FUSION OF FACE NETWORKS THROUGH THE SURVEILLANCE OF PUBLIC SPACES TO ADDRESS SOCIOLOGICAL SECURITY RECOMMENDATIONS

SK Alamgir Hossain, Abu Saleh Md Mahfujur Rahman, Abdulmotaleb El Saddik

Multimedia Communications Research Laboratory,
University of Ottawa, Ottawa, Canada
Email:{skahossain, kafi, abed}@mcrlab.uottawa.ca

ABSTRACT

Researchers around the world are trying to address the ever increasing security requirements by bringing new approaches to surveillance specifically in public places like school, railway, subway station, air port etc. To establish and sustain security in public spaces, surveillance plays a key role in technology-dependent governance common to many countries in the world. Traditionally, through the routine surveillance, an automated security system gains knowledge about people and their activities in a certain space. In this paper we are proposing a fusion algorithm to aggregate surveillance parameters from more than one such spaces. Inspired by existing works on social network analysis based on human photos, we propose a new face network structure model. These face network structures are later fused to obtain sociological parameters of a person of interest and gather recommendations about the circle of associates of that individual. We believe these type of recommendations are helpful in comprehensive investigation purposes.

Index Terms— Surveillance, Face Recognition, Face Co-occurrence, Weighted graph

1. INTRODUCTION

The problem of visual inspection of public space surveillance (e.g., airports, railway stations, roads, etc.) has received growing attention in the last couple of years [1][2][3]. Public space surveillance, which works in real time, is able to detect, localize, track, and classify multiple objects moving in a surveilled area.

Traditionally, through the routine surveillance, an automated security system gains knowledge about people and their activities in a certain space, where myriad scene specific events and sensory data are available. The automated fusion of the data can provide more comprehensive view of those events. In this paper we are proposing a method to aggregate surveillance parameters from more than one such spaces. The integration of video technology and sensor networks constitute the fundamental infrastructure for new generation of multimedia surveillance systems, where many different media streams such as audio, video, images, and sensor signals will concur to provide an automatic analysis of the controlled environment and a real-time interpretation of the space.

Secondary information channel with proximity, distance, night-vision, audio etc. sensors are often used to augment and enhance this exploration [1][3][4]. Applications regard surveillance of wide indoor and outdoor spaces [2] and particularly people surveillance : in this case, multimedia surveillance systems can be enriched with biometric technology; the best views of detected persons and their extracted visual features e.g. faces, voices, and trajectories [5] can be exploited for people identification. In practice however, face detection [6][7] and recognition [8][9][10] contributes hugely in most of the modern surveillance systems. By using the presence of faces in a visual media, associative face co-occurrence networks can be established for recommending friends in social networks [11]. In our presented surveillance framework we leverage similar face co-occurrence model in a surveillance scenario in order to identify a person of interest and gather knowledge about the social networks of that individual. In this model, we propose a new face network structure specific to certain scene. Later, these face network structures are fused to obtain sociological parameters of a person of interest and gather recommendations about the circle of associates of that individual.

Our contribution in this paper is two-fold. First, we propose a method for constructing a face network in a surveillance scenario, where the face network can be used for recommending person of the same interest. Second, we presented an algorithm by which the system can fuse the face networks by using the stored faces to gather sociological recommendations.

The remainder of this paper is organized as the following. At first, in Section 2 we present discussion on some research works on surveillance and social networking that are related to our approach. We illustrate the various components of the proposed system for constructing face network from a scene in a public place surveillance in Section 3. Further, in Section 4 we present some key experimental results and evaluate the

978-1-61284-350-6/11/$26.00 ©2011 IEEE

performance of the system. At the end we provide conclusion of the paper in Section 5 and state some possible future work directions.

2. RELATED STUDY

Many researchers around the world are working on public surveillance and recognition of behaviour based on the captured images, videos, and audios. Some of those works are based on direct captured video and event based surveillance. A notable work is [4], which is a open and extensible framework designed and developed by Exploratory Computer Vision Group in IBM T.J. Watson Research Center. The system is a middleware offering for use in surveillance systems and provides video based behavioural analysis capabilities. The prototype system has two main components Smart Surveillance Engine (SSE), which has front-end video analysis capabilities and the second one is Middleware for Large Scale Surveillance (MILS), which provides data management capabilities.

For the advent of face recognition and face matching technology [12][8][9][5], it is possible to detect and recognize human face image in real time through a video camera [12]. Most of the current face recognition and matching use neural-network or fuzzy pattern matching approach [7] to recognize the human face images. Now it is also possible to track multiple people from multiple camera system. One of the recognized work proposed by N. Katja et. al. [13], where the author presents a multi-view tracker, meant to operate in smart rooms that are equipped with multiple cameras. This prototype system also demonstrates a virtual classroom application, where the system automatically selects the camera with the 'best' view on the face of a person moving in the room. Real-time object tracking, which is needed to achieve this, is implemented by means of color-based particle filtering.

Although privacy is a main concern in video or image based surveillance, many prototype system blur the important part (e.g. human face) or conceal regions of interest (ROIs) of an image or video at real time specially when capture a public place. Some of the notable work related to privacy protection mechanism in video surveillance is [14][15]. Moreover, photo tagging [10], face co-occurrence network [11], and automatic face annotation [16] can enhance the recognition of people of interest both in network. Existing work specially [11] the important of face images for recommending friend list in social network scenario. Motivating of this existing work we are applying the idea in public surveillance scenario, in which our new approach can give a people list to the security authority based on the stored images and the face network that we presented in the next Section 3.

3. PROPOSED FRAMEWORK

In this section we present the detail illustration of various components of the system. At the beginning in Section 3.1 we present the general description of the proposed system. We illustrate the face network construction process in Section 3.2. Later in Section 3.3 we present Minimized Face Co-occurrence Network (*MFCN*) and detail the fusion algorithm of the face network structures. Lastly, in Section 3.4 we present the social parameters extraction steps based on the Minimized Face Co-occurrence Network.

3.1. General Architecture

In this section we describe the general architecture of the proposed system. The architecture is depicted in Figure 1 as a block diagram. Different camera, audio, sensor (proximity, motion sensor) networks are connected with their subsystems. The subsystems are responsible for fetching the video, audio, and proximity sensory data from the camera, microphone and IR sensor respectively. The subsystem units send or receive command to and from the centralized surveillance system.

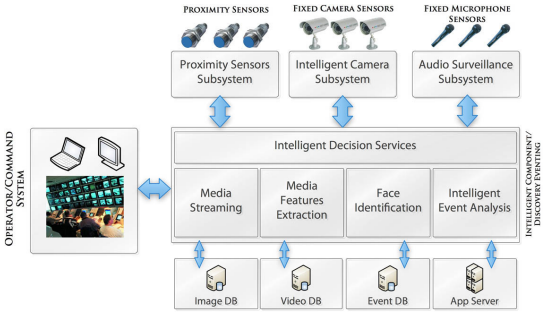


Fig. 1: Proposed face co-occurrence based surveillance architecture.

The centralized surveillance system is composed of different components and manages all the subsystem up & running. Different intelligent components like media streaming service, media feature extraction component, face identification and matching component, event analysis components are placed within the surveillance system. The components are loosely coupled to each other that is we can add a new component with the system or remove the existing components without affecting the whole system. Service Oriented Architecture (SOA) give us this type of coupling flexibility, where Web Service Dynamic Discovery and Web Service Eventing protocol can find a specific service and query the network using multi-cast search messages and also can register to various system events. Thus, changes in the service can be notified to any client without requiring standard polling mechanism. The event delivery is accomplished by using simple asynchronous messaging. In a typical client-server relation, interactions are always from client to server (e.g. invocation of a service method is accomplished by a message always initiated and sent by a client). Servers are typically passive, and are waiting to serve client requests. On the contrary, the eventing mechanism allows servers to notify clients, and become active. When a change in the server occurs, the server initiates a

new communication by sending a message to the clients (Subscribers). The extracted face images are stored into a dedicated image database. The system also incorporates logical databases (within the same physical space) for video, events the streaming video and the event related data. For security reasons we stored the face images in to the database instead of directly storing into the disk.

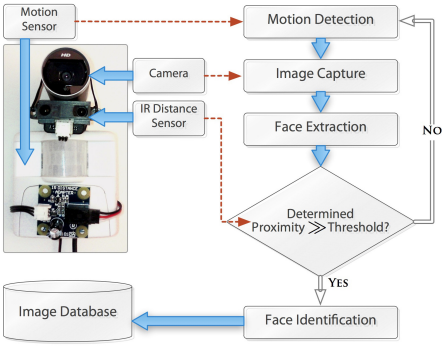


Fig. 2: Flowchart of the image acquisition steps.

A flow chart of our system is shown in Figure 2. In order to conserve energy the camera is inactive if no motion is detected in front of the camera. A motion detector activates the camera whenever a new motion is registered and the scene is captured as an image. After that the faces from the images are extracted for storing into the database. Through the image and proximity data analysis the system approximated the distances between the faces. The distances of the faces of those persons should be within a boundary before the system initiated the face network graph analysis. At this time those extracted faces were stored into the database. For a given face, if any previous entry was found into the database then the system updated that profile entry otherwise a new profile was assigned to the new face before storing the surveillance data such as location, data, time, proximity etc. into the database.

Definition 1: *Face Detection* is the process of finding a 'face' within images or videos and Face Recognition is the process of matching the detected 'face' to one of the many faces known to the system.

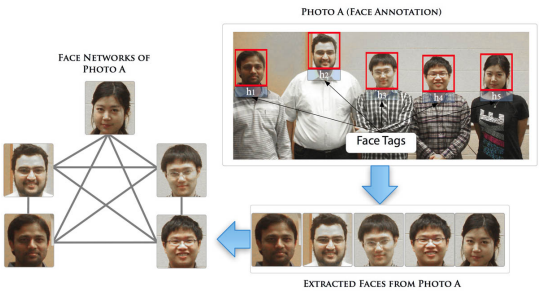
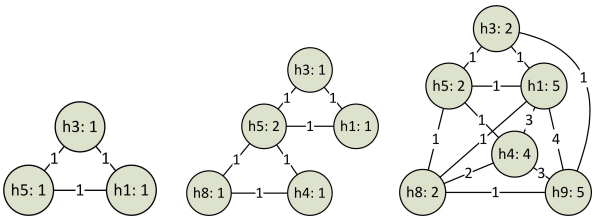


Fig. 3: A sample photo and the extracted face images.



(a) Insert photo 1 (b) Insert photo 2 (c) Insert all photos

Fig. 4: An example face network created from the photos that illustrated in Table 1

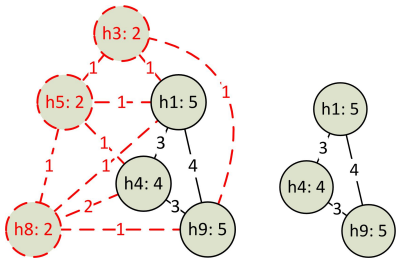
3.2. Building Face Networks

In this section, we explain the construction process of the face network. The face network is an undirected, weighted graph G and is a data structure for representing peoples faces in the network. The graph G contains a set of N nodes called vertices V and a set of edges E. Where each edge E is an undirected weighted link between two vertices V_i and V_j, where $i, j < N$. Each vertex V represent a face image of a person and the associated tag information $FaceId$, $Location$, $Date$, $Time$, $Proximity$, $Count$, where $FaceId$ is an identification number of a face image that will generate to uniquely identify a face, and $Location$, $Date$, $Time$ are the location, date, time information when the face image is collected through the camera, and $Proximity$ is the distance between the camera and the person which face has taken, finally $Count$ stores the occurrence frequency of the person face image in the photos. Each edge E consist of three attributes: $FromNode$, $ToNode$ and $WeightCount$, where the weight between the vertex $FromNode$ and $ToNode$ is $WeightCount$.

Table 1: Example photos taken from a surveillance camera.

Photo ID	Persons in photos	Location
$C_1^{P_1}$	h_1, h_5, h_3	Loc_1
$C_2^{P_2}$	h_5, h_8, h_4	Loc_2
$C_1^{P_3}$	h_4, h_1, h_9	Loc_1
$C_3^{P_4}$	h_9, h_1	Loc_3
$C_2^{P_5}$	h_3, h_9	Loc_2
$C_2^{P_6}$	h_9, h_1, h_4	Loc_2
$C_2^{P_7}$	h_4, h_8, h_1, h_9	Loc_2

Now we are presenting an example of the fusion process of face networks from seven sample photos taken from three different locations (Loc_1, Loc_2, Loc_3). The list of photos are showing in Table 1 where $C_i^{P_j}$, means P_j is a photo taken from the camera C_i. Now for the first photo P_1 that taken from location Loc_1 using camera C_1 has face of person h_1, h_5 and h_3. After converting this photo to the face network the network look like Figure 4a, where h_1 is connected to h_5, h_5 is connected to h_3 and h_3 is connected to h_1. For the second



(a) Final graph (b) The minimized graph

Fig. 5: After minimizing the example network. The red color in 5a is indicating the unnecessary links.

photo P_2 that taken from location Loc_2 using camera C_2 has face of person h_5, h_8, h_4, so if insert the photo to the face network then the combined network looks as shown in Figure 4b, where h_5 is connected to h_8, h_8 is connected to h_4, and h_4 is connected to h_5. As h_5 appear two times so the *count* value of h_5 should be 2. In this way when all the seven photos are inserted into the face network then the network looks like as shown in Figure 4c. Here one thing we want to mention that the edge between h_1 and h_9 is 4 means that both h_1 and h_9 appear 4 times in different photos, also both the *count* value of h_1 and h_9 is 5 means that h_1 appears 5 times and h_9 appears 5 times in different photos.

3.3. Minimized Face Co-occurrence Network (*MFCN*)

The main aim of this work is to give person recommendation to the security personnel based on the stored face network. In our proposed system the edge weight of a face network is the most important information but the method that we presented in Section 3.2 have some edge weight that are less useful compared to some edge weight. For example in Figure 4c the weight between h_3 and h_5 is 1 that means both h_3 and h_5 appear only one time, this weight is significantly negligible value compared to the link between h_1 and h_9, h_4 and h_9. So our next step was to minimize the face network based on the weight. For this requirement we are introducing a new value T called the threshold value. The threshold value T is used to construct a new network called Minimized Face Co-occurrence Network (*MFCN*), in which all the weight value is greater than or equal to T. The *MFCN* graph is presented in Figure 5b.

Definition 2: Minimized Face Co-occurrence Network (*MFCN*) is a graph $G^{'}$ of $N = (V^{'}, E^{'})$ tuple of two sets, such that $V^{'} = v_1^{'}, v_2^{'}, v_3^{'}, ..., v_k^{'}$ is the set of vertices and $E^{'}$ is the set of edges between two vertices, where the weight $W^{'} \geq T, V^{'} \subset V$, and $E^{'} \subset E$.

3.4. People Recommendation based on *MFCN*

The idea behind our work is if two people are in a certain place, in a certain moment, with certain time then there may be some common interest exists between those people. Our system stores the captured images from the camera in a certain interval and store the images into the database. When any suspected event happen then the system can construct the *MFCN* graph by using Algorithm 1. The Algorithm 1 construct the graph and filter by the threshold value, definitely it is also possible to filter the result based on the location, date time information. For hiding the complexity in the algorithm, we filtered the result by the threshold value.

Algorithm 1: *MFCN* Graph Construction

Input: A threshold: T
Output: MFCN Graph: g

```
1  begin
2      g ← new Graph()
3      g.nodeList ← create empty node list
4      g.edgeList ← create empty edge list
5      photoList ← fetch the photo list from the database
6      foreach photo p ∈ photoList do
7          faces ← getFacesByPhotoId(p)
8          totalFaces ← count(faces)
9          for i ← 0 to totalFaces-1 do
10             fromNode ← createNode(faces[i])
11             if fromNode is not in g.nodeList then
12                 add fromNode to g.nodeList
13             for j ← i + 1 to totalFaces-1 do
14                 toNode ← createNode(faces[j])
15                 if toNode is not in g.nodeList then
16                     add toNode to g.nodeList
17                 if Edge(fromNode,toNode) is not in
                   g.edgeList then
18                     add Edge(fromNode, toNode) to
                       g.edgeList with value 1
19                 else
20                     increment Edge(fromNode,
                       toNode) in g.edgeList by 1
21     foreach Edge e ∈ g.edgeList do
22         if e.weight < T then
23             remove e from g.edgeList
24     remove all the node from g.nodeList which is not
       in the g.edgeList
25     return g
26 end
```

4. IMPLEMENTATION AND RESULT

This section describes the implementation details of the prototype system that reflects the architecture presented in Section 3. Microsoft Visual C++ version 6.0 platform and SQL server 2005 database server was adopted to develop the system. In the following, Section 4.1 describes the experimental setup

and the different devices that are connected inside the environment. Further, Section 4.2 presents the test results based on the experiment. Finally, Section 4.3 describes the performance details of our proposed method.

4.1. Experimental Setup

This section describes the experimental setup that we conducted in our laboratory environment. Figure 2 shows the camera and sensory device combos that were installed in two (2) different locations in the test environment. Two (2) of the device combos were placed inside our multimedia lab and the rest two (2) were placed at the lab corridor. For the real-time surveillance we used high definition Microsoft $LifeCam$ 3.0 camera, $X10$ Pro $Occupancy$ motion sensor and $Sharp$ IR distance sensor (see Figure 2). The camera had built-in microphone and we utilized that to capture the sound. The prototype system took images periodically with certain triggered events. The events were triggered by monitoring the microphone sound as well as the motion tracking IR devices. For the validation of our algorithm the prototype system stored the captured images into the disk. For the face recognition we used image based face recognition algorithm [12], which used the algorithm called Principal Component Analysis method (PCA). In our experiment, 15 volunteers participated by roaming in front of the test devices in different postures and in different groups.

Table 2: Overview of our experimental result.

Total video length (4 cameras) in hours	288
Total num of photos	24560
Total num of stored photos into the database	9021
Total number of faces	17452
Total number of distinct faces	15
Num. of connections when T = 10	104

4.2. Results

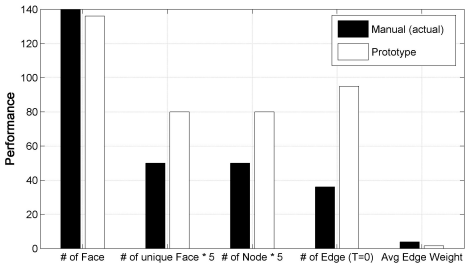
We monitored our laboratory area for three (3) subsequent days with the test subjects and stored their images into the database. The overview of the experimental data size are shown in Table 2. For manual calculation we designed a test case containing a list of 25 photos that were collected from our image database. We prefer those photos in which most number of faces were visible. The overall distribution of faces in the collected photos in the test case is shown in Figure 6b.

Table 3: Test case overview.

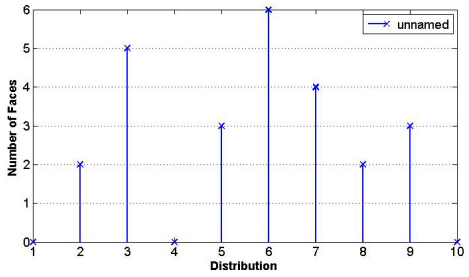
# of photos	25
# of faces	140
# of unique faces	10
maximum connection possible	45

4.3. Evaluation

In order to test the validity of the proposed approach we manually created the *MFCN* graph and in the mean time constructed a *MFCN* graph by using Algorithm 1. The comparison result is depicted in Figure 6a. In this figure the black vertical bars represent the manual calculation result and the white vertical bars represent the result achieved by the prototype.



(a) Performance comparison



(b) Face distribution

Fig. 6: Parameters of the face network evaluation of the Test Case.

In the case of manual calculation we observed that the test case had 140 face images in which the unique human face image was 10. However, the prototype showed some errors in its calculation. In the later case of the prototype calculated that the test case had a total of 136 images and the number of unique face images were 16. The discrepancy was due to the fact that the image recognition did not perform accurately and suffered some poor face recognition in its analysis. As a result, the system failed to identify faces in some images. The percentage of error for this test case was approximately 2.8% and 60% for image recognition and matching respectively. Further, in our manual calculation we found that the number of edges between the nodes were 36 and the average calculated weight was 3.8. But in the case of our prototype system, the total number of deduced edges were 95 and the average weight was 1.5.

5. CONCLUSION

In this paper we presented a method for constructing a face network in a surveillance scenario, where the face network was used for recommending person of interest. We also presented an algorithm for constructing a Minimized Face Co-occurrence Network (*MFCN*) graph by fusing a set of face

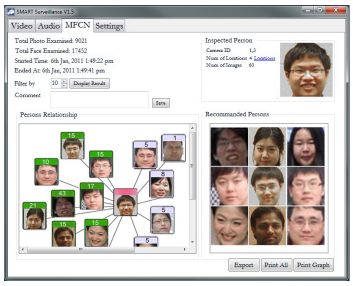


Fig. 7: The prototype system is recommending persons and showing the sociological relationships.

networks obtained from different locations. We developed a prototype system using the proposed methodology. The experimental result showed that our system was working efficiently in a medium to low crowded places (e.g., school lab, library, home etc). In future we want to evaluate the system further to measure the performance of the system with a large-scale dataset. We believe that the proposed approach of recommendation in surveillance based on the *MFCN* will bring new direction in the public space monitoring research.

6. REFERENCES

[1] P. Perez and B. Gidas, "Motion detection and tracking using deformable templates," in *Image Processing, 1994. Proceedings. ICIP-94., IEEE International Conference*, Nov. 1994, vol. 2, pp. 272 –276 vol.2.

[2] G.L. Foresti, "A real-time system for video surveillance of unattended outdoor environments," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 8, no. 6, pp. 697 –704, 1998.

[3] Rita Cucchiara, "Multimedia surveillance systems," in *Proceedings of the third ACM international workshop on Video surveillance & sensor networks*, New York, NY, USA, 2005, VSSN '05, pp. 3–10, ACM.

[4] Chiao-Fe Shu, A. Hampapur, M. Lu, L. Brown, J. Connell, A. Senior, and Yingli Tian, "Ibm smart surveillance system (s3): a open and extensible framework for event based surveillance," in *Advanced Video and Signal Based Surveillance, 2005. AVSS 2005. IEEE Conference on*, 2005, pp. 318 – 323.

[5] Ismail Haritaoglu, David Harwood, and Larry S. Davis, "Hydra: Multiple people detection and tracking using silhouettes," *Visual Surveillance, IEEE Workshop on*, vol. 0, pp. 6, 1999.

[6] Qian Chen, Haiyuan Wu, and M. Yachida, "Face detection by fuzzy pattern matching," in *Computer Vision, 1995. Proceedings., Fifth International Conference on*, June 1995, pp. 591 –596.

[7] Haiyuan Wu, Qian Chen, and M. Yachida, "Face detection from color images using a fuzzy pattern matching method," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 21, no. 6, pp. 557 –563, June 1999.

[8] S. Lawrence, C.L. Giles, Ah Chung Tsoi, and A.D. Back, "Face recognition: a convolutional neural-network approach," *Neural Networks, IEEE Transactions on*, vol. 8, no. 1, pp. 98 –113, Jan. 1997.

[9] Shakhnarovich Lee Darrell, G. Shakhnarovich, L. Lee, and T. Darrell, "Integrated face and gait recognition from multiple views," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2001, pp. 439–446.

[10] Jae Young Choi, Seungji Yang, Yong Man Ro, and Konstantinos N. Plataniotis, "Face annotation for personal photos using context-assisted face recognition," in *Proceeding of the 1st ACM international conference on Multimedia information retrieval*, New York, NY, USA, 2008, MIR '08, pp. 44–51, ACM.

[11] Heung-Nam Kim, Jin-Guk Jung, and Abdulmotaleb El Saddik, "Associative face co-occurrence networks for recommending friends in social networks," in *Proceedings of second ACM SIGMM workshop on Social media*, New York, NY, USA, 2010, WSM '10, pp. 27–32, ACM.

[12] Intel Corporation, "Open source computer vision library, http://opencv.willowgarage.com/wiki/," Tech. Rep., Intel, Last accessed, Feb 2011.

[13] Katja Nummiaro, Esther Koller-Meier, Tomš Svoboda, Daniel Roth, and Luc Gool, "Color-based object tracking in multi-camera environments," in *Pattern Recognition*, vol. 2781 of *Lecture Notes in Computer Science*, pp. 591–599. Springer Berlin / Heidelberg, 2003.

[14] F. Dufaux and T. Ebrahimi, "Scrambling for privacy protection in video surveillance systems," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 18, no. 8, pp. 1168 –1174, 2008.

[15] A. Senior, S. Pankanti, A. Hampapur, L. Brown, Ying-Li Tian, A. Ekin, J. Connell, Chiao Fe Shu, and M. Lu, "Enabling video privacy through computer vision," *Security Privacy, IEEE*, vol. 3, no. 3, pp. 50 – 57, 2005.

[16] Jae Young Choi, W. De Neve, Y.M. Ro, and K.N. Plataniotis, "Automatic face annotation in personal photo collections using context-based unsupervised clustering and face information fusion," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 20, no. 10, pp. 1292 –1309, 2010.