

Recognizing persons in images by learning from videos

Eva Hörster, Jochen Lux, Rainer Lienhart

Angaben zur Veröffentlichung / Publication details:

Hörster, Eva, Jochen Lux, and Rainer Lienhart. 2007. "Recognizing persons in images by learning from videos." In *Multimedia Content Access: Algorithms and Systems - ELECTRONIC IMAGING 2007*, 28 January - 1 February 2007, San Jose, CA, United States, edited by Alan Hanjalic, Raimondo Schettini, and Nicu Sebe, 65060D. Bellingham, WA: SPIE.
<https://doi.org/10.1117/12.705200>.

Nutzungsbedingungen / Terms of use:

licgercopyright

Dieses Dokument wird unter folgenden Bedingungen zur Verfügung gestellt: / This document is made available under these conditions:

Deutsches Urheberrecht

Weitere Informationen finden Sie unter: / For more information see:

<https://www.uni-augsburg.de/de/organisation/bibliothek/publizieren-zitieren-archivieren/publiz/>



Recognizing Persons in Images by Learning from Videos

Eva Hörster, Jochen Lux, Rainer Lienhart

Multimedia Computing Lab
University of Augsburg
Augsburg, Germany

ABSTRACT

In this paper, we propose an approach for automatically recognizing persons in images based on their general outer appearance. Therefore we build a statistical model for each person. Large amounts of training data are collected and labeled automatically by using a visual sensor array capturing image sequences containing the person to be learnt. Foreground-background segmentation is performed to separate the person from background, thus enabling to learn the persons appearance independent of the background. Color and gradient features are extracted representing the segmented person. Person recognition of incoming photos is carried out using (k)-Nearest Neighbor(s) classification and the normalized histogram intersection match value is used as distance measure. Reported experimental results show that the presented approach performs well.

Keywords: Object recognition, object identification, person recognition, person identification

1. INTRODUCTION

Digital cameras and camcorders are nowadays widely used to generate huge amounts of photos and video clips. This media data needs to be organized efficiently. One attractive index is to organize pictures and video clips based on the people they picture. If individuals can be recognized reliably, the user can ask the system to list all images and videos showing some person X .

One way to build such a system is by manually labelling each photo either at creation time, during upload to the PC, or at a later point in time. This, however, is a very time consuming and not scalable endeavour. Instead, a system is needed that automatically can learn the appearances of individuals by passively observing them over a long period of time in order to capture their variability in appearance under changeable conditions.

From this massive amount of observed video data, a compact detection and recognition model needs to be learned for each person. Each model will allow recognition of a specific individual in new photos and video clips. These models can be shared among users of the same user group such as friends, team mates, and family members.

In general, the task of people detection and recognition in unconstrained environments raises a number of difficult challenges due to the similarity between images of different people under the same conditions (e.g., same clothes and poses) and the large dissimilarities between images of the same individual under pose variations, changes in clothes, as well as illumination and background changes. To capture those appearance variations a statistical model is needed that has learned the numerous and diverse visual manifestations of a person.

In our work we assume that a person, we want recognize, has a similar appearance in the test images and video as he/she had during the learning phase. Therefore we need to observe each individual in many different situations over days and weeks to get enough training data that covers realistically and representatively the variability in visual appearance of the individual such as his/her complete wardrobe, body expressions, and poses as well as general conditions such as illumination variations.

Consequently, we have to address two major issues here:

Further author information: (Send correspondence to Eva Hörster)

Eva Hörster: E-mail: hoerster@informatik.uni-augsburg.de, Telephone: +49 (821) 598 4368

Jochen Lux: E-mail: lux@informatik.uni-augsburg.de, Telephone: +49 (821) 598 4386

Rainer Lienhart: E-mail: lienhart@informatik.uni-augsburg.de, Telephone: +49 821 598 5803

Multimedia Content Access: Algorithms and Systems, edited by Alan Hanjalic, Raimondo Schettini, Nicu Sebe,
Proc. of SPIE-IS&T Electronic Imaging, SPIE Vol. 6506, 65060D, © 2007 SPIE-IS&T · 0277-786X/07/\$15

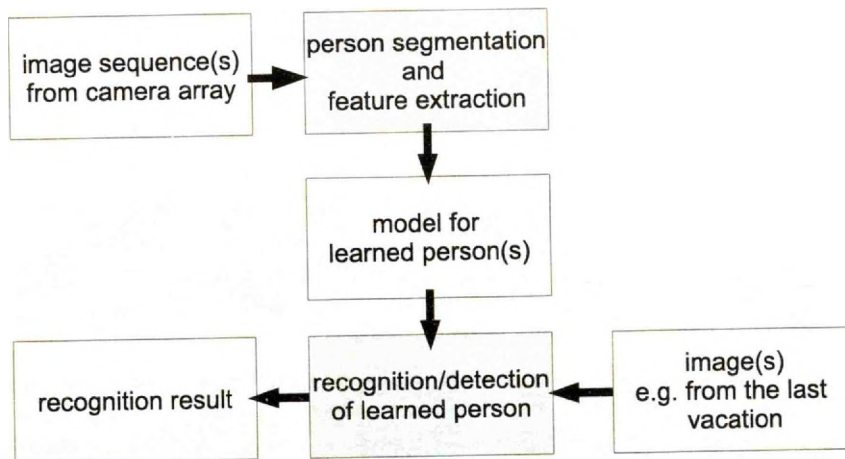


Figure 1. System overview

1. How to obtain easily and almost effortlessly our training data and
2. How to learn from that data a reliable statistical model.

For the former, we use a visual sensor array consisting of several cheap web cameras in a room, which is mostly used by the person whose visual appearance we want to learn. This room could, for instance, be his/her office. Thus we assume that the person captured by our cameras is the person to be learnt. Since he/she must have entered the room by foot, foreground-background segmentation algorithms can be used to segment the visual appearance of him/her from the static background. This enables us to separate the person's appearance from the background making our learning task simpler.

This scenario has several advantages. First of all we do not need to manually label a large amount of data. Additionally, we can easily observe the person over many days and weeks and thus capture his/her full variability in appearance. By capturing with multiple cameras at the same time we enlarge the variety in the training data set that occurs due to differences in lighting conditions, pose, and capture properties of the cameras.

1.1. System Overview

The system we present in this paper consists of two modules: (1) Data acquisition and feature extraction and (2) person recognition. Figure 1 shows an outline of our system. Each captured image from our camera array is forwarded to the foreground segmentation and feature extraction module. We use the outer appearance of the human body for the recognition task, thus we extract color and gradient features describing the appearance from the segmented person. Even though clothes are exchangeable it is still a valuable measure to detect and recognize people, as long as the clothes have been already learned. For recognition an image is input to the recognition module, features are extracted of the entire image, as foreground segmentation is not possible. Adequate matching with the existing person models is performed for recognition. We have tested our system with images taken with different background and illuminations. These images have been taken weeks and month after the image sequences for learning have been captured. Thus our recognition results show that the systems is not sensitive to slight changes in outer appearance and illumination as well as background. We also propose a method to handle images containing multiple as well as unknown persons.

The paper is organized as follows. In Section 2 we introduce related work and discuss some of their shortcomings. Then, Section 3 describes our data acquisition setup and processing infrastructure, before Section 4 details the feature extraction. The model learning and classification methods are presented in Section 5. Experimental results are reported in Section 6 and Section 7 concludes the paper and presents directions of future research.

2. RELATED WORK

Several systems have been described for person recognition in image sequences as well as photos. A commonly used camera-based recognition method of individuals is face recognition. Reference methods for face recognition are eigenfaces,¹ elastic graphic matching,² and Bayesian methods.³ Face recognition however requires performing face detection first, which currently only works reliable for frontal faces in unconstraint environments. It also requires that a face is visible at all. For instance, a person pictured from behind cannot be recognized by face recognition methods. Face recognition techniques are also sensitive to pose, expression and illumination changes. Most importantly the faces have to be captured with sufficient resolution in order to be recognizable. If this cannot be guaranteed face recognition should be combined with or replaced by some model of a person's complete outer appearance to enable robust recognition.

Therefore an appearance-based approach is presented in this paper. An individual's appearance consists of many aspects: the face, the hair and the clothes. Research in this area has been done by Suh et al.,⁴ Zhang et al.,⁵ Nakajima et al.,⁶ Hähnel et al.,⁷ Elgammal et al.,⁸ and Sivic et al.⁹

The works of Suh et al.⁴ and Zhang et al.⁵ used clothing colour to recognize people with detected faces in image collections. A statistic model for torso colour was computed from a rectangular region below the detected frontal faces. These methods require again performing face detection and work only for frontal faces in unconstraint environments.

In the work of Nakajima et al.,⁶ the identity and the pose of eight persons was recognized by applying several colour and shape features. Support vector machines and k-Nearest Neighbour matching were used as classification methods. In Hähnel's work,⁷ colour and texture features were extracted for each person, and an RBF network classifier as well as a Nearest-Neighbour classifier is used for recognition. Image sequences have been used for learning and recognition in both systems which simplifies the task as the person can be segmented also for recognition. The videos for learning and recognition were taken by the same camera under the same conditions such as distance to the persons, lightning, etc. Both systems assume further, that the person in the images matches at least one model. This is a disadvantage, because those systems can neither handle unknown persons nor individuals in cluttered images.

The work of Elgammal et al.⁸ proposes an approach for identifying people in surveillance videos. A model for each person is learned from foreground regions when persons are isolated. Then the recognition task is to classify foreground regions as either containing known persons and if so assigning names to each of them.

The work of Sivic et al.⁹ is most similar to our approach. Here the goal was to find all occurrences of a particular person in a sequence of photographs taken over a short period of time. The authors propose a two stage algorithm, where a colour-based pictorial structure model is used to find occurrences of individuals in images no frontal face could be detected. The proposed approach works well assuming that the background is similar for all images in the sequence. Thus it is not practical in situations where a person needs to be identified in unconstraint environment and thus the specific background may not be learned in advance.

3. DATA ACQUISITION

3.1. Physical Setup

In order to acquire a large amount of positive training data for each person and to reduce the influence of a specific camera sensor on our training data, we use a visual sensor array consisting of multiple cheap web cameras. The cameras are distributed over the entire room at various positions and heights to ensure high coverage and variability in lighting and pose of the captured data. Especially some cameras are taking pictures from below the eye-level, at the eye-level, and above the eye-level. A schematic of the physical setup we used for data acquisition is shown in Figure 2.

Since the number of active (i.e., capturing) USB cameras under Windows XP is usually restricted to two and since most desktop computers cannot compress more than two video streams at 25 fps in real-time, our six USB cameras are connected to a set of four networked desktop computers. Connectivity is provided by wired Gigabit-Ethernet connections. The capture control of this distributed video sensor setup is described in the following subsection.

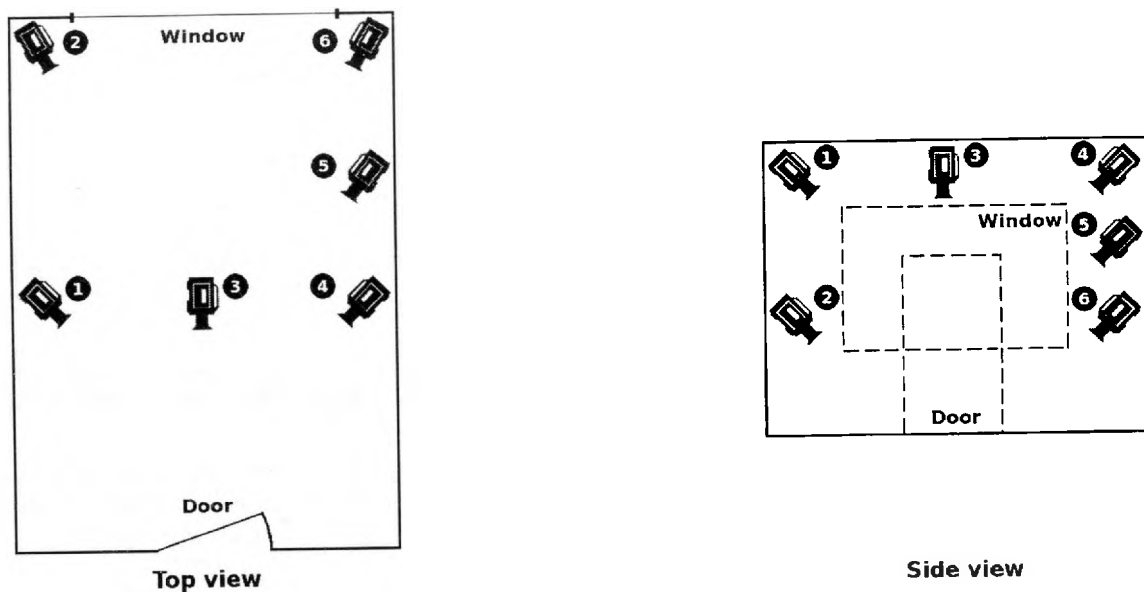


Figure 2. Video capture setup shown from the top view (left image) and the frontal view (right image). Camera positions are designated by camera icons together with their associated camera number.

3.2. Video Capture Process

The video capturing process can be controlled from a single computer, the master. This is achieved by distributing the sensor array to a number of connected, non-synchronized PCs. Theoretically, there exists no upper limit for the number of video devices in the network, but only up to two cameras can be connected to a single PC. Camera control is accomplished by using the Intel's Universal Plug and Play (UPnP) technology as a communication protocol. The UPnP technology provides a distributed, open networking architecture that employs TCP/IP and other Internet technologies.¹⁰ It is platform and device independent, thereby ensuring the possibility of using any visual sensor or personal computer hardware, as long as a camera driver and a UPnP protocol implementation is available for the platform in question.

Client PCs with installed camera devices are referred to as "UPnP devices" and run a background process, that multicasts their ID and installed "UPnP services" when joining the network. In our implementation, UPnP services represent the visual sensors that are connected to each UPnP device. The master PC runs a control program, which receives these UPnP messages and sends commands to each of the registered devices on user request. The client's software implementation is multi-threaded to prevent the capturing process from blocking incoming commands of the master PC. There is no other precondition for the choice of the specific master except that the PCs with the camera devices installed have to be reachable on the UPnP network ports.

In order to remotely control all camera recording functions from the master PC, our implementation features the following functions:

- Recording of a video of specified duration and beginning.
- Shooting a specified number of photos at given time and interval-time between shots.
- Requesting of information about the status of the registered devices.

Based on the the functional requirements listed above, device and service descriptions were designed on template basis and include the actions listed in table 1.

Action	Argument Direction	Argument Type	Argument Name
createSnapshot	Input	Integer	start
	Input	Integer	waitingPeriod
	Input	Integer	anzahl
	Input	String	acUserName
	Return	Boolean	success
startRecording	Input	Integer	startTime
	Input	Integer	duration
	Input	Integer	username
	Return	Boolean	success
stopRecording	Return	Boolean	success
getCameraCount	Return	Integer	cameraCount

Table 1. Implemented UPnP Actions

Before the video capturing process is started, the user is requested to input the name of the person who is moving into the field of view of the camera array. Together with the assumption, that the person of interest is the only moving object in the video, it is not necessary to do any additional labelling of the training data, thus ensuring minimal user effort.

After the video data has been captured, it is encoded by a lossy MPEG-4 Part 2 compression algorithm and stored at a central location.

3.3. Foreground segmentation

As a prerequisite to building models for identification, it is necessary to first separate the moving person from the background. We assume that only one person, the person to be learnt, is moving inside the room and that the background is static without sudden illumination changes. Thus, the moving foreground parts represent the person.

There exist many algorithms for background/ foreground segmentation (e.g.,¹¹¹²). The various approaches can be coarsely divided into (a) approaches which use frame by frame difference between images and (b) approaches which build statistical models of the background and use background subtraction to determine the foreground regions. We perform foreground segmentation using the algorithm presented in.¹³ This algorithm showed good performance in the VSSN05 foreground/ background segmentation algorithm competition.¹⁴ At every time instance the algorithm outputs a frame mask which specify the pixels of the foreground object (i.e., person). An example of a frame mask is shown in Figure 2.

During feature calculation we first compute the local features (see Section 4) of the entire video frame. Then we filter the local features by keeping only those whose centres are within the foreground regions determined by the frame mask. Thus we are keeping only features describing the person moving. It should be noted, that the segmentation results usually include quite some noise. Thus regularly small parts of the current background are included in our feature space of a person. As test images will usually be taken in front of different background, this should not pose a problem.

4. FEATURE EXTRACTION

The use of global features to describe a person's outer appearance has been proposed before. However, global features are only applicable if assumed that during recognition, a person or multiple persons in question can be roughly segmented from the background. In our usage scenario – such as in recognizing people in vacation photos using person models learned at home or in an office environment – we are not able to segment a candidate person from the background in these images during the recognition phase. Thus the use of local features becomes necessary.

We represent our images by a sparse set of points and a description of their respective neighbourhood. Salient points are determined in each image using the algorithm presented in.¹⁵ These salient points (called SIFT points)

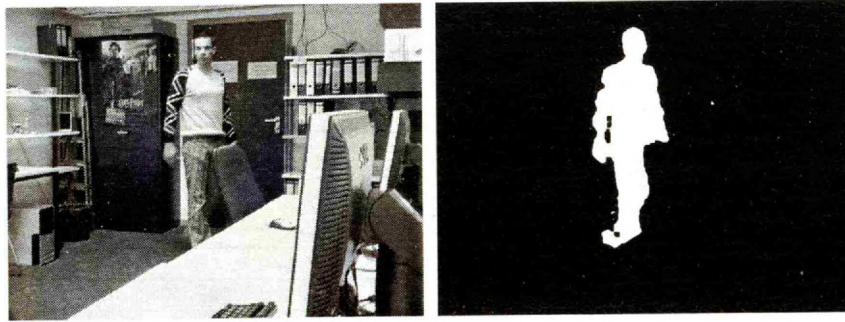


Figure 3. Result of the foreground segmentation: left the original video frame; right the frame mask (foreground is shown in white)

are robustly located in space, scale and orientation, thus making the representation invariant to changes in location, scale and orientation. Local features are then computed from the neighbourhood of each salient point. The radius of the neighbourhood is determined by the scale of the respective feature point, thus ensuring scale invariance of the neighbourhood description.

Two different types of local features are computed for person learning and recognition:

1. SIFT features introduced by Lowe¹⁵ and
2. colour histograms.

SIFT features are computed from histograms of gradients. The resulting feature description is a 128 dimensional vector, that is invariant to changes in orientation as well as partly invariant to illumination and affine/perspective transformations. Local RGB and HS colour histograms are computed for the determined local regions and normalized. In RGB colour space all three channels are combined into one histogram, whereas the HS histogram is build from the H and S channel of the HSV colour space. Those features are also rotation invariant. The dimension of RGB and HS feature vectors is 126 respectively 84.

5. MODEL COMPUTATION AND CLASSIFICATION

We model each instance of a person as a collection of parts, describing only the appearances of the parts while ignoring their relative spatial structure. With each part a visual word from a codebook is associated. A model vector is formed by creating the histogram of the visual word occurrences from the codebook in the image region of that person. Such a vector model is called a bag-of-visual words model and referenced as the document vector in text indexing.

The visual codebook, i.e. the visual vocabulary, is built by quantizing the local descriptors into so called visual words. Representative visual words are determined by clustering a random subset of part descriptors from all persons and people-free background images and videos into k clusters. Clustering is performed using the k -means clustering algorithm for each feature type separately, i.e. a visual vocabulary is build for each feature type separately. Each cluster C_i is represented by its mean vectors μ_i . The set of all visual words μ_i form the codebook for one feature type.

During the training phase, a model vector is computed for a frame from a training video by first performing foreground segmentation as described in Section 3.3. Then each foreground feature is assigned to its nearest visual word μ_i from the codebook followed by the computation of the histogram describing the co-occurrence of visual words in this image. The nearest visual word can be determined by calculating the $L1$ or $L2$ distance to the visual words μ_i . Another method is to compute also the covariance matrix for each cluster and use the Mahalanobis distance.

# persons	# clusters	# videos/person	# frames/video	feature type	reccgnition rate
2	100	2	800	RGB	75.71%
2	150	2	300	HS	70.00%
4	250	4	150	HS	67.14%
3	250	2	300	HS	77.14%
2	100	2	300	SIFT	61.42%

Table 2. Results obtained by the recognition system for different parameter configuration

Finally a person is represented by N model vectors derived from N randomly selected video frames of that person. These N model vectors are referred to as the model of a person. Persons' models are built independently for each feature type and the performances are compared in Section 6.

Our classification module is based on Nearest Neighbour search. Local features are first extracted from the incoming image and a co-occurrence vector of visual words is build as described in the previous paragraph. Note that foreground segmentation is non-applicable in our usage scenario, thus the resulting vector representation of the image describes the depicted individual as well as the background. In order to compute the vector representation we proceed as described above using the trained codebook.

In order to classify the image, the model vector with the minimum distance to the vector representation of the test image is determined. As mentioned, the co-occurrence vector of the test image contains visual words representing the pictured person(s) as well as features arising from background clutter. Thus we define the closest vector to the actual test vector as the model vector with maximum normalized histogram intersection.¹⁶ The normalized histogram intersection I between the test histogram/vector H_t and the model vector H_m is derived by:

$$I(H_t, H_m) = \frac{\sum_{j=1}^M \min(H_t^j, H_m^j)}{\sum_{j=1}^M H_m^j} \quad (1)$$

where M denotes the number of histogram bins in H_c and H_t , i.e. in our case the codebook size. Instead of Nearest Neighbour the K -Nearest Neighbour classification could have been used to boost classification performance results.

The described classification procedure does not account for images containing no (known) person or several known persons. The approach can be extended covering also these cases by thresholding the histogram intersection value I . If I is larger than the specified threshold, then the according person is pictured, otherwise no or unknown persons are pictured. Applying the approach to sub-windows of the whole image would definitely further improve the performance, while being computationally significantly more expensive.

In our experiments we neither have considered unknown or no person being present in an image (i.e. background only) nor have we considered multiple persons being depicted in the same image.

6. EXPERIMENTS

6.1. Experimental Setup

The camera array in our test room consists of six cheap web cams of type Philips SPC 600NC. Each two of those are connected to one PC using USB. The distributed camera network design is described in Section 3. In order to experimentally evaluate our proposed approach, videos of four different persons are captured over weeks at different times of the same day as well as at different days to capture sufficient variability in our training data (clothes, light effects, hair style, etc.). The captured videos are of variable length; each video consists of 1000 to 3000 frames. Captured video frames have a resolution of 640x480 pixels.



Figure 4. The four persons used for experimental evaluation



Figure 5. Examples of test images used for experimental evaluation

6.2. Experimental Evaluation

We used between two and four persons for our experiments and varied the number of training videos per person. All four persons used for the experimental evaluation are shown in Figure 3. The number N of training frames per person has been varied, as well as the size of the computed codebook, i.e. the number of visual words. Features were assigned to visual words by calculating the L1-norm and determining the mean vector μ_i with the minimum distance (see Section 5).

The test set consisted of 35 images per person. Test images have been taken between two weeks to five month after capturing the training videos. Persons were wearing similar clothes in test images and training sequences. Some example images of our test set are shown in Figure 4.

Results of our proposed approach for different parameter configurations are presented in Table 2. The reported recognition rates indicate that our system works well. It should be noted that due to the stochastic aspect in our model computation (we sample a number of representative histograms randomly from training videos), the recognition rates may differ between several runs, even though parameters remain unchanged.

Results indicate also that both colour features, local RGB and local HS histograms, perform almost equally well, whereas SIFT features show lower recognition rates. SIFT features are very distinctive and, as peoples' outer appearance changes significantly due to pose and expression variations, they may be too specific for our person recognition task.

Our recognition results can be improved by using k-Nearest Neighbour classification. One example experiment was set up using two persons and two videos per person, HSV histograms as basic features, 300 training frames per person and a codebook size of 200. Using Nearest Neighbour matching we obtained a resulting recognition rate of 90.00%, whereas using k-Nearest Neighbour matching for $k = 5$, the recognition rate increased to 95.71%.

The system fails if the background produced many features and thus the bag-of-visual words representation of the image is dominated by the background. An improvement of the recognition rate may be achieved in such cases by using a sliding window technique.

7. CONCLUSION

In this paper we have presented an approach for recognizing people in images by learning their appearance from image sequences. A visual sensor array is used to collect training sequences. Training data is labelled with minimal user effort. Moving persons are segmented and various features are extracted in order to represent a person's outer appearance by a bag-of-visual words model. Recognition of persons in new images is carried out using (k)-Nearest Neighbour(s) matching with normalized histogram intersection as the distance measure. Our experimental results show the practicability of our approach. For future work it is planned to use also texture features and combination of features to improve the recognition rates as well as including the geometric relations of parts in the model.

REFERENCES

1. L. Wiskott, J.-M. Fellous, N. Krüger, and C. von der Malsburg, "Face recognition by elastic bunch graph matching," in *Intelligent Biometric Techniques in Fingerprint and Face Recognition*, pp. 355–396, CRC Press, 1999.
2. M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal Cognitive Neuroscience* **3**(1), pp. 71–86, 1991.
3. A. P. B. Moghaddam, T. Jebara, "Bayesian face recognition," *Pattern Recognition* **33**(11), pp. 1771–1782, 2000.
4. B. Suh and B. Bederson, "Semi-automatic image annotation using event and torso identification," in *Tech Report HCIL-2004-15, University of Maryland, College Park, MD*, 2004.
5. L. Zhang, L. Chen, M. Li, and H. Zhang, "Automated annotation of human faces in family albums," in *Proceedings of the eleventh ACM international conference on Multimedia*, pp. 355–358, 2003.
6. C. Nakajima, M. Pontil, B. Heisele, and T. Poggio, "Full-body person recognition system.," *Pattern Recognition* **36**(9), pp. 1997–2006, 2003.
7. M. Hähnel, D. Klünder, and K.-F. Kraiss, "Color and texture features for person recognition," in *International Joint Conference on Neural Networks*, pp. 647–652–86, 2004.
8. A. M. Elgammal and L. S. Davis, "Probabilistic framework for segmenting people under occlusion.," in *ICCV*, pp. 145–152, 2001.
9. J. Sivic, C. L. Zitnick, and R. Szeliski, "Finding people in repeated shots of the same scene," in *Proceedings of the British Machine Vision Conference*, 2006.
10. "[http://intel.com/technology/upnp/.](http://intel.com/technology/upnp/),"
11. L. Li, W. Huang, I. Y. H. Gu, and Q. Tian, "Foreground object detection from videos containing complex background.," in *Proceedings of the eleventh ACM international conference on Multimedia*, pp. 2–10, 2003.
12. C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking.," in *CVPR*, pp. 2246–2252, 1999.
13. J. Lluís, X. Miralles, and O. Bastidas, "Reliable real-time foreground detection for video surveillance applications," in *VSSN '05: Proceedings of the third ACM international workshop on Video surveillance & sensor networks*, pp. 59–62, 2005.
14. R. Lienhart, "Vssn 2005 open source algorithm competition," in *VSSN '05: Proceedings of the third ACM international workshop on Video surveillance & sensor networks*, 2005.
15. D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. Journal Comput. Vision* **60**(2), pp. 91–110, 2004.
16. M. J. Swain and D. H. Ballard, "Color indexing," *International Journal of Computer Vision* **7**(1), pp. 11–32, 1991.