Use of Affine Invariants in Locally Likely Arrangement Hashing for Camera-Based Document Image Retrieval

Tomohiro Nakai, Koichi Kise, and Masakazu Iwamura

Graduate School of Engineering, Osaka Prefecture University
1-1 Gakuen-cho, Sakai, Osaka, 599-8531 Japan
nakai@m.cs.osakafu-u.ac.jp, kise@cs.osakafu-u.ac.jp,
masa@cs.osakafu-u.ac.jp

Abstract. Camera-based document image retrieval is a task of searching document images from the database based on query images captured using digital cameras. For this task, it is required to solve the problem of "perspective distortion" of images, as well as to establish a way of matching document images efficiently. To solve these problems we have proposed a method called Locally Likely Arrangement Hashing (LLAH) which is characterized by both the use of a perspective invariant to cope with the distortion and the efficiency: LLAH only requires $O(N)$ time where N is the number of feature points that describe the query image. In this paper, we introduce into LLAH an affine invariant instead of the perspective invariant so as to improve its adjustability. Experimental results show that the use of the affine invariant enables us to improve either the accuracy from 96.2% to 97.8%, or the retrieval time from 112 msec./query to 75 msec./query by selecting parameters of processing.

1 Introduction

Document image retrieval is a task of searching document images relevant to a user's query. For meeting diverse needs from users, a wide variety of queries have been employed [1]. With document images as queries, the task of finding similar or equivalent document images has been considered. For scanned documents it is called "document image matching" or "duplicate detection" [2, 3]. This paper concerns a kind of document image matching with camera captured documents as queries. We call this task "camera-based document image retrieval". The technique of camera-based document image retrieval can be used as bases of several applications. For example, we can extract annotations of documents by retrieving original documents based on captured query images with annotations and comparing them.

In order to deal with camera captured images, various kind of problems including perspective distortion, uneven lighting and focusing should be solved [4, 5]. We are concerned here with the problem of perspective distortion. An ordinary way of dealing with the distortion is to normalize the image by estimating parameters of perspective transformation [6]. However, the normalization relies heavily on wide justified text regions that are not necessarily included in or successfully extracted from camera captured images.

Another way is to retrieve images regardless of perspective distortion. Geometric hashing (GH) [7] is a well-known way of indexing and retrieval of images regardless of geometric distortion. GH employs feature points extracted from images to register images in the database as well as to retrieve images using queries. A drawback of GH is that its computational complexity is far beyond linear to the number of feature points: it is $O(N^5)$ where N is the number of feature points in the image under perspective distortion. Thus it is difficult to apply GH for the retrieval of images with a lot of feature points such as document images.

To solve this problem, we have proposed a method of indexing and retrieval for images represented by coplanar points [8]. In this paper we call this method Locally Likely Arrangement Hashing (LLAH). In LLAH, each feature point is registered in the hash table with features defined based on arrangements of neighboring feature points. The method is called "locally likely arrangement" because possible arrangements of neighboring points in a local area are enumerated and registered. Since the computational complexity of LLAH is $O(N)$, efficient retrieval has been realized.

In [8], we have employed a perspective invariant called the cross-ratio for calculation of features from the feature points. Calculation of the cross-ratio requires five feature points, which limit adjustability of balancing computational complexity and accuracy of the method. In this paper, we introduce features calculated from less feature points (four points) to improve the adjustability. It is based on the fact that the transformation of feature points in the local area can be approximated as affine transformation even under perspective transformation. As an invariant, therefore, we utilize an affine invariant. From the experimental results using 10,000 database images and 235 query images, it is shown that, as compared to the cross-ratio, use of the affine invariant enables us to improve either the accuracy from 96.2% to 97.8%, or the retrieval time from 112 msec./query to 75msec./query by selecting parameters of processing.

2 Locally Likely Arrangement Hashing

2.1 Geometric invariants

In LLAH, we use geometric invariants calculated from f coplanar points. Geometric invariants are the values which keep unchanged under geometric transformation. There are several types of geometric invariants depending on the types of geometric transformation. Different types of geometric invariants require different numbers of f.

1. **Cross-Ratio**
   The cross-ratio is known as an invariant of perspective transformation. It is used in our previous work [8] since query images captured by digital cameras suffer from perspective transformation. The cross-ratio is calculated using coordinates of five coplanar points ($f = 5$) ABCDE as follows:

$$\frac{P(A,B,C)P(A,D,E)}{P(A,B,D)P(A,C,E)} \tag{1}$$

   where P(A,B,C) is the area of a triangle with apexes A, B, and C [9]. Since the cross-ratio is a perspective invariant, its value keeps unchanged even if coordinates of points ABCDE change by perspective distortion.

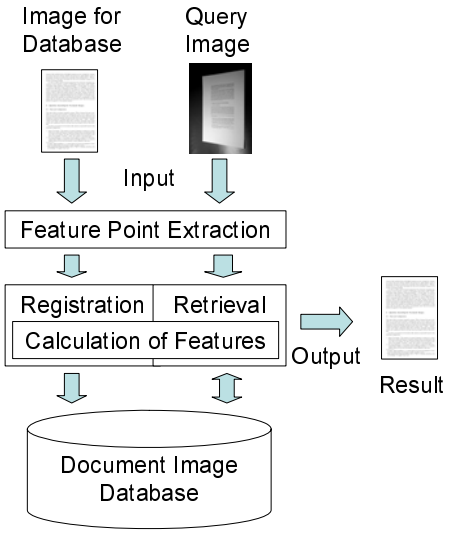


Fig. 1. Overview of processing.

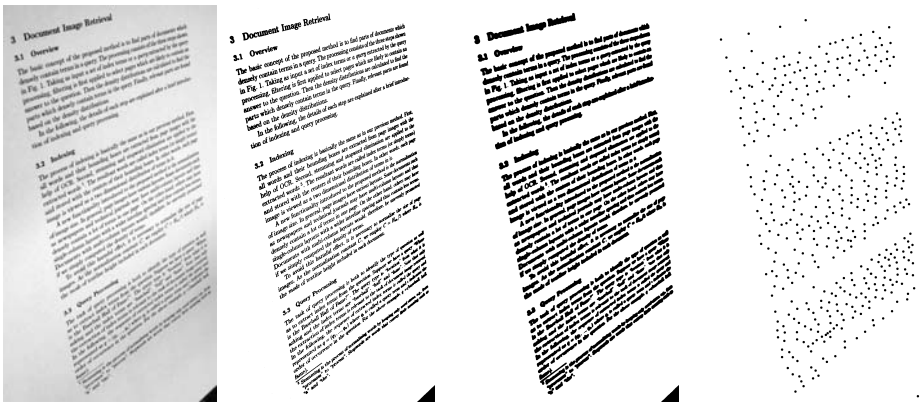
2. **Affine invariant**
   We also have invariants for affine transformation. Affine transformation is more restrictive than perspective transformation since it preserves the parallelism of lines. Because perspective transformation of a small limited area can be approximated as affine transformation, it would be possible to apply an affine invariant instead of the cross-ratio.
   In this paper we utilize an affine invariant defined using four coplanar points ($f = 4$) ABCD as follows:

$$\frac{P(A,C,D)}{P(A,B,C)} \tag{2}$$

Although the values of invariants are continuous, they must be converted into discrete values in order to be used as indices of the hash table. In LLAH, continuous values are converted into k discrete values by taking into account their frequency: the discretization step is finer for values occurring more frequently. To be precise, discrete values are assigned in proportion to the frequency of values of invariants using a histogram of values of invariants obtained in a preliminary experiment.

2.2 Overview of processing

Figure 1 shows the overview of processing. At the step of feature point extraction, a document image is transformed into a set of feature points. Then the feature points are inputted into the registration step or the retrieval step. These steps share the step of calculation of features. In the registration step, every feature point in the image is registered into the document image database using its feature. In the retrieval step, the



(a) Input image. (b) Binarized image. (c) Connected components. (d) Feature points.

Fig. 2. Feature point extraction.

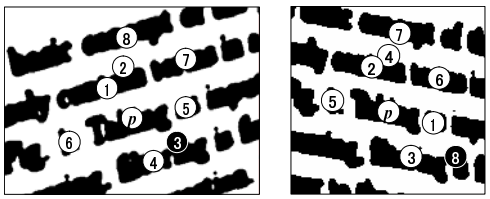


Fig. 3. Change of nearest feature points by perspective distortion.

document image database is accessed with features to retrieve images by voting. We explain each step in the following.

2.3 Feature point extraction

An important requirements of feature point extraction is that feature points should be obtained identically even under the perspective distortion, noise, and low resolution. To satisfy this requirement, we employ centroids of word regions as feature points.

The details of processing are as follows. First, the input image (Fig. 2(a)) is adaptively thresholded into the binary image (Fig. 2(b)). Next, the binary image is blurred using the Gaussian filter whose parameters are determined based on an estimated character size (the square root of the mode of areas of connected components). Then, the blurred image is adaptively thresholded again (Fig. 2(c)). Finally, centroids of word regions (Fig. 2(d)) are extracted as feature points.

2.4 Calculation of features

The feature is a value which represents a feature point of a document image. In order to realize successful retrieval, the feature should satisfy the following two requirements.

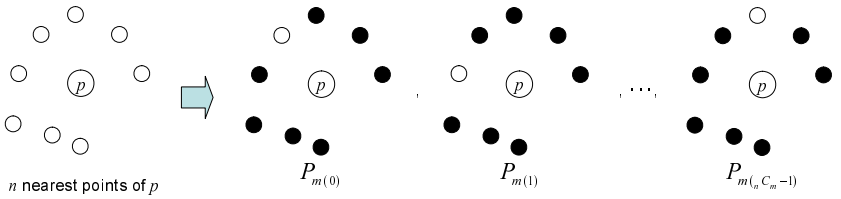


Fig. 4. All possible combinations of $m(=7)$ points from $n(=8)$ nearest points are examined.

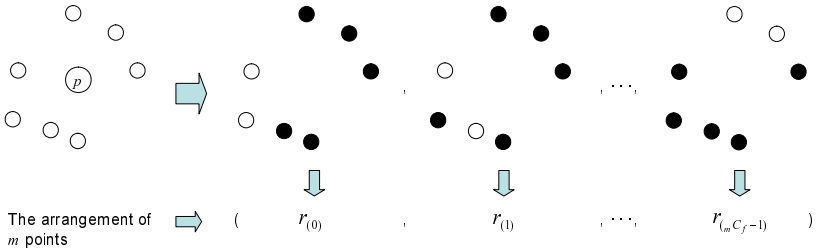


Fig. 5. The arrangement of $m(=7)$ points is described as a sequence of invariants calculated from all possible combinations of $f(=5)$ points.

One is that the same feature should be obtained from the same feature point even under distortion. If different features are obtained from the same feature point at registration and retrieval processes, the corresponding document image cannot be retrieved. We call this requirement "stability of the feature". The other requirement is that different features should be obtained from different feature points. If the same feature is obtained from different feature points, not only the corresponding document image but also other document images are retrieved. We call this requirement "discrimination power of the feature". Both two requirements, the stability and the discrimination power, have to be satisfied for successful retrieval.

Let us explain the stability first. The simplest definition of the feature of a feature point p is to use f nearest feature points from p ($f = 5$ for the cross-ratio, and $f = 4$ for the affine invariant). However, nearest feature points can be changed by the effect of perspective distortion as shown in Fig. 3. Hence the invariant from f nearest points is not stable. In order to solve this problem, we utilize feature points in a broader local area. In Fig. 3, it is shown that 7 points of 8 nearest points are common. In general, we assume that common m points exist in n nearest neighbors under some extent of perspective distortion. Based on this assumption, we use common m points to calculate a stable feature. As shown in Fig. 4, common m points are obtained by examining all possible combinations $P_{m(0)}, P_{m(1)}, \cdots, P_{m({}_nC_m-1)}$ of m points from n nearest points. As long as the assumption holds, at least one combination of m points is common. Thus a stable feature can be obtained.

1: **for each** $p \in$ {All feature points in a database image} **do**
2: $P_n \leftarrow$ The nearest n points of p (clockwise)
3: **for each** $P_m \in$ { All combinations of m points from P_n } **do**
4: **for each** $P_f \in$ { All combinations of f points from P_m } **do**
5: $r_{(i)} \leftarrow$ The invariant calculated with P_f
6: **end for**
7: $H_{\mathrm{index}} \leftarrow$ The hash index calculated by Eq. (3).
8: Register the item (document ID, point ID, $r_{(0)}, \cdots, r_{({}_mC_f-1)}$) using H_{index}
9: **end for**
10: **end for**

Fig. 6. Registration algorithm.

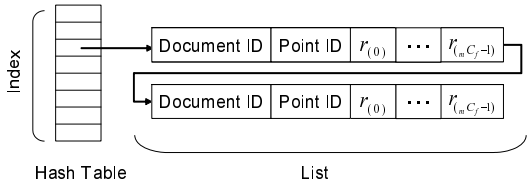


Fig. 7. Configuration of the hash table.

Let us move to the second requirement. The simplest way of calculating the feature from m points is to set m=f and calculate the cross-ratio or the affine invariant from f points. However, such a simple feature lacks the discrimination power because it is often the case that similar arrangements of f points are obtained from different feature points. In order to increase the discrimination power, we utilize feature points of a broader area. It is performed by increasing the number $m(> f)$. As m increases, probability that different feature points have similar arrangement of m points decreases. As shown in Fig. 5, an arrangement of m points is described as a sequence of discretized invariants $(r_{(0)}, r_{(1)}, \cdots, r_{({}_mC_f-1)})$ calculated from all possible combinations of f feature points taken from m feature points.

The following is the summary of calculation of features. For each feature point, its n nearest points are obtained. Then all possible ${}_nC_m$ combinations of m points are generated from n points. Features are defined as sequences of invariants by taking ${}_mC_f$ combinations from m points in a certain fixed order.

2.5 Registration

Let us turn to the registration step. Figure 6 shows the algorithm of registration of document images to the database. In this algorithm, the document ID is the identification number of a document, and the point ID is that of a point.

Next, the index H_{index} of the hash table is calculated by the following hash function:

$$H_{\mathrm{index}} = \left(\sum_{i=0}^{{}_mC_f-1} r_{(i)} k^i \right) \bmod H_{\mathrm{size}} \tag{3}$$

1: **for each** $p \in$ {All feature points in a query image} **do**
2: $P_n \leftarrow$ The nearest n points of p (clockwise)
3: **for each** $P_m \in$ { All combinations of m points from P_n } **do**
4: **for each** $P'_m \in$ { Cyclic permutations of P_m } **do**
5: **for each** $P_f \in$ { All combinations of f points from P'_m } **do**
6: $r_{(i)} \leftarrow$ The invariant calculated with P_f
7: **end for**
8: $H_{\mathrm{index}} \leftarrow$ The hash index calculated by Eq. (3).
9: Look up the hash table using H_{index} and obtain the list.
10: **for each** Item of the list **do**
11: **if** Conditions 1 to 3 are satisfied **then**
12: Vote for the document ID in the voting table.
13: **end if**
14: **end for**
15: **end for**
16: **end for**
17: **end for**
18: Return the document image with the maximum votes.

Fig. 8. Retrieval algorithm.

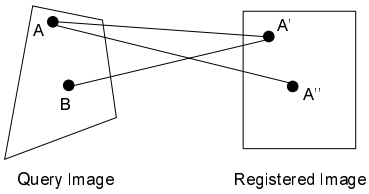


Fig. 9. Incorrect correspondence.

where $r_{(i)}$ is the discrete value of the invariant, k is the level of quantization of the invariant, and H_{size} is the size of the hash table.

The item (document ID, point ID, $r_{(0)}, \cdots, r_{({}_mC_f-1)}$) is registered into the hash table as shown in Fig. 7 where chaining is employed for collision resolution.

2.6 Retrieval

The retrieval algorithm is shown in Fig. 8. In LLAH, retrieval results are determined by voting on documents represented as cells in the voting table.

First, the hash index is calculated at the lines 5 to 8 in the same way as in the registration step. At the line 9, the list shown in Fig. 7 is obtained by looking up the hash table. For each item of the list, a cell of the corresponding document ID in the voting table is incremented.

However, the sequence of invariants $r_{(0)} \cdots r_{({}_mC_f-1)}$ is not necessarily identical for items with the same value H_{index} of the hash function. In order to remove items with different sequences of invariants, the following condition is employed.

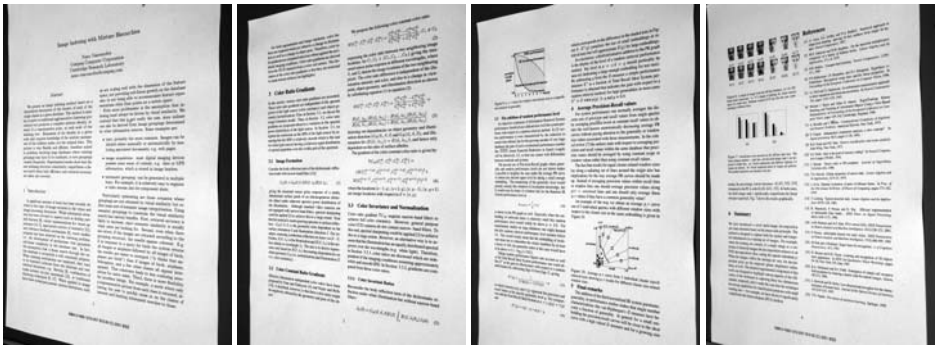


Fig. 10. Examples of query images.

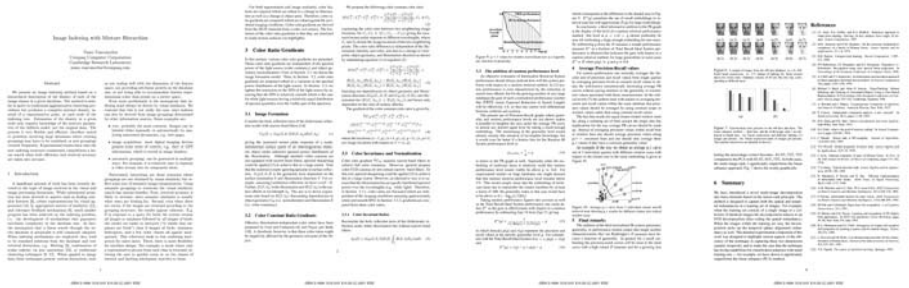


Fig. 11. Examples of images in database.

Condition 1: All values of $r_{(0)} \cdots r_{({}_mC_f-1)}$ in the item are equal to those calculated at the lines 5 to 7 for P'_m.

If only the condition 1 is employed, we face the following two types of inconsistency shown in Fig. 9: (Type 1) A point (A) in the query image corresponds to more than one point (A' and A") in a registered image. (Type 2) A point (A') in a registered image corresponds to more than one point (A and B) in the query image. In order to avoid such inconsistent correspondences, following conditions are employed.
Condition 2: It is the first time to vote for the document ID with the point p.
Condition 3: It is the first time to vote for the point ID of the document ID.
The conditions 2 and 3 are aimed at removal of types 1 and 2 inconsistency, respectively.

3 Experimental results

3.1 Overview

In order to examine effectiveness of the affine invariant, we measured accuracy and processing time. Query images were captured from a skew angle using a digital camera CANON EOS Kiss Digital (also known as EOS-300D; 6.3 million pixels) with

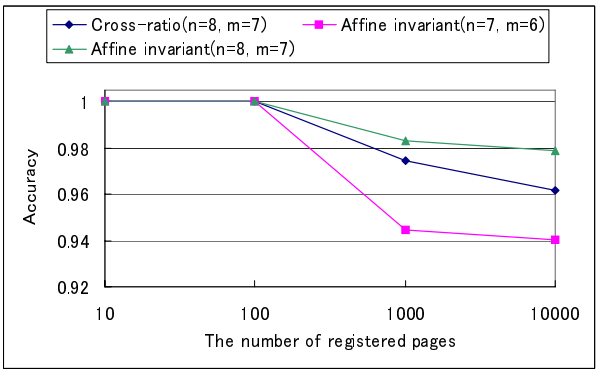


Fig. 12. Accuracy of retrieval.

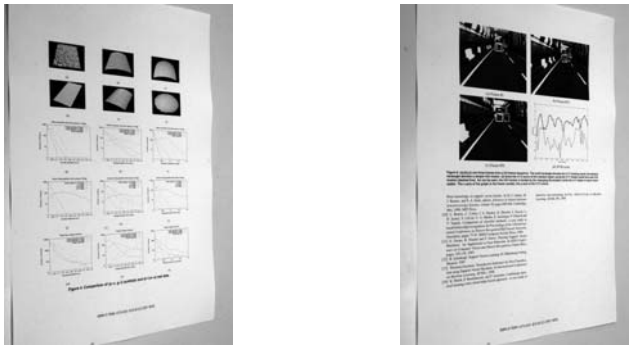
a lens EF-S 18-55mm USM. The size of query images is 2,048 $\times$ 3,072. The number of query images was 235 (We added 185 images to 50 images in [8]). Figure 10 shows examples of query images. As documents in the database we employed 10,000 page images converted with 200 dpi from PDF files of single- and double-column English papers collected mainly from CD-ROM proceedings. Their size is about 1,700 $\times$ 2,200. Figure 11 shows examples of images in the database. Note that the pages in the database look quite similar because most pages are from scientific papers formatted according to the same style file. Experiments were performed on a workstation with AMD Opteron 1.8GHz CPUs and 4GB memory. We used some sets of parameters n, m, k with which both high accuracy and short processing time were realized in a preliminary experiment. The value of k is set to the best with n and m. A set of parameters is described as cross-ratio(n, m) (cross-ratio with parameters n and m) or affine(n, m) (affine invariant with parameters n and m). As for the cross-ratio, parameters were set to $n = 8, m = 7, k = 18$ (cross-ratio(8, 7)). As for the affine invariant, parameters were set to $n = 7, m = 6, k = 25$ (affine(7, 6)) and $n = 8, m = 7, k = 7$ (affine(8, 7)). H_{size} was set to 1.28×10^8.

3.2 Accuracy of retrieval

We first analyzed the relationship between the size of the database (the number of registered pages) and the accuracy of retrieval (the rate that the correct page receives the maximum votes).

Figure 12 shows the results. Both the cross-ratio and the affine invariant yielded high accuracy. The highest accuracy was obtained with affine(8, 7). Second was cross-ratio(8, 7) and third was affine(7, 6). This is due to the discrimination power of the feature. Affine(8, 7), cross-ratio(8, 7), and affine(7, 6) have ${}_7C_4 = 35$, ${}_7C_5 = 21$, and ${}_6C_4 = 15$ invariants in their feature, respectively. More invariants in the feature resulted in higher accuracy.

Figure 13(a) shows an examples of query images which caused a failure on retrieval. For this query image, the correct image in the database was not retrieved with cross-



(a) Failed on all sets of parameters. (b) Failed on cross-ratio(8, 7) and affine(7, 6).

Fig. 13. Erroneous cases.

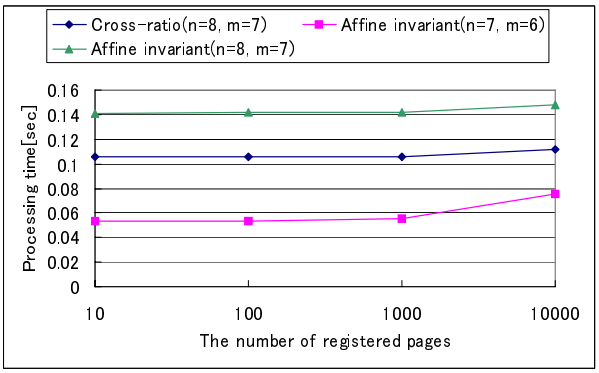


Fig. 14. The relationship among the number of registered pages and processing time.

ratio(8, 7), affine(8, 7), and affine(7, 6). This is because this query image is covered with many figures and little text regions. Since the current feature point extraction utilizes centroids of word regions, it does not work well to such images. Figure 13(b) shows another example of erroneous cases. For this image, retrieval was failed with cross-ratio(8, 7) and affine(7, 6). However, thanks to the discrimination power of affine(8, 7), the correct image was obtained in spite of the small text regions.

As shown in Fig. 12, the growth of the number of pages decreases the accuracy. This is because a larger database is more likely to have different pages with the same features.

3.3 Processing time

Figure 14 shows results of experiments about processing time. Contrary to the case of accuracy, affine(7, 6) showed the highest performance. Second was the cross-ratio(8, 7)

and third was affine(8, 7). We consider this was caused by the difference of computational complexity for calculating features. This figure also indicates that the increase of the number of registered pages extended processing time. We consider this was due to the increase of collision in the hash table.

3.4 Discussion

From the experimental results, it is shown that the affine invariant can be used as an invariant for the retrieval of perspectively distorted document images. This is because each part of the document images is approximately affine transformed. Therefore it is expected that the cross-ratio or the affine invariant can be used as an invariant for the retrieval of non-perspectively distorted document images if the distortion of each local area is approximated as perspective or affine transformation.

4 Related work

LLAH can be considered to be a method of planar object recognition if registered document images are viewed as object models. There have been many methods of object recognition which utilize invariants as LLAH does. In this section, we describe similar methods and differences from them.

4.1 Geometric hashing

In GH, all feature points of models are registered into the hash table using 2 to 4 selected points for defining a local coordinate basis. The number of points b for the basis depends on the kind of invariance: $b = 2$ for similarity, $b = 3$ for affine, and $b = 4$ for perspective transformation. Registration is performed on every possible basis. Retrieval is performed by looking up the hash table using selected bases and voting. GH is similar to the LLAH in the following points.

- invariant indexing of feature points,
- registration of an object by registering all feature points,
- utilization of the hashing and the voting techniques.

However, LLAH is superior to GH in terms of computational complexity. In LLAH, features are calculated from limited neighboring points for each feature point. Hence the computational complexity of the LLAH is $O(N)$ where N is the number of feature points in each model. On the other hand, in GH each feature point is registered using every possible basis. Hence the computational complexity of the registration process is $O(N^{b+1})$. For example, for the case of perspective transformation, the computational complexity of GH is $O(N^5)$ since four points are necessary for the basis. For example, since a document image in the database for the experiments has 630 feature points on average, use of GH requires $630^5 = 10^{14}$ times of point registration to the hash table for each registered image. Thus GH is prohibitive for retrieval with many feature points such as document images.

4.2 Other methods

Many invariant-based object recognition methods such as [10] and [11] have so far been proposed. However, improvement of the discrimination power of the feature is not employed in these methods. For example, the feature is simply a cross-ratio of five connected line segments in [11]. It is difficult in our case to adopt such a simple indexing, because a huge number of points have similar values of the invariant. In order to avoid this problem, the arrangement of points in the broader area is employed in LLAH; this discriminative feature realizes both high accuracy and computational efficiency.

5 Conclusion

We have proposed a method of camera-based document image retrieval. The method is characterized by the ways of improving both the stability and the discrimination power of the feature defined based on the invariants. High accuracy and efficiency of LLAH were shown by the experimental results. It is also shown that the affine invariant can be used in LLAH. The affine invariant is not invariant under perspective transformation which occurs on camera captured images. However, in methods which focus on local areas such as LLAH, the affine invariant can be used as an approximated invariant. Since the affine invariant requires fewer points than the cross-ratio, its use enables us to make retrieval system adjustable: accuracy oriented or speed oriented. Future work includes experiments with partially captured images of queries and an extension of the method to object retrieval in scene images.

References

1. D. Doermann. The Indexing and Retrieval of Document Images: A Survey. *Computer Vision and Image Understanding*, **70**, 3, pages 287–298, 1998.
2. J. J. Hull. Document image matching and retrieval with multiple distortion-invariant descriptors. *Document Analysis Systems*, pages 379–396, 1995.
3. D. Doermann, H. Li and O. Kia. The detection of duplicates in document image databases. *Proc. ICDAR'97*, pages 314–318, 1997.
4. D. Doermann, J. Liang and H. Li. Progress in camera-based document image analysis. *Proc. ICDAR'03*, pages 606–616, 2003.
5. P. Clark and M. Mirmehdi. Recognising text in real scenes. *IJDAR*, **4**, pages 243–257, 2002.
6. S. Pollard and M. Pilu. Building cameras for capturing documents. *IJDAR*, **7**, pages 123–137, 2005.
7. H. J. Wolfson and I. Rigoutsos. Geometric hashing: an overview. *IEEE Computational Science & Engineering*, Vol. 4, No. 4, pages 10–21, 1997.
8. T. Nakai, K. Kise and M. Iwamura. Hashing with Local Combinations of Feature Points and Its Application to Camera-Based Document Image Retrieval. *Proc. CBDAR'05*, pages 87–94, 2005.
9. T. Suk and J. Flusser. Point-based projective invariants. *Pattern Recognition*, 33, pages 251–261, 2000.
10. B. Huet and E. R. Hancock. Cartographic indexing into a database of remotely sensed images. *WACV96*, pages 8–14, 1996.
11. C. A. Rothwell, A. Zisserman, D. A. Fosyth and J. L. Mundy. Using projective invariants for constant time library indexing in model based vision. *Proc. BMVC*, pages 62–70, 1991.