Finding People Frequently Appearing in News

Derya Ozkan and Pınar Duygulu

Bilkent University, Department of Computer Engineering
06800, Ankara, Turkey
{deryao, duygulu}@cs.bilkent.edu.tr

Abstract. We propose a graph based method to improve the performance of person queries in large news video collections. The method benefits from the multi-modal structure of videos and integrates text and face information. Using the idea that a person appears more frequently when his/her name is mentioned, we first use the speech transcript text to limit our search space for a query name. Then, we construct a similarity graph with nodes corresponding to all of the faces in the search space, and the edges corresponding to similarity of the faces. With the assumption that the images of the query name will be more similar to each other than to other images, the problem is then transformed into finding the densest component in the graph corresponding to the images of the query name. The same graph algorithm is applied for detecting and removing the faces of the anchorpeople in an unsupervised way. The experiments are conducted on 229 news videos provided by NIST for TRECVID 2004. The results show that proposed method outperforms the text only based methods and provides cues for recognition of faces on the large scale.

1 Introduction

Finding specific people in news videos is important and the challenge is also acknowledged by NIST in TRECVID video retrieval evaluation [1]. Searching for the names of the people in the speech transcript text is a common approach for accessing the related video shots. However, only text based systems are likely to produce incorrect results since the shots associated with the text may include the appearances of many other people and especially the anchorperson or reporter besides the query name. On the other hand, recognizing faces is a long standing and difficult problem [2,3]. The noisy and complicated nature of news videos and the variety of poses, expressions and illumination conditions make the face recognition even more challenging.

Recently, it is shown that the performance of person queries can be improved by integrating name and face information [4,5,6,7,8]. Yang et al. [9] show that text-based search results can be improved by modeling the timing between names and appearances of people in news videos. In [10], Berg et al. proposed a method for associating the faces in the news photographs with a set of names extracted from the captions, and then clustering in appropriate discriminant coordinates to correct the mistakes in labeling and to identify incorrectly labeled faces.

H. Sundaram et al. (Eds.): CIVR 2006, LNCS 4071, pp. 173–182, 2006.

In this study, we propose a method for improving the performance of person queries in news videos by combining name and face information. We use the observation that, faces of a query name will appear more frequently when his/her name is mentioned in the speech transcript text, and limit our search space for a query name by choosing the shots around which the name appears. Although, there may be faces in this search space corresponding to other people in the story, or some non-face images due to the errors of the face detection method used, the faces of the query name are likely to be the most frequently appearing ones than any other person in the same space. Our assumption is that, even if the expressions or poses vary, different appearances of the face of the same person tend to be more similar to each other than to the faces of others.

If a similarity measure between any two faces can be assigned, then this measure can be used to find the similarities among all the faces in the search space of a query name. Then, this search space represents a graph structure in which nodes are faces and edges correspond to similarities. The problem transforms into a graph problem in which we aim to find the densest component corresponding to the group of most similar faces, which are the faces belonging to the query name.

In [11], we apply a similar method on news photographs data set collected from the Web by Berg et al. [5]. Due to the higher noise level and lower resolution, news videos is a harder data set to work with. Also, there is usually a time shift between the appearance of a name and the appearance of the face belonging to that name. Therefore, using a single shot temporally aligned with the text may yield incorrect results. Another problem in news videos, which is more important, is that the most frequent face usually corresponds to the anchorperson or reporter rather than the face of the query name (See Fig. 1).

The time shift problem can be handled by taking a window around the name. The solution is, rather than searching the faces only on the shots including the name of the person, also to include the preceding and succeeding shots. In order to handle the problem due to anchorperson faces, we add a mechanism to detect and remove the anchorpeople. Since, the anchorperson are the most frequently appearing people in the news, we take each video separately and apply the densest component algorithm to each of them.

In the following, we first explain the data set used in experiments. Then, we describe the similarity measures and the densest component algorithm. After presenting the method for finding the anchorpeople, the methods for integrating the name and face information for improving the person queries are presented.

2 Data Set

The data set used in the experiments is the broadcast news videos provided by NIST for TRECVID video retrieval evaluation competition 2004 [1]. It consists of 229 movies (30 minutes each) from ABC and CNN news. The shot boundaries and the key-frames are provided by NIST. Speech transcripts extracted by LIMSI [12] are used to obtain the associated text for each shot.

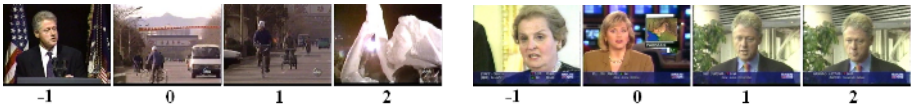

Fig. 1. Key-frames from two different videos. The numbers below each image show the distance to shot, in which the name 'Clinton' is mentioned. Note that in both cases, Clinton does not appear visually in the shot in which his name is mentioned but appears in preceding (left image) or succeeding shots (right image).

For the experiments, we choose 5 people, namely Bill Clinton, Benjamin Netanyahu, Sam Donaldson, Saddam Hussein and Boris Yeltsin. In the speech transcript text, their names appear 991, 51, 100, 149 and 78 times respectively.

The face detection algorithm provided by Mikolajcyzk [13] is used to extract faces from key-frames. Due to high noise levels and low image resolution quality, the face detector produces many false alarms. On randomly selected ten videos, in 2942 images, 1395 regions are detected as faces but only 790 of them are real faces and 580 faces are missed. In total, 31,724 faces are detected over the whole data set.

3 Graph Based Person Finding Approach

Faces of a particular person tend to be more similar to each other than to faces of other people. If we can define a similarity measure among the faces in a set and represent the similarities in a graph structure, then the problem of finding the most similar faces corresponding to the instances of query name's face can be tackled by finding the densest component in the graph. In the following two subsections we explain the similarity measure used and the greedy graph algorithm to find the densest component. The details of the algorithm can be found in [11].

3.1 Constructing the Dissimilarity Graph of Faces

The similarity of faces are defined using the interest points extracted from the detected face areas. Lowe's SIFT operator [14], which have been shown to be successful in recognizing objects and faces, are used for extracting the interest points.

The dissimilarity of two faces are computed based on the matching interest points. To find the matching interest points on two faces, each point on one face is compared with all the points on the other face and the points with the least Euclidean distance are selected. Since this method produces many matching points including the wrong ones, we apply two constraints to obtain only the correct matches, namely the geometrical constraint and the unique match constraint.

Geometrical constraint expects the matching points to appear around similar similar positions on the face when the normalized positions are considered. The matches whose interest points do not fall in close positions on the face are eliminated. Unique match constraint ensures that each point matches to only a single point by eliminating multiple matches to one point and also by removing one-way matches. Example of matches after applying these constraints are shown in Fig. 2.

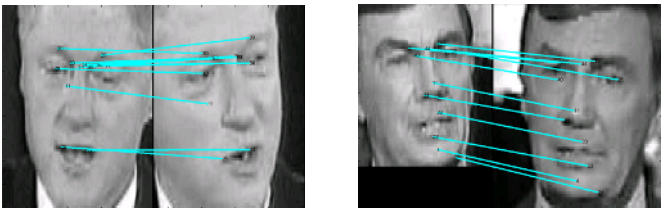

Fig. 2. Examples for matching points. Note that, even for faces with different size, pose or expressions the method successfully finds the corresponding points.

After applying the constraints, the distance between the two faces is defined as the average distance of all matching points between these two faces. A dissimilarity graph for all the faces in the search space is then constructed using these distances.

3.2 Finding the Densest Component in the Graph

In the dissimilarity graph, faces represent the nodes and the distances between the faces represent the edge weights. We assume that, in this graph the nodes of a particular person will be close to each other (highly connected) and distant from the other nodes (weakly connected). Hence, the problem can be transformed in to finding the densest subgraph (component) in the entire graph. To find the densest component we adapt the method proposed by Charikar [15] where the density of subset S of a graph G is defined as

$$f(\ S) = \frac{|\ E(\ S)\ |}{|\ S\ |},$$

in which $E(\ S) = \{i, j \in E : i \in S, j \in S\}$ and E is the set of all edges in G and $E(S)$ is the set of edges induced by subset S. The subset S that has maximum $f(S)$ is defined as the densest component.

Initially, the algorithm presented in [15] starts from the entire graph and in each step, the vertex of minimum degree is removed from the set S. The $f(S)$ value is also computed for each step. The algorithm continues until the set S is empty. Finally, the subset S with maximum $f(S)$ value is returned as the densest component of the graph.

In order to apply the above algorithm to the constructed dissimilarity graph, we need to convert it into a binary form, in which 0 indicates no edge and 1

indicates an edge between the two nodes. This conversion is carried out by applying a threshold on the distance between the nodes. For instance, if 0.5 is used as the threshold value, then edges in the dissimilarity graph having higher value than 0.5 are assigned as 0, and others as 1. In other words, the threshold can be thought of an indicator of two nodes being near-by and/or remote.

The success of our algorithm varies with the threshold that is chosen while converting the weighted dissimilarity graph to a binary one. In order to determine a reasonable threshold, we randomly selected 10 videos and recorded recall-precision values of different thresholds for anchorperson detection. These values are plotted in Fig. 3. Further in our experiments, we select the point marked with a cross in the recall-precision curve, which corresponds to threshold of 0.6. The same threshold is used both for anchorperson detection and for person queries.

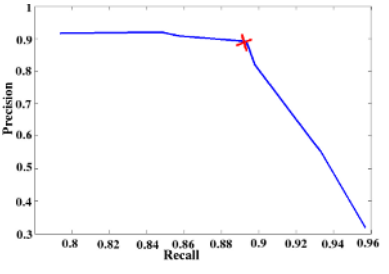

Fig. 3. Recall-precision values for randomly selected 10 videos for threshold values varying between 0.55 and 0.65

4 Integrating Names and Faces

The probability of a person appearing on the screen is high when his/her name is mentioned in the speech transcript text. Thus, looking for the shots in which the name of the query name is mentioned is a good place to start search over people. However, there can be a time shift or there can be some anchorperson/reporter scenes. Anchorpeople can be removed as will be explained in the next section. In order to handle the alignment problem we can also look for also preceding and succeeding shots.

Recently, it has been showed that the frequency of a person's visual appearance with respect to the occurrence of his/her name can be assumed to have a Gaussian distribution [9]. We use the same idea and search for the range where the face is likely to appear relative to the name. As we experimented on "Clinton" query, we see that taking the ten preceding and the ten succeeding shots together with the shot where the name is mentioned is a good approximation to find most of the relevant faces(See Fig. 4).

However, the number of faces in this range (which we refer to as [-10,10]) can still be large compared to the instances of the query name. For better understanding of the distributions, we plot the frequency of faces relative to the

Table 1. Number of faces corresponding to the query name over total number of faces in the range [-10,10] and [-1,2]

Range	Clinton	Netanyahu	Donaldson	Saddam	Yeltsin
[-10,10]	213/6905	9/383	137/1197	18/1004	21/488
$[-1, 2]$	160/2457	6/114	102/330	14/332	19/157

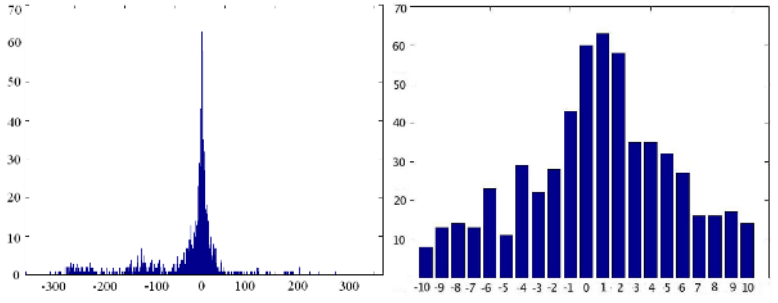

Fig. 4. The figure shows frequency of Bill Clinton's visual appearance w.r.t the distance to the shot in which his name is mentioned. **Left:** when the whole data set is considered, **right:** when the faces appearing around the name within the preceding and the following ten shots are considered. Over the whole data set Clinton has 240 faces and 213 of them appear in the selected range.

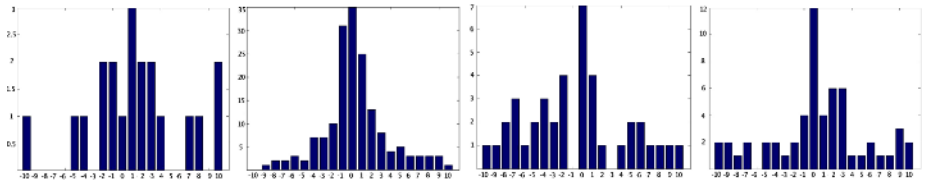

Fig. 5. The relative position of the faces to the name for Benjamin Netanyahu, Sam Donaldson, Saddam Hussein, and Boris Yeltsin respectively

position of the names for the five people that we have chosen for our experiments in Fig. 5. It is seen that taking only one preceding and two following shots (which we refer to as [-1,2]) is also a good choice. Table 1 shows that, most of the correct faces fall into this selected range by removing many false alarms.

5 Anchorperson Detection and Removal

When we look at the shots where the query name is mentioned in the speech transcript, it is likely that the anchorperson/reporter might be introducing or wrapping up a story, with the preceding or succeeding shots being relevant, but not the current one. Therefore, when the shots including the query name

are selected, the faces of the anchorperson will appear frequently making our assumption that the most frequent face will correspond to the query name wrong. Hence, it is highly probable that the anchorperson will be returned as the densest component by the person finding algorithm. The solution is to detect and remove the anchorperson before applying the algorithm.

In [6], a supervised method for anchorperson detection is proposed. They integrate color and face information together with speaker-id extracted from the audio. However, this method has some disadvantages. First of all, it highly depends on the speaker-id, and requires the analysis of audio data. The color information is useful to capture the characteristics of studio settings where the anchorperson is likely to appear. But, when the anchorperson reports from another environment this assumption fails. Finally, the method depends on the fact that the faces of anchorpeople appear in large sizes and around some specific positions, but again there may be cases where this is not the case.

In this study, we use the graph based method to find the anchorpeople in an unsupervised way. The idea is based on the fact that, the anchorpeople are usually the most frequently appearing people in broadcast news videos. For different days there may be different anchorpeople reporting, but generally there is a single anchorperson for each day.

We apply the densest component based method to each news video separately, to find the people appearing most frequently, which correspond to the anchorpeople. We run the algorithm on 229 videos in our test set, and obtained average recall and precision values as 0.90 and 0.85 respectively. Images that are detected as anchorperson in ten different videos are given in Fig. 6.

When the anchorpeople are detected, the next step is to remove them from the search space to improve the person queries as will be explained in the following sections.

Fig. 6. Detected anchors for 6 different videos

6 Improving Person Queries

After selecting the range where the faces may appear we apply the densest component algorithm to find the faces corresponding to the query name. We have recorded the number of true faces of the query name and total number of images retrieved as in Table 2. The first column of the table refers to total number of true images retrieved vs. total number of true images retrieved by using only the speech transcripts -selecting the shots within interval [-1,2]. The numbers after removing the detected anchorpeople by the algorithm from the text-only results are given in the second column. And the last column is for applying the algorithm to this set, from which the anchorpeople are removed. The precision values are given in Fig. 8. Some sample images retrieved for each person are shown in Fig. 7.

Table 2. Numbers in the table indicate the number of correct images retrieved/ total number of images retrieved for the query name

Query name	Clinton	Netanyahu	Sam Donaldson	Saddam	Yeltsin
Text-only	160/2457	6/114	102/330	14/332	19/157
Anchor removed	150/1765	5/74	81/200	14/227	17/122
Method applied	109/1047	4/32	67/67	9/110	10/57

Fig. 7. Sample images retrieved for five person queries in experiments. Each row corresponds to samples for Clinton, Netanyahu, Sam Donaldson, Saddam, Yeltsin queries respectively.

As can be seen from the results, we keep most of the correct faces (especially after anchorperson removal), and we get reject many of the incorrect faces. Hence the number of images presented to the user is decreased. Also, our improvement in precision values are relatively high. Average precision of only text based results increases by 29% after ancherperson removal, and by 152% after applying the proposed algorithm.

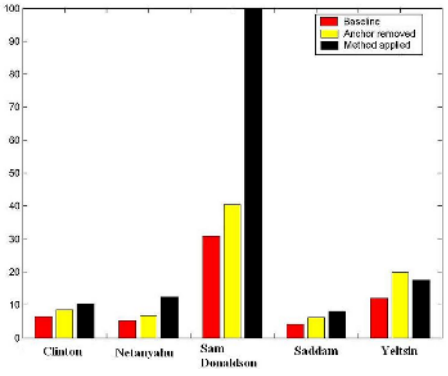

Fig. 8. Precisions values achieved for five people used in our tests

7 Conclusion

This paper addresses the problem of finding a specific person in news videos. We fist use the speech transcripts and select the neighboring shots in which the name of the query name appears to limit our search space. Applying the proposed person finding algorithm on each video separately, we detect the anchorperson in each video. Then, we remove detected anchorperson from the search space of the query name and apply the algorithm to the remaining images.

Experiments are conducted on 229 broadcast news videos archive, which is a difficult set due to large variations in pose, illumination and expressions in data. Experiments show that we improve person search performances relative to only text based results. Average precision values of only text based results are increased by 29% after ancherperson removal, and by 152% after applying the proposed algorithm. The person finding algorithm also performs well for anchorperson detection without requiring any supervision.

In [16] sets of face exemplars for each person are gathered automatically in shots for tracking. A similar approach can be adapted and instead of taking a single face from each shot by only considering the key-frames, face detection can be applied to all frames to obtain more instances of the same. This approach can help to find better matching interest points and more examples that can be used in the graph algorithm.

Acknowledgement

This research is partially supported by TÜBİTAK Career grant number 104E065 and grant number 104E077.

References

1. : Trec video retrieval evaluation http://www-nlpir.nist.gov/projects/trecvid/ (2004)
2. Gross, R., Baker, S., Matthews, I., Kanade, T.: Face recognition across pose and illumination. In Li, S.Z., Jain, A.K., eds.: Handbook of Face Recognition, Springer Verlag (2004)
3. Zhao, W., Chellappa, R., Phillips, P., Rosenfeld, A.: Face recognition: A literature survey. ACM Computing Surveys **35**(4) (2003) 399–458
4. Satoh, S., Kanade, T.: Name-it: Association of face and name in video. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition(CVPR). (1997)
5. Berg, T., Berg, A.C., Edwards, J., Forsyth, D.: Who is in the picture. In: Neural Information Processing Systems (NIPS). (2004)
6. Chen, M.Y., Hauptmann, A.: Searching for a specific person in broadcast news video. In: International Conference on Acoustics, Speech, and Signal Processing (ICASSP'04), Montreal, Canada (2004)
7. P. Duygulu, A.H.: What's news, what's not? associating news videos with words. In: The 3rd International Conference on Image and Video Retrieval (CIVR 2004) Ireland. (July 21-23, 2004)
8. Ikizler, N., Duygulu, P.: Person search made easy. In: The Fourth International Conference on Image and Video Retrieval (CIVR 2005), Singapore (2005)
9. Yang, J., Chen, M.Y., Hauptmann, A.: Finding person x: Correlating names with visual appearances. In: International Conference on Image and Video Retrieval (CIVR'04), Dublin City University Ireland (2004)
10. Berg, T., Berg, A.C., Edwards, J., Maire, M., White, R., Teh, Y.W., Learned-Miller, E., Forsyth, D.: Faces and names in the news. In: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). (2004)
11. Ozkan, D., Duygulu, P.: Interesting faces in the news. In: to Appear in IEEE Conf. on Computer Vision and Pattern Recognition. (2006)
12. Gauvain, J., Lamel, L., Adda, G.: The limsi broadcast news transcription system. Speech Communication **37**(1-2) (2002)
13. Mikolajczyk, K.: Face detector. INRIA Rhone-Alpes (2004) Ph.D Report.
14. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision **60**(2) (2004)
15. Charikar, M.: Greedy approximation algorithms for finding dense components in a graph. In: APPROX '00: Proc. of the 3rd International Workshop on Approximation Algorithms for Combinatorial Optimization, London, UK (2000)
16. Sivic, J., Everingham, M., Zisserman, A.: Person spotting: video shot retrieval for face sets. In: International Conference on Image and Video Retrieval (CIVR 2005), Singapore. (2005)