A HIERARCHICAL ASM/AAM APPROACH IN A STOCHASTIC FRAMEWORK FOR FULLY AUTOMATIC TRACKING AND RECOGNITION

Sascha Schreiber, Andre Störmer and Gerhard Rigoll

Institute for Human-Machine-Communication
Technische Universität München
{schreiber, stoermer, rigoll}@mmk.ei.tum.de

ABSTRACT

This paper deals with the fully automatic extraction of classifiable person features out of a video stream with challenging background. Basically the task can be split in two parts: Tracking the object and extracting distinctive features. In order to track a person, a system composed of an Active Shape Model embedded in a particle filter framework has been built. The output - a shape representing the position and the geometry of the human's head - serves as an initial guess for the following Active Appearance Model, which enables high precision matching of the head's texture. In this way raw features are transformed into appearance parameters, which finally can be used for a variety of classification tasks. The novelty of this framework is the hierarchical combination using the similarities of the models as well as exploiting their differences to enhance robustness and performance in complex scenarios.

Index Terms— Image processing, Tracking, Face recognition

1. INTRODUCTION

Due to a steady increase of computational power associated with a dramatic price decline of electronic devices, image video processing tasks enter and facilitate our everyday life. In this context a wide spectrum of applications has arisen, comprising topics like video surveillance, medical image processing as well as intelligent indoor spaces. For most of these applications numerous steps in a long chain of processing tasks like object localization, object tracking, feature extraction and classification have to be executed to derive the desired result. In the last two decades there has been a lot of progress in each of these steps in research and diverse algorithms have been developed, especially in the field of face and head tracking as well as face recognition. Basically all tracking approaches can be grouped by the fundamental cues these techniques are applying. One of the most popular cues for tracking heads and faces is probably the color information extracted from skin/hair regions [1], [2]. Other more high level approaches are based on template matching [3], facial features [4], contour analysis [5], optical flow [6] or exploit a combination of these features [7]. Also face recognition is a widely researched topic. The first popular approach has been a facial description using linear subspaces called Eigenfaces [8]. Later the Elastic Bunch Graph Matching [9] has risen much attention. It was able to detect and recognize faces within the same framework using gabor wavelets to encode facial features in a flexible graph. Active Appearance Models were used in [10] and it was suggested to solve different classification tasks using their parameters. There is a huge variety of different approaches in face recognition, a survey on these can be found in [11].

In this paper a system for a fully automatic tracking and recognition of a person's identity near real-time is presented. In Section 2 the fundamental procedure to localize and track a human is explained. Based on the tracked position an appearance based approach is used to generate a feature set. This set is the input to a classifier which is described in Section 3. Followed by Section 4 promising recognition results are presented and finally a summarization of our idea is given.

2. FEATURE EXTRACTION

2.1. Active Shape Tracking

As a first step it is explained how to track a person's head using a modified version of an active shape model (ASM). While in the standard ASM approach the gray values of the pixels are observed along the normal of the contour to detect trained histogram characteristics (as described in detail by [5]), our enhanced method is directly applied on the gradient image and thus is enabled to incorporate an additional feature - the direction of the edges in the image - to adapt the shape to the image data. Thus the iterative approach to improve the fit of the instance now looks like the following:

- Compute gradient image by applying a Sobel filter in x- and y-direction.
- Calculate the normal vectors at each landmark of the shape.

Proc. ICIP 2006, Atlanta, USA. IEEE

- Compute the angle between the normal vector and the gradient vector at each of the p pixel coordinates along the normal. Choose the pixel with the smallest corresponding angle as the new position for each landmark.

- Update the model parameters (position, scale, rotation) by least squares fitting to fit the new landmarks best.

- Repeat until the shape does not change significantly any more.

Compared to the standard ASM this approach does not aim to minimize the Mahalanobis distance between two samples (histogram characteristics) for each landmark, but tries to minimize the angle between two vectors by maximizing the dot product. A quality score θ for the complete shape can be obtained by summing the maximum dot product of each landmark. Using this technique our approach is enabled to determine new landmark positions corresponding to the orientation of the edges appearing in the image.

For the further stabilization of the tracked shape, the described method is embedded in a stochastic particle filtering framework called ICondensation [12, 13]. The idea behind this approach is to generate several different hypotheses representing the diverse shape appearances and modeling the probability distribution $\mathbf{w}_t$ for heads at each time step t. Based on the observations $\mathbf{z}_t$, representing the image features, the aim is to track the position of the persons throughout the posterior probability $p(\mathbf{w}_t|\mathbf{z}_{1:t})$. In most cases, there is no functional representation available for this conditional probability, but it can be derived iteratively by

$$p(\mathbf{w}_t|\mathbf{z}_{1:t}) \propto p(\mathbf{z}_t|\mathbf{w}_t) \int p(\mathbf{w}_t|\mathbf{w}_{t-1}) p(\mathbf{w}_{t-1}|\mathbf{z}_{1:t-1}) d\mathbf{w_{t-1}} \quad (1)$$

Updating the posterior distribution $p(\mathbf{w}_{t-1}|\mathbf{z}_{1:t-1})$ from the previous time step by prediction with dynamics $p(\mathbf{w}_t|\mathbf{w}_{t-1})$ leads to the effective prior $p(\mathbf{w}_t|\mathbf{z}_{1:t-1})$ for the actual time step. Finally multiplying the prior distribution with our measurement $p(\mathbf{z}_t|\mathbf{w}_t)$ derived from the quality score θ of the ASM results in the current state density $p(\mathbf{w}_t|\mathbf{z}_{1:t})$. Caused by the particle filter the precision requirements of the ASM detector can be reduced for the benefit of an optimized computation time. The usage of multiple hypotheses leads to a higher overall tracking performance than by using only a plain ASM structure.

2.2. Active Appearance Model based Feature Extraction

It has been shown, that Active Appearance Models (AAMs) are suited to model complex non-rigid objects like faces [5]. The major drawback of AAMs is the iterative matching algorithm which only works properly, if a good initialization near the optimum is given. In our approach this problem is solved by using the results of the ASM matching as initial guess for the AAM search algorithm. To create the AAM based on k labelled images (see Fig. 1) the following steps are done:

- Compute the main modes of variation of the shape $\mathbf{E}_s = (\mathbf{e}_{s,1}, \ldots, \mathbf{e}_{s,n})$ by applying Principal Component Analysis (PCA) to the shape describing vectors $(\mathbf{s}_1, \ldots, \mathbf{s}_k)$, which contain the landmarks.

- Warp the texture of all images within the training set to the mean shape, to derive shaped normalized textures.

- Compute the main modes of variation of the shape normalized texture $\mathbf{E}_t = (\mathbf{e}_{t,1}, \ldots, \mathbf{e}_{t,m})$ by applying PCA to the texture describing vectors $(\mathbf{t}_1, \ldots, \mathbf{t}_k)$.

With the given shape and texture eigenvectors $(\mathbf{E}_s, \mathbf{E}_t)$ each image I with known landmark positions $\mathbf{s}$ can be described by a parameter vector describing the shape projection $\mathbf{b}_s$ on the main shape modes and the projection $\mathbf{b}_t$ of the shape normalized texture on the main texture modes. Also with a given set of parameters $\mathbf{b}^T = (\mathbf{b}_s^T, \mathbf{b}_t^T)$ an approximation of each image I can be synthesized by firstly synthesizing the texture $\hat{\mathbf{t}}$ by a linear combination of the texture modes $\hat{\mathbf{t}} = \mathbf{E}_t^T \mathbf{b}_t + \overline{\mathbf{t}}$, synthesizing the shape $\hat{\mathbf{s}}$ by linear combination of the shape modes $\hat{\mathbf{s}} = \mathbf{E}_s^T \mathbf{b}_s + \overline{\mathbf{s}}$ and finally warping of the synthesized texture from the mean shape to the synthesized shape. $\overline{\mathbf{t}}$ denotes the mean texture and $\overline{\mathbf{s}}$ the mean shape, which are computed from the set of examples used to build the model.

An automatic search algorithm to estimate $\mathbf{b}$ is derived by computing a linear predictor based on examples of known displacement in shape and texture [14]. This means the parameters of the training examples are varied by known values (e.g. ± 0.5 standard derivations), shape and texture are synthesized by using the modified parameters. Then the residuals between the synthesized texture and the texture of the original image, warped from the synthesized to the mean shape, are computed for each training example and stored in matrix $\mathbf{R}$. The predictor $\mathbf{P}$, which estimates the parameter displacement, is computed by multivariate linear regression between the residuals in matrix $\mathbf{R}$ and the known parameter displacements which are stored in matrix $\Delta\mathbf{B}$.

$$\Delta\mathbf{B} = -\mathbf{P}\mathbf{R} \quad (2)$$

To compute the linear predictor $\mathbf{P}$ we compute the pseudoinverse and multiply it from the right side:

$$\mathbf{P} = -\Delta\mathbf{B}\mathbf{R}^T(\mathbf{R}\mathbf{R}^T)^{-1} \quad (3)$$

Using $\mathbf{P}$, an iterative matching routine works as follows:

- Compute the prediction of the parameter correction based on the residual $\mathbf{R}$ of the actual estimate and the given image.

- Apply parameter correction and synthesize new texture based on texture parameters as well as warp image texture from the the new shape position to the mean shape.

- Compute new residual; if energy of residual is minimized keep changes, else restore old estimate and change the step width of the parameter correction.

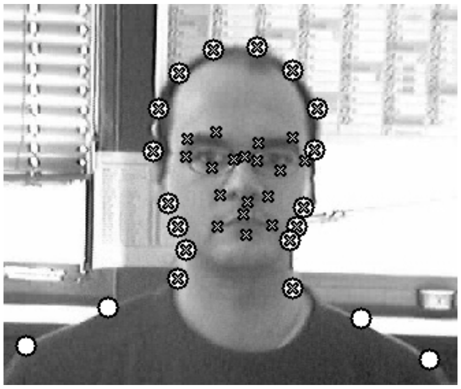

Fig. 1. Landmarks used for the models, circles denote landmarks used by ASM, crosses denote those used by the AAM. There is an overlap on a major part of the outline, these landmarks are used by both models.

- If a given set of step widths has been processed and no further improvements have been achieved declare convergence, else start the next iteration.

A detailed description on the model building and matching procedure can be found in [15]. A difference in our current work is that two different linear predictors are computed, one for a wide parameter displacement and another one for a small change in parameters. These two predictors are alternated to increase the robustness of the overall matching procedure.

2.3. Tied ASM and AAM Matching Scheme

The overall concept of the tied search method is to use the fast and robust ASM approach to localize and fit the object shape in the image data. Based on the result of the search a precise localization of the outline can be provided as an initial guess for the iterative AAM matching procedure. The Active Appearance Model is built up with a similar landmark set to the one used to generate the ASM. Lankmarks within the face, like corners of the eyes and mouth are added to this set while those on the shoulders are disregarded (see Figure 1).

The missing lankmarks can be estimated by projecting the subshape $\mathbf{s}_*$ given by the ASM on the shape eigenvectors of the AAM, omitting the elements referring to the coordinates of the unknown landmarks. This matrix of modified eigenvectors is denoted by E_*.

$$\mathbf{b}_s = \mathbf{E}_*^T \mathbf{s}_* \qquad (4)$$

The resulting weights $\mathbf{b}_s$ are used to resynthesize the shape, this time using the complete eigenvector matrix $\mathbf{E}_s$. With this an initial guess of the complete shape used by the AAM is derived. This principle works similar to the handling of the AAM search result to the ASM.

After the iterative matching of the AAM has finished, the quality of the result is measured by a normalized cross correlation between the original image frame and the resynthesized model. This leads to values between 0 and 1, the nearer to 1 the better the matching. Only for values above a threshold, the classification steps (as described in 3) are done and the AAM continues tracking. If the quality is below the threshold, the control is given back to the ASM which is able to track view independent. After a number of frames the AAM tests again if a better matching quality can be achieved.

This overall concept leads to a robust tracking combined with a high confidential classification, which is only done if the fitting results are good enough. In the office scenario this means, the approach is able to identify a person if the AAM fits well, else the approach is able to bridge this by shape tracking until a classifiable state is reached again.

3. NN-BASED CLASSIFICATION

The parameters $\mathbf{b}$ which are delivered by the Active Appearance Model are used as input for classification, if the above mentioned quality measurement is fulfilled. The identity of a person is classified by a multilayer perceptron (MLP). It is trained with the appearance parameters which were computed during the model building. A standard backpropagation algorithm is used to train the model. For our testings a MLP consisting of 168 input neurons, 85 neurons in the hidden layer and an output unit for every of the 25 different identities, has been created. The experiments have been executed with the Stuttgart Neural Network Simulator (SNNS) [16].

4. RESULTS

The results which were achieved are very promising. Videos of a typical office scenario with a challenging background were used to evaluate the performance. They showed persons sitting in front of a working station with a monitor mounted webcam. The persons should behave quite natural, i.e. they were allowed to rotate on the swivel chair, further increasing the difficulty of our scenario. In Fig. 2 some exemplary frames of one sequence are depicted. Results are drawn into the original images. In the graph shown in Fig. 3 the results of the experiments are depicted. As intended, the recognition rate increases if the threshold on the cross correlation quality measurement is higher, but the number of frames used for classification decreases. A recognition rate up to 100% is achieved if a very high quality threshold is used. If 15% of the frames are used for classification, a recognition rate of 92.5% is achieved. For practical applications this amount of frames used for classification is enough to obtain a continuous identification of persons visible in the video stream. The overall concept works near realtime.

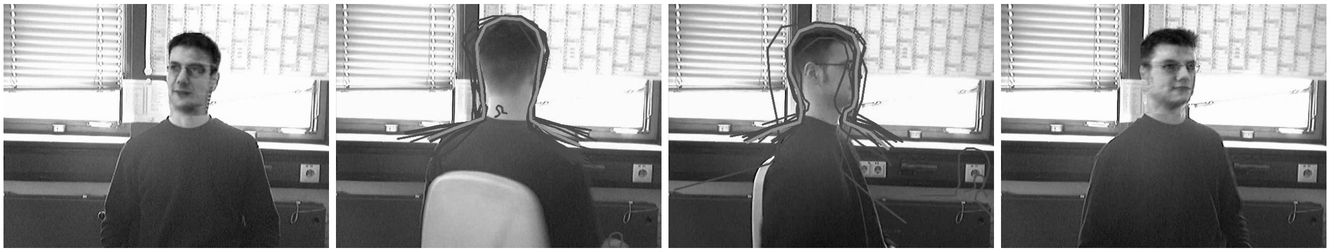

Fig. 2. A typical tracking output: In suited frames (left and right image) the AAM fitting reaches the quality threshold, the resynthesized estimates are drawn in; for the remaining frames tracking hypotheses are plotted in dark gray, the mean of the hypotheses is plotted in light gray.

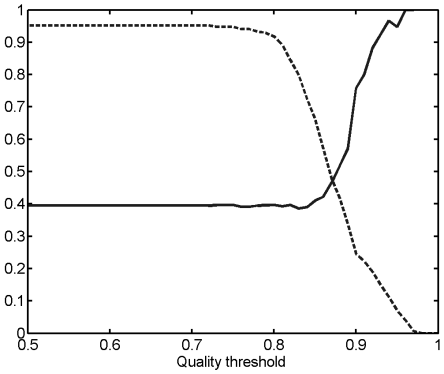

Fig. 3. Results on the office scenario database: The dotted line indicates the ratio of the classified frames to the overall number of frames in the sequence, the solid line shows the recognition rate (rank 1 Cumulative Matching Characteristics) of the identity.

5. CONCLUSIONS

In this paper a combination and improvement of known techniques is presented to make a step forward in the direction of fully automatic video processing. A robust tracking mechanism (Particle filtered gradient based ASM) is combined with a precise feature extraction method (AAM). Our approach is able to keep the track in difficult and complex scenarios and automatically uses suited frames for classification. This idea is demonstrated on the task of identifying tracked people. Future research will deal with the classification of other person properties like gender, emotion and age using the same framework, applying a refinement on the Point Distribution Models (ASM, AAM).

6. REFERENCES

[1] P. Fieguth and D. Terzopoulos, "Color based tracking of heads and other mobile objects at video frame rates," in *CVPR*, 1997.

[2] V. Vezhnevets, V. Sazonov, and A. Andreeva, "A survey on pixel-based skin color detection techniques," 2003.

[3] Y. Zhong, A. K. Jain, and M.-P. Dubuisson-Jolly, "Object tracking using deformable templates," in *ICCV*. 1998, p. 440, IEEE Computer Society.

[4] Y. Tian, K. Kanade, and J. Cohn, "Multi-state based facial feature tracking and detection," Tech. Rep., 1999.

[5] T. Cootes and C. Taylor, "Statistical models of appearance for computer vision," Tech. Rep., 2004.

[6] Y. Zhang and C. Kambhamettu, "3d head tracking under partial occlusion," in *Pattern Recognition*, 2002, vol. 35, pp. 1545–1557.

[7] S. Birchfield, "Elliptical head tracking using intensity gradients and color histograms," in *CVPR*, 1998.

[8] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3(1), pp. 71–86, 1991.

[9] L. Wiskott, J.-M. Fellous, N. Krüger, and C. von der Malsburg, "Face recognition by elastic bunch graph matching," in *CAIP*, 1997, pp. 456–463.

[10] G. Edwards, C. Taylor, and T. Cootes, "Interpreting face images using active appearance models," in *FG*, 1998, pp. 300 – 305.

[11] W. Zhao, R. Chellappa, A. Rosenfeld, and P. Phillips, "Face recognition: A literature survey," Tech. Rep., 2000.

[12] M. Isard and A. Blake, "Condensation – conditional density propagation for visual tracking," in *International Journal of Computer Vision*, 1998, vol. 29(1), pp. 5–28.

[13] M. Isard and A. Blake, "ICONDENSATION: Unifying low-level and high-level tracking in a stochastic framework," *Lecture Notes in Computer Science*, vol. 1406, pp. 893–908, 1998.

[14] T. Cootes, G. Edwards, and C. Taylor, "A comparative evaluation of active appearance model algorithms," in *BMVC*, 1998, vol. 2, pp. 680–689.

[15] A. Störmer and J. Stadermann, "Constructing faces using active appearance models and evaluating the similarity to the original image data," in *SOAVE*, 2004, pp. 47–56.

[16] A. Zell, N. Mache, R. Huebner, M. Schmalzl, T. Sommer, and T. Korb, "SNNS: Stuttgart neural network simulator," Tech. Rep., 1992.