# A GRAPH BASED APPROACH FOR FINDING PEOPLE IN NEWS

A THESIS

SUBMITTED TO THE DEPARTMENT OF COMPUTER ENGINEERING

AND THE INSTITUTE OF ENGINEERING AND SCIENCE

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

By

Derya Ozkan

July, 2007

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Asst. Prof. Dr. Pinar Duygulu Şahin(Advisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Asst. Prof. Dr. Selim Aksoy

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Prof. Dr.

Approved for the Institute of Engineering and Science:

Prof. Dr. Mehmet B. Baray
Director of the Institute

# ABSTRACT

## A GRAPH BASED APPROACH FOR FINDING PEOPLE IN NEWS

Derya Ozkan
M.S. in Computer Engineering
Supervisor: Asst. Prof. Dr. Pinar Duygulu Şahin
July, 2007

Along with the recent advances in technology, large quantities of multi-modal data has arisen and became prevalent. Hence, effective and efficient retrieval, organization and analysis of such data constitutes a big challenge. Both news photographs on the web and news videos on television forms this kind of data by covering rich sources of information. People are mostly the main subject of the news; therefore, queries related to a specific person are often desired.

In this study, we propose a graph based method to improve the performance of person queries in large news video and photograph collections. We exploit the multi-modal structure of the data by associating text and face information. On the assumption that a person's face is likely to appear when his/her name is mentioned in the news, only the faces associated with the query name are selected first to limit the search space for a query name. Then, we construct a similarity graph of the faces in this limited search space, where nodes correspond to the faces and edges correspond to the similarity between the faces. Among these faces, there could be many faces corresponding to the queried person in different conditions, poses and times. There could also be other faces corresponding to other people in the news or some non-face images due to the errors in the face detection method used. However, in most cases, the number of corresponding faces of the queried person will be large, and these faces will be more similar to each other than to others. To this end, the problem is transformed into a graph problem, in which we seek to find the densest component of the graph. This most similar subset (densest component) is likely to correspond to the faces of the query name. Finally, the result of the graph algorithm is used as a model for further recognition when new faces are encountered. In the paper, it has been

shown that the graph approach can also be used for detecting the faces of the anchorpersons without any supervision.

The experiments are performed on two different data sets: news photographs and news videos. The first set consists of thousands of news photographs from Yahoo! news web site. The second set is 229 broadcast news videos provided by NIST for TRECVID 2004. Images from the both sets are taken in real life conditions and, therefore, have a large variety of poses, illuminations and expressions. The results show that proposed method outperforms the text only based methods and provides cues for recognition of faces on the large scale.

*Keywords:* Face recognition, face retrieval, SIFT features.

# ÖZET

## HABERLERDEKI ONEMLI YUZLER: ONEMLI KISILER VE ONEMLI YUZ NOKTALARI

Derya Ozkan
Bilgisayar Mühendisliği,, Yüksek Lisans
Tez Yöneticisi: Yard. Doç. Dr. Pinar Duygulu Şahin
Temmuz, 2007

Bu almada, haber fotoraflarndan oluan geni veri kmelerinde kiilerin sorgulanmasn salayan bir yntem sunulmutur. Yntem isim ve yzlerin ilikilendirilmesine dayanmaktadr. Haber baslnda kiinin ismi geiyor ise fotorafta da o kiinin yznn bulunaca varsaymyla, ilk olarak sorgulanan isim ile ilikilendirilmi, fotoraflardaki tm yzler seilir. Bu yzler arasnda sorgu kiisine ait farkl koul, poz ve zamanlarda ekilmi, pek ok resmin yannda, haberde ismi geen baka kiilere ait yzler yada kullanlan yz bulma ynteminin hatasndan kaynaklanan yz olmayan resimler de bulunabilir. Yine de, ou zaman, sorgu kiisine ait resimler daha ok olup, bu resimler birbirine dierlerine olduundan daha ok benzeyeceklerdir. Bu nedenle, yzler arasndaki benzerlikler izgesel olarak betimlendiinde , birbirine en ok benzeyen yzler bu izgede en youn bileen olacaktr. Bu almada, sorgu ismiyle ilikilendirilmi, yzler arasnda birbirine en ok benzeyen alt kmeyi bulan, izgeye dayal bir yntem sunulmaktadr.

*Anahtar sözcükler*: Yüz tanıma, yüz sorgulama, SIFT.

# Acknowledgement

I would first like to express my gratitude to my advisor Pinar Duygulu Şahin for her guidance and support throughout my studies. Besides her valuable comments and teaching, she was my main source of morale support by highly motivating me during my graduate study. I am honoured to be her first masters student.

I am very thankful to Selim Aksoy for his suggestions and valuable comments.

I was pleased to be a part of the RETINA team, and having such a nice friendship with the group members. The two years with them in EA 522 will be a very pleasant memory for me.

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

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Motivation

Along with the recent advances in technology, large quantities of multi-modal data has become more available and widespread. With its emergence, effective and efficient retrieval, organization and analysis of such kind of big data has become a challenging problem and aroused interest. News photographs on the web and news videos on television are two examples of this type of data. They acquire rich sources of information wherein. Hence, accessing them is especially important and its importance has also been acknowledged by NIST in TRECVID video retrieval evaluation [1].

People are usually the main subject of the stories in the news. Therefore, queries related to a specific person is often a desired task. In general these people visually appear around when his/her name is mentioned in the news. On this account, the common way to retrieve information related to a person is to search using his/her name. However, such an approach is likely to yield incorrect results due to the following reasons: there may be other people in the same story that also appear visually besides the query name, and the query may not appear in the associated picture even if his/her name is mentioned in the story. So to get the true images of the query person, a face detection should be applied to extract

the desired faces of the person. But, this may also cause some non-face images to arise in the search results, due to the errors of face detection algorithm used. Figure 1.1 exemplifies some of the detected faces in news photographs that are associated with the query name *President George W. Bush*. Even if there appears the faces of the query person, there also appears the faces of other people and some non-face images that are also associated with the same name.

Figure 1.1: Sample detected faces in news photographs [8], for which the name *President George W. Bush* appears in the associated caption.

News photographs appear with an associated caption on the web (see Figure 1.2). There could be more than one face on the photograph or more than one name in the caption. Thence, it is indeterminate that which name goes with which face. Similarly in news videos, there is a speech transcript aligned with each shot (see Figure 1.3). The face-name association problem is also encountered for these images. Moreover, in news videos 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 speech may yield incorrect results. Another problem, 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 Figure 1.4). On these grounds, in order to retrieve the correct images of a particular person, visual information must be incorporated and the faces of the person need to be recognized.

On the other hand, face recognition is a long standing and a difficult problem (see [23, 54, 24, 49, 48] for recent surveys). Although many different approaches have been proposed for recognizing faces, most of the current face recognition methods are evaluated only in controlled environments and for limited data sets. However, for larger and more realistic data sets like news photographs and/or videos, face recognition is still difficult and error-prone due to the noisy and complicated nature of these sets. These sets contain large variations in pose,

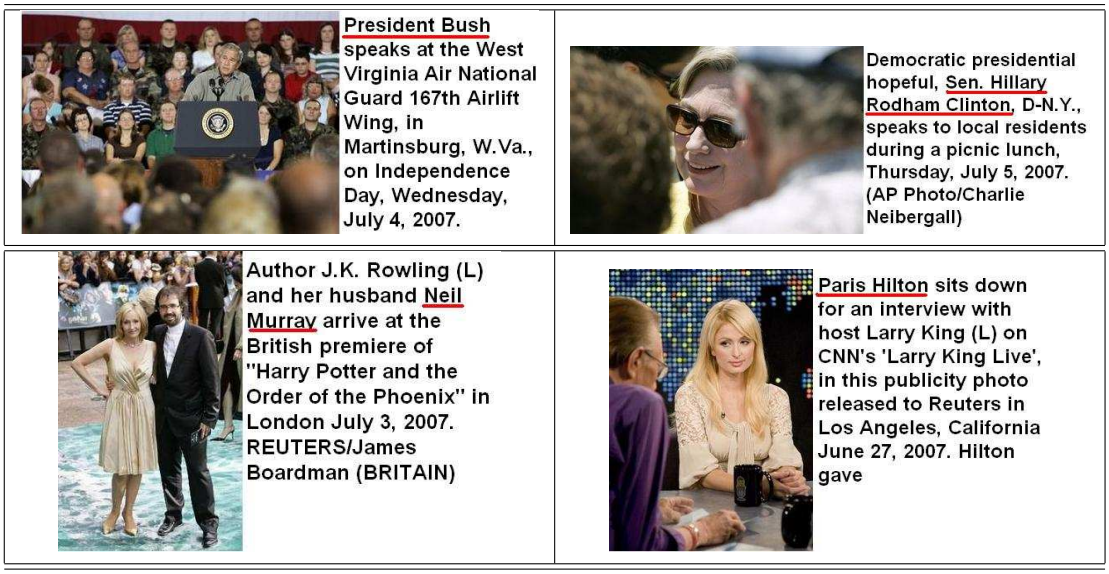


Figure 1.2: Sample news photographs and their associated captions. There can be more than one face on the photograph or more than one name in the caption.

illumination and facial expression, which cause the face recognition problem even more difficult.

Recently, it has been shown that the performance of person queries can be improved by integrating name and face information [17, 19, 27, 4, 13, 47]. When text information is provided together with the visual appearance, the face recognition problem can be simplified and transformed into the problem of finding associations between names and faces [44, 8, 9].

In this study, we propose a method for improving the performance of person queries in news datasets by exploiting from both text and visual information. A search is started first by looking for the name of a query person in the caption or in the speech transcript text. Based on this search, we limit our search space for the query person to the images/frames that are associated with the query name. Although, there may be faces corresponding to other people in the story, or some non-face images due to the face detection algorithm used, the faces of the query person are likely to be the most frequently appearing ones than any other person in its limited search space. 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. In other words, faces of the query person


Figure 1.3: Sample shots from news videos and the speech transcript texts aligned with those shots.

forms the largest group of similar faces in its limited search space.

In this context, the problem of finding the faces of the query person in the limited search space is dual to the graph problem of finding the largest densest component. To this end, we first find the similarities between faces in the search space and construct a similarity graph of faces, where nodes correspond to faces and edges correspond to face similarities. Then, we transform our problem to a graph problem of finding the largest densest component of the graph. This largest densest component refers to the biggest set of highly connected nodes in the graph; thus largest group of similar faces corresponding to the faces of the query person. When we find the faces of the query person with the proposed approach, the returned solution is also used as a model for recognizing new faces.

Different from the most of current face recognition systems, we find similarity between the two faces based on the SIFT features extracted from those faces. The proposed method exploits from the scale and illumination invariance characteristics of SIFT features. Besides, it also overcomes the problems in holistic appearance or local facial feature based face recognition models by being less


Figure 1.4: 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 (up image) or succeeding shots (bottom image).

sensitive to variations in noise, occlusion, and illumination; and by working also in the absence of any of the facial features.

The proposed method is not a solution to the general face recognition problem. Rather, it is a method to increase the retrieval performance of the person queries in the large data sets where names and faces appear together and where traditional face recognition systems cannot be used. It does not require a training step for a specific person and therefore, there is no limit on the number of people queried.

In the following two section, we briefly describe the overall approach, and then present the organization of thesis.

## 1.2 Summary of Contribution

Our person finding approach is based on first limiting the search space of a query person by using text information and then solving the problem by transforming

it to a graph problem. After finding the faces of the query person, we can use the result as a model for recognizing new faces. The overall approach consists of four steps: constructing a limited search space for a query person by using text and name information, defining similarities between faces in this search space to form a similarity graph of faces, finding the densest component of this graph which corresponds to the faces of the query person, and finally using the result as a model further in recognizing new faces. The main steps of the proposed method are given in Figure 1.6.

In the first step, we use the text information to limit our search space for a query name. It consists of searching for the query name in the caption or in the speech transcript, and choosing the images/frames that are associated with the query name. As stated earlier, there is usually a time shift between the appearance of a name and the appearance of the face belonging to that name in news videos. This problem is 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.

We assume that in this limited search space, faces of the query person appear more than faces of any other person. Also, faces of the same person tend to be more similar to each other than to the faces of others. Therefore, the faces of the query person forms the largest group of similar faces in the limited search space. Hence, in the second step, we assign a similarity measure between each pair of faces in the limited search space and represent these similarities among faces in a graph structure. In this similarity graph, nodes correspond to the faces and the edge weights correspond to the assigned similarities.

In this study, the similarity between faces is represented by using interest points extracted from the face. We use Lowe's SIFT features, which have been shown to be successful in recognizing objects [35, 39, 26] and faces [28]. Different from the original matching metric of SIFT, we assign the interest points having the minimum distance as the initial matching points. Then, we apply two constraint on these matched points: geometrical constraint and unique match constraint, to eliminate the false matches and select the best matching points.

Finally, the average distance of matching points between two faces is used to assign the distance between these two faces. Using the SIFT features, we both exploit from the scale and illumination invariant property of these features and are able to assign a distance metric between two faces even in the absence of any of facial features.

In the constructed similarity graph, the nodes corresponding to the faces of the query person will be more strongly connected to each other than to other nodes corresponding to other faces in the graph. Moreover, the query person is the one whose face usually appears the most frequently in its search space. Considering all these, the problem is transformed into a graph problem in the third step and solved by a greedy graph algorithm that returns the largest densest component of the graph. This largest densest component refers to the set of highly connected nodes in the graph; thus the set of most similar faces corresponding to the faces of the query person.

The final step involves using the result of the graph algorithm as a model to recognize new faces. With the proposed graph algorithm, we can automatically obtain the labeled training set for learning the model in recognizing new faces. We propose two different techniques to use the result based on: degree, distance, and match number modeling. In all those techniques, the model is trained by using the returned graph as the training set.

The method has been applied on two different news datasets: news photographs and news videos. The first data set, namely the news photographs collected by Berg *et al.* [8], is quite different from most of the existing data sets (see Figure 1.5). It consists of large number of photographs with associated captions collected from Yahoo! News on the Web. Photographs are taken in real life conditions rather than in restricted and controlled environments. Therefore, they represent a large variety of poses, illuminations and expressions. They are taken both indoors and outdoors. The large variety of environmental conditions, occlusions, clothing, and ages make the data set even more difficult to be recognized.

The second data set is broadcast news videos provided by NIST for TRECVID

Figure 1.5: Sample faces from news photographs [8] (on the top), and news videos (on the bottom).

2004 video retrieval evaluation [1]. Due to the higher noise level and lower resolution, news videos is a harder data set to work with. In order to handle the problem due to anchorperson faces, we add a mechanism to detect and remove the anchorpeople. The anchorperson is the one who appears the most frequently in each news video. Hence, we apply the densest component algorithm to each video seperately and automatically detect the anchorperson, since the faces of the anchorperson will correspond to the biggest densest component of the graph formed by the faces in that video.

## 1.3 Organization of the Thesis

The rest of the paper is organized as the following:

In Chapter 2, we summarize related previous works on the similar problems. in Chapter 3, we present a detailed explanation of the overall graph approach: association of names and faces to limit the search space for a query name, definition of the similarity measure to construct the similarity graph, finding the densest component of the graph, and using the obtained result in recognizing new faces. The method is also applied on news videos for detecting the anchorpeople automatically. In Chapter 4, experimental results are given along with the description of the datasets used. In Chapter 5, the proposed method is compared with some other possible methods. Finally, we summarize the overall work in Chapter 6 and

conclude with future research.

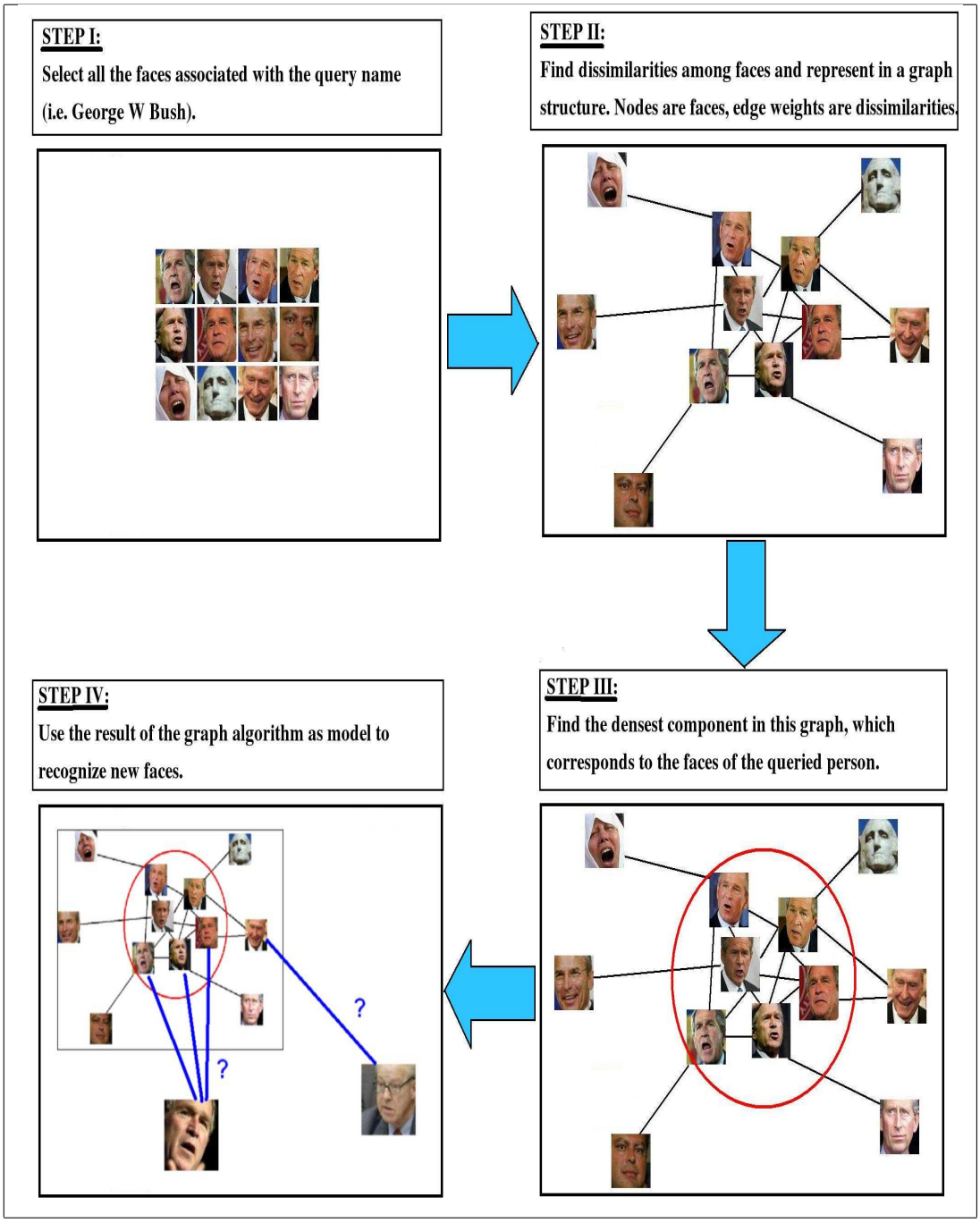


Figure 1.6: The four steps of our overall approach. **Step 1:** Limit the search space of a query person to the images that are associated with the query name. **Step 2:** Construct a similarity graph of faces in this search space. **Step 3:** Find the largest densest component of the graph corresponding to the faces of the query person. **Step 4:** Use the result of the previous approach as a model in recognizing new faces

# Chapter 2

# Background

In this thesis, we propose method for finding and recognizing faces in news by integrating names and faces with a graph based approach.. In the following sections, we first discuss some of the previous work on the use of name and text information. Then, we define the problem of face recognition and importance of interest points in literature on solving the recognition problem. Later in the last section, we briefly talk on the use of graph theoretical methods in computer vision.

## 2.1 On Integration of Names and Faces

News video and news photograph collections possess different types of resources, such as text, speech, and visuality. Recently, it has been shown that effective and efficient accessing and utilization of such multi-modal data can be simplified by integrating the different types of resources. In many of the previous works in literature, names and faces are associated for better query results.

In [52], Yang et al. showed that the combination of text and face information allows better retrieval performances in news videos. In that work, they avoid the difficulties of face recognition by using the text information for selecting

some shots as the initial results and applying face recognition on those shots. By this way, they aim to reduce the number of faces to be recognized, hence improve the accuracy. The timing between names and appearances of people is modeled by propagating the similarity scores from the shots containing the query person's name to the neighboring shots in a window. The Eigenface algorithm is used for face recognition. In the paper, an anchor detector has also been built by combining three information resources: color histogram from image data, speaker ID from audio data, face info from face detection. They use Fisher's Linear Discrimant (FLD) to select the distinguishing features for each source of data. Then, they unite the selected features into a new feature vector to be used in classification. Upon a similar approach, [16] uses the text and image features together to iteratively narrow the search for browsing and retrieval of web documents. [14] also unifies the textual and visual statistics in a single indexing vector for retrieval of web documents.

Berg et al. [9] proposed a method for associating the faces in the news photographs with a set of names extracted from the captions. In that paper, they first perform kernel principal component analysis (kPCA) to reduce the dimensionality of the image data and linear discriminant analysis (LDA) to project the data into its discriminant coordinates. Each image is then represented with both a vector gained after the kPCA and LDA processes, and a set of associated names extracted from the caption. A modified version of k-means clustering is used to assign a label for each image. Clusters that are far away from the mean are removed from the data, and discriminant coordinates are re-estimated. Finally, clusters which show high facial similarity are merged. The results given in this work, are then improved in [8] by analyzing language more carefully. In the latter work, they also learn a natural language classifier that can be used to determine who is pictured from text alone.

[19] integrates names and faces using speech transcripts, and improves the retrieval performance of person queries on TRECVID2004. It first searches over the speech transcript text and selects the key-frames that are aligned with the query name. Then, it applies Schneiderman-Kanade's face detection algorithm on each key-frame. However, many false positives are produced with such an

approach. Hence, skin color information is used to eliminate the false positives. The probability of a pixel being a skin pixel is modeled using a Gaussian probability distribution on HSV color space. Then, three features (color feature, PCA, ICA) are extracted from the faces to be use in grouping similar faces. While grouping, it uses G-means clustering and starts from small number of clusters, K, and increases K iteratively if some clusters fail the Gaussianity test (Kolmogorov-Simirov test) [25]. After grouping, each cluster is represented by one representative face: the one that is the closest to the mean of its cluster. Hence, the proposed method increases the speed of the system by reducing the number of images provided to the user. Anchor filtering is also experimented by selecting the anchor representatives and removing them from the rest of the cluster.

In [2] a method for automatically labeling of the characters in TV or film by using both visual and textual information. First, it aligns the subtitles and the script for tagging subtitles with speaker identity. It detects faces and then tracks to compose face tracks. While obtaining the face tracks, the number of point tracks which pass through faces is counted, and if this number is large relative to the number of point tracks which are not in common to both faces, a match is declared. To represent the appearance of a face, nine facial features are located and two local feature descriptors are used: sift and simple pixel-wised descriptor formed by taking the vector of pixels in the elliptical region and normalizing to obtain local photometric invariance. Further to use the additional cues, a bounding box is predicted for each face, which is expected to contain the clothing of the corresponding character. Speaker are detected by a simple lip motion detection algorithm. Then, each unlabeled face track is represented by a set of face and clothing descriptors. Finally, a probabilistic model is used to classify these tracks based on the weighted probabilities of the face and cloth appearance features.

## 2.2 On Face Recognition

Face recognition is a long-standing and a difficult problem, which has received great attention due to its demand on different fields like commercial and law enforcement applications. The problem has aroused interest in various science domains such as pattern recognition, computer vision, image analysis, psychology, and neurosciences. There is a direct relation among the findings in any of those science. As stated in [54], there exists still many ambiguous questions involved in the process of human perception of faces that psychologists and neuroscientists work on, such as:

- Is face recognition a unique and different from object recognition?
- Do we recognize faces as a whole or by individual features?
- Which features are more important for face recognition?
- Is human face perception invariant to changes in view-point, lighting and/or expressions?

In the context of computer vision, a broad statement of the face recognition problem is declared as the this: Given a still or a video image, identify or verify one or more persons in the image by using a stored database of faces( [54, 49]). There are three main steps involved in the configuration of a generic face recognition system (see Figure 2.1): detection of the face region in the image, feature extraction from the face region, recognition.

In [54] and in [48], the face recognition methods have been put into one of the three categories: holistic methods, local methods or hybrid methods. We give a brief summary of the those three approaches in the next three subsections.

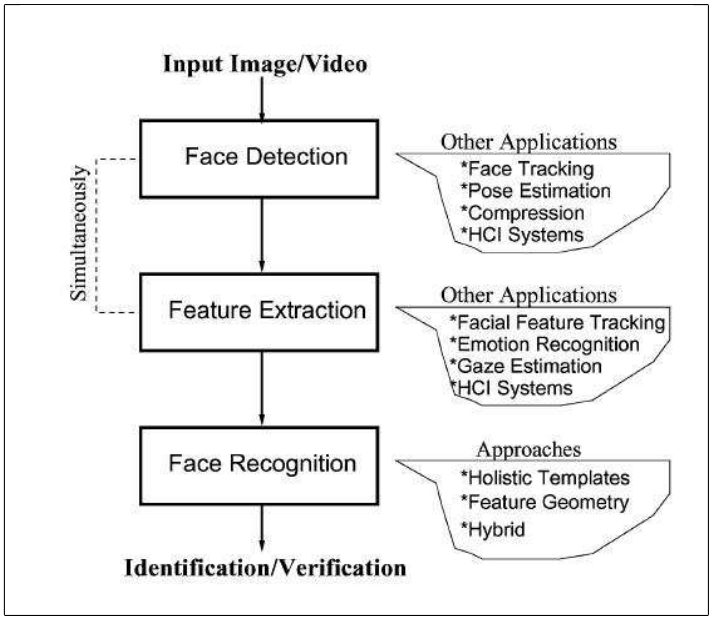


Figure 2.1: The main steps in the face recognition process (taken from [54]).

### 2.2.1 Holistic Methods

Holistic methods use te whole face as the raw input to the system. In those methods, each face image is represented with a vector composed by concatenating the gray-level pixel values of the image. Among the holistic methods, the most declared techniques used are Principal Component Analysis (PCA), Fisher Discriminant Analysis (FDA), Linear Discriminant Analysis (LDA), and Independent Component Analysis (ICA).

The main idea in PCA is to project the training data into a sub-dimensional feature space, in which basis vectors correspond to the maximum variance direction in the original image space. Each PCA basis vector was referred as an "eigenface" in [36, 37] that can be displayed as a sort of ghostly face. In the classification stage, the input face image is first projected into the subspace spanned by the eigenfaces; where each face is represented by a linear combination of the eigenfaces. Then, the new face is classified according its position in the face space to the positions of the known individuals. Later in literature, extensions of the Principal Component Analysis have been proposed as in [53] by using two-dimensional PCA and in [50] by selecting discriminant eigenfaces for face

recognition.

In [5], Bartlett et al. claimed that ICA representations of faces are superior to PCA; hence ICA can perform better performances across sessions and changes in expression. In ICA, data is projected onto some basis that are statistically independent. It also attempts to minimizes second-order and higher-order dependencies in the input data.

Holistic methods can be advantageous in the context of covering the global appearance of the faces. One other edge of the holistic methods is that they maintain the diffuse texture and shape information; hence can differentiate the faces. However, representing each face image by a feature vector makes the holistic methods sensitive to variations in appearance caused by occlusion, changes in illumination and/or expressions. To overcome this problem, local feature representations have been developed in literature that are less sensitive to changes in appearance.

### 2.2.2 Local Methods

The importance of hair, eyes and mouth in face human perception has been highlighted in psychology and neurosciences ( [29, 11]). It has almost been claimed in [11] that nose plays an important role in face recognition. On these grounds, local methods have been presented that represent each face image by a set of low dimensional feature vector, which usually correspond to the facial features like eyes, mouth, and nose.

Mainly, there are two approaches used in local methods: local feature-based methods and local appearance-based methods. The geometric relations of the features are considered in local feature-based methods. In some primitive studies of this types, only the geometric measures –such as distance between the eyes or the size of the eyes– have been considered [32, 31]. Then, in [38] Manjunath et al. proposed a method that preserves both the local information and the global topology of the face. The method stores both the local information and the

feature information of the detected facial features; and constructs a topological graph using these features. Then, the classification stage, the problem is solved by a graph matching scheme.

Even if the method in [38] is advantageous since it considers both the similarity of the local features and the global topology, it is sensitive variations that causes any change in this topology. Consequently, Lades et al. [12] proposed Elastic Bunch Graph Matching Scheme (EBGM) that is based on a deformable topology graph. Even though the method resolves the problem caused by the changes in appearance, it cannot remove the problem caused by the occlusion of any of facial features.

Motivated by the foundings in psychology that a set of simple lines can characterize the structure of an object, hence is sufficient to recognize its shape; a face is represented by the Face-ARG in [42]. Face-ARG consists of a set of nodes that corresponds to line features, and binary relations between them. Using a Face-ARG, all the geometric quantities and structural information of the face can be encoded in an Attributed Relational Graph (ARG) structure. Then, in the classification stage, partial ARG matching is applied on the constructed Face-ARG's of the test and reference faces. The advantage of the method over recent methods is that it can still work in presence of occlusion and expression changes.

Most of the local feature-based methods are sensitive to accurate feature point localization; which is still an unresolved problem. Hence, in local appearance-based methods, facial feature regions are detected first and features (such as Gabor wavelet [38] and Haar wavelet [34]) are extracted from these regions in the image. Different classifiers are applied on these features finally in the classification step.

### 2.2.3 Hybrid Methods

It is still a question if we recognize faces as a whole or by individual features. Holistic methods cover the global face appearance and can maintain the diffuse

texture and shape information. However, they are very sensitive to the changes in appearance that may be caused by occlusion, changes in illumination and/or expressions. Local methods is less sensitive to those changes, however accurate localization of facial feature points still remains that can affect the performance of recognition. Especially for images of small size, facial feature detection is even more problematic; hence the local methods cannot be applied in that case. How to use the local features to without disregarding the global structure is another open problem. To overcome those problems and benefit from different advantages of both approaches, hybrid methods can be used. Hybrid methods aim to use both holistic and local approaches. However, how to use which features and which classifiers in hybrid methods is still indeterminate and not much investigated in literature yet.

## 2.3 On the Use of Interest Points

Recently, it has been shown that local image features provide a good representation of the image for recognition and retrieval [6, 41], while global features tend to be sensitive to image variations in viewpoint, pose and illumination. Among the local features, Scale Invariant Feature Transform (SIFT) technique has been shown to be successful in recognizing objects [35, 39, 26] and faces [28, 10]. The technique, which has first been presented in [35], provides distinctive invariant features that can be used in reliable matching between different images of an object or scene. The most powerful aspect of these features is that they are invariant to image scale and rotation, and partially invariant to change in illumination and 3D camera viewpoint.

In [10], application of the SIFT approach in the scope of face authentication has been invested. There different matching schemes are proposed in the paper: 1. Minimum pair distance, 2. Matching eyes and mouth, 3. Matching on a regular grid. The first scheme proposes to compute the distances between all pairs of keypoints in two images and assign the minimum distance as the matching score. Using the that face and mouth regions provide most of the information for face

recognition, only the features in these regions are used in the second scheme. Finally, feature locations are considered in the third scheme by dividing the face image into grids and matching the features of corresponding grids.

Sivic et al. has proposed a person retrieval system in [28] that represents each face image as a collection of overlapping local SIFT descriptors placed at the five facial feature locations (eyes, mouth, nose, and middle point between the eyes). They first use tracking to associate faces into face-tracks within a shot to obtain multiple exemplars of the same person. Then, they represent each face-track with a histogram of precomputed face-feature exemplars. This histogram is used for matching the face-tracks; hence retrieving sets of faces across shots.

## 2.4 On the Use of Graph Theoretical Methods in Computer Vision

Graph theoretical approaches have recently been used in computer vision problems due to their representational power and flexibility. They allow vision problems to be cast in a strong theoretical area and access to the full depot of graph algorithms developed in computer science and operations research. The most common graph theoretical problems used in computer vision include maximum flow, minimum spanning tree, maximum clique, shortest path, maximal common subtree/ subgraph, graph partitioning, graph indexing, graph matching, etc. [18].

Graph partitioning algorithms that have been used in [22, 30, 45], target the two typical applications of computer vision: image segmentation and perceptual grouping. They address the problem of making cuts in a weighted graph according to an appropriate minimum weight criterion. In these works, data elements (i.e. image pixel points) correspond to the vertices in the graph, and similarity between any two vertices correspond to the edge weight between those vertices. In [7], the problem of content based image retrieval has been solved by a graph matching scheme. The main idea used was to represent an image query as an attributed relation graph, and select a small number of model image graphs that are similar

to the query image graph.

[3] has proposed a graph theoretic clustering method for image grouping and retrieval. The motivation of the work was that an efficient retrieval algorithm should be able to retrieve images that are not only close (similar) to the query image but also close to each other. However, most of the existing feature extraction algorithms cannot always map visually similar images to nearby locations in the feature space. Hence in the retrieval step, it is often to retrieve irrelevant images (or not to retrieve relevant images) simply because they are close to the query image (or a bit far away from the query image). In this context, they retrieve best $N$ matches for a query image, and best $N$ matches of each of the retrieved images. A graph is constructed with all those retrieved images, in which nodes corresponds to the images and edge weights correspond to the similarities. Then, the retrieval problem is transformed into and solved by the problem of finding the set of nodes in the graph, that are not as dense as major cliques but are compact enough within user specified thresholds.

# Chapter 3

# Graph Based Person Finding Approach

## 3.1 Overview

It is likely that in the news, a person will appear when his/her name is mentioned. Following up this cue, we start search for a person by first looking for the name of the query person and limit our search space to the images that are associated with that name. Although there might be the faces of other people or non-face images in this limited search space, mostly query person will be the one that appears more than any other individual. Visually, faces of a particular person tend to be more similar to each other than to faces of other people. Based on these assumptions, we transform the problem to a graph problem, in which the nodes correspond to faces and the edges correspond to the similarities between faces, and we seek to find the largest densest component in this graph. Hence, if we can define a similarity measure among the faces in the limited search space 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 three sections, we first explain the steps of the graph based

person finding approach: integrating names and faces to limit the search space, defining a similarity measure between faces to construct the similarity graph, and the greedy graph algorithm to find the largest densest component of the graph. Following those sections, we will explain the use of person finding approach for automatic anchorperson detection in news videos. Then, in the last section we define two methods to use the output of the three step person finding scheme later in recognizing new faces.

## 3.2 Integrating Names and Faces

The first step of our algorithm involves integrating name and face information. In this step, we use the name information to limit our search space to the images around which the name of the query person appears. To this end, we look for the name of the query person in the caption or in the speech transcript; and choose the images that are associated with the query name.

On the web, the news photographs appear with the captions. However, there can be more than one face in a photograph and more than one name in the corresponding caption. Therefore, it is not known which face goes with which name. Using the assumption that a person is likely to appear in a photograph when his/her name is mentioned in the caption, we reduce the face set for a queried person by only choosing the photographs that include the name of that person in the associated caption. However, a person's name can appear in different forms. For example, the names *George W. Bush, President Bush, U.S. President*, and *President George Bush*, all correspond to the same person. We merged the set of different names used for the same person to find all faces associated with all different names of the same person. A detailed list of different forms of names corresponding to each query person used in our experiments is given in the appendix.

For the news videos, 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, it is problematic since there can be a time shift between the appearance of the person visually and the appearance of his/her name.

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 [52]. 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 Figure 4.9). 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. As we will explain in the experiments, it is seen that taking only one preceding and two following shots (which we refer to as [-1,2]) is also a good choice.

Another problem in news videos is that usually the faces of the anchorperson appear around a name. For solution to this problem, we use an anchorperson detection method based on our graph based approach as we will be explained later.

Integrating names and faces produces better retrieval performances compared to solely text-based methods. However, the resulting set may still contain many false faces due to the following reasons: the query person may not appear visually even if his/her name is mentioned, there may be other people in the same story that also appear visually together with the query person, there may be non-face images returned by the face detection algorithm used. However, the faces of the query person are likely to be the most frequently appearing ones than any other person in the same space. 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. In the following sections, we explain our strategy to increase retrieval performance by finding the correct faces by using visual similarities.

## 3.3 Constructing Similarity Graph of Faces

We represent the faces with the interest points extracted from the images using the SIFT operator [35]. Lowe's SIFT operator [35], have recently been shown to be successful in recognizing objects [39, 26] and faces, [28]. The SIFT technique consists of four main steps: 1. Scale-space extrema detection, 2. Keypoint localization, 3. Orientation Assignment, 4. Keypoint descriptor.

In the first step, potential interest points are extracted by looking for all scales and image locations. A scale space of the image is constructed first by convolving the image with variable-scale Gaussian function. Let the input image be $I(x, y)$ then, the Gaussian-blurred image $L(x, y, \sigma)$ can be represented by:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y),$$

where,

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}.$$

Difference-ofGaussian function (DoG) is used to detect stable keypoint locations in this scale space. DoG of two nearby scales seperated by constant k is giveb by:

$$\begin{aligned} DoG(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma). \end{aligned}$$

Local maxima and/or minima of DOF gives the candidate keypoint location, and is computed by comparing each sample point with both the eight neighbor points on the same scale and the nine neighbor points on two images in the neighbor scale.

In the second step, each candidate keypoint location is fit to a nearby data to determine its location, scale and ratio of principal curvatures. First, a threshold

on minimum contrast is applied to remove the unstable extrema with low contrast. Then, a second threshold is applied on the ratio of principle curvatures to eliminate the strong responses of difference-of- Gaussian function along edges which are poorly determined.

An orientation is assigned to the keypoint locations in the third step, based on local image gradient directions. An orientation histogram is formed from gradient orientations of the sample points around a keypoint by precomputing their pixel differences in a scale invariant manner. Peaks in the histogram denominate dominant directions of local gradients; thus the 80 per cent of the highest peak in the histogram is used to assign the orientation of a keypoint.

In the last step, a descriptor for each keypoint is computed from image gradients. Gradients of points within an array around the keypoint are weighted by a Gaussian window and its content is summarized into a descriptor array, using orientation histograms. These features are then normalized to unit length and a threshold is applied to this unit vector values to reduce the effects of illumination change.

We first use a minimum distance metric to find all matching points and then remove the false matches by adding some constraints. For each pair of faces, the interest points on the first face are compared with the interest points on the second face and the points having the least Euclidean distance are assumed to be the correct matches. However, among these there can be many false matches as well (see Figure 3.1).

In order to eliminate the false matches, we apply two constraints: the geometrical constraint and the unique match constraints. Geometrical constraint expects the matching points to appear around 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. In the next two subsections, we give the details of how those constraints are applied.

### 3.3.1 Geometrical Constraint

We expect that matching points will be found around similar positions on the face. For example, the left eye usually resides around the middle-left of a face, even in different poses. This assumption presumes that the matching pair of points will be in close proximity when the normalized coordinates (the relative position of the points on the faces) are considered.

To eliminate false matches which are distant from each other, we apply a geometrical constraint. For this purpose, we randomly selected a set of images of 10 people (5 faces for each person). Then, we manually assigned true and false matches for each comparison and used them as training samples to be run on a quadratic Bayes normal classifier ( [43, 51]) to classify a matched point as true or false according to its geometrical distance. The geometrical distance corresponding to the $i^{th}$ assignment refers to $\sqrt{X^2 + Y^2}$ where

$$X = \frac{locX\,(i)}{sizeX\,(image1)} - \frac{locX\,(match\,(i))}{sizeX\,(image2)},$$

$$Y = \frac{locY\,(i)}{sizeY\,(image1)} - \frac{locY\,(match\,(i))}{sizeY\,(image2)},$$

and $locX$ and $locY$ hold $X$ and $Y$ coordinates of the feature points in the images, $sizeX$ and $sizeY$ hold $X$ and $Y$ sizes of the images and $match(i)$ corresponds to the matched keypoint in the second image of the $i^{th}$ feature point in the first image.

In Figure 3.1, matches before and after the application of this geometrical constraint are shown for an example face pair. Most of the false matches are eliminated when the points that are far away from each other are removed.

The relative angle between the points could also be used as a geometric constraint. However, since the closer points could have very large angle differences, it is not reliable.

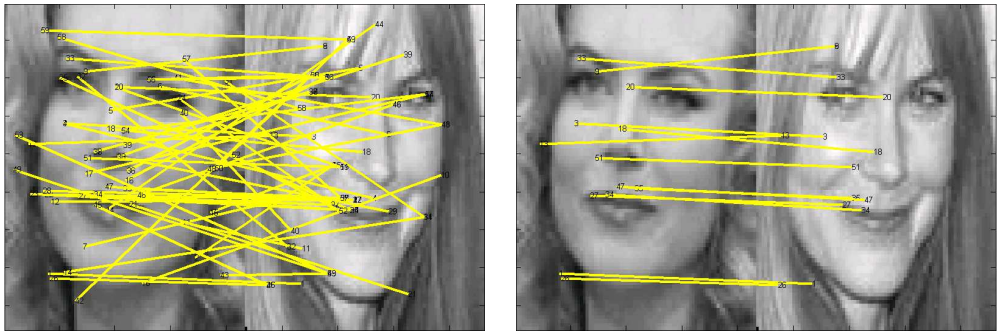

Figure 3.1: The first image on the left shows all the feature points and their matches based on the minimum distance. The second image on the right shows the matches that are assigned as true after the application of geometrical constraints.

### 3.3.2 Unique Match Constraints

After eliminating the points that do not satisfy the geometrical constraints, there can still be some false matches. Usually, the false matches are due to *multiple assignments* which exist when more than one point (e.g, $A_1$ and $A_2$) are assigned to a single point (e.g, $B_1$) in the other image, or to *one way assignments* which exist when a point $A_1$ is assigned to a point $B_1$ on the other image while the point $B_1$ is assigned to another point $A_2$ or not assigned to any point (Figure 3.2). These false matches can be eliminated with the application of another constraint, namely the unique match constraint, which guarantees that each assignment from an image $A$ to another image $B$ will have a corresponding assignment from image $B$ to image $A$.

The false matches due to multiple assignments are eliminated by choosing the match with the minimum distance. The false matches due to one way assignments are eliminated by removing the links which do not have any corresponding assignment from the other side. An example showing the matches before and after applying the unique match constraints are given in Figure 3.3 and in Figure 3.4.

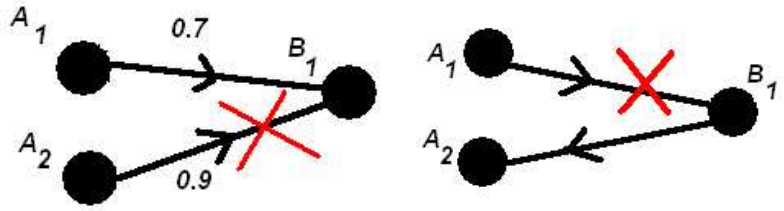


Figure 3.2: For a pair of faces $A$ and $B$, let $A_1$ and $A_2$ be two points on $A$; and $B_1$ is a point on $B$ with the arrows showing the matches and their direction. On the left is a *multiple assignment* where both points $A_1$ and $A_2$ on $A$ match $B_1$ on $B$. In such a case, the match between $A_2$ and $B_2$ is eliminated. On the right is a *one way match* where $B_1$ is a match for $A_1$, whereas $B_1$ matches another point $A_2$ on $A$. The match of $A_1$ to $B_1$ is eliminated. The match of $B_1$ to $A_2$ remains the same if $B_1$ is also a match for $A_2$; otherwise it is eliminated.

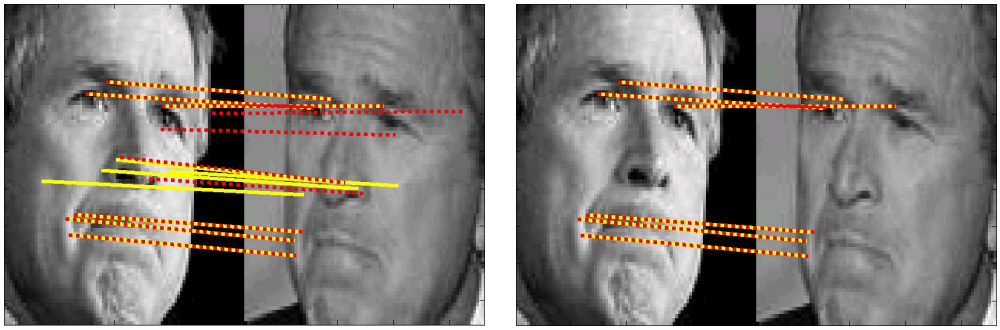

Figure 3.3: An example for unique match constraint. Matches from the left to the right image are shown by red, dashed lines, whereas matches from right to left are shown by yellow lines. The left image shows the matches assigned after applying geometrical constraints, but before applying the unique match constraints. The right image shows the remaining matches after applying the unique match constraints.

### 3.3.3 Similarity Graph Construction

After applying the constraints and assuming that the remaining matches are true matches, we define the distance between the two faces $A$ and $B$ as the average value of all matches.

$$dist(A, B) = \frac{\sum_{i=1}^{N} D(i)}{N},$$

where $N$ is the number of true matches and $D(i)$ is the Euclidean distance

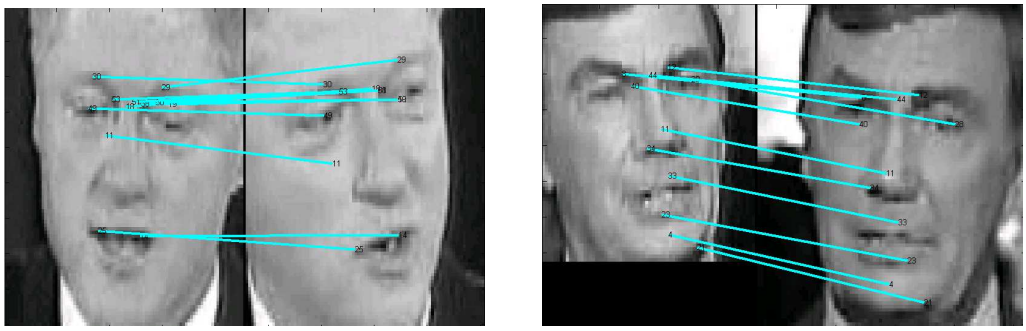

Figure 3.4: Examples for matching points. Note that, even for faces with different size, pose or expressions the method successfully finds the corresponding points.

between the SIFT descriptors of the two points for the $i^{th}$ match.

A similarity graph for all faces in the search space is then constructed using these distances. We can represent the graph as a matrix as in Figure 3.5. The matrix is symmetric and the values on the diagonal are all zero. For a more clear visual representation, the distances for the faces corresponding to the person we are seeking are shown together. Clearly, these faces are more similar to each other than to the others. Our goal is to find this subset which will correspond to the densest component in the graph structure.

## 3.4 Greedy Graph Algorithm for Finding the Densest Component

In the constructed similarity 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

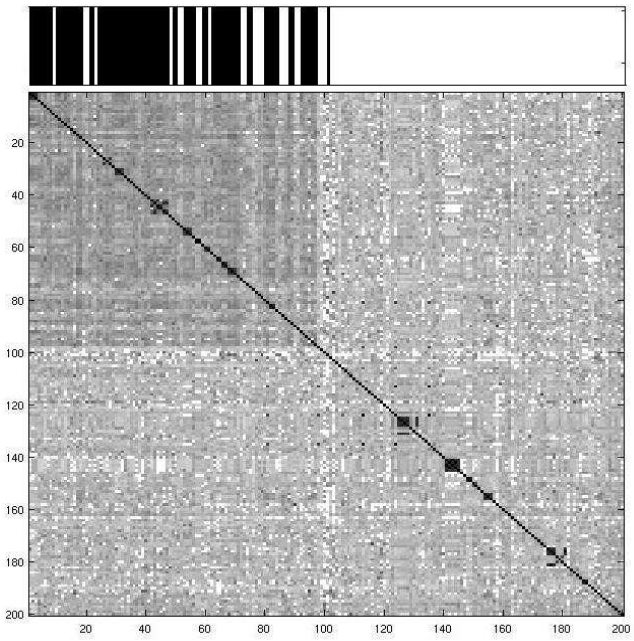


Figure 3.5: Dissimilarity matrix for 200 images in the search space for the name *Hans Blix*. In this search space, 97 of the images are true *Hans Blix* images, and the remaining 103 are not. For visualization, the 97 true *Blix* images are put on the top left of the matrix. Dark colors correspond to larger similarity values.

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

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

Our goal is to find the subgraph $S$ with the largest average degree that is the subgraph with the maximum density. Initially, the algorithm presented in [15] starts with the entire graph $G$ and sets $S = G$. Then, in each step, the vertex with the minimum degree is removed from $S$. The algorithm also computes the value of $f(S)$ for each step and continues until the set $S$ is empty. Finally, the set $S$, that has maximum $f(S)$ value, is returned as the densest component of the graph.

In order to apply the above algorithm to the constructed similarity graph, we need to convert it into a binary form, since the algorithm described above only works well for binary graphs. Thus, before applying it, we convert our original dissimilarity values into a binary form, in which 0 indicates no edge and 1 indicates an edge between two nodes. This conversion is carried out by applying a threshold on the distance between the nodes. This threshold also connotes what we define as near-by and/or remote. An example of such a conversion is given in Figure 3.6. In the example, assume that 0.65 is defined as our proximity threshold. In other words, if the distance between two nodes is less than or equal to 0.65 then these two nodes are near-by; therefore we put an edge between them. Otherwise, no edge is maintained between these nodes, since they are far away from each other.

## 3.5 Anchorperson Detection and Removal for News Videos

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 (see Figure 3.7). The solution is to detect and remove the anchorperson before applying the algorithm

In [17], 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. Hence, we apply the densest component based method to each news video separately, to find the people appearing most frequently, which correspond to the anchorpeople.

## 3.6 Dynamic Face Recognition

The overall scheme explained in the previous sections returns a set of images classified as the query person (densest component) and the rest as others (outliers). Also, the graph algorithm works on the whole set of images in the search space of the query person. Thus, when a new face is encountered, the algorithm needs to be re-run on the whole set to learn the label of the new face–query person or outlier. However, since the scheme returns the classified images, this result can be used as a model to recognize new faces dynamically and check if it belong to the faces of the queried person. Moreover, this task can be achieved without any supervision, since the scheme provides us the training data labeled automatically.

In the next two subsections, we explain how the output of the person finding approach can be used in recognizing new faces. We model the returned solution in two ways to learn the thresholds for: average degree and average distance.

### 3.6.1 Degree Modeling

As explained in 3.4,the greedy densest component algorithm works iteratively by removing one node from the graph until there is one last node left in the graph. Average density of each subgraph is computed in each iteration and

finally the subgraph with the largest average density is assigned as the densest component. Among these iterations, there is one *lastnode* removed from the current subgraph that results in the densest component in the next iteration. This last node can be thought of as the breaking point, and indicate an evidence for the maximum number of total connections (edges) from an outlier node to all the nodes in the densest component. This total number–degree of the nearest outlier to all the faces recognized as the query person–can used as a threshold in further recognition. When a new face is encountered, its degree to all the faces in the densest component is computed first. Then, the face is labeled as the query person if its degree is greater than the found degree threshold.

### 3.6.2 Distance Modeling

In this method, average distance of true-true and false-true matches are used. For each node in the graph, its average distance to all the nodes in the densest component– hence the faces of the query person–are computed. If a node was in the densest component, then its average distance is labeled as a true-true match distance, else a false-true match distance. These distances are then trained with the quadratic Bayes normal classifier ( [43, 51]) to learn the average distance threshold and classify new test images based on its average distance to true images in the training set.

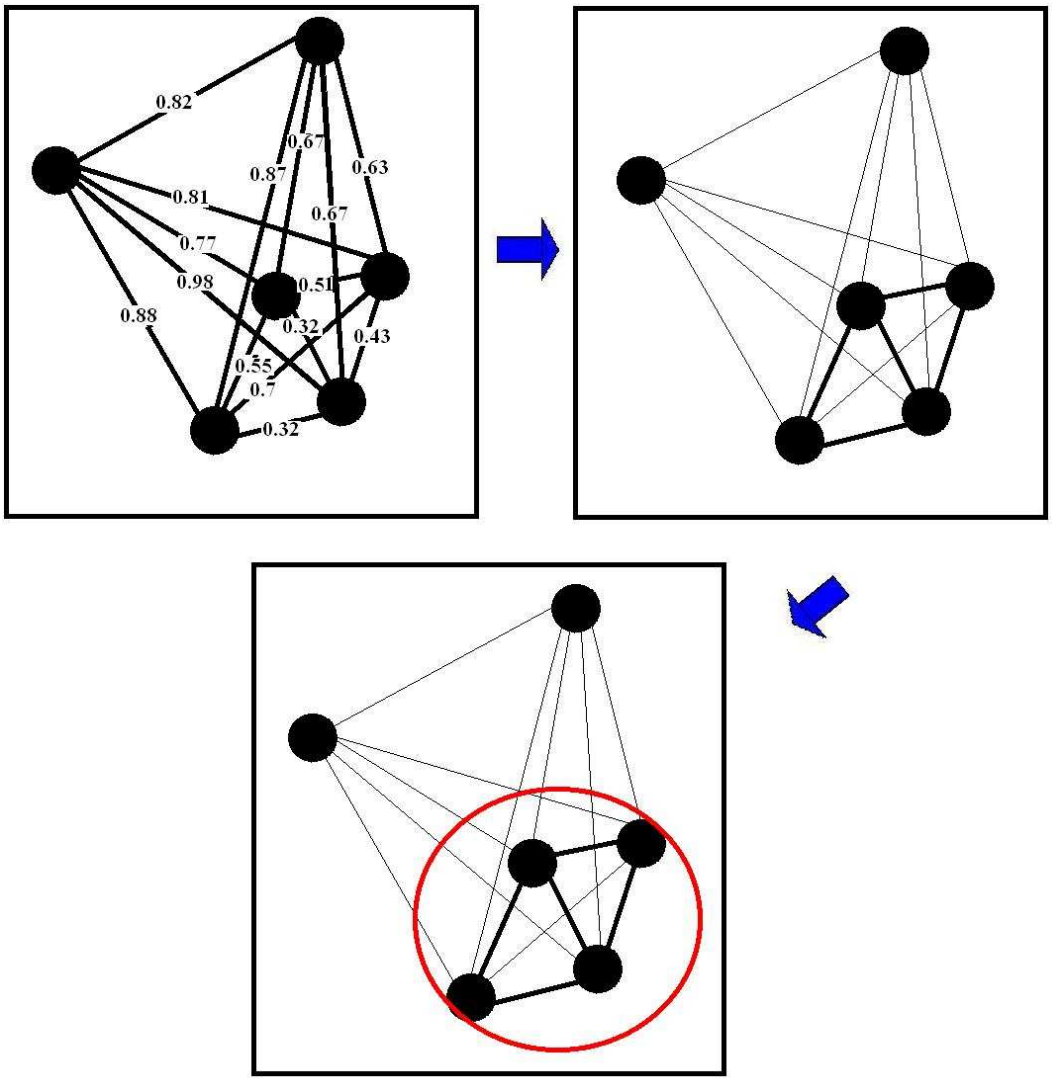


Figure 3.6: Example of converting a weighted graph to a binary graph. Nodes and their distances are given in the first image. The resulting graph after applying 0.65 as the proximity threshold is given in the second image. Bold edges are the edges that remain after conversion. The final densest component of this graph is circled in the last image.

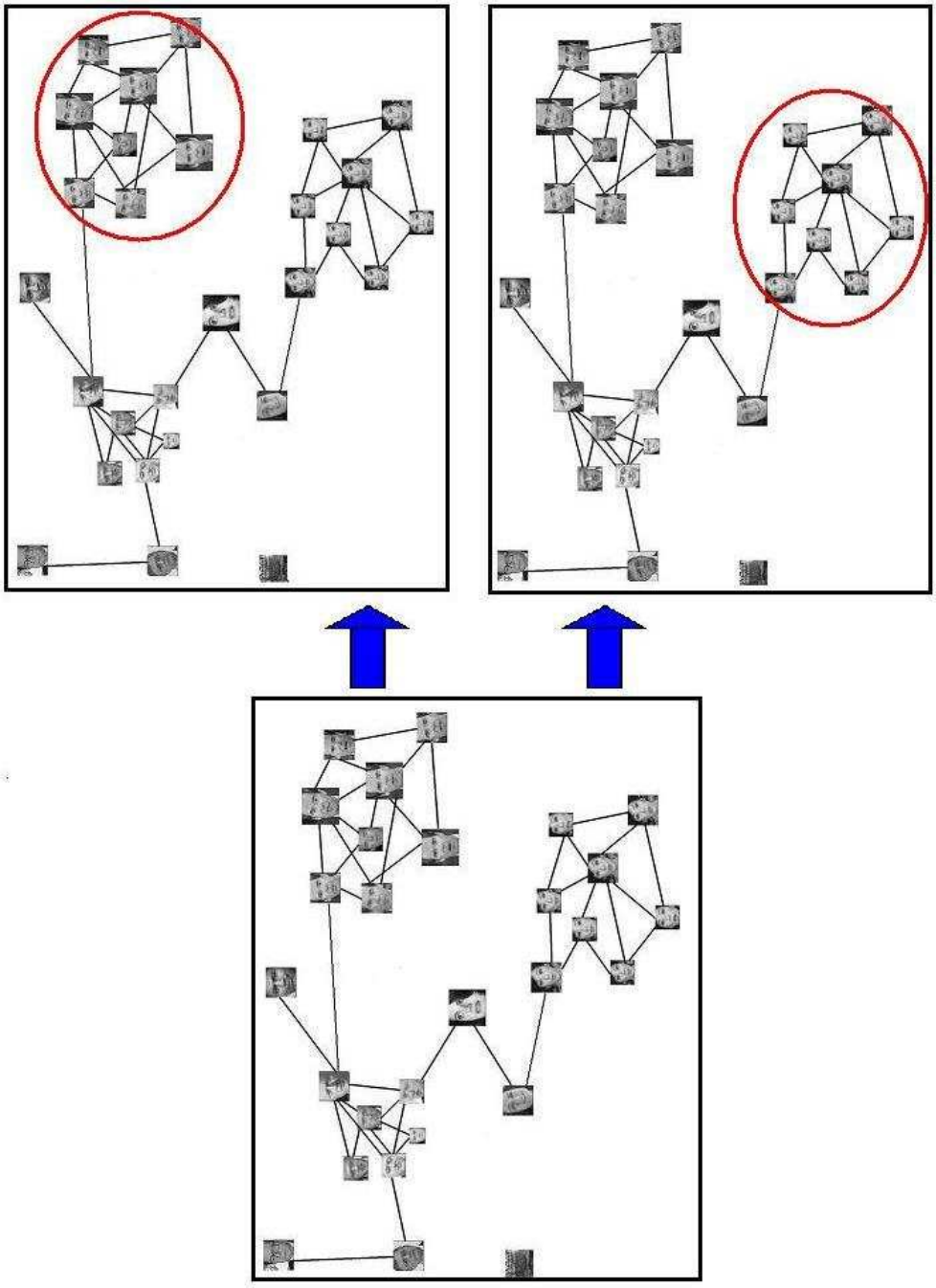

Figure 3.7: Anchorperson problem in news videos. After integrating names and faces for a query person, it is highly probable that either one of the anchorpersons will be returned as the densest component of the limited search space. The first figure on the left corresponds to a sample

# Chapter 4

# Experiments

## 4.1 Data Sets

The method proposed in this thesis was tested on two different data sets: news photographs on the web and broadcast news videos on television. In the next two subsections, we briefly describe both data sets used in our experiments.

### 4.1.1 News Photographs

The data set constructed by Berg *et al.* originally consists of about half a million captioned news images collected from Yahoo! News on the Web. After applying a face detection algorithm and processing the resulting faces, they were left with a total of 30,281 detected faces [8]. Each image in this set is associated with a set of names. A total of 13,292 different names are used for association. However more than half (9,609) of them are used only once or twice. Also, as we mentioned previously, a particular person may be called by different names. For example, the names used for *George W Bush* and their frequency are: *George W (1485); W. Bush (1462); George W. Bush (1454); President George W (1443); President Bush (905); U.S. President (722); President George Bush (44); President Bushs (2); President George W Bush (2); George W Bush (2).* We merge the set of

different names used for the same person and then take the intersection to find faces associated with different names of the same person. A detailed list of different forms of names corresponding to each query person used in our experiments is given in the appendix.

Generally, the number of faces in the resulting set is less than the number of all names since a caption may include more than one instance of the referred name. For example, for *Bush* the number of faces is 2,849 while the total number of all referred names is 7,528. In the experiments, the top 23 people appearing with the highest frequencies (more than 200 times) are used. Figure 4.1 shows the total number of faces associated with the given name and the number of correct faces for the 23 people used in the experiments.

### 4.1.2 News Videos

The second 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 [21] are used to obtain the associated text for each shot.

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 [40] 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.

## 4.2 Evaluation Criteria

For evaluation, we give the experimental results based on recall and precision values. These values are computed as follows: Let $N$ be the total number of faces returned by the algorithm as the faces of the query person. Among those $N$, let $n$ be the total number of faces that really belong to that person. Then, the precision value of this result is:

$$precison = \frac{n}{N}.$$

If there is a total of $m$ faces in the whole dataset that belong to the query person, then the recall value of the result is:

$$recall = \frac{n}{m}.$$

We should also denote that as the baseline, we use the images returned by the face detector. Hence, $m$ (ground truth of the query person) is computed among those detected faces.

After finding the recall and precision values for each query person individually, we finally compute the weighted results for average recall and precision values. What we mean with weighted is that recall and/or precision value of each person is weighted by the number of images that appear in his/her limited search space. Let $recall(i)$ be the recall value for the $i^{th}$ person, and $S(i)$ be the total number of images in its limited search space. Then, weighted average recall is:

$$w_avg_recall = \frac{\sum S(i) * recall(i)}{\sum S(i)}.$$

In the following two subsections, we give the results of these experiments on both sets separately:

## 4.3 Experimental Results on News Photographs

### 4.3.1 Matching Points

As the first step, the points having the minimum distance according to their SIFT descriptors are defined as the matching points. These points are further eliminated using the two constraints. After this elimination process, 73% of all possible true matches are kept and we lose only 27% of true matches. Among these assignments, we achieved a correct matching rate of 72%.

### 4.3.2 Graph Approach

The success of our algorithm varies with the threshold that is chosen while converting the weighted dissimilarity graph to a binary one. For the news photographs data set, average recall and precision values by varying the threshold between 0.55 and 0.65 is given in Figure 4.2. Based on these values, the threshold 0.575 is chosen to represent the recall and precision values for each person.

The threshold 0.575 is chosen to represent the recall and precision values for each person These values are given in Figure 4.3 for this threshold. Average precision value is obtained as 48% for the baseline method which assumes that all the faces appearing around the name is correct. With the proposed, method we achieved 68% recall and 71% precision values on the average. The method can achieve up to 84% recall- as for *Gray Davis*- and 100% precision - as for *John Ashcroft, Hugo Chavez, Jiang Zemin and Abdullah Gul*. We had initially assumed that, after associating names, true faces of the queried person appear more than any other person in the search space. However, when this is not the case, the algorithm gives bad retrieval results. For example, there is a total of 913 images associated with name *Saddam Hussein*, but only 74 of them are true *Saddam Hussein* images while 179 of them are *George Bush* images.

To show that our system works also on individuals appearing in a small number of captions, we performed experiments on 10 people appearing less than 35 times and obtained average recall and precision values 85% and 66%. As another experiment, we changed the number of instances of a face by removing some of the correct faces or by adding some incorrect faces. For 4 people having around 200 instances and similar number of true and false images (i) we removed 50 of true images of from each of their search space, (ii) we added 100 false images. Originally, average recall and precision values were 63% and 95%. We obtained 59% recall an 89% precision after (i), and 58% recall and 70% precision after (ii). Although the precision is somewhat affected, results are still acceptable.

Some sample images retrieved and not retrieved for three people from the test set are shown in Figure 4.4.

### 4.3.3 Online Recognition

The recognition methods explained in 3.6 are tested on the news photographs dataset. In these experiments, K percentage of the images in the search space of a query person is selected as the held-out set and the graph algorithm is applied on the remaining images to get the model. Average results of the degree modeling method are given in Figure 4.5 and in Table 4.1. And the average results of the distance modeling method are given in Figure 4.6 and in Table 4.2. For K = 10, the recall and precision values for each 23 person is given in Figures 4.7 and 4.8 respectively. for distance modeling technique.

## 4.4 Experimental Results on News Videos

### 4.4.1 Integrating Faces and Names

For better understanding of the distributions, we plot the frequency of faces relative to the position of the names for the five people that we have chosen for

Table 4.1: Recognition rates of degree modeling for different K values. (K per cent of the images are used as the held-out set.

| K | 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90 |
|---|---|---|---|---|---|---|---|---|---|
| model | | | | | | | | | |
| recall | 0.68 | 0.68 | 0.67 | 0.66 | 0.66 | 0.65 | 0.63 | 0.61 | 0.58 |
| precision | 0.71 | 0.71 | 0.71 | 0.70 | 0.70 | 0.70 | 0.69 | 0.68 | 0.64 |
| test set | | | | | | | | | |
| recall | 0.69 | 0.70 | 0.70 | 0.70 | 0.69 | 0.69 | 0.69 | 0.70 | 0.76 |
| precision | 0.71 | 0.71 | 0.70 | 0.70 | 0.70 | 0.69 | 0.68 | 0.66 | 0.62 |

Table 4.2: Recognition rates of distance modeling for different K values. (K per cent of the images are used as the held-out set.

| K | 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90 |
|---|---|---|---|---|---|---|---|---|---|
| model | | | | | | | | | |
| recall | 0.69 | 0.68 | 0.67 | 0.67 | 0.66 | 0.65 | 0.63 | 0.61 | 0.58 |
| precision | 0.71 | 0.71 | 0.70 | 0.70 | 0.70 | 0.70 | 0.69 | 0.68 | 0.65 |
| test set | | | | | | | | | |
| recall | 0.72 | 0.70 | 0.70 | 0.68 | 0.67 | 0.67 | 0.64 | 0.62 | 0.57 |
| precision | 0.72 | 0.72 | 0.72 | 0.72 | 0.72 | 0.72 | 0.71 | 0.71 | 0.69 |

our experiments in Figure 4.10. 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 4.3 shows that, most of the correct faces fall into this selected range by removing many false alarms.

Table 4.3: 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 |

### 4.4.2 Anchor Detection

We applied 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 Figure 4.11.

### 4.4.3 Graph Approach

In order to determine a reasonable threshold used in converting the weighted similarity graph to a binary one for the news videos, we randomly selected 10 videos and recorded recall-precision values of different thresholds for anchorperson detection. These values are plotted in Figure 4.12. 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.

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 4.4. 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 Figure 4.14. Some sample images retrieved for each person are shown in Figure 4.13.

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

Table 4.4: 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 |

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.

## 4.5 A Method for Finding the Graph Threshold Automaticaly

The normalized cut metric presented in [45] can be used for selecting the threshold to convert the weighted graph to a binary one. Let A be the set of vertices of a cut and V be the set of all vertices in graph G. Then the value of a cut and normalized cut of A are defined as follows:

$$cut(A,V) = \sum_{u \epsilon A, v \epsilon V - A} w(u,v),$$

$$Ncut(A,V) = \frac{cut(A,V)}{assoc(A,V)} + \frac{cut(A,V)}{assoc(V-A,V)},$$

where w(u, v) is a function of similarity between nodes u and v; and $assoc(A,V) = \sum_{u \epsilon A, t \epsilon V} w(u,t)$ is the total connection from nodes in A to all nodes in the graph.

We obtain different binary graphs by choosing different threshold in the range 0.55 and 0.65. Then graph algorihm is run on these graphs seperately and the

normalized cut value for each graph is calculated as defined as above. Among all, the thresholds applied in constructing the graph with minimum Ncut value is selected. The overall weighted recall and precision values are achieved as 74.01 and 68.55 respectively.

## 4.6 Performance Analysis

The performance of our system is mainly based on computing the similarity values since we compare each face with all other faces in the search space. Hence, the time complexity of constructing the similarity matrix is $O(N * N)$, where N is the total number of images in the limited search space of a query name. The time complexity of the greedy graph algorithm is $O(N)$; and it takes constant time to check if a new image belongs to the query person after the result of the graph approach is modeled.

To form an example, the similarity matrix of a search space with 200 pictures is constructed in 9 minutes on a Pentium IV 3 GHz machine with 2 GB memory; and it takes less than 1 second to partition this graph with the densest component algorithm.

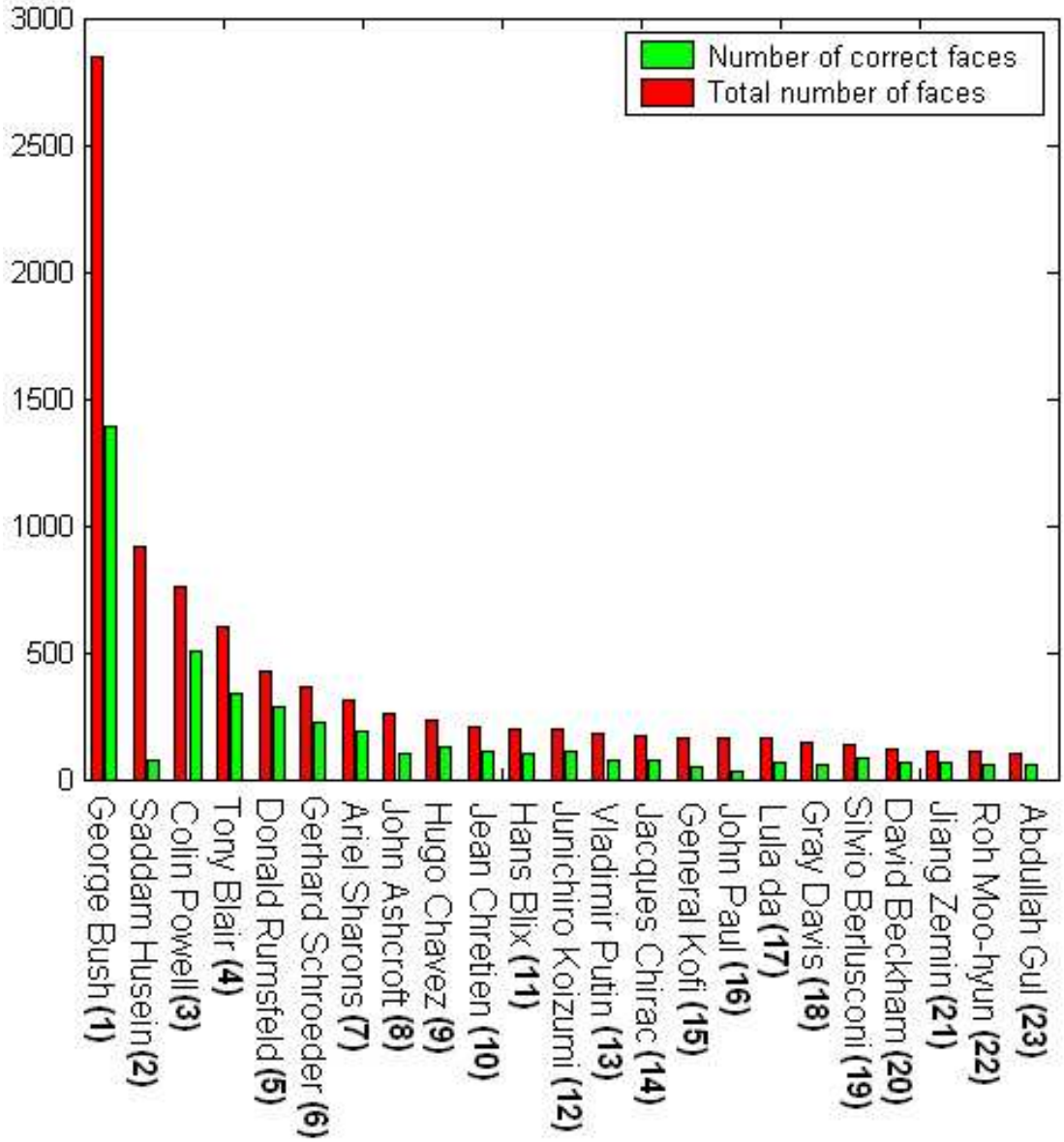


Figure 4.1: Names of 23 people are used in the experiments. The total number of faces associated with a name is represented by red bars and number of correct faces by green bars.

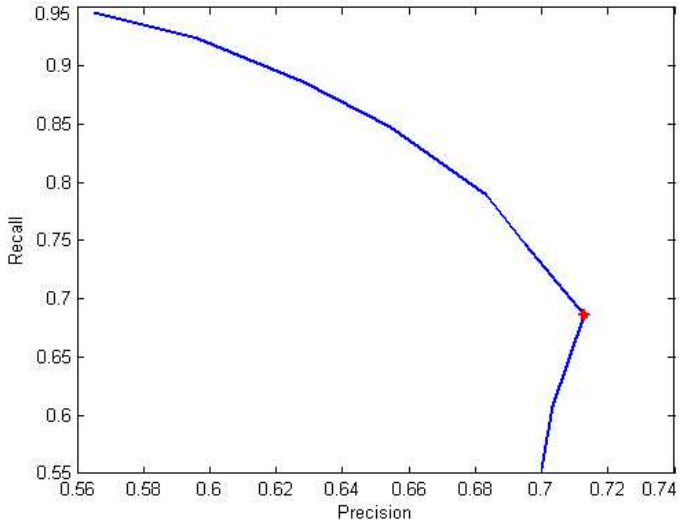


Figure 4.2: Recall-precision curve of 23 people in the news photographs data set. Precision and recall values change depending on the threshold. We used threshold values between 0.55 and 0.65 to show the effect. The threshold used in the rest of our experiments is marked with red.

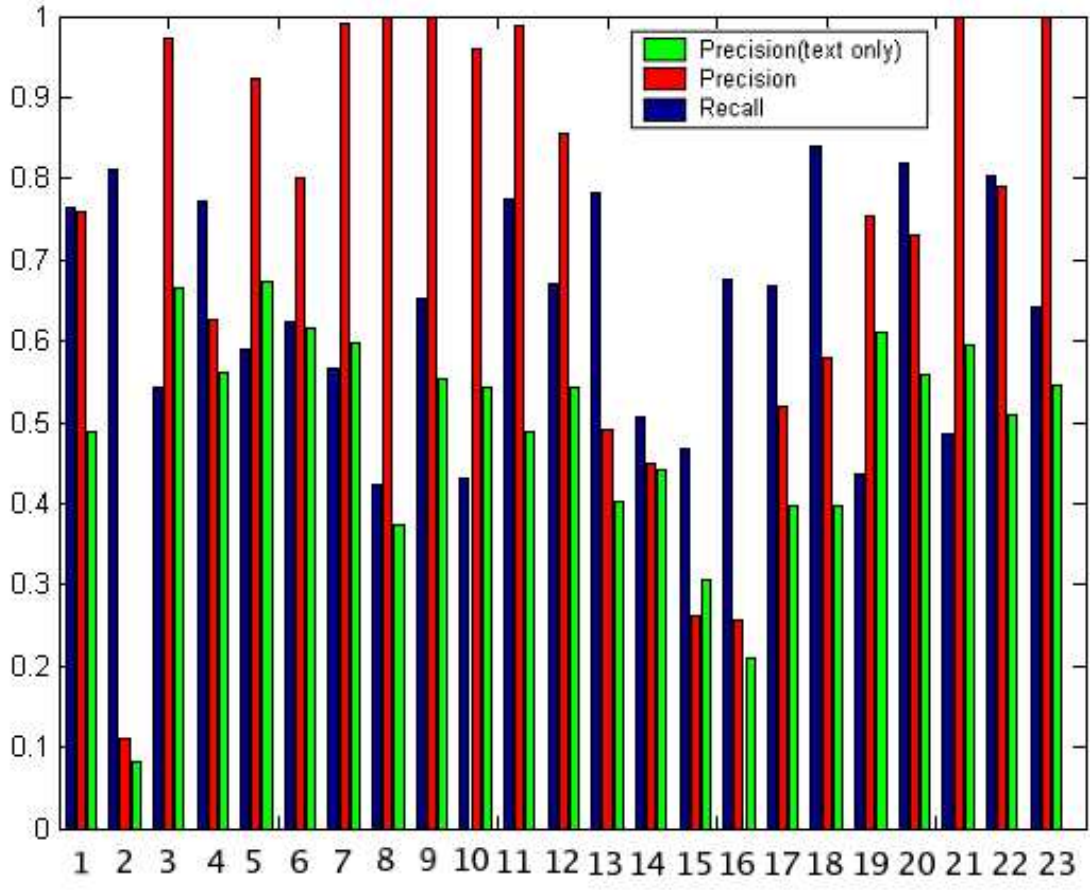


Figure 4.3: Recall and precision values for 23 people for graph threshold value of 0.575. Blue bars represent recall and red bars represent precision values that are achieved with the proposed method. Green bars are precision values for the baseline method, which does not use the visual information and retrieves the faces when name appears in the caption.

Figure 4.4: Sample images retrieved (on the left) and sample images not retrieved (on the right) for the query names: George Bush (top), Colin Powell (middle), Hans Blix(bottom).

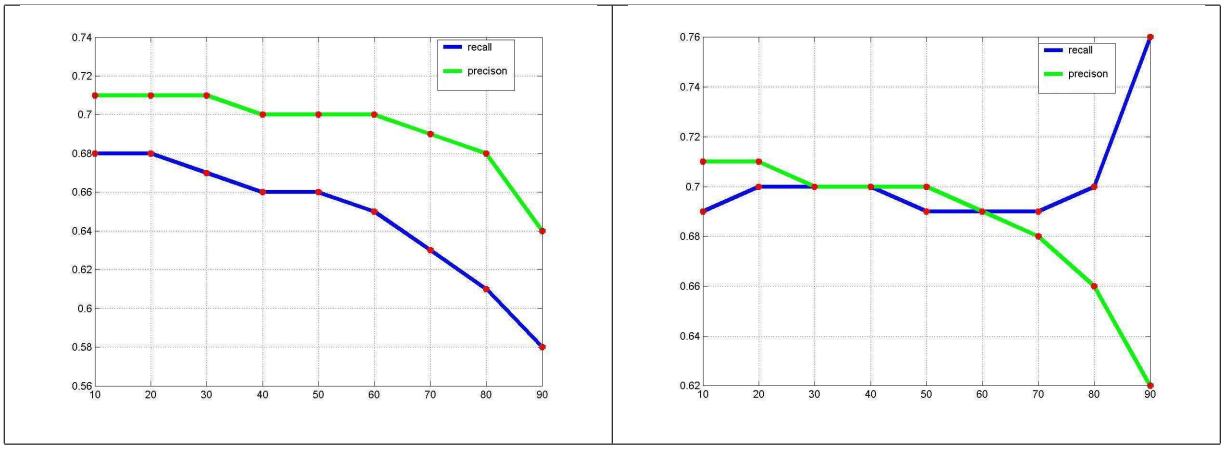


Figure 4.5: Recall and precison values of the held-out set (on the left) and the constructed model (on the right) for online recognition with degree modeling method.

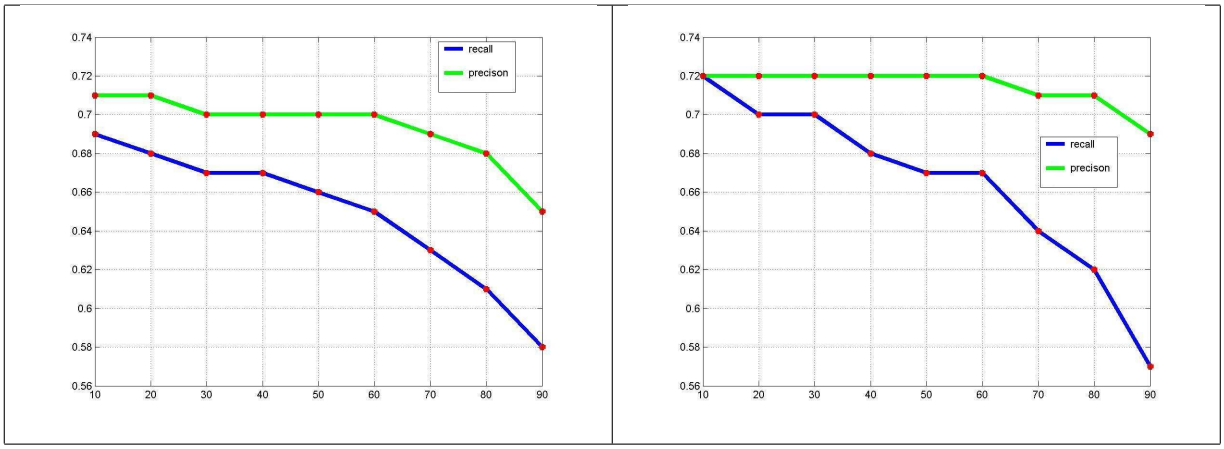


Figure 4.6: Recall and precison values of the held-out set (on the left) and the constructed model (on the right) for online recognition with distance modeling method.

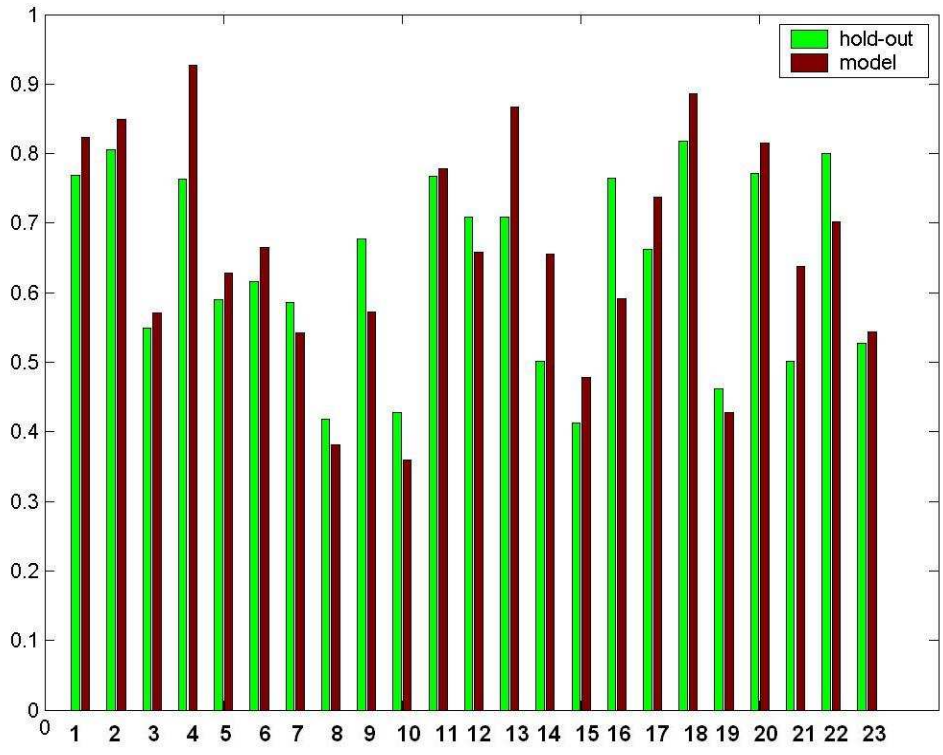


Figure 4.7: Recall values of the held-out set and the constructed model for each person for K = 10.

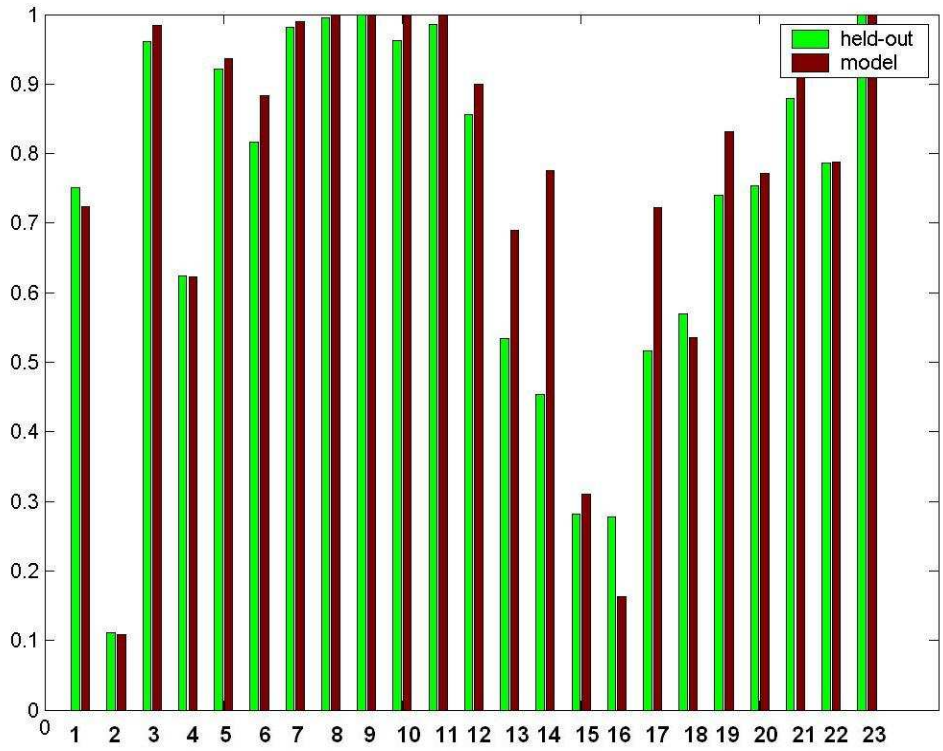


Figure 4.8: Precison values of the held-out set and the constructed model for each person for K = 10.

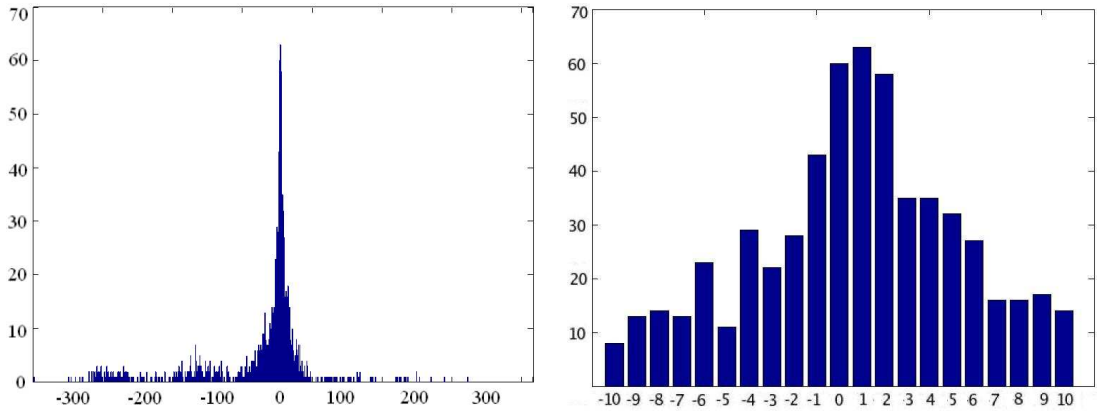


Figure 4.9: The figure shows frequency of Bill Clinton's visual appearance with respect to the distance to the shot in which his name is mentioned. **Left:** when the whole data set is considered, **right:** when the faces appearing around the name within the preceding and the following ten shots are considered. Over the whole data set Clinton has 240 faces and 213 of them appear in the selected range.

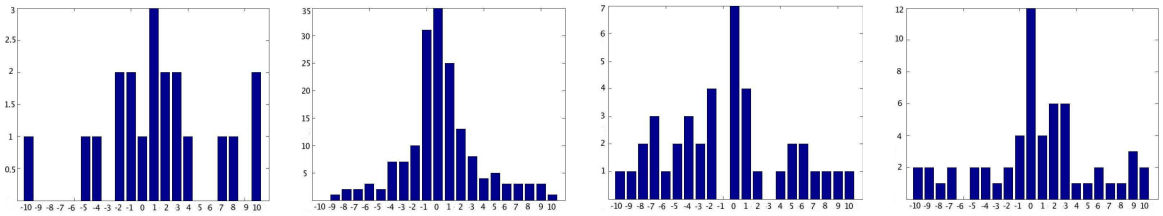

Figure 4.10: The relative position of the faces to the name for Benjamin Netanyahu, Sam Donaldson, Saddam Hussein, and Boris Yeltsin respectively.

Figure 4.11: Detected anchors for 6 different videos.

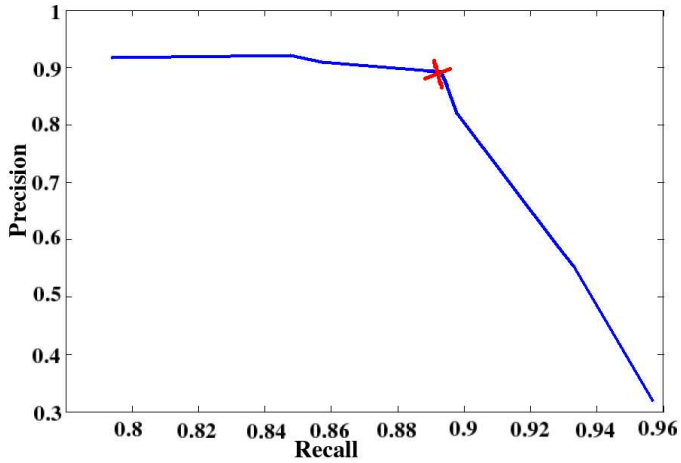


Figure 4.12: Recall-precision values for randomly selected 10 news videos for threshold values varying between 0.55 and 0.65.

Figure 4.13: Sample images retrieved for five person queries in experiments. Each row corresponds to samples for Clinton, Netanyahu, Sam Donaldson, Saddam, Yeltsin queries respectively.

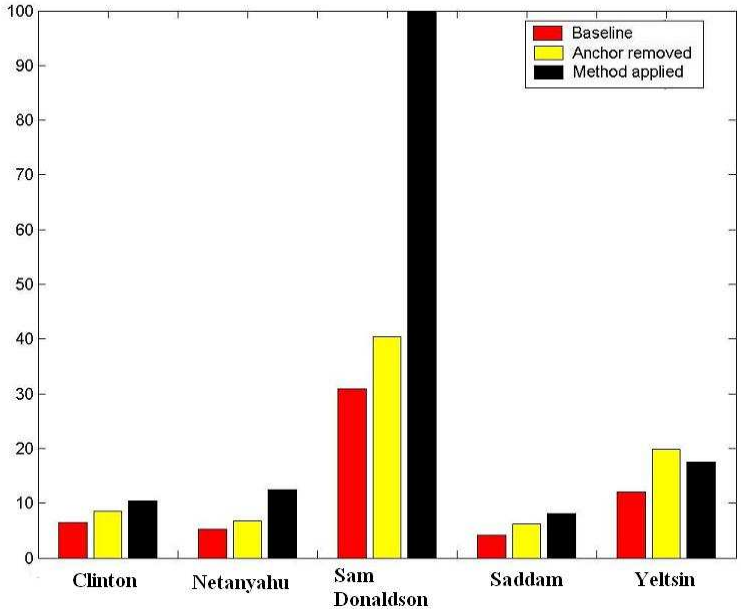


Figure 4.14: Precisions values achieved for five people used in our tests.

# Chapter 5

# Comparison

There are two main issues to be analyzed in the overall person finding approach: definition of the similarity measure in graph construction and the densest component algorithm in partitioning this graph. For comparison reasons, we first analyze these two issues in the following two sections, and then compare the results with the previous approach on the same dataset.

## 5.1 Baseline Method

The principle component analysis (pca) is a well-known method that has also been used in face recognition as eigenfaces [36]. As to compare with a baseline method, we have applied it on the news photographs data, to give an idea of the performance of the traditional face recognition methods. The experiments are conducted on the ground truth faces of the top 23 people used in experiments. For each person, (100-K) percent of the images are selected for training. Remaining K percent of the images are then classified as being one of these people in the train set. The algorithm is run K times with different random groups of images for testing and training. Recognition rate is calculated by diving the truly labeled images by total number of images tested. The average recognition rates of both test and train set for different K are given in Table 5.1 and in 5.1, which shows

off low rates for test images.

Table 5.1: Recognition rates of the eigenface method for for different K values.

| K | 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90 |
|---|---|---|---|---|---|---|---|---|---|
| test | 51.90 | 51.83 | 50.41 | 48.91 | 46.33 | 43.48 | 39.64 | 34.65 | 28.13 |
| train | 98.56 | 98.57 | 98.58 | 98.74 | 98.80 | 98.89 | 99.12 | 99.30 | 99.45 |

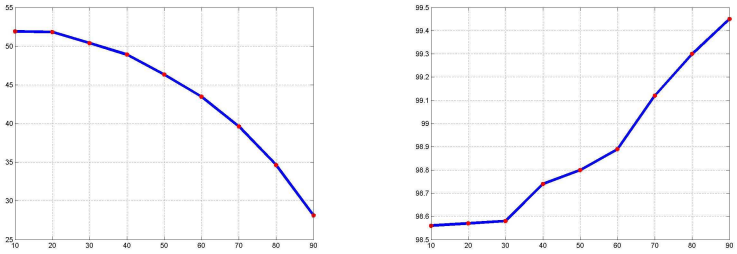

Figure 5.1: Recognition rates of the eigenface method for different K values.

## 5.2 Feature Selection and Similarity Matrix Construction

### 5.2.1 Finding True Matching Points

As stated earlier, we have not used the original matching metric of SIFT, since it does not work well for faces. (See Figure 5.2 for sample matches found by using the original matching metric and the proposed method are shown.) To see how well the original metric performs, we have constructed the similarity graph with using the matches of the original metric and then applied the densest component algorithm. A sample similarity matrix can be seen in Figure 5.3. The performance results for different graph thresholds are plotted in Figure 5.4. The recall and precision values for the same threshold in our original tests (0.575) are recorded as 0.91 and 0.59 respectively.

In a second experiment, we have applied the Ransac algorithm [20] to find the

homography matrix and assign true matches among all. Results of this techniques are given in Figure 5.5. As it can be perceived from the results, affine constraints does not work well due to deformability property of the faces.

## 5.3 Extracting Similar Group of Faces

We compare the greedy graph algorithm for finding the densest component with one well-known approach: k-nearest neighbor and the one-class classification methods in the following two subsections. All the experiments are conducted on news photographs dataset, where we had achieved 68% recall and 71% precision values on the average.

### 5.3.1 k-nn Approach

In the last experiments a method similar to the k-nearest neighbors classification has been used. For each face in the test set, we find the distances of that face to all the faces in the training set, and select the nearest k faces (k-neighbors). If number of true faces are greater than the number of false ones in this k-neighbors, then the test face is classified as a true face. The tests are conducted with different number of training and testing sets. The results are given in Table 5.2 and in Figure 5.6 for different K, where K indicates percentage of the images are used for testing. The results show that the greedy densest component algorithm outperforms the k-nn approach.

### 5.3.2 One-class Classification

Given a set of data items, one-class classification methods aim to find a target class against the outliers( **??**). In other words, given a test sample, it is either accepted as belonging to the target class or rejected. Hence, one-class classification approach differs from any other traditional multi or two-class classification

Table 5.2: Recognition rates of supervised method for different K values. (K is the percentage of the images that are used in testing; k is the number of neighbours used.

| K | 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90 |
|---|---|---|---|---|---|---|---|---|---|
| k(neighbours) = 11 | | | | | | | | | |
| recall | 0.53 | 0.52 | 0.51 | 0.49 | 0.47 | 0.45 | 0.41 | 0.36 | 0.33 |
| precision | 0.36 | 0.36 | 0.35 | 0.34 | 0.34 | 0.33 | 0.32 | 0.31 | 0.31 |
| k(neighbours) = 5 | | | | | | | | | |
| recall | 0.61 | 0.61 | 0.60 | 0.58 | 0.56 | 0.54 | 0.51 | 0.47 | 0.40 |
| precision | 0.38 | 0.38 | 0.38 | 0.38 | 0.37 | 0.37 | 0.36 | 0.35 | 0.34 |
| k(neighbours) = 3 | | | | | | | | | |
| recall | 0.68 | 0.67 | 0.65 | 0.64 | 0.62 | 0.59 | 0.56 | 0.52 | 0.45 |
| precision | 0.41 | 0.40 | 0.40 | 0.40 | 0.39 | 0.39 | 0.38 | 0.37 | 0.36 |

approaches by holding only the information of the target class. With the greedy densest component algorithm [15], we also seek to find only the nodes belonging to the densest component (hence the faces of the query person) and assume all others are outliers. In this context, the most similar problem to the problem of finding the densest component of the graph is one-class classification problem. To this end, we compare the one-class classification methods in **??** with the graph algorithm used in this study.

The similarity graph constructed as described in **??** keeps only the distances among faces. Hence, the one-class classification methods cannot be applied to this graph. So to compare the greedy graph densest component method with any of those methods, we used the Bag-Of-Features approach as in [46, 33] for graph construction. Then, we applied both the greedy graph method and two of the one-class classification methods (nearest neighbor data description method and k-nearest neighbor data description method) that gave us the best results among all.

To construct the graph, we first extracted sift features from each face image and clustered these features using k-means clustering into 50 clusters. Then, a histogram of the size of number of clusters (50) is formed for each image showing

the distribution of clusters. In Information Retrieval, the frequencies of the clusters are weighted by 'term frequency inverse document frequency (tf-idf)' which is computed as:

$$tfidf = \frac{n_{id}}{n_d} \log \frac{N}{n_i} \tag{5.1}$$

where, $n_{id}$ is number of occurrences of term $i$ in document $d$, $n_d$ is total number of terms in document $d$, $N$ is the total number of documents in database and $n_i$ is the number of documents in database containing term $i$.

Adapting the same approach, we find the weighted frequencies of the clusters and use them as the final feature vector for each image. The one-class classification methods can then be applied on these features. To apply the densest component graph algorithm, the similarity among the faces are found by normalized scalar product of tfidf vectors by using the following equation:

$$distance(f_1, f_2) = \frac{tfidf(f_1) * tfidf(f_2)}{norm(f_1) * norm(f_2)} \tag{5.2}$$

where, $tfidf(f_1)$ is tf-idf vector of face image $f_1$ and $norm(f_1)$ is the norm of tfidf vector of face image $f_1$. The similarity matrix constructed as described above for the query name *Hans Blix* is shown in Figure 5.7. The precision-recall curve of the densest component graph algorithm for varying graph thresholds is given in Figure 5.8. Recall and precision values of one-class-classification methods for different K-fold validation tests are given in Table 5.3. The results indicate that the once-class classification methods is not superior to the greedy densest component algorithm used in our experiments for finding the most similar set of faces.

Table 5.3: Recall-precision rates of two one class classification methods: w1 (nearest neighbor data description method) and w2 (k-nearest neighbor data description method) (applied on tfidf's).

| training set | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | K = 10 | | K = 20 | | K = 30 | | K = 40 | | K = 50 | |
| | rec | pre | rec | pre | rec | pre | rec | pre | rec | pre |
| w1 | 1.00 | 0.54 | 1.00 | 0.53 | 1.00 | 0.52 | 1.00 | 0.52 | 1.00 | 0.51 |
| w2 | 0.90 | 0.57 | 0.90 | 0.55 | 0.90 | 0.54 | 0.90 | 0.54 | 0.90 | 0.53 |
| test set | | | | | | | | | | |
| | K = 10 | | K = 20 | | K = 30 | | K = 40 | | K = 50 | |
| | rec | pre | rec | pre | rec | pre | rec | pre | rec | pre |
| w1 | .90 | 0.50 | 0.91 | 0.50 | 0.90 | 0.50 | 0.90 | 0.50 | 0.90 | 0.49 |
| w2 | 0.84 | 0.53 | 0.88 | 0.54 | 0.87 | 0.53 | 0.86 | 0.53 | 0.86 | 0.52 |

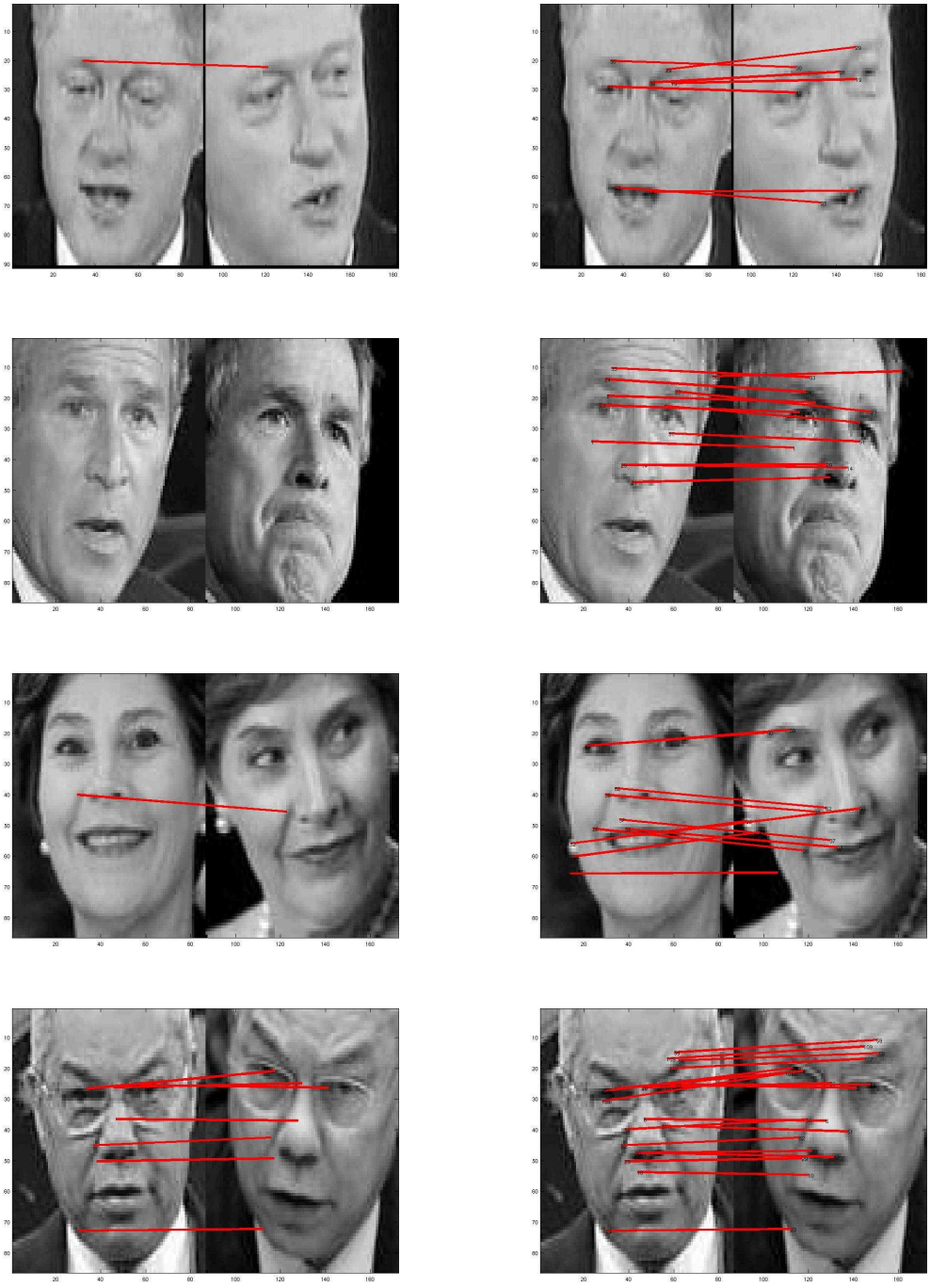

Figure 5.2: Examples for matching points. First column for matches found by using the original matching metric of sift. Second column for matches found by applying the proposed method.

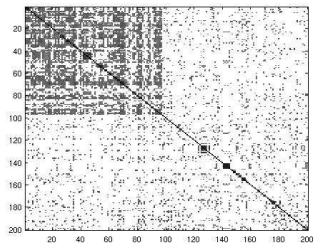


Figure 5.3: Similarity matrix for 201 images in the search space for the name *Hans Blix*. In this search space, 98 of the images are true *Hans Blix* images, and the remaining 103 are not. For visualization, the 98 true *Blix* images are put on the top left of the matrix. Dark colors correspond to larger similarity values.

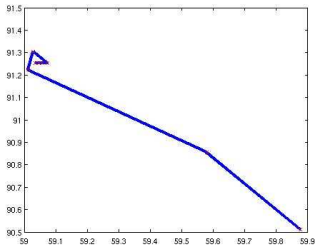


Figure 5.4: Recall-precision curve of 23 people in the test set. Precision and recall values change depending on the threshold.

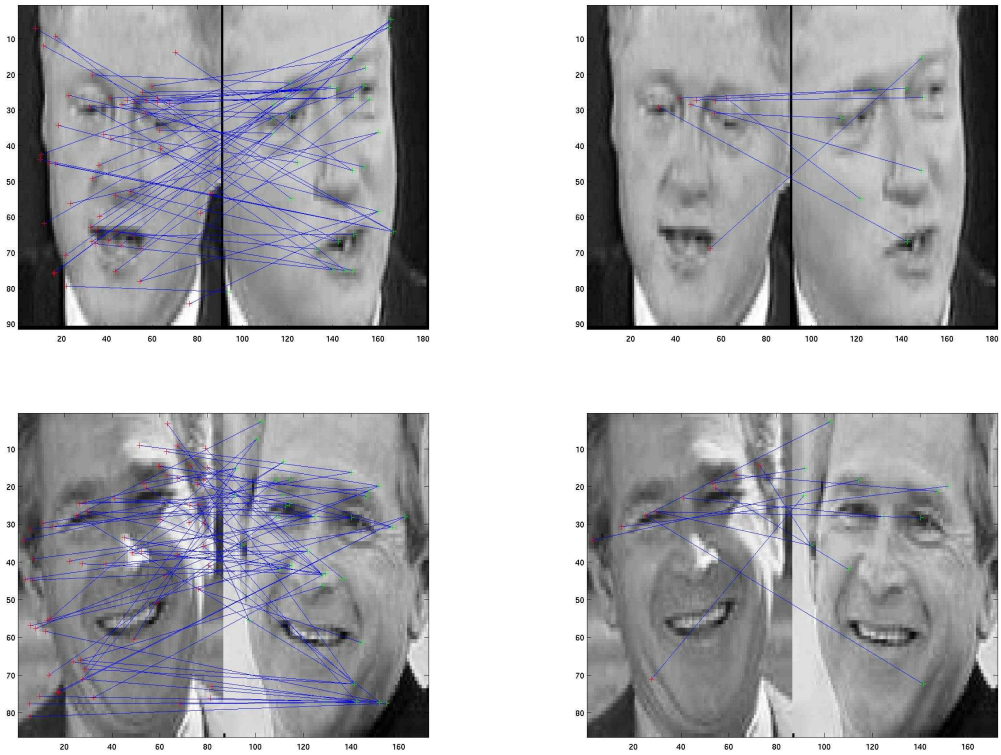

Figure 5.5: Examples for matching points assigned by the homography matrix found after ransac.

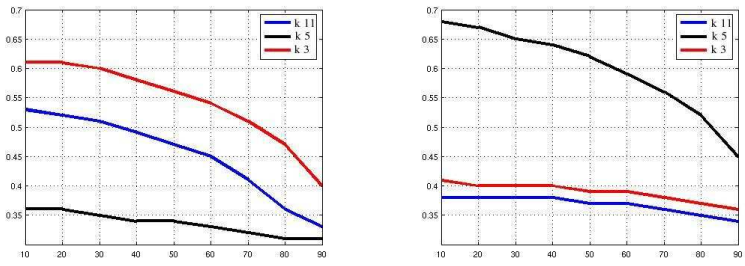


Figure 5.6: Recognition rates of the k-nn approach for different K and k values. Recall values on the left and precision on the right. K is the percentage of the images used for testing, and k is the number of neighbors in k-nn.

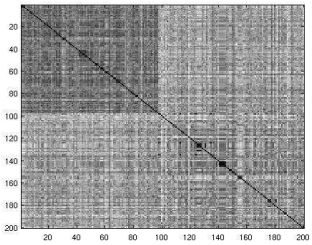


Figure 5.7: Similarity matrix for 201 images in the search space for the name *Hans Blix*. In this search space, 98 of the images are true *Hans Blix* images, and the remaining 103 are not. For visualization, the 98 true *Blix* images are put on the top left of the matrix. Dark colors correspond to larger similarity values.

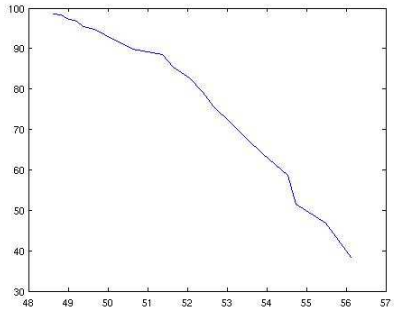


Figure 5.8: Recall-precision curve of 23 people in the test set. Precision and recall values change depending on the threshold.

# Chapter 6

# Conclusions and Future Work

## 6.1 Conclusions

In this paper, we propose a graph based method for querying people in large news photograph and video collections with associated captions or speech transcript texts. Given similarity measures between the face images in a data set, the problem is transformed into a graph problem in which we seek the largest densest component of the graph corresponding to the largest group of similar faces. We use SIFT descriptors [35] to represent each face image and define the similarity values by using the average distances of the matching interest points. Then, we apply a greedy graph algorithm [15] to find the densest component of the graph corresponding to the faces of the query person. In the paper, we also propose two different methods to use the results of the graph based person finding approach in futher recognition of new faces.

For large realistic data sets, face recognition and retrieval is still a difficult and an error-prone problem due to large variations in pose, illumination and expressions. In this study, we have described a multi-modal approach for querying large numbers of people in such data sets. The method does not require training for any specific person and thus it can be applied to any number of people. With this property, it is superior to any supervised method which requires labeling of

large number of samples. The results achieved are also very close to supervised methods.

The experiments are conducted on two different news data sets. The first set consists of thousands of news photographs with associated captions collected from Yahoo! news. The captions are used to limit the number of images for a query name and only the images associated with the name are selected. In this data, over 20% increase in precision is achieved compared to solely text-based methods. For individuals, up to 84% recall and 100% precision values can be obtained.

The second experiments are conducted on 229 broadcast news videos archive. 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 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 anchorperson removal, and by 152% after applying the proposed algorithm. The person finding algorithm also performs well for anchorperson detection without requiring any supervision.

The proposed method is an overall scheme to find and recognize faces in large news photograph and video collections. Each step of the method can be commited with another tecniques, for instance similarities between faces can be assigned with a different approach or the densest component can be extracted with another graph partitioning algorithm. However, experiments show that our similarity definition works better than other traditional approaches for this dataset; and the greedy graph algorithm is comparable to one other the most possible approach, namely one-class classification.

One of the important remarks to be made on the method is that even if it is not a face recognition scheme on the whole, it is instrumental in reducing the number of images presented to the user by improving the retrieval performance

of baseline methods.

## 6.2 Future Work

Before applying the greedy densest component algorithm, we convert our weighed graph consisting of dissimilarity values into a binary graph. However, this ignores some of the information. A method, which does not violate the weighted property of the graph, may yield better results. In this study, SIFT descriptors are used to represent the similarity of the faces. Other representations or similarity measures can also be used to construct the graph structure.

In [28] sets of face exemplars for each person are gathered automatically in shots for tracking in video. 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.

Since the proposed approach is an overall scheme, it can also be applied to other problems such as object recognition or image region annotation. For instance, in the context of region annotation, annotations of images can be used for limiting the search space for a region. Similary, in this limited space, the region of interest is expected to form the largest similar group of regions.

# Bibliography


[1] Trec video retrieval evaluation
http://www-nlpir.nist.gov/projects/trecvid/, 2004.

[2] J. Sivic . Everingham and A. Zisserman. Hello! my name is... buffy – automatic naming of characters in tv video. In *Proceedings of the British Machine Vision Conference*, 2006.

[3] S. Aksoy and R. M. Haralick. Graph-theoretic clustering for image grouping and retrieval. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 63–69, 1999.

[4] K. Barnard, P. Duygulu, N. de Freitas, D. A. Forsyth, D. Blei, and M. Jordan. Matching words and pictures. *Journal of Machine Learning Research*, 3, 2003.

[5] M. Bartlett, H. Lades, and T. Sejnowski. Independent component representations for face recognition, 1998.

[6] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts, 2001.

[7] S. Beretti, A. Del Bimbo, and E. Vicario. Efficient matching and indexing of graph models in content-based retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10):1089–1105, 2001.

[8] T. Berg, A. C. Berg, J. Edwards, and D.A. Forsyth. Who's in the picture. In *Neural Information Processing Systems (NIPS)*, 2004.

[9] T. Berg, A. C. Berg, J. Edwards, M. Maire, R. White, Y.-W. Teh, E. Learned-Miller, and D.A. Forsyth. Names and faces in the news. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2004.

[10] M. Bicego, A. Lagorio, E. Grosso, and M. Tistarelli. On the use of sift features for face authentication. In *CVPRW '06: Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop*, page 35, Washington, DC, USA, 2006. IEEE Computer Society.

[11] V. Bruce. Recognizing faces, 1988. Lawrence Erlbaum Associates, London, U.K.

[12] J.M. Fellous C. von der Malsburg, N. Kruger and L. Wiskott. Face recognition by elastic bunch graph matching. In *Computer Analysis of Images and Patterns 1997*, pages 456–463, 1997.

[13] C. Carson, S. Belongie, H. Greenspan, and Jitendra Malik. Blobworld: Image segmentation using expectation-maximization and its application to image querying. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(8), 2002.

[14] M. La Cascia, S. Sethi, and S. Sclaroff. Combining textual and visual cues for content-based image retrieval on the world wide web. In *Proc. IEEE Workshop on Content-Based Access of Image and Video Libraries*, Santa Barbara CA USA, June 1998.

[15] M. Charikar. 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] F. Chen, U. Gargi, L. Niles, and H. Schuetze. Multi-modal browsing of images in web documents. In *Proceedings of SPIE Document Recognition and Retrieval VI*, 1999.

[17] M-Y. Chen and A. Hauptmann. Searching for a specific person in broadcast news video. In *International Conference on Acoustics, Speech, and Signal Processing (ICASSP'04)*, Montreal, Canada, May 17-21 2004.

[18] S. Dickinson, M. Pelillo, and R. Zabih. Introduction to the special section on graph algorithms in computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10):1049–1052, 2001.

[19] P. Duygulu and A. Hauptmann. What's news, what's not? associating news videos with words. In *The 3rd International Conference on Image and Video Retrieval (CIVR) Ireland*, July 21-23, 2004.

[20] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981.

[21] J.L. Gauvain, L. Lamel, and G. Adda. The limsi broadcast news transcription system. *Speech Communication*, 37(1-2), 2002.

[22] Y. Gdalyahu, D. Weinshall, and M. Werman. Self-organization in vision: Stochastic clustering for image segmentation, perceptual grouping, and image database organization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10):1053–1074, 2001.

[23] R. Gross, S. Baker, I. Matthews, and T. Kanade. Face recognition across pose and illumination. In Stan Z. Li and Anil K. Jain, editors, *Handbook of Face Recognition*. Springer Verlag, 2004.

[24] R. Gross, J. Shi, and J. Cohn. Quo vadis face recognition? In *Third Workshop on Empirical Evaluation Methods in Computer Vision*, 2001.

[25] G. Hamerly and C. Elkan. Learning the $k$ in $k$-means. In *Advances in Neural Information Processing Systems*, volume 17, 2003.

[26] S. Helmer and D.G. Lowe. Object recognition with many local features. In *Workshop on Generative Model Based Vision 2004(GMBV)*, Washington D.C., 2004.

[27] N. Ikizler and P. Duygulu. Person search made easy. In *The Fourth International Conference on Image and Video Retrieval (CIVR)*, Singapore, 2005.

[28] M. Everingham J. Sivic and A. Zisserman. Person spotting: video shot retrieval for face sets. In *International Conference on Image and Video Retrieval (CIVR), Singapore*, 2005.

[29] G. M. Davies J. W. Shepherd and H. D. Ellis. Studies of cue saliency, 1981. In Perceiving and Remembering Faces, G. M. Davies, H. D. Ellis, and J. W. Shepherd, Eds. Academic Press, London, U.K.

[30] I. Jermyn and H. Ishikawa. Globally optimal regions and boundaries as minimum ratio weight cycles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10):1075–1088, 2001.

[31] Y. Kaya and K. Kohayashi. A basic study on human face recognition, 1972. Frontiers of Pattern Recognition, S. Watanabe, ed., p. 265.

[32] Michael David Kelly. *Visual identification of people by computer*. PhD thesis, 1971.

[33] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, 2006.

[34] H. Le and H. Li. Recognizing frontal face images using hidden markov models with one training image per person. In *ICPR '04: Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04) Volume 1*, pages 318–321, Washington, DC, USA, 2004. IEEE Computer Society.

[35] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2), 2004.

[36] A. Pentland M. Turk. Eigenfaces for recognition. *Journal of Cognitive Neurosicence*, 3(1):71–86, 1991.

[37] A.P. Pentland M.A. Turk. Face recognition using eigenfaces. In *IEEE Conference on Computer Vision and Pattern Recognition*, 1991.

[38] B. S. Manjunath, R. Chellappa, and C. von der Malsburg. A feature based approach to face recognition. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 1992.

[39] K. Mikolajczk and C. Schmid. A performance evaluation of local descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2003.

[40] K. Mikolajczyk. Face detector. INRIA Rhone-Alpes, 2004. Ph.D Report.

[41] K. Mikolajczyk and C. Schmid. Indexing based on scale invariant interest points. In *International Conference on Computer Vision (ICCV)*, pages I: 525–531, 2001.

[42] B. Park. Face recognition using face-arg matching. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(12):1982–1988, 2005. Member-Kyoung-Mu Lee and Member-Sang-Uk Lee.

[43] P.E. Hart R.O. Duda and D.G. Stork. In *Pattern classification*. John Wiley and Sons, 2001.

[44] S. Satoh and T. Kanade. Name-it: Association of face and name in video. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, 1997.

[45] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[46] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *ICCV '03: Proceedings of the Ninth IEEE International Conference on Computer Vision*, page 1470, Washington, DC, USA, 2003. IEEE Computer Society.

[47] C.G.M. Snoek and M. Worring. Multimodal video indexing: A review of the state-of-the-art. *Multimedia Tools and Applications*, 25(1), 2005.

[48] X. Tan, S. Chen, Z. Zhou, and F. Zhang. Face recognition from a single image per person: A survey. *Pattern Recognition.*, 39(9):1725–1745, 2006.

[49] R. Chellappa W. Zhao. Image-based face recognition: Issues and methods. *Image Recognition and Classification*, pages 375–402, 2002.

[50] J. Wang, K. N. Plataniotis, and A. N. Venetsanopoulos. Selecting discriminant eigenfaces for face recognition. *Pattern Recognition Letters*, 26(10):1470–1482, 2005.

[51] A. Webb. In *Statistical Pattern Recognition*. John Wiley and Sons, 2002.

[52] J. Yang, M-Y. Chen, and A. Hauptmann. Finding person x: Correlating names with visual appearances. In *International Conference on Image and Video Retrieval (CIVR)*, Dublin City University Ireland, 2004.

[53] J. Yang, D. Zhang, A. F. Frangi, and J. Yang. Two-dimensional pca: A new approach to appearance-based face representation and recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(1):131–137, 2004.

[54] W. Zhao, R. Chellappa, P.J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM Computing Surveys*, 35(4):399–458, 2003.

# Appendix A

# Different Forms of Names in News Photographs

| | |
|---|---|
| George Bush | George W (1485), W. Bush (1462), George W. Bush (1454),<br>President George W (1443), President Bush (905),<br>U.S. President (722), President George Bush (44),<br>President Bushs (2), President George W Bush (2),<br>George W Bush (2) |
| Saddam Hussein | Saddam Hussein (911), President Saddam (356),<br>President Saddam Hussein (351), President Saddam Hussien (3),<br>President Saddam Hussiein (1), Minister Saddam (1) |
| Colin Powell | Colin Powell (618), Secretary of State Colin Powell (606)<br>Secretary Colin Powell (4), Collin Powell (3),<br>Secretary General Colin Powell (2), Secretary Powell (1),<br>Secretary of State Powell (1) |
| Tony Blair | Tony Blair (502), Prime Minister Tony Blair (472),<br>Premier Tony Blair (2), Prime Minister Blair (2), Mr Blair (1),<br>Prime Minister Tony Bair (1) |
| Jean Chretien | Prime Minister Jean (158), Jean Chretien (155),<br>Minister Jean Chretien (145), Prime Minister Jean Chretien (145),<br>Prime Minister John Chretien (2), Prime Minister Chretien (2) |
| Gerhard Schroeder | Gerhard Schroeder (311), Chancellor Gerhard Schroeder (283)<br>Chancellor Schroeder (1), Chancellor Gerhard Schroeders (2)<br>Chancellor Gerhard Schroder (1), Chancellor Gerhard Schoeder (1) |

| | |
|---|---|
| John Ashcroft | John Ashcroft (147), General John Ashcroft (146)<br>Attorney General John Ashcroft (143), U.S. Attorney (106)<br>U. S. Attorney (2), U.S Attorney (1) |
| Donald Rumsfeld | Donald Rumsfeld (279), Donald H (47),<br>Secretary Donald Rumsfeld (84), Donald H. Rumsfeld (44),<br>Secretary Donald H (26), H. Donald (13),<br>Secretary of State Donald Rumsfeld (4), Secretary Rumsfeld (6) |
| Ariel Sharons | Minister Ariel (249), Prime Minister Ariel (248)<br>Prime Minister Ariel Sharons (2) |
| Junichiro Koizumi | Junichiro Koizumi (156), Prime Minister Junichiro (151)<br>Prime Minister Junichiro Koizumi (149), Prime Minister Koizumi (1) |
| Hugo Chavez | Hugo Chavez (194), President Hugo Chavez (186),<br>President Hugo Chaves (1), President Chavez (3) |
| General Kofi | General Kofi (124), Secretary General Kofi (61)<br>Secretary-General Kofi (60), General General Kofi (1)<br>Secretary-Genaral Kofi (1), Annan , U.N. (1), U.N. Secretary (57)<br>U.N. Secretary- (39), U.N. General (9) |
| Roh Moo-hyun | Roh Moo-hyun (86), Roh Moo- (93), President Roh Moo-hyun (55)<br>President-elect Roh Moo-hyun (10), President Roh (61),<br>President Roh Moo- (61), President-Elect Roh Moo (1) |
| Lula da | Lula da (119), President Luiz Inacio Lula (30),<br>President-elect Luiz Inacio Lula (19), President Lula (7),<br>President Lula Da (5), President-elect Lula Da (2)<br>President Luiz Lula Da (1), President Luis Inacio Lula (1)<br>President-elect Luis Inacio Lula (1), Luiz Inacio (105), Luis Inacio (4) |
| Jacques Chirac | Jacques Chirac (143), President Jacques Chirac (138)<br>President Chirac (4), President Jaques Chirac (3) |
| Vladimir Putin | Vladimir Putin (146), President Putin (4)<br>President Vladimir Putin (136) |
| Abdullah Gul | Abdullah Gul (84), Minister Abdullah Gul (57)<br>Prime Minister Abdullah Gul (47), Premier Abdullah Gul (9)<br>Minister Abdullah (74) |
| Jiang Zemin | Jiang Zemin (94), President Jiang (85),<br>President Jiang Zemin (85), General Secretary Jiang Zemin (2) |
| John Paul | John Paul (135), John Paul II (57), John Paul II (42) |
| Silvio Berlusconi | Silvio Berlusconi (113), Prime Minister Silvio Berlusconi (81)<br>Premier Silvio Berlusconi (22), Prime Minister Sivlio Berlusconi (1) |
| David Beckham | David Beck (94), David Beckham (93),<br>captain David Beckham (17), captain Beckham (5) |
| Gray Davis | Gray Davis (109), Gov. Gray Davis (73), Governor Gray Davis (26) |
| Hans Blix | Hans Blix (168), Inspector Hans Blix (17), Dr. Hans Blix (13),<br>U.N. Hans Blix (3) |