CamShift-Based Tracking in Joint Color-Spatial Spaces

Bogdan Kwolek

Rzeszów University of Technology,
W. Pola 2, 35-959 Rzeszów, Poland
bkwolek@prz.rzeszow.pl

Abstract. This paper presents a visual tracking algorithm that is based on CamShift. Both the face and upper body are utilized simultaneously to perform tracking. They are first tracked independently by applying two separate CamShifts which continue tracking from the locations determined in the last time step and use only color probability images. Next, the candidate locations are subjected to CamShift which operates on distributions reflecting additionally geometrical relations between the face and the body. The aim of the CamShift-based searching in the joint color-spatial space is to find the mode. Experimental tracking results on meeting video recordings are presented. They demonstrate that this algorithm is superior over traditional CamShift. Furthermore, it is very simple and computationally fast.

1 Introduction

The goal of tracking is to establish a stable track for each object of interest in successive frames. It can be seen as a problem of assigning consistent identities to objects of interest. The tracking of people is very important component of many present and near-future applications of computer vision. A number of authors have previously considered the problem of tracking objects in video [1][2][3][8].

There are several computationally inexpensive visual techniques for face tracking. One of earliest attempts to track the face in live video sequences was made by Yang and Waibel [11]. They limited the number of CPU cycles needed for realization of efficient tracking by using color information to extract desirable skin-like regions. Bradski's CamShift is very interesting because it is very fast and requires minimal training. It can deal with irregular object motion arising due to perspective, uncalibrated lenses, image noise and so on. The major advantage of the algorithm is that it can work with cheap desktop cameras. The algorithm is representative of a group of algorithms that exploit the color cue to locate and subsequently track a human face in a video sequence. It is based on a robust non-parametric technique called Mean Shift to seek the nearest mode of probability distribution. A Mean Shift-based tracker by Comaniciu et al. [3] also exploits color distributions. The algorithm requires that the new target center remains within the kernel centered on the previous location of the

target. A relatively computationally inexpensive tracker of Birchfield [1] simultaneously utilizes a gradient-based elliptical outline fitted to the oval shape of the head and the color distribution enclosed. The algorithm operates through a deterministic searching in 3D space. The particle filter-based tracker [9] utilizes color information. The filter performs a random seeking guided by a probabilistic motion model to estimate the posterior probability density distributions of general non-linear and non-Gaussian systems. The algorithm uses a multi-part color modeling to take into account a rough spatial layout. The discussed work demonstrates that splitting of considered entity into two parts with specific color models improves tracking performance.

In this work we present a CamShift-based tracking algorithm. The face and upper body are utilized simultaneously to improve the tracking performance. They are first tracked independently by applying two separate CamShifts to final positions determined in the last time step. The candidate locations of rectangles with the interior color distributions most similar to distributions of the color models of face and body are determined. The final face location is then computed by CamShift acting on joint color-spatial distributions. The algorithm has been tested using the PETS-ICVS-03 meeting recordings.

The paper is organized as follows. The next section contains a review of the CamShift algorithm. In section 3. we describe our algorithm and present some tracking results which were obtained on meetings recordings. Some conclusions are drawn in the last section.

2 Object Tracking Using CamShift

The Continuously Adaptive Mean-Shift (CamShift) algorithm has been developed to perform efficient tracking of head and face in a perceptual user interface [2]. The algorithm is a generalization of the Mean Shift algorithm [5], which can only deal with static distributions. The Mean Shift algorithm provides a way to find the density modes without estimating the density. The CamShift is designed for dynamically changing distributions. The size and location of the probability distribution changes during tracking due to object movement, changing illumination conditions, viewing angle, shadows, etc. The algorithm uses color information to generate a probability distribution which is utilized to locate and then to subsequently track an object in a video sequence. It finds the mean (mode) of the distribution by iterating in the direction of maximum increase in probability density. The probability density is recomputed in each frame on the basis of the histogram back-projection [10][2]. Each pixel in the probability image represents a probability that the color of the considered pixel from an input image belongs to the object of interest. Spatial moments are used during iterations towards the mode of the distribution. This differs the CamShift algorithm from the conventional Mean Shift where the target and the candidate distributions are used to iterate towards the mode.

A variety of parametric and non-parametric statistical methods can be utilized to represent color distributions of homogeneous colored areas. The his-

togram is the oldest and most widely applied non-parametric density estimator. It is computed by counting the number of pixels in a region of interest that have a given color. The colors are quantized into bins. This operation allows similar color values to be clustered as single bin. The quantization into bins reduces the memory and computational requirements. The unweighted histogram is computed in the following manner:

$$q_u = \sum_{i=1}^{n} \delta[c(x_i) - u] \tag{1}$$

where the function $c : \Re^2 \rightarrow \{1, ..., m\}$ associates the value of pixel at location x_i to the bin number, n is the number of pixels, and δ is the Kronecker delta function. Due to their statistical nature color histograms can only reflect the content of images in a limited way [10]. Therefore such characterization of an object is tolerant to the noise. Histogram-based techniques are effective only when the number of bins can be kept relatively low and where sufficient data amounts are available.

The color distribution of an object represents a feature that is relatively stable under object rotation and scaling. It is also robust to partial occlusions while edge-based methods are ineffective. The major drawback with modeling the color distribution with histograms is the lack of convergence to the true density if the data set is small. In certain applications the color histograms are invariant to object translations and rotations. They vary slowly under change of angle of view and with the change in scale.

The original implementation of the CamShift algorithm uses the HSV color space [2]. A shadow cast does not change significantly the hue color component. Shadow decreases mainly the illumination component and changes the saturation component. Since the algorithm is intended to spend the lowest number of CPU cycles as possible, the color model is created by taking only a 1-D histogram of the hue component. This algorithm may fail to track the object when hue alone cannot be sufficient to distinguish the targets from the background.

The probability density image $P(x, y)$ is extracted on the basis of the histogram back-projection. This operation replaces the pixel values of the input image with the value of corresponding bin of the histogram. The value of each pixel in the probability image represents the probability that the pixel belongs to the object of interest. In order to provide the range of probability values between 0 and 255 the histogram bin values are linearly rescaled according to the following formula:

$$p_u = \min\left(\frac{255}{q_{max}} q_u, 255\right), \qquad u = 1, ..., m, \qquad q_{max} = \left\{\max\left(q_u\right)\right\}_{u=1}^{m}. \tag{2}$$

The mean location of the distribution within the search window is computed using moments [6][2]. It is given by:

$$x_1 = \frac{\sum_x \sum_y xP(x, y)}{\sum_x \sum_y P(x, y)}, \quad y_1 = \frac{\sum_x \sum_y yP(x, y)}{\sum_x \sum_y P(x, y)} \tag{3}$$

where x, y range over the search window. The eigenvalues (major length and width) of the probability distribution are calculated as follows [6][2]:

$$l = 0.707\sqrt{(a+c) + \sqrt{b^2 + (a-c)^2}},\ \ w = 0.707\sqrt{(a+c) - \sqrt{b^2 + (a-c)^2}} \quad (4)$$

where

$a = \frac{M_{20}}{M_{00}} - x_1^2,\ \ b = 2\frac{M_{11}}{M_{00}} - x_1 y_1,\ \ c = \frac{M_{02}}{M_{00}} - y_1^2,\ \ M_{00} = \sum_x \sum_y P(x,y),$

$M_{20} = \sum_x \sum_y x^2 P(x,y),\ \ M_{02} = \sum_x \sum_y y^2 P(x,y).$

The object orientation can be estimated as follows:

$$\theta = 0.5 * \arctan \frac{b}{a-c}. \quad (5)$$

The algorithm repeats the computation of the centroid and repositioning of the search window until the position difference converges to some predefined value, that is, changes less than some assumed value. Relying on the zero-th moment M_{00} the CamShift adjusts the search window size in the course of its operation. It requires the selection of the initial location and size of the search window. The algorithm outputs the position, dimensions, and orientation of an object undergoing tracking. It can be summarized in the following steps [2]:

1. Set the search window at the initial location (x_0, y_0).
2. Determine the mean location in the search window (x_1, y_1).
3. Center the search window at the mean location computed in Step 2, set the window size to zero-th moment M_{00}.
4. Repeat Steps 2 and 3 until convergence.

The CamShift algorithm has been tested using the PETS-ICVS-03 meeting recordings. For cameras 1 and 2 in scenario C there are maximum of 3 people sitting in front of each camera, see Fig. 1. The images of size 720x576 have been converted to size of 320x240 by subsampling (consisting in selecting odd pixels in only odd lines) and bicubic-based image scaling. The tracker has been initialized in frame #10949 with the number of bins m equals 30, Smin=10 and Vmin=10. In frames #11224, #11233, and #13669 we can observe how the window size is influenced by skin colored pixels from outside of the face. In frame #13670 the track was lost and the algorithm started tracking an other head which influenced the size and the location of the window.

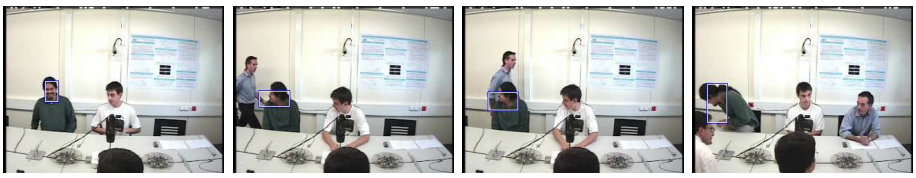

Fig. 1. Head tracking using CamShift. Frames #10969, #11224, #11233, #13669.

3 Tracking in Joint Color-Spatial Distributions

The tracking algorithm we present here follows the idea of person tracking through considering face-body relations, which has been presented in our previous paper [7]. The algorithm works by applying two probabilistic detectors of person's face and shirt colors. The probability images have been used to segment the candidates of person's face and shirt from the background. The ratio of areas, coordinates of gravity centers and geometrical relations between the labeled skin-like regions and shirt-like regions have been then used in extraction of the person from the background. The Kalman filter has been utilized to perform tracking the person within an image sequence.

In this work both the face and upper body are also utilized simultaneously to perform tracking. The face and body are first tracked independently by applying two separate CamShifts which continue tracking from the locations determined in the last time step. This operation finds the candidate locations of rectangles where the interior color distributions are most similar to distributions of the color models of face or body. A refined face location is then computed by CamShift which operates on distributions reflecting also geometrical relations between the face and the body.

Denote by $X_f = (x_f, y_f)$ and $X_b = (x_b, y_b)$ the position of the rectangles surrounding the face and body, respectively. The difference $X_f - X_b = (x_f - x_b, y_f - y_b) = (x_{fb}, y_{fb})$ reflects the configuration between face and body. The probability that $X_f - X_b$ represents the human H can be expressed by product of two Gaussians:

$$p\left(X_f - X_b \mid H\right) = G(x_f - x_b, \mu_x, \sigma_x) G(y_f - y_b, \mu_y, \sigma_y) \tag{6}$$

where μ_x, σ_x, μ_y, σ_y can be determined in advance from training samples.

Denote by $\rho(X_f)$ and $\rho(X_b)$ the similarity of the model color distributions of face and body to the candidate face or body color distributions, which are surrounded by rectangles at positions X_f and X_b, respectively. In order to compare two color distributions we need a metric of similarity or dissimilarity. In the discussed algorithm we have implemented the histogram intersection technique [10]. For a given pair of histograms I and M, each containing n values, the intersection of the histograms is defined as follows: $\rho = \sum_{i=1}^{N} \min(I_i, M_i)$. The terms I_i, M_i represent the number of pixels inside the i-th bin of the current and the model histogram, respectively, whereas N the total number of bins. The result of the intersection of two histograms is the number of pixels that have the same color in both histograms. To obtain a similarity measure with values between the zero and one the intersection has been normalized.

Bayes rule states that:

$$p\left(H \mid \rho(X_f), \rho(X_b), X_f - X_b\right) \propto p\left(\rho(X_f \mid H)\right) p\left(\rho(X_b \mid H)\right) p\left(X_f - X_b \mid H\right) p(H) \tag{7}$$

The best location corresponding to the local maximum of the probability can be obtained through a time-consuming search in the 4D space $S = (X_f, X_b) = (x_f, y_f, x_b, y_b)$:

$$S^* = \arg\max_{S_i \in S} p\left(\rho(X_f \mid H)\right) p\left(\rho(X_b \mid H)\right) p\left(X_f - X_b \mid H\right). \tag{8}$$

Instead of time consuming deterministic searching in the space S in order to find the extremum, we construct in each step a joint color-spatial distributions and apply the CamShift alternately to distributions in order to find two modes. The joint color-spatial distributions are created as the product of color probability images and Gaussian distributions reflecting the final face-body configuration in the last iteration. The aim of CamShift iterations is to find such locations of two 2D Gaussians in the color probability images, where two successive locations of the face or body mode in joint color-space distribution differ less than some predefined value.

Having the candidate face location $X_f = (x_f, y_f)$ we can extract the product of corresponding raw probability image of the body P_b and a 2D Gaussian $G(\mu_b, \Sigma_b)$, where $\mu_b = (x_f - x_{fb}, y_f - y_{fb})$, $\Sigma_b = \begin{pmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{pmatrix}$, x_{fb} and y_{fb} are determined by face-body configuration from the last iteration. Using such a modified body probability image we utilize CamShift in order to find the mode. Next, taking the location X_b corresponding to this mode we can extract the product of the raw probability image of the face P_f and a 2D Gaussian $G(\mu_f, \Sigma_f)$, where $\mu_f = (x_b + x_{fb}, y_b + y_{fb})$, $\Sigma_f = \begin{pmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{pmatrix}$. Finally, using the modified probability image of the face with joint color-spatial information we utilize CamShift to find the mode. At the end of each step we have a new candidate face location X_f which has been found by CamShift operating on joint color-spatial distributions. Using the raw probability images and the new face location X_f we repeat such recomputing of the raw probability images as well as CamShift-based searching until a distance between two successive face locations computed by CamShift converges to some predefined value.

Starting from the candidate body location $X_b = (x_b, y_b)$, which has been determined by one of the CamShifts working independently, a similar searching has been conducted. The upper images in Fig. 2. depict the locations of the face and body that were obtained with the searching initialized from X_f, whereas the images in the middle row show the locations which were obtained with the initialization at X_b. Having in disposal two face-body locations we computed the similarities of color distributions to the original face or body distributions and chosen the more similar face-body. The locations of face and body extracted in such a way estimate the locations S^* given by (8). The images which constitute the bottom row in the Fig. 2. demonstrate the locations of the rectangle surrounding the tracked face. The frames #11233 and #13669 demonstrate improved tracking capabilities of the proposed approach, see also Fig. 1.

To deal with situations where the evidence of one component of the face-body structure is weak or even missing, we generated additional Gaussian sub-distributions in the raw probability images. They have been constructed using information about the location of the corresponding face or body component as well as face-body geometrical configuration, which had been determined in the last frame. Thanks to such recovery parts in the distributions the algorithm can continue the tracking when the evidence of only one part of the face-body structure is relatively strong. Experiments demonstrated that such complementary

Fig. 2. Tracking using joint color-spatial distribution.

distributions improve also the overall performance of tracking. The mentioned above operation has been realized before the computations in the joint color-spatial distributions.

Figure 3. depicts the number of iterations which were needed for convergence in each time step. Typically, the average number of iterations in each call of CamShift is less than four. The picture (a) reported in Fig. 4. demonstrates the candidate rectangles on the image with maximal number of iterations, whereas the picture (c) shows the candidate rectangles on the last image in the sequence. The locations of the rectangles have been then refined using CamShift operating on joint color-spatial distributions (pictures b and d). The processing time is 100 msec on average for 320x240 images on an ordinary Pentium III-based PC.

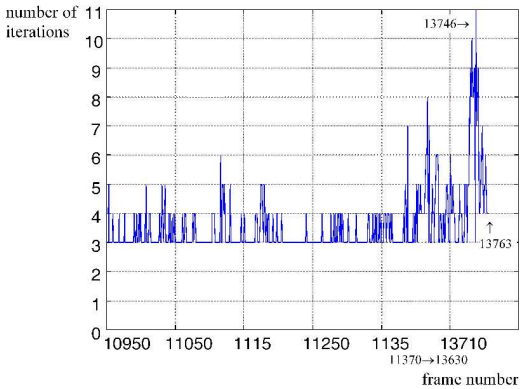

Fig. 3. Number of iterations versus frame number.

Fig. 4. Convergence of the algorithm. Frame #13746 (a), (b). Frame #13763 (c), (d).

4 Conclusion

The superiority of CamShift-based tracking using joint color-spatial distributions over the traditional CamShift tracking arises because the geometrical relations between face and body yield useful information. As a consequence we developed the modified CamShift tracking method. The method is computationally fast. Further improvements to the algorithm could be made through integrating the tracking algorithm with the background subtraction.

References

1. Birchfield, S.: Elliptical Head Tracking Using Intensity Gradients and Color Histograms, In Proc. of the IEEE Conf. on Comp. Vision and Pattern Recognition (1998) 232–237.
2. Bradski, G. R.: Computer Vision Face Tracking as a Component of a Perceptual User Interface, In Proc. of the IEEE Workshop on Applications of Comp. Vision, (1998) 214–219.
3. Comaniciu, D., Ramesh, V., Meer, P.: Real-Time Tracking of Non-Rigid Objects Using Mean Shift, In Proc. of the IEEE Conf. on Comp. Vision and Pattern Recognition (2000) 142–149.
4. Elgammal, A., Harwood, D., Davis, L.: Non-parametric Model for Background Subtraction, European Conf. on Computer Vision, vol. 2 (2000) 751–767.
5. Fukunaga, K.: Introduction to Statistical Pattern Recognition, sec. ed., Acad. Press (1990).
6. Horn, B. K. P.: Robot Vision, The MIT Press (1986).
7. Kwolek, B.: Color Vision Based Person Following with a Mobile Robot, In Proc. of the 3rd Int. Workshop on Robot Motion and Control (2002) 375–380.
8. Kwolek, B.: Stereovision-Based Head Tracking Using Color and Ellipse Fitting in a Particle Filter, 8th European Conf. on Computer Vision, LNCS, 3024 (2004) 192–204.
9. Perez, P., Hue, C., Vermaak, J., Gangnet, M.: Color-Based Probabilistic Tracking, European Conf. on Computer Vision (2002) 661–675.
10. Swain, M. J., Ballard, D. H.: Color Indexing, Int. Journal of Computer Vision, vol. 7, no. 1 (1991) 11–32.
11. Yang, J., Waibel, A.: A Real-Time Face Tracker. In Proc. of the IEEE Workshop on Applications of Comp. Vision (1996) 142–147.