WWW.WAWO.NET: A CBIR FOR FACIAL SIMILARITIES

A. Georgakis, H. -S. Le, J. -P. Kouma, R. Cabral and H. Li

Digital Media Laboratory (DML),
Department of Applied Physics and Electronics,
Umeå University, SE-90187 Umeå Sweden
medialab@tfe.umu.se

ABSTRACT

This paper demonstrates an experimental content based facial image retrieval system. The proposed system can be queried by a user using a persons photo; either a frontal or near frontal facial image. The proposed system utilizes state-of-the-art face recognition technology which exploits the inherent advantages of single HMM scheme based on 1D discrete HMM.

Index Terms— Face detection, focused search engine, face similarity.

1. INTRODUCTION

This paper presents a content based image retrieval system that specializes on images depicting persons. The system under consideration has been under development in the Digital Media Lab in Umeå University over the past five years and has now reached the general availability release stage [1]. It relies on human facial characteristics and it uses a database of non-annotated images, where some humans may or may not appear. Given the above, the proposed system can search for a particular person when the initial query is another image (content based searching). The search engine is publicly accessible though the following URL: `www.wawo.net`.

2. RELATED WORK

Currently, there are some systems available the exhibit similar characteristics to our system. These systems either have the same aim or similar objectives to `www.wawo.net`:

- Riya.com [2]
- Online Face Alignment Demo [3]
- Betaface [4]
- Pittsburg Pattern Recognition [5]
- Polar Rose [6]
- Cydral [7]

All the above systems are currently publicly available. There is also a plethora of stand-alone implementations in the form of applications or DLLs that can be accessed through the usage of an appropriate API.

3. SYSTEM DESCRIPTION

The system under consideration is depicted in Fig. 1. From the figure we can see that the system has a distributed architecture. In this setting two servers is the minimum amount of machines needed to operate the CBIR system. These servers provide the following functionalities:

- HTTP and MySQL server
- Image similarity module (based on Matlab)

We should note here that the first server can also be substituted by two independent servers, a web server and an SQL server.

The distributed nature of the system can be fully utilized with the usage of additional image similarity modules. In this case the web server can be used for load balancing. When a new user query is received the Apache server looks at the workload of each of the Matlab clients and assigns the query to the first free server. The distributed architecture allows us to scale the system by adding an infinite number of Matlab and Apache servers. This solution is highly efficient and this is the reason why it is being used by content providers like YouTube, search engines like Google and distributed computing platforms like Akamai.

3.1. Face Detection

Face detection is an integral element for the proposed search engine. Its goal is to locate regions in images that contain, with high probability, faces [8]. For such task, a statistical model (classifier) is used to detect faces [9]. Statistical modeling employs multiple instances of facial images ("positive samples) and "negative samples, i.e., images that do not

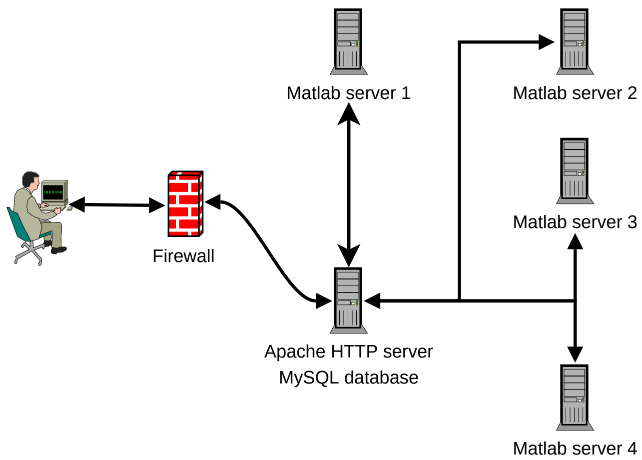

Fig. 1. Architecture of the proposed system.

contain faces (*training set*). During training, features are extracted from the training set using Haar-like features and a cascade of boosted tree classifiers [10].

The classifier is trained on non-annotated images of fixed size. Detection is performed by sliding a search window over the image and checking whether an image region resembles a face or not. A cascade of simple "weak classifiers uses single features to classify the image region as face or non-face. Each feature is described by its template, its coordinate relative to the search window origin and the size (scale factor) of the feature [9, 10].

Figure 2 depicts an face detection operation applied on a query image. The figure depicts the image of `Anitra Steen` to the left with a set of blue markers over her face. These markers correspond to two different facial regions. These regions correspond to automatic and manual face detection. In the right side of the figure in question we can see the detected facial regions (`Objects in this image`).

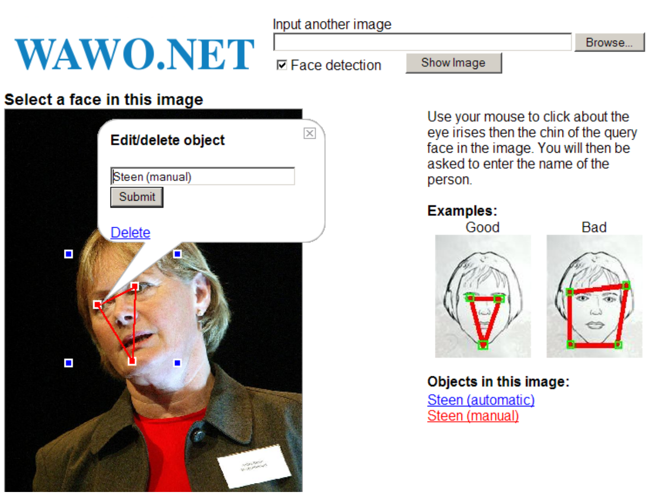

Fig. 2. Automatic and manual face detection.

It must be noted here that manual face marking is not necessary but since automatic detection is not always accurate user input can always elevate the performance of the system. Manual marking can also be seen in Fig. 2. In this case the face is marked with three marker point that the user must place on the person's eyes and chin.

After the detection phase (automatic or manual) the user can proceed with the face recognition. In doing so she must select the face of preference to be submitted to the system and press the `Submit` button (seen in Fig. 2). This action leads to the generation of the result page that is depicted in Fig. 3.

3.2. Face Recognition

Face recognition is based on HMM that have the advantage to be able to learn from as many samples as possible. The system needs no information about the pose of the face in the images under consideration. However the computational cost of the system depends linearly on the number of images in the database. Fortunately results reported in [11] showed that the system can function equally well with a limited number of small pose changes between the input and the query images. The proposed system along with the training of the HMM and recognition procedures are briefly described in the following subsections.

3.2.1. HMM Modeling and Training

An HMM is usually defined as a triplet $\lambda = (\pi, A, B)$. The model used is an ergodic 1D-DHMM with randomly initialized parameter matrices. Due to the discrete nature, the system needs some retraining whenever it is provided new image(s). That means that the system depends on the number of persons as well as the number of sample images per person. To solve this problem a system has been developed using only one HMM, which is used not to model any specific person, but to capture the relationship of observation blocks. In other words the system does not depend on the number of observations nor on the number of HMMs.

In modeling, the sample images are transformed, preprocessed and decomposed to observation vectors to form a codebook as follows:

- Decomposed into overlapping vertical strips by horizontally scanning from left to right. Each vertical strip is assigned an ordering number. The amount of vertical overlap between consecutive strips is permitted up to one pixel less than the strip width.
- Each strip is decomposed into overlapping blocks. The amount of overlap between consecutive blocks is permitted up to one pixel less than the block height.
- The extracted blocks are normalized
- The normalized feature blocks are arranged columnwise to form the observation vectors.

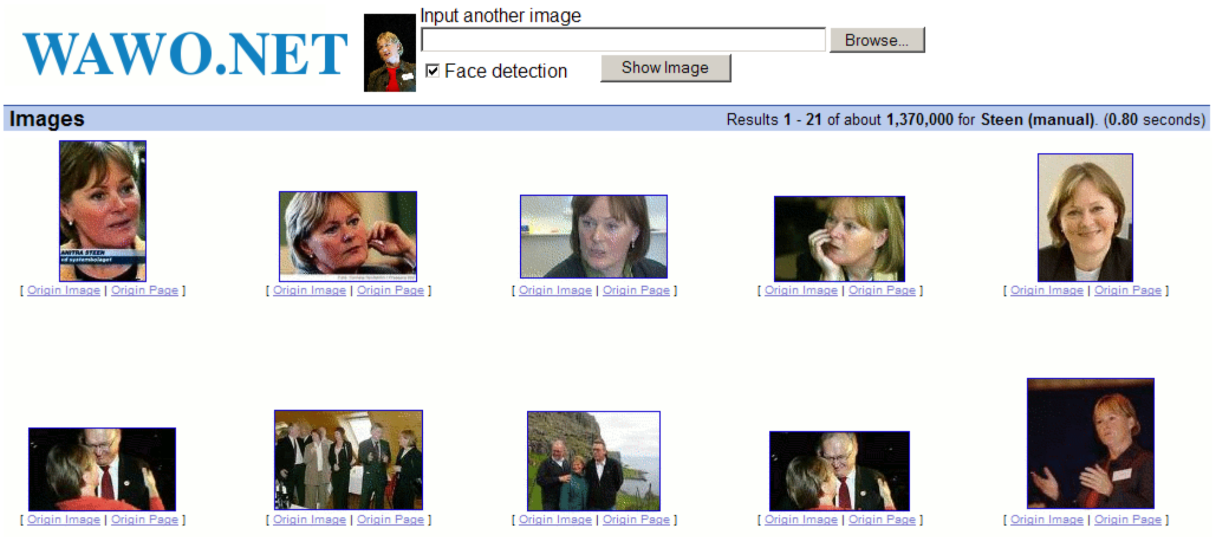

Fig. 3. A sample result page from `www.wawo.net`.

Unlike any previous HMM based face recognition approaches, the only HMM in the system is trained by an abstract observation sequence.

3.2.2. *Face recognition and HMM*

In the usual HMM based approach, where there are specific observations to represent and a dedicated HMM to model each individual person, the observation vectors are labeled once to form an observation sequence and then the sequence is put into each HMM to compute a similarity score. The proposed HMM system however has just one set of abstract observations and a single HMM. But this HMM does not model any specific face. Instead different observation sequences are generated and used as a means to distinguish between persons. In the context of measuring similarity between a query image and a person in the database, if the matched observation vector belongs to the person then it will be labeled as one of the abstract observations. Otherwise it will be labeled as non-relevant. The previous implies that the comparison has no meaning. The abstract observation is essentially exploited as a common tag specifying a new observation.

The main procedure to compute the similarity between the query face and each face in the database in the single HMM system can be summarized as follows:

1. The observation vector is given a label of observation.
2. Those observation vectors that have the same index form an observation sequence.
3. The sequences are broken down to subsequences of length two.
4. Each subsequence is used as input testing observation sequence for the HMM to compute the likelihood. The higher the probability of the observation subsequence, given the model λ, the more likely the subsequence belongs to the face under consideration.
5. The summation of the likelihood over all subsequences for the query gives the matching score between the query and each face independently.

4. SYSTEM SCALABILITY

Currently the system is based on a static collection of images. The next logical step in the evolution of the system is its transformation into a full-scale web search engine. In doing so Internet crawlers and indexers are needed. This will enable the system to increase the size of the image database and become fully autonomous.

Figure 4 depicts the face crawler. The main components of the face crawler are:

- URL server
- Web crawlers or spiders
- Mark-up language processing
- URL resolvers
- Name identifiers
- Face detector
- Indexer

From the above components it is worth spending some time on the following ones:

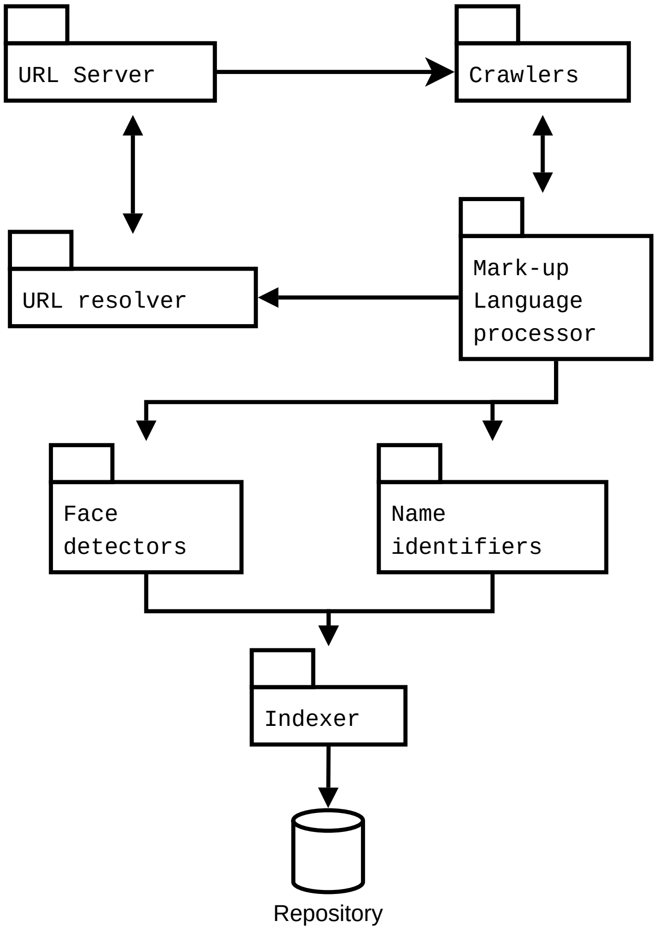

Fig. 4. The architecture of the face crawler.

- URL server: This server stores all the URL of the visited pages along with time information. This way we know when was the last time a particular web page was last seen from the spiders.
- Mark-up Language processor: They are responsible for the cleaning of the HTML code.
- URL resolvers: Recognition of URL, elimination of the ones that do not need to be visited and up-date of the URL server.
- Face detector: If there are images in the web page under consideration then the face detector tries to determine whether there are humans depicted in this image. If there are faces then the photo is stored in the Repository otherwise it is cast away.
- Name identifier: In case that there are some photos depicting humans then the module Name identifier using part-of-speech tagging tries to identify proper nouns (Names) in the web pages under consideration. This nouns can reveal the name of the depicted person.

5. REFERENCES

[1] H. S. Le and H. Li, "Can the surveillance system run pose variant face recognition in real time?," in *Proc. of ICCV Workshop in Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS)*, 2005.

[2] "Riya: The visual search engine," http://www.riya.com/index?btnSearch=people, (accessed 070212).

[3] "Online face alignment demo," http://facealignment.-ius.cs.cmu.edu/alignment/webdemo.html, (accessed 070116).

[4] "Betaface: Face detection and recognition," http://www.betaface.com/links.aspx, (accessed 070111).

[5] "Pittpatt: Pittsburgh pattern recognition, inc.," http://demo.pittpatt.com/.

[6] "Polar rose ab," http://www.polarrose.com/.

[7] "Cydral: The visual search engine," http://www.cydral.com/, (accessed 070209).

[8] P. Viola and M. J. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. of IEEE Comp. Society Conf. on Computer Vision and Pattern Recognition (CVPR'01)*, 2001, vol. I, pp. 511–518.

[9] A. Kuranov, R. Lienhart, and V. Pisarevsky, "Empirical analysis of detection cascades of boosted classifiers for rapid object detection," in *Proc. of DAGM 25th Pattern Recognition Symposium*, 2003, vol. 2781, pp. 297–304.

[10] R. Lienhart and J. Maydt, "An extended set of haar-like features for rapid object detection," in *Proc. of Int. Conf. on Image processing (ICIP'02)*, 2002, vol. I, pp. 901–903.

[11] H. S. Le and H. Li, "Face identification system using single hidden markov model and single sample image per person," in *Proc. of IEEE Int. Joint Conf. on Neural Networks (IJCNN'04)*, 2004, pp. 455–459.