HAL
open science

Combining face detection and people tracking in video sequences

Etienne Corvee, François Bremond

To cite this version:

Etienne Corvee, François Bremond. Combining face detection and people tracking in video sequences. The 3rd International Conference on Imaging for Crime Detection and Prevention - ICDP09, Dec 2009, Kingston Upon Thames (London), United Kingdom. pp.1. inria-00502749

HAL Id: inria-00502749

https://inria.hal.science/inria-00502749v1

Submitted on 15 Jul 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Combining face detection and people tracking in video sequences

E. Corvee and F. Bremond

INRIA, Sophia Antipolis, France, *etienne.corvee, francois.bremond@sophia.inria.fr*

Keywords: face detection, people tracking, histogram of oriented gradients, K-mean clustering.

Abstract

Face detection algorithms are widely used in computer vision as they provide fast and reliable results depending on the application domain. A multi view approach is here presented to detect frontal and profile pose of people face using Histogram of Oriented Gradients, i.e. HOG, features. A K-mean clustering technique is used in a cascade of HOG feature classifiers to detect faces. The evaluation of the algorithm shows similar performance in terms of detection rate as state of the art algorithms. Moreover, unlike state of the art algorithms, our system can be quickly trained before detection is possible. Performance is considerably increased in terms of lower computational cost and lower false detection rate when combined with motion constraint given by moving objects in video sequences. The detected HOG features are integrated within a tracking framework and allow reliable face tracking results in several tested surveillance video sequences.

1 Introduction

A large variety of video applications require objects to be detected, recognised and tracked in a particular scene in order to extract semantic information about scene activity and human behaviour. In particular, most video surveillance applications rely on the detection of human activities captured by static cameras. In this domain, although cameras remain mostly fixed, many issues occur. For example, outdoor scenes can display varying lighting conditions (e.g. sunny/cloudy illumination, shadows), public spaces can be often crowded (e.g. subways, malls) and images can be obtained with a low resolution and can be highly compressed. Hence, detecting and tracking objects in such complex environment remains a delicate task to perform. Although the techniques presented in the state of the art of this domain show great results, their success is relative to the evaluation context.

In this paper, attention is focused in detecting and tracking faces in videos. Rather than to try to get better performances than state of the art face detection algorithms presented in section 2, the approach is designed to combine face detection with a tracking algorithm as illustrated in figure 1. Tracking solely frontal views of people faces means tracking moving objects coming toward the direction of the camera observing them. These trajectories show unidirectional patterns of people trajectory and occur in restricted scenarios such as in corridors. However, while a person moves across the camera field of view, both the person frontal and profile view of his face are likely to be captured.

The proposed face detection algorithm learns object features using a generic approach. Objects are represented by feature vectors in section 3 and a description on how different poses of facial features can be learned using a K-mean algorithm is given in section 4. Due to the large complexity occurring in grey level distribution of people face, a cascade of classifiers is designed in section 5 to speed up the detection process. In the proposed approach, the fact that most surveillance cameras are static and viewing static background scenes allows relatively fast segmentation of 2D moving objects where faces are assumed to occur, hence restricting the face searching process. Faces are tracked by integrating the detected features within a tracking framework presented in section 6. Results of detected and tracked faces are shown in section 7.

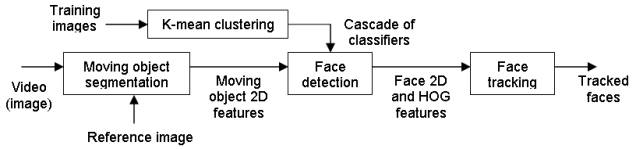

Figure 1. Overview of the face tracking algorithm structure.

2 State of the art

Face detection has been studied for many decades with a much higher interest this last decade since face detection and recognition algorithms are getting performant in terms of processing cost and can be used for a large variety of applications such as in security or multimedia applications. The algorithm developed by Viola and Jones [19] and distributed by the OpenCv library [9] is widely used for face detection. They use Haar features to represent faces and an Adaboost [21] algorithm to build a cascade of fast classifiers. Great performances are generally obtained when Haar features are combined with boosting techniques for the detection of faces [16, 10]. Haar features take advantage of the grey level differences between regions of the face. For example, Suguna *et al* [17] first quickly extract face candidates using the most significant face feature i.e. the eyes, and then use a SVM classifier on histogram equalised

candidates to determine if they really are faces or not. Many techniques exist as reviewed in the survey of Yang *et al* [11] with performances fairly comparable depending on the applications and their use. For example, McKenna *et al* [18] filter out false detections and increase the speed of their algorithm by detecting objects of interest in moving image regions. They use a Gabor Wavelet Transform to model features of faces and extract the eigen poses of faces captured under different rotation angles and many different lighting orientation with respect to the camera position.
Many face tracking algorithms take advantage of fast colour segmentation of the skin to build a fast face tracking algorithm. However, the majority of these techniques deal with only one class of skin colour [7, 8, 6]. Faces can be tracked using a Kalman filtering technique to model face trajectories [20, 4, 13] with application for example in the compressed video domain. For instance, in [20] people need to be facing the camera in a closed up view for the system to be able to extract eyes and mouth locations as tracking inputs.

Another important features used for object detection is provided by the calculation of Histogram of Oriented Gradients i.e. HOG. Pedestrians, faces and bicycles are successfully detected when represented by HOG [12, 1]. Such as with Haar based detectors, a boosting technique is also often used to model and rapidly detect objects [5] such as humans [15]. HOG features are extracted from selected areas of the image and compared to the trained models for object classification. HOG features can also be tracked independently without having to classify objects [1]. The detection of objects can be constrained with object motion information given, for example by optical flow of pedestrians [12].

3 Face representation

A face is here represented as a set of HOG feature vectors calculated over a set of $N_c = 9$ regular non overlapping cells. A HOG contains $N_b = 8$ bins covering the 360 degrees range of pixel gradient orientations estimated by a simple Sobel kernel operator. Hence, each bin i.e. each feature of a HOG represents $\Theta = 360/N_b$ and the visual signature of each cell of a face image can be represented by a N_b dimension vector. Each feature is normalised according to the edge magnitude response corresponding to the edge orientation of the feature. The integral image technique [19] is used for fast HOG computation.

4 Training

The training of face images is performed on a set of 2429 frontal and 428 profile views of face images referred to as the positive dataset and on a set of 4548 non face images referred to as the negative dataset. These datasets are provided by [3]. A K-mean algorithm extracts K clusters from the HOG features given by the positive dataset. The feature vector of a sample image indexed i containing N_c cells is expressed in equation 1 and the mean vector of the kth cluster is expressed in equation 2:

$$[\mathbf{h}_1^T(i), ..., \mathbf{h}_{N_c}^T(i)]^T \tag{1}$$

$$[\mathbf{m}_{1,k}^T, ..., \mathbf{m}_{N_c,k}^T]^T \tag{2}$$

An error term $E(i)$ is associated with each sample i defined as the minimum weighted squared magnitude of vector difference between the feature vector and its closest trained mean vector:

$$E(i) = \mathbf{min}_{k=[1:K]} \left(\frac{\sum_{c=1}^{N_c} w_c e_{c,k}(i)}{N_c \sum_{c=1}^{N_c} w_c} \right) \tag{3}$$

where $e_c(i)$ is the error of the sample feature i corresponding to the area of cell c calculated by:

$$e_{c,k}(i) = (\mathbf{h}_c(i) - \mathbf{m}_{c,k})^T (\mathbf{h}_c(i) - \mathbf{m}_{c,k}) \tag{4}$$

Expression $E(i)$ gives a face error from the combination of the HOG vectors given by all image cells. This global face coherency is weighted by a local clustering technique which aims to discriminate between cells which are likely to belong to a face feature and the ones that aren't. Thus, the weight w_c of a cell c tells how much a cell represents a positive sample cell. This weighting procedure is achieved by applying the K-mean algorithm exclusively on each cell feature vectors of the positive samples independently from the other cells. The weight w_c on cell x is defined as the minimum ratio, over the cell clusters indexed k, between the number of positive samples i.e. $\sum_{i=1}^{N} \delta_{c,k}(i)$ and the total number of samples i.e. $\sum_{i=1}^{N'} \delta_{c,k}(i)$ belonging to each cell cluster k:

$$w_c = \frac{\sum_{i=1}^{N} \delta_{c,k}(i)}{\sum_{i=1}^{N'} \delta_{c,k}(i)} \tag{5}$$

where $\delta_{c,k}(i)$ is to 1 when the cell c of sample i has for closest cluster the cluster indexed k or 0 otherwise. In the previous equation, N' represents the totality of the training samples. For more robust error estimation, each difference of HOG vector values (as illustrated in equation 4) is normalised by the standard deviation obtained along with each mean clustered vector values.

5 Cascade of classifiers and face detection

In order to detect faces given a large variability of faces in the training dataset, a cascade of classifiers is required to quickly and iteratively discriminate between face and non face as described in section 5.1. Faces are detected using the cascade in section 5.2.

5.1 Cascade of classifiers

People face appearance can greatly differ from one person to another. The variability of face images get even more complex when lighting exposure (e.g. half face shadowed) varies and when people do not face the camera (e.g. high camera location or people bending their head). Therefore face detection algorithms cannot easily handle this large variability and cannot discriminate between training negative and positive samples using a single thresholding operation. Cascade of classifiers are often used as in the works of [19] to deal with this

issue and is used as a boosting system such as the Adaboost system [21].

A cascade of classifiers is used to train the process described in section 4. Each classifier is associated with a threshold which is obtained from the error calculation given by equation 3 for the positive samples. The threshold is given by the maximum error among the 99% of the best (i.e. minimum) errors. All samples giving an error above this threshold are discarded from the next classifying stage where a new training is performed on the remaining samples. The discarded samples are assumed to be non-face samples.

Using the dataset of [3], 64 classifications are necessary for the system to converge when using 9 cells of 8 dimensions feature vectors (i.e. a face is represented by a 72 dimension vector) and for a 10 clusters K-mean algorithm. As example, 54% of the negative samples are discarded from the training dataset in the first iteration, and 10% in the second iteration. The system converges until the maximum number of negative samples is thresholded out. For comparison, about 250 clusters are needed without any use of cascade making the system computationally expensive. Therefore, for an image to be detected as a valid face candidate, it needs to pass all the 64 stages of the iterating process otherwise it is detected as a non-face candidate. The face detection algorithm can be formulated by equation 6 where an image indexed i is considered to be a face if $F(i) = N_s$ where $N_s = 100$ is the number of classifiers:

$$F(i) = \sum_{s=1}^{N_s} C(i, s) \tag{6}$$

The sth classifier $C(i, s)$ classifies image i as a face if its face error measured in equation 3 is below the classifier threshold $TH(s)$ (defined above):

$$C(i, s) = \begin{cases} 1 \text{ if } E(i) < TH(s) \\ 0 \text{ else} \end{cases} \tag{7}$$

5.2 Face detection

Faces present in an image are detected by a squared scanning window of 20x20 pixels. This relatively small window size allows small faces to be detected which often occur in video surveillance videos. The content of the image bounded by a scanning window represents the image to be detected as face or non face by the cascade of classifiers using equation 6. To avoid large computational cost of the search procedure and to avoid multiple overlapping results, the scanning window looks for faces every 5 pixels across the entire image. In order to detect faces with sizes bigger than the scanning window dimension, a hierarchical search is deployed where faces are searched in sub sampled versions of the original image: the image is successively sub sampled with a ratio of 15%.

To enhance the speed of the search process, the search is performed in areas which are pre-classified as moving regions (see figure 1). A foreground object detector performs this task by segmenting and grouping foreground pixels using a thresholding operation from a background reference image [anonymous]. Overlapping faces, i.e. the overlapping areas which have been classified as being face candidates (rescaled to the original image size scale) are filtered so that only the face candidate associated with the minimum face error (see equation 3) remains.

6 Face tracking

There exist many ways to track a set of feature vectors across successive images of a video. We here analyse how faces are tracked wihtin a tracking framework [anonymous]. This tracking algorithm builds a history of face candidates over the last 10 processed images. A set of possible paths is assigned to each last detected faces where a path represents a possible trajectory that a face undergoes across the last 10 images. Hence a mobile path consists of multiple sets of links between mobiles in the history.

A link between two mobiles is associated with a matching score between the two mobiles calculated from their 2D dimension similarities, their 2D distance closeness and their HOG feature vector difference. Therefore, two mobiles have a high probability to match each other if their dimensions are similar, if they are close to each other and if the magnitude of the two HOG vector difference is small. The matching probabilities given by the links forming a path are combined to give an overall probability factor to the path. The most probable path for each last detected face represents the trajectory of the face. Trajectories are then updated with the new trajectories obtained when new face objects are detected. This technique allows us to filter out noisy objects (due to false detection) associated with trajectory probability too small as shown in the next section.

7 Experimental results

The training face dataset provided by [3] is used to compare the obtained results with standard face detection techniques. The evaluation performed by Castrillon *et al* [10] shows that the best face detectors give a detection rate between approximately 58 to 73 % for less than 500 false detections for 40 processed images. The total processing time of these detectors does not exceed 45 seconds. For example, the Viola and Jones detector implementated in OpenCv [9] gives a true detection rate of 70% in 42 seconds for 40 test images with only 7 false detections while our face detection algorithm provided a detection rate of 61% with 376 false detections in 220 seconds. An example of detected faces at the many different resolutions in figure 2. Each bounding box colour represent a resolution. This example shows how noisy the system can be but alos that faces are associated with many overlapping detections. In this example, 100 % of the 7 faces are successfully detected with 15 false detections (i.e. noise). A simple thresholding operation on the number of overlapping bounding boxes could be applied to efficiently filter out false detections. For example, 86% true detections with 2 false detections can be obtained with a rule of 2 minimum overlapping bounding boxes as a face candidate. And 57% true detection without any noise can be obtained with at least 3 overlapping bounding boxes. However, in

this framework, we wish to keep the maximum true detection rate and to filter out noisy detections by temporal analysis and not with spatial overlapping rules. This explains why the Viola and Jones technique, which performs spatial filtering, obtains such a low false detection rate compared to our technique.

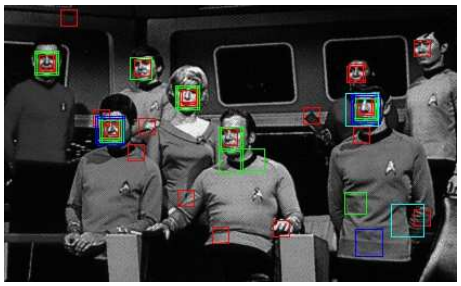

Figure 2. Multi resolution face detection.

Examples of detected faces are shown in figure 3 where a face represent at least one resolution detection. The two top and the bottom left images show frontal view of detected faces with false detections. The bottom right image of figure 3 shows results of profile views of faces obtained when adding 428 training samples of profile face views in the training dataset.

The proposed HOG face detection approach is tested against different scene scenarios as shown in figure 4. The top images show results obtained from a video captured in 2008 in the context of the TRECVID project [2] and the bottom images show results obtained with images captured in the context of the GERHOME project [14]. In both cases, the false detections shown on the left images are successfully rejected in the right figures when the face searching process is performed on moving regions represented by a gray bounding box.

For fair comparison, the algorithm results are compared with the results given by the Viola and Jones algorithm [16] for a sequence a 46 images each containing one frontal face view and no profile view. The algorithm obtains a score of 57% true positive detection rate with 5 false detections in total whereas the Viola and Jones approach obtains a score of 41% with 9 false detections. The tracking algorithm correctly tracks the person face using the detected face candidates (true positive and false positive) given by both detectors as inputs. However, one extra noisy trajectory is extracted using faces detected by the Viola and Jones approach (the HOG vector dissimilarity is not taken into account).

In terms of computational speed, the face detector processes each 720x576 pixels image in approximately 4 seconds which makes the algorithm useless for tracking faces in real time. However, when the detector focuses on moving regions, this cost falls below one second when the total searching area does not exceed 360x288 pixels. Assuming the camera is calibrated and the moving object 3D dimensions (width and height, the depth is discarded) fit the dimension of a 3D person model (e.g. 170 ± 30cm in height and 60 ± 20cm in width) then the search area can be constrained to the most probable area where face occur: e.g. 2/5 of the upper part of the 3D moving object as shown by the white rectangle in the right images of figure 4.

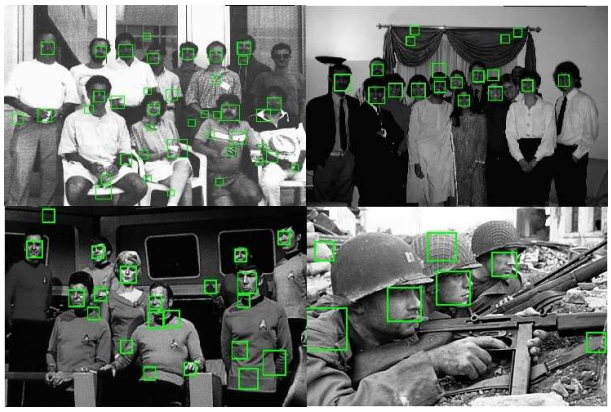

Figure 3. Examples of detected faces.

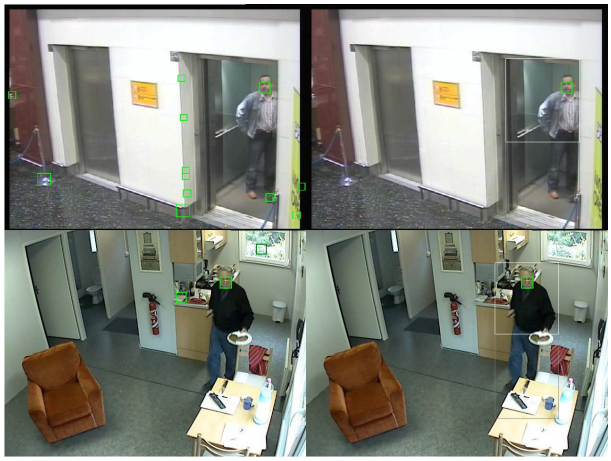

Figure 4. Examples of motion filtered faces.

The multiple view face tracking algorithm is evaluated for 200 captured images of a video captured in a lab with a framerate of 5 fps. The camera always sees the evolving individual in either its frontal or profile view. A detection rate of 82.5% is obtained with 29 false detections for the 200 frames. The majority of the mis-detected faces is observed to occur when the individual purposely tilted his head whose extracted HOG vectors are not trained for.

Using the moving regions as face searching areas reduces the number of false detection from 29 to 20. Each detected faces (including the falsely detected faces) are analysed by the tracking algorithm which extracts trajectories as shown in the top left image of figure 6 where the face is continuously tracked for 33 frames. Misdetection becomes an issue whenever it occurs for too long period of times where the tracking algorithm fails to find good matches in its temporal tracking window. This is illustrated in the top right figure where a new trajectory is assigned to the face after it undergoes significant head tilt for a too long period of time. Further examples of detected and tracked faces are given in the bottom images of figure 6: while a false detection occurs close to the face, the tracking algorithm prevents this false detection from being a valid face trajectory. The results show that the face is successfully tracked but with

two associated trajectories during the 220 frames with 14 occurrences of noisy trajectories due to false detections which have all been successfully filtered out.

A temporal analysis of the tracked face is performed in figure 5 where the mean deviation between two successive HOG vectors is plotted for 70 frames. The areas of the different activity patterns correspond to the following actions. A: the individual moves from right to left. His profile is detected. B: the individual quickly turns 90 degrees twice. The two error peaks correspond to the large difference in the detected pose, here from profile to frontal. The low error between the two peaks explains that the individual remained relatively static in front of the camera for a short period of time. C: the individual moves toward the camera from left to right; the face view is continuously detected as profile. D: the individual slowly and continuously turns his head. The face is detected as frontal before being detected back to profile. E: the individual moves from right to left of the camera in a closed up view. His face is detected as profile during this period.

8 Conclusion

A Histogram of Oriented Gradients (HOG) technique is designed to detect objects. The technique is here tested on the detection of multiple views of people face captured in video sequence. A standard clustering technique is deployed to model facial features. The evaluation of the algorithm with state of the art algorithms shows that similar performances are obtained with a limitation concerning the computational cost. This limitation is successfully dealt with when faces are constrained to be detected within moving regions of objects segmented from a reference background scene. Moreover, this motion constraint also allows false detection rate to be significantly reduced.

The presented works show a great interest in the temporal analysing of HOG feature vector that deserves more investigation in terms of tracking multiple views of objects. Hence, more objects shall be learned, for instance the combination of tracked several body shall provide relevant information of people behaviours in videos. The proposed face detection algorithm uses a multi resolution approach to detect different size faces. However, an alternative searching method can be proposed by investigating the detection in the 2D or in the 3D domain. For example, a 3D calibrated environment would use to 3D human height face dimension to constraint the 2D image areas to be searched for faces. In uncalibrated scenes, an offline face detection process can learn the detected face size to be searched with respect to the image plane location.

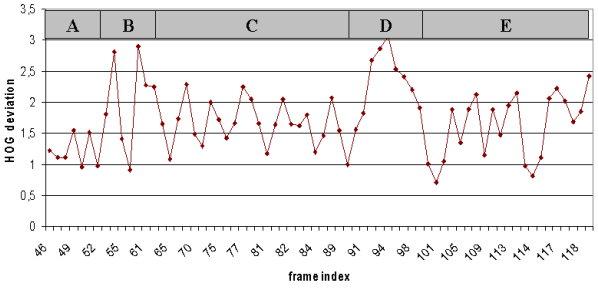

Figure 5. HOG deviation of a tracked face.

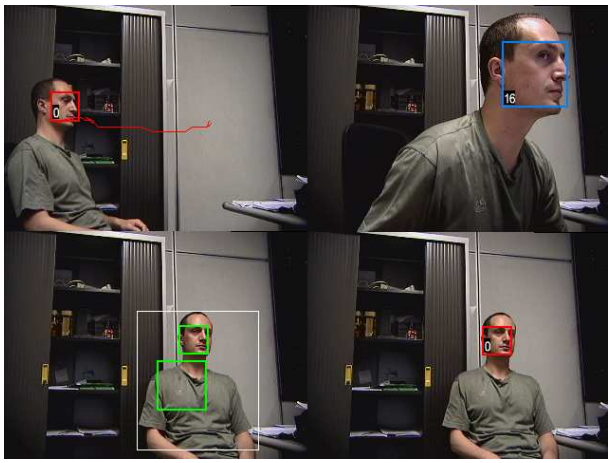

Figure 6. Examples of a tracked face.

References

[1] Adam A., Rivlin E., and Shimshoni I. Robust fragment-based tracking using integral histogram. In *Computer Vision and Pattern Recognition - CVPR*, 2006.

[2] Smeaton A.F., Over P., and Kraaij W. Evaluation campaigns and TRECVid,. In *MIR'06: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, 2006.

[3] provided by the face reserch group at CMU. CMU/VASC Image Database. Face Datasets. In .

[4] Harold H.W., Stone H.S., and Chang S-f. FaceTrack: Tracking and summarizing faces from compressed video. In *SPIE Photonics East, Conference on Multimedia Storage and Archiving Systems*, 1999.

[5] Laptev I. Improvements of object detection using boosted histograms. In *Proceedings of the British Machine Vision Conference*, 2006.

[6] Varona J., Buades J.M., and Perales F.J. Hands and face tracking for VR applications. *Computers and Graphics*, 29(2):179–187, 2005.

[7] Schwerdt K. and Crowley J.L. Robust Face Tracking using Color. In *4th IEEE International Conference on Automatic Face and Gesture Recognition*, 2000.

[8] Jordao L., Perrone M., Costeira J.P., and Santos-Victor J. Active Face and Feature Tracking. In *International Conference on Image Analysis and Processing - ICIAP*, 1999.

[9] Intel Open Source Computer Vision Library. . In .

[10] Castrillon Santana M., Deniz-Suarez O., Anton-Canalis L., and Lorenzo-Navarro J. Face and Facial Feature Detection Evaluation - Performance Evaluation of Public

Domain Haar Detectors for Face and Facial Feature Detection. In *International Conference on Computer Vision Theory and Applications - VISAPP*, Portugal, January 2008.

[11] Yang M-H. and Kriegman D.J. Narendra A. Detecting Faces in Images: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(1):34–58, 2002.

[12] Dalal N and Triggs B. Histograms of Oriented Gradients for Human Detection. In *Computer Vision and Pattern Recognition - CVPR*, 2005.

[13] Fonseca P.M. and Nesvadba J. Face Tracking in the Compressed Domain. *EURASIP Journal on Applied Signal Processing*, 2006.

[14] GERHOME project. Digital services enhaning independence of the elderly at home. In .

[15] Zhu Q., Avidan S., Yeh M-c., and Cheng K-t. Fast human detection using a cascade of histograms of oriented gradients. In *Computer Vision and Pattern Recognition - CVPR*, 2006.

[16] Lienhart R. and Maydt J. An Extended Set of Haar-Like Features for Rapid Object Detection. In *IEEE ICIP 2002*, 2002.

[17] Suguna R., Sudha N., and Sekhar C.C. A fast and efficient face detection technique using support vector machine,. *N.R. Pal et al. (Eds.): ICONIP 2004, LNCS 3316*, 2004.

[18] McKenna S.J., Gong S., and Collins J.J. Face Tracking and Pose Representation. . In *BMVC*, 1996.

[19] Jones M.J. Viola, P. Robust real-time face detection. *International Journal of Computer Vision*, 57(2):151–173, 2004.

[20] Vieux W.E., Schwerdt K., and Crowley J.L. Face-Tracking and Coding for Video Compression. In *Lecture Notes in Computer Science*, volume 1542, pages 151–161, 1999.

[21] Freund Y. and Schapire R.E. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 1997.