Instance Search Experiment for TRECVID 2010

Masami Shishibori, and Kenji Kita

Dept. of Information Science and Intelligent Systems, Faculty of Engineering, The University of Tokushima,
2-1 Minami-Josanjima-cho, Tokushima-shi, Tokushima, 770-8506, JAPAN

0. STRUCTURED ABSTRACT

1. *Briefly, what approach or combination of approaches did you test in each of your submitted runs?*

- KB_lab: This system extracts the facial image from each cut scene frame using the Haar-like operator, and then eliminates noise images (non-facial image) using SVM. Next, SIFT features are detected from the true facial image, and the similarity of the facial image is calculated using the SIFT features.

2. *What if any significant differences (in terms of what measures) did you find among the runs?*

none.

3. *Based on the results, can you estimate the relative contribution of each component of your system/approach to its effectiveness?*

Estimation is the same as above.

4. *Overall, what did you learn about runs/approaches and the research question(s) that motivated them?*

SIFT-based approach seems to be promising, but the cost of the retrieval time becomes high. Thus, the efficient retrieval algorithm must be considered.

1. INTRODUCTION

This is the first TRECVID participation for Tokushima University. This year, we have participated in the instance search (INS) pilot. For the INS task, our main focus was to apply SIFT-based image retrieval method with facial images.

2. INSTANCE SEARCH SYSTEM

2.1 Outline of this system

The overview of our retrieval system is shown in Figure 1. At the registration phase, cut scene frames are detected from the video data, and then facial images are extracted from each cut frame using the Haar-like operator [1]. Next, noise images (non-facial image) in the extracted facial images are eliminated using the SVM (Support Vector Machine) [2]. Finally, SIFT (Scale-Invariant Feature Transform) features [3] are detected from true facial images, and these features are registered in the facial database.

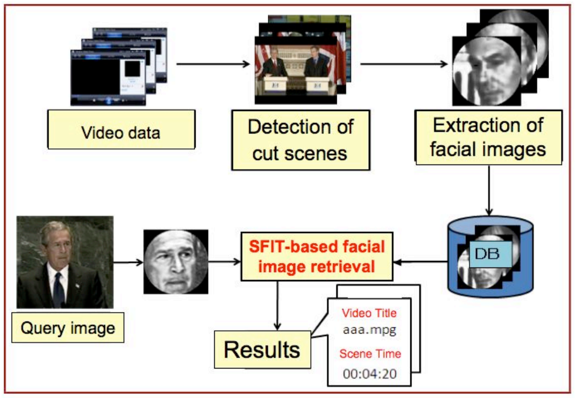


Figure 1: Outline of the instance search system.

At the retrieval phase, the user specifies the query image. The person whom the user wants to retrieve is reflected in this image. First, the facial area of the query image is extracted using the same operator as the registration phase. Next, SIFT features are detected from the query facial image. After that, the similarity between query facial image and the facial database is calculated based on 128 dimensional SIFT features. As the retrieval result, the title and the cut scene time of the video where the query person is reflected are obtained.

2.2 Extraction of facial images

On this system, the face extraction tool in the OpenCV [4] is utilized. This tool is very useful, however the extraction result includes some noise images. Figure 2 shows the example of extracted facial images using the Haar-like operator. The upper images of Figure 2 are images which were able to detect the facial area correctly. On the other hand, the lower images indicate the noise images. These noise images have a bad influence for the retrieval result.

In order to solve this problem, this system classifies true facial images and noise images using the SVM (Support Vector Machine) [2]. We notice that the position of the SIFT point detected from the noise image and the facial image is different. Figure 3 shows the distribution of the frequencies at the position of the SIFT point detected from each image.

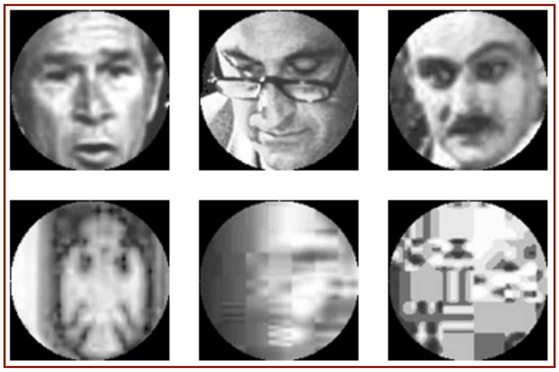


Figure 2: Example of extracted facial images using the Haar-like operator.

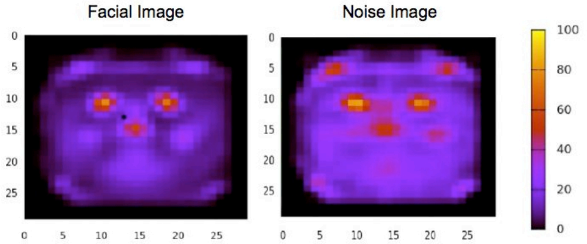


Figure 3: Distribution of the frequencies at the position of the SIFT point detected from each image.

The size of the facial image extracted by the OpenCV becomes 30*30 pixels. The vertical and horizontal axes of both images of Figure 3 indicate the corresponding pixels. The pixel of high frequency is expressed in yellow, and the pixel of low frequency becomes purple and black. From Figure 3, it is found that the distribution of the point to the facial image (left side image) has concentrated on the part of eyes and noses. On the other hand, the points in the noise image (right side image) are distributed more widely than the facial images. From this result, the feature vector used with SVM consists of the frequencies at the position of the SIFT point detected from the image. The each dimensional value corresponds to the frequency at each pixel, and this feature is represented as 900 (30*30) dimensional vector.

2.3 SIFT-based facial image retrieval

We suppose that L facial images are extracted from the video data and M SIFT features are detected from a facial image, $L*M$ SIFT features are registered in the facial database. At the retrieval phase, the similarity between the query facial image and the facial database is calculated based on the SIFT features. If N SIFT features are detected from the query facial image, the k-nearest neighbor search, which top k similarities can be obtained, is executed N times.

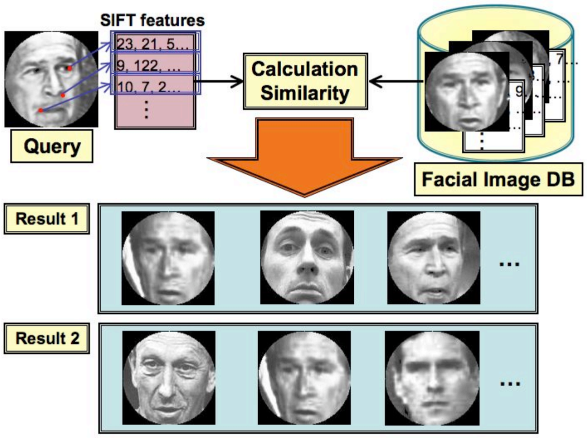


Figure 4: Illustration of the similarity calculation based on SIFT features.

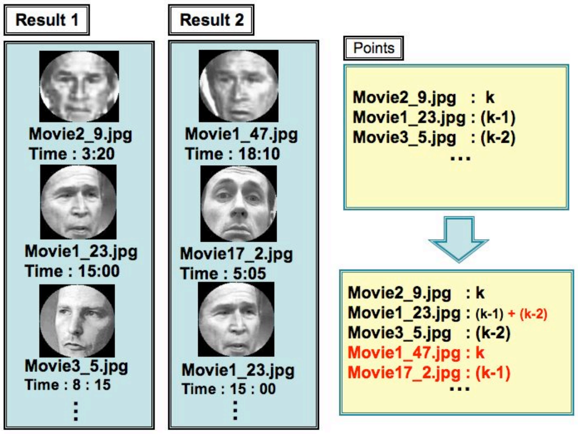


Figure 5: Illustration of the integration of some retrieval results by the vote algorithm.

As a result, N retrieval results can be obtained as shown in Figure 4, where one retrieval result has top k similarity facial images. These N retrieval results must be merged into a final result. The final retrieval result is integrated by the vote algorithm as shown in Figure 5. The vote algorithm is executed as following: First, the point according to the order of the similarity is given to each facial image in the first retrieval result. If the retrieval result has k facial images, the top similarity (most similarity) facial image is set to k point. The second similarity image is k-1 point, and the i-th similarity image is k-i+1 point. Next, the same procedure is repeated for other retrieval results. In Figure 5, black points correspond to the first retrieval result (Result 1), and red points are the second result (Result 2). Finally, the final order of the similarity is calculated by the sum of all the points

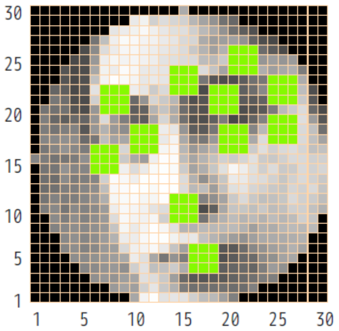


Figure 6: Example of the appearance frequency of SIFT points at the facial image of 30*30 pixels.

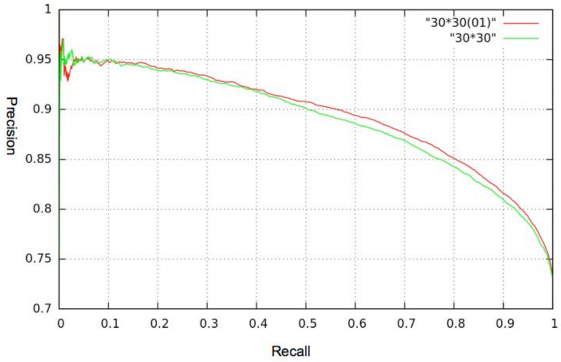


Figure 7: Experimental result of facial image extraction.

3. EVALUATIONS

3.1 Evaluation of the facial image extraction

The video data of TRECVID2010 INS task consists of 400 files, which lengths are from 9 minutes to 60 minutes. About 41,000 cut scenes were detected from the video data, and 20,533 facial images were extracted from the above cut scenes using the Haar-like operator. These images have 15,003 true facial images and 5,530 noise images.

In this section, the evaluation of facial images extraction using the SVM is shown. As for the feature data, the facial image extracted by the Haar-like operator is changed to 30*30 pixels. Then, the appearance frequency of SIFT points at each pixel corresponds to the dimensional value as shown in Figure 6. Then, the feature vector becomes 900 dimensional data. As for the experimental method, true facial images and noise images are divided into 10 groups, which have the same number of images. Then, one group is used as the test data, and the remaining 9 groups are the training data. This evaluation is repeated 10 times for other group patterns. And, the average of ten experimental results is obtained, namely the 10-fold cross validation is performed.

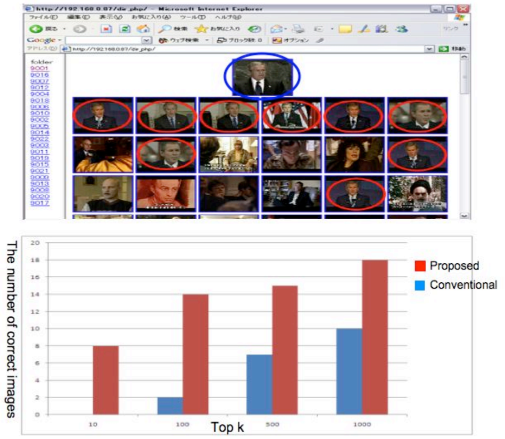


Figure 8: Experimental result of the query A.

Figure 7 shows the experimental result and indicates the precision-recall curve. The vertical axis indicates the precision and the horizontal axis indicates the recall. The green line “30*30” shows the result of the frequency feature, and the red line “30*30 (01)” is the binary frequency feature. The binary frequency means that if the frequency is 0, the binary frequency is set to 0, otherwise the binary frequency is 1. From Figure 7, the accuracy of the binary frequency (red line) is a little better. The number of SIFT points detected from the image changes according to the content of the image. So, it seems that the difference of total frequencies becomes small using the binary frequency.

3.2 Evaluation of instance search

On this system, true facial images (about 15,000 images) were manually selected from the detected images and registered into the facial database, because the accuracy of the facial image extraction was not so good. Experimental results are shown Figure 8, 9, 10 and 11.

The upper part of each figure shows the retrieval result of this system. In the retrieval result, the image in the blue circle corresponds to the query image, and the image in the red circle corresponds to the correct image. In the case of Figure 8, all correct images can be retrieved in top 6 images. The bottom graph shows the experimental result, which means the number of correct facial images including in the top *k* images (*k* = 10, 100, 500, 1000). The vertical axis indicates the number of correct facial images, and the horizontal axis is *k* value. In order to compare with the other method, the face recognition method using eigenfaces [5] is used as the conventional method. In each graph, the blue bin indicates the conventional method, and the red bin is the proposed method. The proposed method can be obtained higher accuracy than the conventional method for any queries.

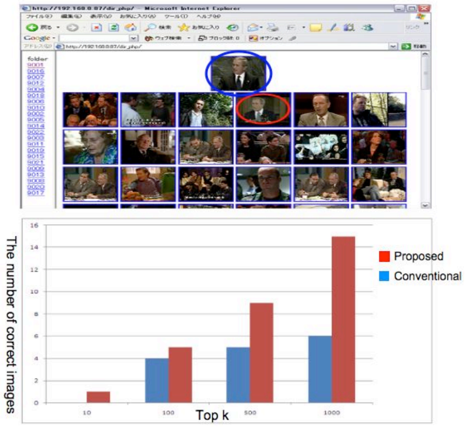


Figure 9: Experimental result of the query B.

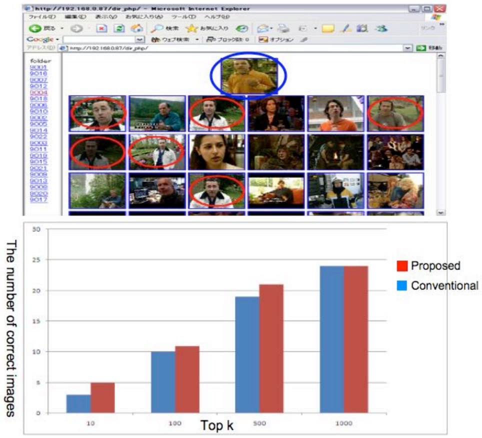


Figure 10: Experimental result of the query C.

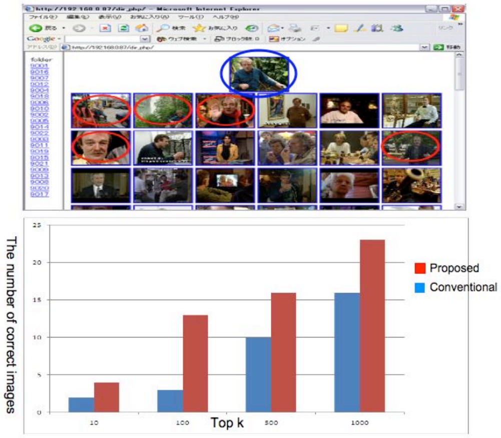
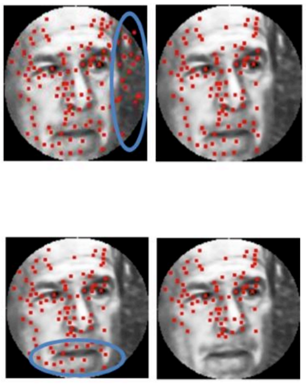


Figure 11: Experimental result of the query D.

In Figure 8 and 9, the query image is a same person, however the accuracy of Figure 9 becomes lower than Figure 8. As the face of the query image in Figure 8 turns to the front, this system can search many images of the same person, where faces of right images in the retrieval result almost turn to the front. On the other hand, the face of the query image in Figure 9 doesn't turn to the front. It is found that this system can search only the same sideways face as the query and can't search other images of the same person.



	Before	After
Top 10 of Results	7	8
Top 100 of Results	12	13

	Before	After
Top 10 of Results	8	7
Top 100 of Results	13	16

Figure 12: Improvement by eliminating SIFT points in the background and around the mouth.

As for the future works, the improvement by eliminating useless SIFT points should be considered. As shown in the upper left images of Figure 12, many SIFT points are allocated in the background. Red points in the blue circle of the image correspond to points of the background. These points are useless and noise to calculate the similarity of the face. After these points are eliminated manually, the number of correct images including in top *k* results increased a little as shown in the right table of Figure 12. The bottom right table of Figure 12 shows the experimental result, which was obtained by eliminating SIFT points of not only the background but also around the mouth. The points around the mouth might be also unnecessary in the calculation of the facial similarity.

REFERENCES

[1] Rainer Lienhart and Jochen Maydt, “An Extended Set of Haar-like Features for Rapid Object Detection”, *IEEE ICIP 2002*, Vol.1, pp. 900-903, Sep. 2002.

[2] V.Vapnik, “The Nature of Statistical Learning Theory”, Springer, 1995.

[3] D.G.Lowe, “Object recognition from local scale-invariant features”, *Proc. of IEEE International Conference on Computer Vision (ICCV)*, pp. 1150-1157, 1999.

[4] http://opencv.willowgarage.com

[5] M. Turk and A. Pentland, “Face Recognition Using Eigenfaces”, *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pp.586-591, 1991.