2011 International Conference on Document Analysis and Recognition

Real-Time Document Image Retrieval for a 10 Million Pages Database with a Memory Efficient and Stability Improved LLAH

Kazutaka Takeda, Koichi Kise and Masakazu Iwamura
Dept. of CSIS, Graduate School of Engineering
Osaka Prefecture University
1-1 Gakuen-cho, Naka, Sakai, Osaka, 599-8531 Japan
takeda@m.cs.osakafu-u.ac.jp, {kise, masa}@cs.osakafu-u.ac.jp

***Abstract*—This paper presents a real-time document image retrieval method for a large-scale database with Locally Likely Arrangement Hashing (LLAH). In general, when a database is scaled up, a large amount of memory is required and retrieval accuracy drops due to insufficient discrimination power of features. To solve these problems, we propose three improvements: memory reduction by sampling feature points, improvement of discrimination power by increasing the number of feature dimensions and stabilizing features by reducing redundancy. From the experimental results, we have confirmed that the proposed method realizes 50% memory reduction, and achieves 99.4% accuracy and 38ms processing time for a database of 10 million pages.**

***Keywords*-Document image retrieval, Real-time 10 million pages processing, LLAH, Large-scale database**

I. Introduction

Recently, camera phones have become very popular devices. Moreover, the quality of mobile phone cameras is comparable to that of ordinary digital cameras. Therefore, we are in the situation that people always carry high resolution digital cameras. For this reason, image retrieval with queries captured by a digital camera is paid more attention. In this research, we especially focus on document image retrieval, which is a task to find a document image from a database corresponding to a query obtained by capturing a document. This technique can provide users with the information which is associated with the retrieved document in the database. In other words, with the help of document image retrieval, paper documents can be viewed as media for accessing various information; images, movies, texts and more.

For this purpose, the document image retrieval method based on a hashing technique called Locally Likely Arrangement Hashing (LLAH) has already been proposed [1]. In this method, images are retrieved based on their features which consist of geometric invariants. LLAH is robust to variation of camera angles since geometric invariants are stable under perspective distortion. LLAH is also known for its fast retrieval which enables a real-time processing [2]. Moreover, LLAH has already been extended for retrieval of documents in various languages [3]. Owing to the above property, LLAH has been applied to Augmented Reality [4] and Camera-Pen [5].

However, LLAH has two problems for scaling up the database. First, a large amount of memory is required; 150GB memory is needed to realize high accuracy on a 10 million pages database. This inefficiency of memory space restricts scalability of LLAH. Second, retrieval accuracy decreases when the size of database is large because similar features are more likely to be extracted from a larger database. In order to overcome this problem, we have to increase the discrimination power of the feature.

In this paper, we propose some approaches for improvements of LLAH to solve the problems stated above. The basic idea to reduce memory is to sample feature points stored in the database. In order to increase the discrimination power, we increase the number of dimensions of features. However, as the number of dimensions increase, features become unstable. Therefore, we improve the stability of features by removing redundant dimensions. From the experimental results, we have confirmed that the proposed method realizes 50% memory reduction. It has also achieved retrieval accuracy of 99.4% and processing time of 38ms for the database of 10 million pages.

II. Document image retrieval with original LLAH

A. Overview of processing

Figure 1 shows the overview of processing of LLAH. First, a document image is transformed into a set of feature points. Then, features are calculated from arrangements of the feature points. In the storage step, every feature point in the image is stored into the document image database using its feature. In the retrieval step, the document image database is accessed with features to retrieve images by voting. We explain each step in the following.

B. Feature point extraction

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

1520-5363/11 $26.00 © 2011 IEEE
DOI 10.1109/ICDAR.2011.213

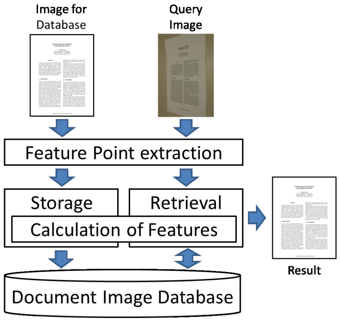


Figure 1. Overview of processing.

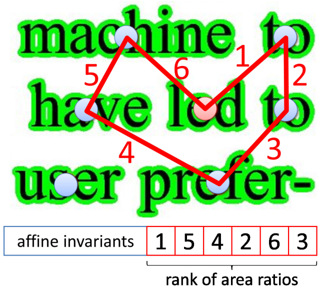


Figure 2. Additional feature.

perspective transformation of a small limited area is weak, the centroids could be considered as approximate invariants.

The process is as follows. First, the input image is adaptively thresholded into the binary image. Next, the binary image is blurred using the Gaussian filter. The blurred image is adaptively thresholded again. Finally, centroids of word regions are extracted as feature points.

C. Calculation of features

The feature of LLAH has the following four characteristics. The first one is that a feature is defined for each feature point in order to realize robustness and availability under occlusion. The second one is that a feature is calculated using geometric invariants in order for invariance to perspective distortion which occurs in camera-captured images. In concrete term, we utilize the affine invariant defined with four coplanar points ABCD as follows:

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

where $P(A,B,C)$ is the area of a triangle with apexes A, B and C. The third one is that a feature consists of multiple affine invariants calculated from multiple feature points so as to increase discrimination power of the feature. In concrete, a sequence of discretized affine invariants $(r_{(0)}, \cdots, r_{(\binom{m}{4}-1)})$ is calculated from m nearest points including the point in question. In fact, the invariants calculated from all possible combinations of four points from m points are utilized as a feature. The last one is that multiple features per feature point are calculated from nearest $n(> m)$ points. Specifically, $\binom{n}{m}$ features are calculated from all possible combinations of m points from n points. In the original LLAH, $n = 7$, $m = 6$ are utilized.

Moreover, the rank of area ratio of word regions is also employed as an additional feature. Figure 2 shows an example. For example, 1 indicates the area ratio of the word region of "led" to that of "to". As shown in Fig. 2, the additional feature is concatenated to the affine invariants. Therefore, the number of dimension of the feature is $(\binom{m}{4} + m)$.

D. Storage

Every feature point is stored in the database in accordance with its feature. The index H_{index} of the hash table is calculated by the following hash function:

$$\left(\sum_{i=0}^{(\binom{m}{4}+m)} r_{(i)} d^i\right) = QH_{\text{size}} + H_{\text{index}} \tag{2}$$

where $r_{(i)}$ is a discrete value of the invariant, d is the level of quantization of the invariant, and H_{size} is the size of the hash table. Q is uniquely determined for a feature. Therefore, the item (document ID, point ID and Q) is stored into the hash table where chaining is employed for collision resolution. Consequently, we can make sure that each item of the list has the same feature by using Q in place of the feature in retrieval.

E. Retrieval

In LLAH, the result of retrieval is determined by voting on documents represented as cells in the voting table.

H_{index} and Q are calculated for each feature point of a query image in the same way as in the storage step. The list of items (document ID, point ID and Q) is obtained by looking up the hash table. For each item, a cell of the corresponding document ID in the voting table is incremented if it has the same Q. Eventually, the document obtained the maximum votes is returned as the retrieval result.

III. Proposed Method

A. Reduction of required amount of memory

In the original LLAH, all extracted feature points are stored in the database. However, there is no need to store all points since retrieval results are determined by voting. Therefore, we reduce memory consumption by sampling points stored in the hash table.

When we sample feature points, it is important that sampled feature points are evenly distributed in a document. If the distribution of sampled points is uneven, the document has densely both sampled regions and sparsely sampled regions. In such a case, the system cannot retrieve documents

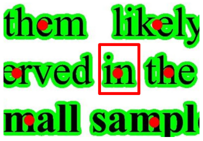


Figure 3. Sampling feature points.

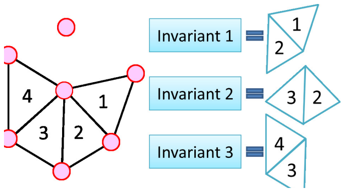


Figure 4. Invariants and their redundancy.

by capturing sparse regions because the number of points in the regions is not enough.

In the proposed method, we focus on area of word regions in order to sample points which are evenly distributed. To be precise, we sample the feature points extracted from word regions whose area is smaller than that of surrounds. In Fig. 3, the feature points extracted from "in" is sampled. Since such points are evenly distributed in a document, we can satisfy the requirement stated above. Moreover, when the area of the word region is smallest, it is more robust against perspective distortion since the distances to the n nearest points become shorter. However, the number of sampled points in this way is too small. Thus we store k nearest points from each sampled point. In the propose method, k is determined so that the number of feature points in a document is 200. If the number of all extracted feature point is less than 200, we do not sample points.

B. Additional descriptor

When we apply the original version of LLAH to a large-scale database, the number of collisions occurs in the hash table increases. This causes a number of erroneous votes in the retrieval process. Therefore, we need to modify features so that they can be more discriminative to reduce the number of collisions.

The idea to modify features is to increase the dimensions of features. The feature of original LLAH has 21 dimensions when the parameters are $n = 7$, $m = 6$. This number of dimensions has enough discrimination power in original LLAH because the database has only 10,000 pages. In the proposed method, we increase the dimensions by modifying the values to $n = 8, m = 7$. Consequently, the number of affine invariants is $\binom{7}{4} = 35$ and the number of area ratio features is 7, resulting in the total number of dimensions 42.

However, the above method has a problem on the stability of features. In LLAH, it is required that all dimensions of a feature must be identical to be considered to correspond in retrieval. However, if the number of dimension increases, there is an increasing possibility that different invariants are calculated due to the fluctuation of invariants. From this reason, the stability of features decreases in exchange for increasing the discrimination power. To solve this problem, we reduce the redundant dimensions of features. As show Fig. 4, we utilize an identical triangle to calculate two different invariants, which causes correlation. In Fig. 4, the invariant 2 has relationship to invariant 1 and invariant 3. This redundancy can be resolved by deleting the invariant 2. As shown in the above example, we select triangles for calculating invariants in the way that no triangles share invariants. As a result, the number of dimensions becomes 24.

Figure 5. Example of images in the database.

Figure 6. Example of query images.

IV. Experiments

A. Ex.1: Scalability

In order to examine effectiveness of improvements introduced in this paper, we tested the original and the improved versions of LLAH. Effectiveness of the improvements was measured by the required amount of memory, processing time per query and accuracy of three versions of LLAH: the original LLAH, the memory reduced version and the proposed method in this paper. Note that the memory reduced version was improved by only sampling feature points mentioned in III-A and the number of sampled feature point was up to 200 per document.

For the experiments, we made four databases which include different number of pages: 0.01 million, 0.1 million, 1 million and 10 million. An example of images in the database is shown in Fig. 5. Query images were captured from an elevation angle of 60 degrees using a digital camera with 1,200 million pixels. The number of query images was 1,003, whose example is shown in Fig. 6. Since the angle with which query images are captured (60 degrees) is different from that of the images in the database (90 degrees), experiments performed with these query images and the database would demonstrate robustness of the proposed method to perspective distortion. Experiments were performed on a workstation with AMD Opteron 2.8GHz CPUs and 128GB memory. H_{size} was set to $2^{30} - 1$.

1) Required amount of memory: Figure 7 shows the amount of required memory of the three versions of LLAH. These values include the size of hash table (8GB). The original LLAH can not deal with a 10 million pages database. Therefore, the value of original LLAH with a 10 million pages database is an estimated value. The memory reduced version of LLAH achieved 50% memory reduction. On the other hand, the proposed method of LLAH required a larger amount of memory than the memory reduced version. This is because the number of features calculated from one feature

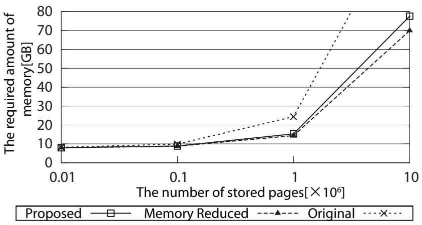


Figure 7. Relationship between the number of stored pages and the required amount of memory.

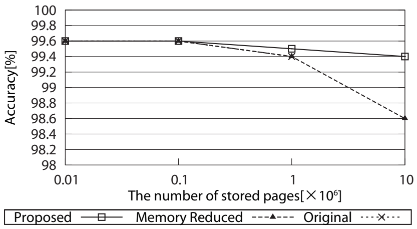


Figure 8. Relationship between the number of stored pages and accuracy.

point is $\binom{8}{7} = 8$ in the proposed method, while $\binom{7}{6} = 7$ in the memory reduced method.

2) Accuracy: Figure 8 shows retrieval accuracy of each version of LLAH. With the 10 million pages database, the proposed method demonstrated higher performance than the memory reduced version. In the proposed method, the average number of erroneous votes for each query was 144, while it was 443 in the memory reduction version. Since the number of erroneous votes was decreased by increase of discrimination power, the proposed method achieved high accuracy. Figure 9 shows an example of query images which could not be retrieved. This query image consists of many figures and few text regions. Since the current feature point extraction utilizes centroids of word regions, it does not work well with such images.

3) Processing time of retrieval: Figure 10 shows processing time for each version of LLAH. The proposed version of LLAH realized processing time of 38ms for the database of 10 million pages. Therefore, the proposed method can retrieve documents in real-time.

When the number of pages stored in the database is small, the original version of LLAH shows higher performance. This is because of the difference of computational complexity. In the original method $\binom{7}{6} = 7$ features per feature point and $\binom{6}{4} = 15$ invariants per feature are calculated. In the proposed method $\binom{8}{7} = 8$ features per point and $\binom{7}{4} = 35$ invariants per feature are calculated. As a result, the original has faster since its computational complexity is less than that of the proposed method. However the proposed method realized higher performance with the large-scale database. This is because of the difference of the number of collision in the hash table. Figure 11 shows the average length of lists whose length is not 0. For the database of 1 million pages, the average length of the original LLAH is twice as compared to that of the proposed method. Therefore, the time of referring to lists becomes longer. As a result, the original LLAH has longer processing time than the proposed method. Note that the reason why the length of list explodes in 10 million is that the hash table is saturated. In the proposed method, more than 99.6% hash bins were occupied.

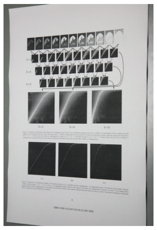


Figure 9. Example of images which caused retrieval errors.

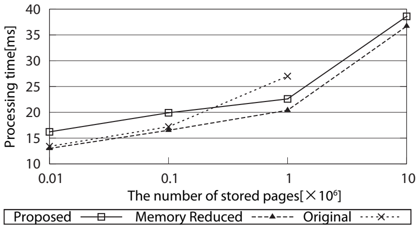


Figure 10. Relationship between the number of stored pages and processing time per query.

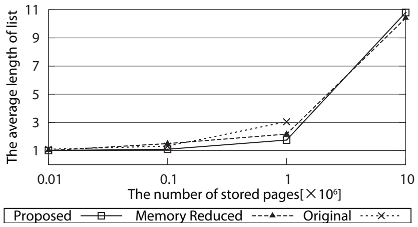
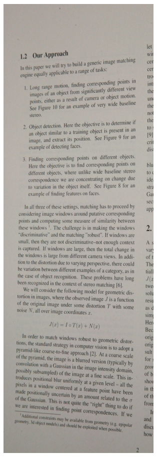


Figure 11. Relationship between the number of stored pages and the average length of lists.

B. Ex.2 : Cropping

Recall that the original LLAH indexes all feature points while the proposed method only keeps sampled points. Thus the proposed method may perform worse than the original LLAH if the number of feature points is not enough. This happens if queries capture narrow regions of pages. In this experiment, we examined the relationship between the area of captured regions and the accuracy.

First, we selected images which were not covered with many figures from the query images employed in Ex.1,



1.2 Our Approach

In this paper we will try to build a generic image matching engine equally applicable to a range of tasks:

1. Long range motion, finding corresponding points in images of an object from significantly different view points, either as a result of camera or object motion. See Figure 10 for an example of very wide baseline stereo.
2. Object detection. Here the objective is to determine if an object similar to a training object is present in an image, and extract its position. See Figure 9 for an example of detecting faces.
3. Finding corresponding points on different objects. Here the objective is to find corresponding points on different objects, where unlike wide baseline stereo correspondence we are concentrating on change due to variation in the object itself. See Figure 8 for an example of finding features on faces.

In all three of these settings, matching has to proceed by considering image windows around putative corresponding points and computing some measure of similarity between these windows [1]. The challenge is in making the windows "discriminative" and the matching "robust". If windows are small, then they are not discriminative–not enough context is captured. If windows are large, then the total change in the windows is large from different camera views. In addition to the distortion due to varying perspective, there could be variation between different examples of a category, as in the case of object recognition. These problems have long been recognized in the context of stereo matching [6].

We will consider the following model for geometric distortion in images, where the observed image J is a function of the original image under some distortion T with some noise N, all over image coordinates x.

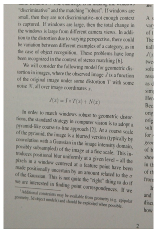
$$J(x) = I \circ T(x) + N(x)$$

In order to match windows robust to geometric distortions, the standard strategy in computer vision is to adopt a pyramid-like coarse-to-fine approach [2]. At a coarse scale of the pyramid, the image is a blurred version (typically by convolution with a Gaussian in the image intensity domain, possibly subsampled) of the image at a fine scale. This introduces positional blur uniformly at a given level – all the pixels in a window centered at a feature point have been made positionally uncertain by an amount related to the σ of the Gaussian. This is not quite the "right" thing to do if we are interested in finding point correspondences. If we

[1] Additional constraints may be available from geometry (e.g. epipolar geometry, 3d object models) and should be exploited when possible.

2

Figure 12. 1/2.



"discriminative" and the matching "robust". If windows are small, then they are not discriminative–not enough context is captured. If windows are large, then the total change in the windows is large from different camera views. In addition to the distortion due to varying perspective, there could be variation between different examples of a category, as in the case of object recognition. These problems have long been recognized in the context of stereo matching [6].

We will consider the following model for geometric distortion in images, where the observed image J is a function of the original image under some distortion T with some noise N, all over image coordinates x.

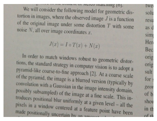
$$J(x) = I \circ T(x) + N(x)$$

In order to match windows robust to geometric distortions, the standard strategy in computer vision is to adopt a pyramid-like coarse-to-fine approach [2]. At a coarse scale of the pyramid, the image is a blurred version (typically by convolution with a Gaussian in the image intensity domain, possibly subsampled) of the image at a fine scale. This introduces positional blur uniformly at a given level – all the pixels in a window centered at a feature point have been made positionally uncertain by an amount related to the σ of the Gaussian. This is not quite the "right" thing to do if we are interested in finding point correspondences. If we

[1] Additional constraints may be available from geometry (e.g. epipolar geometry, 3d object models) and should be exploited when possible.

2

Figure 13. 1/4.



We will consider the following model for geometric distortion in images, where the observed image J is a function of the original image under some distortion T with some noise N, all over image coordinates x.

$$J(x) = I \circ T(x) + N(x)$$

In order to match windows robust to geometric distortions, the standard strategy in computer vision is to adopt a pyramid-like coarse-to-fine approach [2]. At a coarse scale of the pyramid, the image is a blurred version (typically by convolution with a Gaussian in the image intensity domain, possibly subsampled) of the image at a fine scale. This introduces positional blur uniformly at a given level – all the pixels in a window centered at a feature point have been made positionally uncertain

Figure 14. 1/8.

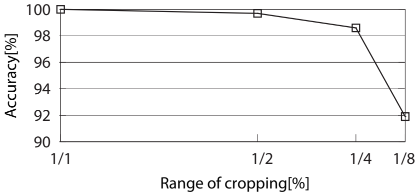


Figure 15. Relationship between the range of cropping and accuracy.

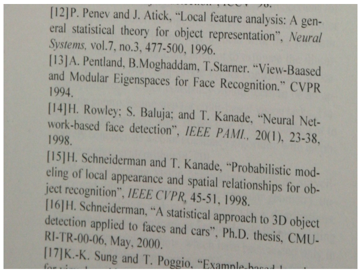
whose number was 989. Next, we cropped images partially. Eventually, the partially cropped images were utilized as query images in this experiment. The sizes of cropping from the entire page were: $1/2$, $1/4$, and $1/8$, whose examples are shown in Fig. 12, 13, and 14, respectively. The database included 10 million pages.

Figure 15 shows the relationship between the range of cropping and accuracy. When the range is small, the accuracy decreases. Since the number of feature points extracted from smaller query images is small, the correct image could not obtain the sufficient number of votes. Figure 16 shows an example of query images which caused a failure. This document has many dots ([.]) and quotations (["]). In this case, stable feature points could not be extracted because [.] and ["] easily became noise as shown in Fig 17. In order to solve the problem, we need a way of stabilizing feature points.

The result showed the proposed method has achieved 92% accuracy with $1/8$ size of entire page and more than 98% with $1/4$ size. Therefore, the proposed method is robust to scale change.

V. Conclusion

In this paper, we have introduced three improvements to the original LLAH. The amount of memory was decreased by sampling feature points stored in the database. The discrimination power and stability of features were improved



[12]P. Penev and J. Atick, "Local feature analysis: A general statistical theory for object representation", *Neural Systems,* vol.7, no.3, 477-500, 1996.

[13]A. Pentland, B.Moghaddam, T.Starner. "View-Baased and Modular Eigenspaces for Face Recognition." CVPR 1994.

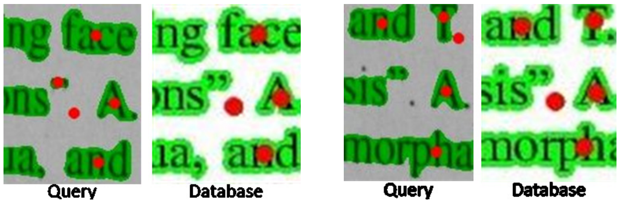
[14]H. Rowley; S. Baluja; and T. Kanade, "Neural Network-based face detection", *IEEE PAMI.,* 20(1), 23-38, 1998.

[15]H. Schneiderman and T. Kanade, "Probabilistic modeling of local appearance and spatial relationships for object recognition", *IEEE CVPR,* 45-51, 1998.

[16]H. Schneiderman, "A statistical approach to 3D object detection applied to faces and cars", Ph.D. thesis, CMU-RI-TR-00-06, May, 2000.

[17]K.-K. Sung and T. Poggio, "Example-based

Figure 16. Example of images which caused retrieval errors.



(a) Added feature point in the query (b) Disappeared feature point

Figure 17. Examples of unstable feature points.

by increasing the number of dimensions and removing redundancy. From the experimental results, we have confirmed achieves 99.4% accuracy and 38ms processing time for the database of 10 million pages. Our future work includes further reduction of memory requirement for making a large library available for document image retrieval.

Acknowledgment

This work was supported in part by JST, CREST, and the Grant-in-Aid for Scientific Research (B) (20300049) from Japan Society for the Promotion of Science (JSPS).

References

[1] T. Nakai, K. Kise, and M. Iwamura, "Use of affine invariants in locally likely arrangement hashing for camera-based document image retrieval," *Lecture Notes in Computer Science (7th International Workshop DAS2006)*, vol. 3872, pp. 541–552, feb 2006.

[2] ——, "Camera based document image retrieval with more time and memory efficient llah," *Proc. Second International Workshop on Camera-Based Document Analysis and Recognition (CBDAR2007)*, pp. 21–28, sep 2007.

[3] ——, "Real-time retrieval for images of documents in various languages using a web camera," *Proceedings of the 10th International Conference on Document Analysis and Recognition (ICDAR2009)*, pp. 146–150, jul 2009.

[4] R. T. Azuma, "A survey of augmented reality," *Presence*, vol. 6, no. 4, pp. 355–385, 1997.

[5] K. Kise, M. Chikano, K. Iwata, M. Iwamura, S. Uchida, and S. Omachi, "Expansion of queries and databases for improving the retrieval accuracy of document portions — an application to a camera-pen system," *Proceedings of the 9th IAPR International Workshop on Document Analysis Systems (DAS2010)*, pp. 309–316, jun 2010.