB</th>
</tr>
</thead>
<tbody>
<tr>
<td>I₁</td>
<td>O₁₁</td>
<td>A</td>
<td>0.7</td>
<td>1.4</td>
<td>0.5</td>
<td>1.2</td>
</tr>
<tr>
<td>I₁</td>
<td>O₁₂</td>
<td>B</td>
<td>2.5</td>
<td>3.2</td>
<td>0.6</td>
<td>1.3</td>
</tr>
<tr>
<td>I₂</td>
<td>O₂₁</td>
<td>A</td>
<td>1.1</td>
<td>1.8</td>
<td>0.7</td>
<td>1.4</td>
</tr>
<tr>
<td>I₂</td>
<td>O₂₂</td>
<td>B</td>
<td>1.9</td>
<td>2.6</td>
<td>0.2</td>
<td>0.9</td>
</tr>
<tr>
<td>I₃</td>
<td>O₃₁</td>
<td>A</td>
<td>1.2</td>
<td>1.9</td>
<td>0.6</td>
<td>1.3</td>
</tr>
<tr>
<td>I₃</td>
<td>O₃₂</td>
<td>B</td>
<td>1.9</td>
<td>2.6</td>
<td>0.6</td>
<td>1.3</td>
</tr>
<tr>
<td>I₄</td>
<td>O₄₁</td>
<td>A</td>
<td>1.1</td>
<td>1.8</td>
<td>1</td>
<td>1.7</td>
</tr>
<tr>
<td>I₄</td>
<td>O₄₂</td>
<td>B</td>
<td>1.5</td>
<td>2.2</td>
<td>0.2</td>
<td>0.9</td>
</tr>
<tr>
<td>I₅</td>
<td>O₅₁</td>
<td>A</td>
<td>1.1</td>
<td>1.8</td>
<td>0.9</td>
<td>1.6</td>
</tr>
<tr>
<td>I₅</td>
<td>O₅₂</td>
<td>B</td>
<td>1.35</td>
<td>2.05</td>
<td>0.1</td>
<td>0.8</td>
</tr>
<tr>
<td>I₆</td>
<td>O₆₁</td>
<td>A</td>
<td>1.5</td>
<td>2.1</td>
<td>0.3</td>
<td>1.1</td>
</tr>
<tr>
<td>I₆</td>
<td>O₆₂</td>
<td>B</td>
<td>0.7</td>
<td>1.4</td>
<td>0.7</td>
<td>1.4</td>
</tr>
</tbody>
</table>

Assume now there is a query of finding the images where A is to the left of B. The algorithm proceeds as follows:

Step 1: The objects included in the query are collected. In this example, the query objects are A and B.

Step 2: The definition for the spatial relation “A to the left of B” is:

\[
\frac{XUB(A) - XLB(B)}{[XUB(A) - XLB(A)] + [XUB(B) - XLB(B)]}/2 < 0.
\]

Only one constraint exists in the definition.

Step 3: The images with both objects A and B are extracted. In this example, all the six images in Figure 3 are extracted.

Step 4: Steps 5 to 9 are done for images I₁ to I₆.

Step 5: Since only one constraint exists for the only one basic spatial relation in the query, m is thus 1. The left-side value of the constraint for each possible combination of A and B in each image is then calculated. In this example, only one combination of objects A and B exist for each image. Take image I₁ as an example. The value in I₁ is 1.3-2.5/[[1.3-0.7]+[3.2-2.5]]/2, which is -1.85. The left-side values of the constraint for all the six images are listed in the second column of Table 2.

Table 2: The left-side values and the fuzzy values of the constraint for the matched six images.

<table>
<thead>
<tr>
<th>Image</th>
<th>(XUB(A)−XLB(B))/2</th>
<th>Fuzzy value</th>
</tr>
</thead>
<tbody>
<tr>
<td>I₁</td>
<td>-0.77</td>
<td>1</td>
</tr>
<tr>
<td>I₂</td>
<td>-0.07</td>
<td>0.507</td>
</tr>
<tr>
<td>I₃</td>
<td>0</td>
<td>0.5</td>
</tr>
<tr>
<td>I₄</td>
<td>0.21</td>
<td>0.29</td>
</tr>
<tr>
<td>I₅</td>
<td>0.315</td>
<td>0.185</td>
</tr>
<tr>
<td>I₆</td>
<td>2</td>
<td>0</td>
</tr>
</tbody>
</table>

Steps 6: The left-side values \((XUB(A)−XLB(B))/2\) of the matched images in Table 2 are then transformed into fuzzy values by using the membership function of \(" > 0"\). The fuzzy values for images I₁ to I₆ are listed in the third column of Table 2.

Steps 7 to 9: These steps are omitted in this example since only one constraint must be checked and only one combination of objects A and B exists in each image. The matched degrees of images I₁ to I₆ are thus the same as those shown in the third column of Table 2.

Step 10: Assume the user-defined criterion is above a matching threshold \(\alpha=0.2\). The images I₁, I₂, I₃, and I₄ are thus output since their matching values are larger than or equal to 0.2.

Example 1 above shows a simplest fuzzy query in which only one constraint exists. Below, we give another example to show the processing of a query with more than one combination of matched objects.

Example 2: Assume an image database including six images I₁ to I₆ is shown in Figure 4.

![Figure 4](image-url)  
**Figure 4**: An image database used in Example 2.

The image table describing the information of objects in Figure 4 is shown in Table 3.

Table 3: The image table for the given six images in Figure 4.

<table>
<thead>
<tr>
<th>Image_Name</th>
<th>ObjID</th>
<th>Obj_Name</th>
<th>XLB</th>
<th>XUB</th>
<th>YLB</th>
<th>YUB</th>
</tr>
</thead>
<tbody>
<tr>
<td>I₁</td>
<td>O₁₁</td>
<td>A</td>
<td>0.7</td>
<td>1.4</td>
<td>0.5</td>
<td>1.2</td>
</tr>
<tr>
<td>I₁</td>
<td>O₁₂</td>
<td>B</td>
<td>2.5</td>
<td>3.2</td>
<td>0.6</td>
<td>1.3</td>
</tr>
<tr>
<td>I₂</td>
<td>O₂₁</td>
<td>A</td>
<td>1.1</td>
<td>1.8</td>
<td>0.7</td>
<td>1.4</td>
</tr>
<tr>
<td>I₂</td>
<td>O₂₂</td>
<td>B</td>
<td>1.9</td>
<td>2.6</td>
<td>0.2</td>
<td>0.9</td>
</tr>
<tr>
<td>I₃</td>
<td>O₃₁</td>
<td>A</td>
<td>1.2</td>
<td>1.9</td>
<td>0.6</td>
<td>1.3</td>
</tr>
<tr>
<td>I₃</td>
<td>O₃₂</td>
<td>B</td>
<td>1.9</td>
<td>2.6</td>
<td>0.6</td>
<td>1.3</td>
</tr>
<tr>
<td>I₄</td>
<td>O₄₁</td>
<td>A</td>
<td>1.1</td>
<td>1.8</td>
<td>1</td>
<td>1.7</td>
</tr>
<tr>
<td>I₄</td>
<td>O₄₂</td>
<td>B</td>
<td>1.5</td>
<td>2.2</td>
<td>0.2</td>
<td>0.9</td>
</tr>
<tr>
<td>I₅</td>
<td>O₅₁</td>
<td>A</td>
<td>1.1</td>
<td>1.8</td>
<td>0.9</td>
<td>1.6</td>
</tr>
<tr>
<td>I₅</td>
<td>O₅₂</td>
<td>B</td>
<td>1.35</td>
<td>2.05</td>
<td>0.1</td>
<td>0.8</td>
</tr>
<tr>
<td>I₆</td>
<td>O₆₁</td>
<td>A</td>
<td>1.5</td>
<td>2.1</td>
<td>0.3</td>
<td>1.1</td>
</tr>
<tr>
<td>I₆</td>
<td>O₆₂</td>
<td>B</td>
<td>0.7</td>
<td>1.4</td>
<td>0.7</td>
<td>1.4</td>
</tr>
</tbody>
</table>
Assume now there is a query of finding the images where \( A \) is to the right of \( B \) and \( C \) top-meets \( D \). The proposed algorithm proceeds as follows.

Step 1: The objects included in the query of finding the images where \( A \) is to the right of \( B \) and \( C \) top-meets \( D \) are collected. In this example, the query objects are \( A, B, C \) and \( D \).

Step 2: The definition for the first spatial relation "\( A \) is to the right of \( B \)" includes the following one constraint:

\[
\frac{\text{XUB}(B) - \text{XLB}(A)}{|\text{XUB}(B) - \text{XLB}(A)|} < 0.
\]

The definition for the second spatial relation "\( C \) top-meets \( D \)" includes the following three constraints:

\[
\frac{\text{YUB}(C) - \text{YLB}(D)}{|\text{YUB}(C) - \text{YLB}(D)|} = 0,
\]

\[
\frac{\text{XUB}(D) - \text{XLB}(C)}{|\text{XUB}(D) - \text{XLB}(C)|} > 0,
\]

\[
\frac{\text{XUB}(C) - \text{XLB}(D)}{|\text{XUB}(C) - \text{XLB}(D)|} > 0.
\]

The two spatial relations are then individually evaluated.

Step 3: The images with the four objects \( A, B, C \) and \( D \) are extracted. In this example, all the six images in Figure 4 are extracted.

Step 4: Steps 5 to 9 are done for each image.

Step 5: The left-side values of each constraint for each possible combination in each image is then calculated. Take image \( I_1 \) as an example. All the left-side values of the three constraints in the spatial relation "\( C \) top-meets \( D \)" for all images are shown in Table 4. Note that there are two possible combinations of objects \( C \) and \( D \) for image 3.

Table 4: The left-side values and the fuzzy values of the constraints with "\( C \) top-meets \( D \)" for the matched six images.

<table>
<thead>
<tr>
<th>Image</th>
<th>Combination of objects</th>
<th>Constraint 1 ( f_{AB}^{top-meets} )</th>
<th>Constraint 2 ( f_{AB}^{top-meets} )</th>
<th>Constraint 3 ( f_{CD}^{top-meets} )</th>
<th>( f_{CD}^{top-meets} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( I_1 )</td>
<td>CD</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>( I_1 )</td>
<td>CD</td>
<td>0</td>
<td>1</td>
<td>1.6</td>
<td>1</td>
</tr>
<tr>
<td>( I_1 )</td>
<td>CD</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>( I_1 )</td>
<td>CD</td>
<td>0</td>
<td>1</td>
<td>1.6</td>
<td>1</td>
</tr>
<tr>
<td>( I_1 )</td>
<td>CD</td>
<td>0</td>
<td>1</td>
<td>1.05</td>
<td>1</td>
</tr>
<tr>
<td>( I_1 )</td>
<td>CD</td>
<td>-0.727</td>
<td>0.273</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Similarly, the left-side value in the spatial relation "\( A \) is to the right of \( B \)" for each image is shown in Table 5.

Step 6: The left-side values of the matched images are then transformed into fuzzy values by using the membership functions of "\( = 0 \)" and "\( > 0 \)" in Figure 4.

Table 5: The left-side value and the fuzzy value of the constraint with "\( A \) is to the right of \( B \)" for the matched six images.

<table>
<thead>
<tr>
<th>Image</th>
<th>Combination of objects</th>
<th>Constraint</th>
<th>( f_{AB}^{right-of} )</th>
<th>( f_{CD}^{right-of} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( l_1 )</td>
<td>AB</td>
<td>-0.8</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>( l_2 )</td>
<td>AB</td>
<td>-0.8</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>( l_3 )</td>
<td>AB</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( l_4 )</td>
<td>AB</td>
<td>0</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>( l_5 )</td>
<td>AB</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( l_6 )</td>
<td>AB</td>
<td>0</td>
<td>0.5</td>
<td>0.5</td>
</tr>
</tbody>
</table>

1. The results are also listed in Tables 4 and 5.

Step 7: The fuzzy value of each combination satisfying the second spatial relation "\( C \) top-meets \( D \)" is calculated as \( \min f_{CD}^{top-meets} \). Results are shown in the \( f_{CD}^{top-meets} \) column of Table 4. The fuzzy value of each combination satisfying the first spatial relation "\( A \) is to the right of \( B \)" is the same as that calculated in Step 6 since only one constraint exists for this spatial relation.

Step 8: There are two combinations of \( C \) and \( D \) in image \( I_3 \). The match degree of \( I_3 \) with the spatial relation "\( C \) top-meets \( D \)" is thus the maximum of the fuzzy values of the two combinations. The results are listed in the last column of Table 4.

Step 9: Since the two spatial relations "\( A \) is to the right of \( B \)" and "\( C \) top-meets \( D \)" in the query are connected by an "and" logical operation, the minimum operation is adopted to find the final match degrees. The resulting match degrees are shown in Table 6.

Table 6: The resulting match degrees of the six images.

<table>
<thead>
<tr>
<th>Image</th>
<th>( f_{AB}^{right-of} )</th>
<th>( f_{CD}^{top-meets} )</th>
<th>Match degree</th>
</tr>
</thead>
<tbody>
<tr>
<td>( I_1 )</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>( I_2 )</td>
<td>1</td>
<td>0.9</td>
<td>0.9</td>
</tr>
<tr>
<td>( I_1 )</td>
<td>0</td>
<td>0.5</td>
<td>0</td>
</tr>
<tr>
<td>( I_1 )</td>
<td>0.5</td>
<td>0.9</td>
<td>0.5</td>
</tr>
<tr>
<td>( I_1 )</td>
<td>0</td>
<td>0.5</td>
<td>0</td>
</tr>
<tr>
<td>( I_6 )</td>
<td>0.5</td>
<td>0.273</td>
<td>0.273</td>
</tr>
</tbody>
</table>

Step 10: Assume the user-defined criterion is a matching threshold \( \alpha=0.2 \). The images \( I_1, I_2, I_6 \) and \( I_6 \) are thus sorted and output since their matching values are larger than or equal to 0.2.

6 Conclusions

In this paper, we have proposed a fuzzy image
matching algorithm to calculate the fuzzy match degrees of images with linguistic spatial queries. It consists of two main stages. In the first stage, the objects in a linguistic spatial query are first used to filter the images in the image database, thus avoiding some unnecessary checking in the second stage. The fuzzy match degrees between the converted spatial queries and the promising images are then calculated in the second stage by using the fuzzy operations and the membership functions. The proposed algorithm is thus suitable for images with uncertain objects and satisfies users’ linguistic queries in a flexible and efficient way.

References