CLEAR contribution (only tracker with face detector)

Igor Potúček[1]

[1] Brno University of Technology, Faculty of Information Technology, Božetěchova 2, Brno, 612 66, Czech Republic
potucek@fit.vutbr.cz
http://www.fit.vutbr.cz/~potucek/

1.1 Face detection

The tracking method suggested in [5] is based on the pure skin color segmentation, face detection and tracking which uses the movement prediction. The appearance of the skin-tone color depends on the lightning conditions. Hence we used normalized rg-color space, which is good solution for the problem of varying brightness. Normalized rg-color are computed from RGB values. The *r* and *g* components create 2D color space with normal probability distribution. Various face color pixels are picked manually and then is compute color class Ω_k. A color class Ω_k is determined by its mean vector μ_k and the covariance matrix K_k of its distribution. We need compute probability of each pixel in image by this equation:

$$p(c \mid \Omega_k) = \frac{1}{2\pi\sqrt{\det K_k}} \exp\left(-\frac{1}{2}(c-\mu_k)^T K_k^{-1}(c-\mu_k) \right) \qquad (1)$$

Skin color blobs are extracted by connected component analysis and morphological operations. We are keeping the information about objects for whole sequence of frames. The aim is to attach to an object in frame *t* its correspondent object in the frame *t+1*. The object tracking is based on the object correspondence determination. The information used for this purpose is only movement. We are limited to prediction of the next object position from its previous motion. The motion equations based on basic physics are used for estimating of a new position from previous object movement. Prediction of object position is therefore based only on the positions in past frames. Thus we can define a boundary in which the searched object will occur. The boundary we use is circular and is defined by specified radius. The problem can arise when some of tracked objects disappear – the predicted area is empty or appear on new position – the object is not in any predicted area. The first situation means the tracker termination and the second situation presents the new object in the scene so the new tracker is initialized. If more than one object is in the predicted area, then the nearest is the corresponding one. This kind of object correspondence keeps the consistence between detected objects in all frames.
The face detection algorithm is then applied only on the detected skin colored areas, which dramatically increase the speed of the whole algorithm. The corresponding object sequence, which contains some specified number of objects with detected faces, is labeled as object sequence representing a head. The criterion is presented by

minimal percentual number of detected faces in the sequence which is in our case 10%.
The face detection algorithm is based on the well known AdaBoost [7] learning algorithm. Viola and Jones [4] have used the AdaBoost to find a small set of rectangular features suitable for the classification of face and non-face images. The face classifier is constructed as a linear combination of several weak classifiers (e.g. a simple perceptron) built on features issued from the AdaBoost algorithm. In our case, the simple rectangle image/facial features are replaced by more complex Gabor wavelets [9] and a modified confidence-rated AdaBoost algorithm [8] is used for learning. Each weak classifier is composed of the Gabor wavelet and a decision tree whose output determines "confidence" that the input image is a face. The training algorithm considers fact that there is much more non-face regions then face regions in an image. Therefore, during the classification, the non-face regions are recognized and rejected faster.
Face detector is trained on normalized face images (24x24 pixels). The input image is sub-sampled and rotated first. Then, the face detection is performed, scanning sub-sampled images by the normalized window. After processing of the input image, all possible occurrences of faces are grouped by the help of a clustering algorithm. This helps to stabilize position of detected faces over a video sequence and it also improves overall detection rate. Such algorithm is able to detect faces rotated along the eye-view axis and partially rotated along the vertical/horizontal axis.
For the first training, the CBCL database from MIT was used. This data set contains about 1500 face and 14000 non-face images. Afterwards, hundreds of false positive and false negative samples, obtained from the testing data, were added to the training set.

1.2 KLT tracker

The further proposed method[6] is based on the public domain KLT feature tracker [1], which uses an image pyramid in combination with Newton-Raphson style minimisation to efficiently find a most likely position of features in a new image. We embedded both flocking behavior and color cue into our tracking system. We also designed methods based on progressive background model improvement. The model improvement is done through accumulation of RGB pixel values of current frame in model buffer. Only those pixels evaluated as background are updated. We use RG color model as color cue that could be either predefined or trained when tracker is placed on an object. The skin color detection approach is the same as in previous method. We use this model to discard all features whose color does not match expected object color. This color cue in combination with the flock compactness criterion almost eliminates feature drift of background and non-stationary objects in the scene. Tracker is also resistant to partial occlusions. Although normalized RG color representation is quite insensitive to light intensity changes, it is sensitive to chromatic changes. While using this color cue, this results in almost certain object loss when the tracked object is illuminated by chromatic light source (e.g. data projector). The object can be also lost as a result of large occlusion or unfortunate combination of background color, object rotation and motion. In our case it is necessary to detect these events. To detect the object loss we use the same trained object RG color model

as described earlier. Percentage of area beneath the tracker matching the color model is computed each frame. If the percentage drops below given threshold (e.g. 35 %) tracked object is considered lost. Further we compute sum of mean feature velocities over short history (e.g. 10 frames). If this velocity drops too low we assume tracker has drifted to background object and we also discard it. This leads to falsely discarding trackers over temporary stationary objects, but this isn't such a problem because lost objects are in most cases soon detected again.

The skin color analysis, background subtraction, and connected component analysis are used to extract suitable object for head detection. We use the presumption that faces correspond to compact ellipse-like shapes with distinctive axis aspect ratio in the mask. We use a method of statistical moments to find these components. The result of the head detection is set of a detected head centers. These points are used on higher level to initialize a KLT tracker. KLT tracker is able to track features only up to certain maximum displacement which is derived from the number of image pyramid levels, sub sampling between the levels and feature window size. Increasing the maximum distance results in higher computational cost and/or worse tracker performance. In order to increase maximum displacement we predict future tracker position based on current mean feature velocity and acceleration. This also provides little speed up due to the fact that the search for feature correspondence starts at more probable position resulting in less search iteration cycles at coarsest pyramid level.

References

1. Kölsch, M.,Turk, M., Fast 2D Hand Tracking With Flocks and Multi Cue Integration, Department of Computer Science, University of California, 2005.
2. A. Elgammal, D. Harrwood, L. Davis, Non-Parametric Model for Background Subtraction, European Conference on Computer Vision, 2000
3. J. Shi, C. Thomasi, Good Features to Track, IEEE Conference on Computer Vision and Pattern Recognition, 1994
4. Viola, J., Jones, M., Robust Real-time Object Detection, Technical Report 2001/01, Compaq CRL, February 2001.
5. Potucek, I., Sumec, S., Spanel, M., Participant activity detection by hands and face movement tracking in the meeting room, In: 2004 Computer Graphics International (CGI 2004), Los Alamitos, US, IEEE CS, 2004, s. 632-635, ISBN 0-7695-2717-1.
6. Hradis, M., Juranek, R., Real-time Tracking of Participants in Meeting Video, In: Proceedings of CESCG 2006, Wien, 2006.
7. Freund, Y., Schapire, R. E., A Short Introduction to Boosting. Journal of Japanese Society for Artificial Intelligence, 14(5):771-780, September, 1999.
8. Schapire, R. E., Singer, Y., Improved Boosting Algorithms Using Confidence-rated Predictions. Machine Learning, 37(3):297-336, 1999.
9. Kruger V., Gabor Wavelet Networks for Object Representation. Dissertation thessis, Technischen Fakultat, Christian-Albrechts-Universitat zu Kiel, 2000.

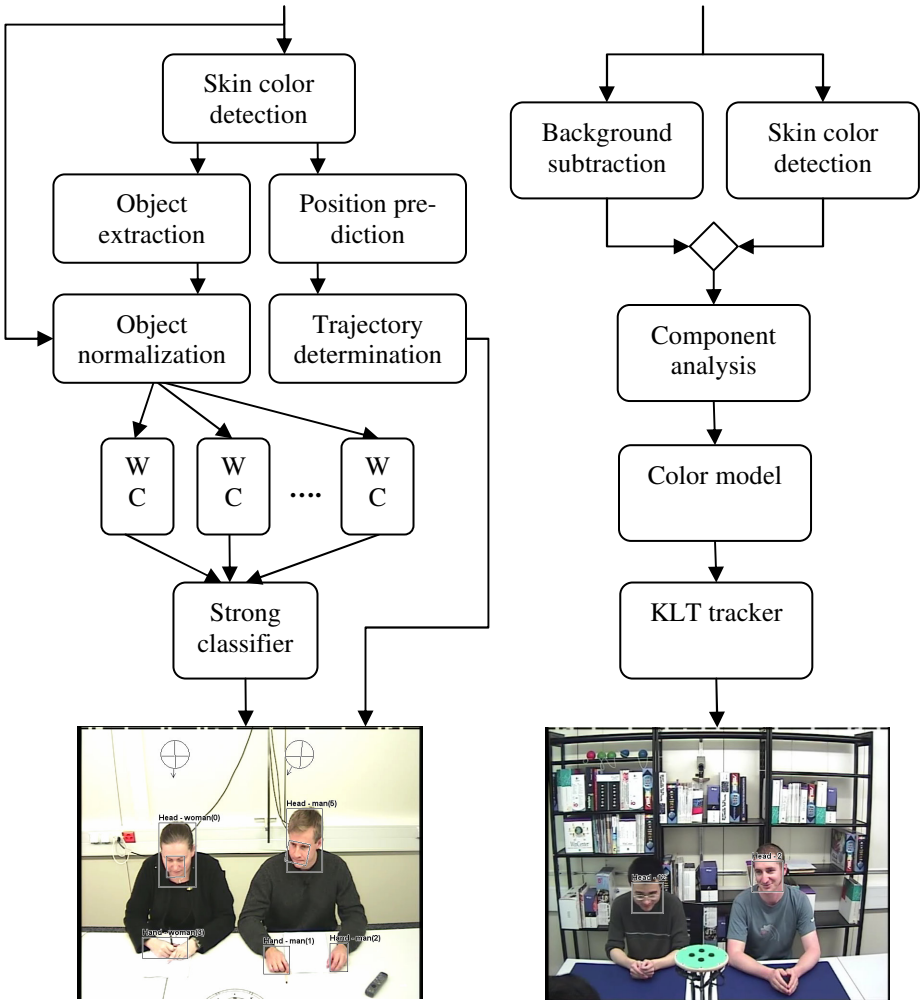

Fig. 1. a) Tracker with face detection system – consist of several weak classifiers (WC) working with Gabor wavelets and decision trees, b) Head tracking system using KLT tracker