SCALE AND SHAPE ADAPTIVE MEAN SHIFT OBJECT TRACKING IN VIDEO SEQUENCES

Katharina Quast and André Kaup

Chair of Multimedia Communications and Signal Processing, University of Erlangen-Nuremberg
Cauerstr. 7, 91058, Erlangen, Germany
{quast,kaup}@LNT.de

ABSTRACT

A new technique for object tracking based on the mean shift method is presented. Instead of using a symmetric kernel like in traditional mean shift tracking, the proposed tracking algorithm uses an asymmetric kernel which is retrieved from an object mask. During the mean shift iterations not only the new object position is located but also the kernel scale is altered according to the object scale, providing an initial adaption of the object shape. The final shape of the kernel is then obtained by segmenting the area inside and around the adapted kernel and distinguishing the object segments from the non-object segments. Thus, the object shape is tracked very well even if the object is performing out-of-plane rotations.

1. INTRODUCTION

Object tracking is an important and challenging task in multimedia technologies. A lot of research has been performed on this topic inducing numerous methods for object tracking. One of the most common and well-known tracking techniques is the mean shift algorithm because of its ease of implementation, computational speed, and robust tracking performance. Mean shift is a nonparametric statistical method which iteratively shifts each data point to the average of data points in its neighborhood [1]. It has been applied to several computer vision tasks such as segmentation [2] and object tracking [3, 4]. In spite of its advantages traditional mean shift has two main drawbacks. The first problem is the fixed scale of the kernel or the constant kernel bandwith. In order to achieve a reliable tracking result of an object with changing size an adaptive kernel scale is necessary. The second drawback is the use of a radial symmetric kernel. Since most objects are of anisotropic shapes a symmetric kernel with its isotropic shape is not a good representation of the object shape. In fact if not specially treated, the symmetric kernel shape may lead to an inclusion of background information into the target model, which can cause even tracking failures.

An intuitive approach of solving the first problem is to run the algorithm with three different kernel bandwiths, former bandwith and former bandwith $\pm 10\%$, and to choose the kernel bandwith which maximizes the appearance similarity ($\pm 10\%$ method) [5]. A more sophisticated method using difference of Gaussian mean shift kernel in scale space has been proposed in [6]. The method provides good tracking results, but is computationally very expensive. And both methods are not able to adapt to the orientation or the shape of the object.

Mean shift based methods which are not only adapting the kernel scale but also the orientation of the kernel are presented in [4, 7, 8]. Scale and orientation of a kernel can be obtained by estimating the second order moments of the object silhouette, but that is of high compuational costs. In [7] mean shift is combined with adaptive filtering to obtain kernel scale and orientation. The estimation of kernel scale and orientation are good but since a symmetric kernel is used no adaption to the actual object shape can be performed. Therefore, in [8] asymmetric kernels are generated using implicit level set functions. Since the search space is extended by a scale and an orientation dimension the method simultaneous estimates the new object location, scale and orientation. However the method can only estimate the objects orientation for in-plane rotations. In case of 3D or in-depth rotations none of the mentioned algorithms is able to adapt to the objects orientation and therewith to the object shape.

Therefore, we propose a mean shift based tracking method which is able to adapt to the object's shape. Our method uses asymmetric kernels which are first obtained from an object mask and are fitted to the object shape through a scale adaption followed by a segmentation process. Thus, a good fit of the object shape is retrieved even if the object is performing a rotation in 3D space.

The rest of the paper is organized as followed. An overview of the mean shift tracking is given in Section 2. In Section 3 the proposed method is described explaining the construction of the object shaped kernel, the execution of the mean shift iterations in the spatial-scale-space, and the final estimation of the kernel shape using a segmentation process. The experiments of the tracking algorithm are then described and results are shown in Section 4. Finally conclusions are drawn in Section 5.

2. MEAN SHIFT TRACKING OVERVIEW

Mean shift tracking discriminates between a target model in frame n and a candidate model in frame $n+1$. For tracking purposes the target model is defined as the color density distribution of the object. The target model is estimated from the discrete density of the objects color histogram $\mathbf{q}(\hat{\mathbf{x}}) = \{q_u(\hat{\mathbf{x}})\}_{u=1\ldots m}$ (whereas $\sum_{u=1}^{m} q_u(\hat{\mathbf{x}}) = 1$).

The probability of a certain color belonging to the object with the centroid $\hat{\mathbf{x}}$ can be expressed as the probability of the feature $u = 1\ldots m$ occuring in the target model. Which is

$$q_u = C \sum_{i=1}^{N} k\big(\| \frac{\mathbf{x}_i - \hat{\mathbf{x}}}{h} \|^2 \big) \delta[b(\mathbf{x}_i) - u] \qquad (1)$$

where δ is the impulse function, h is the kernel bandwith, N is the number of pixels of the target model and normalization constant C is the reciprocal of the sum of values of the kernel function $k(z)$. The kernel K with kernel function $k(z)$ makes

the density estimation more reliable because it provides pixels farther away from the center of the ellipse with smaller weight. Hence the least reliable outer pixels don't influence the density estimation to much.

The candidate model $\mathbf{p}(\hat{\mathbf{x}}_{new}) = \{p_u(\hat{\mathbf{x}}_{new})\}_{u=1\ldots m}$ (whereas $\sum_{u=1}^{m} p_u = 1$) in the following frame and the probability of a certain color appearing in the candidate model

$$p_u(\hat{\mathbf{x}}_{new}) = C\sum_{i=1}^{N} k\left(\left\|\frac{\mathbf{x}_i - \hat{\mathbf{x}}_{new}}{h}\right\|^2\right)\delta[b(\mathbf{x}_i) - u] \quad (2)$$

are defined similarly.

The core of the mean shift method is the computation of the offset from an old object position $\hat{\mathbf{x}}$ to a new position $\hat{\mathbf{x}}_{new} = \hat{\mathbf{x}} + \Delta\mathbf{x}$ by estimating the mean shift vector

$$\Delta\mathbf{x} = \frac{\sum_i K(\mathbf{x}_i - \hat{\mathbf{x}})\omega(\mathbf{x}_i)(\mathbf{x}_i - \hat{\mathbf{x}})}{\sum_i K(\mathbf{x}_i - \hat{\mathbf{x}})\omega(\mathbf{x}_i)} \quad (3)$$

where $\omega(\mathbf{x}_i)$ is the weight of $\mathbf{x}_i$ which is defined as

$$\omega(\mathbf{x}_i) = \sum_{u=1}^{m} \delta[b(\mathbf{x}_i) - u]\sqrt{\frac{q_u(\hat{\mathbf{x}})}{p_u(\hat{\mathbf{x}}_{new})}}. \quad (4)$$

In detail, the problem of localizing the candidate model in the next frame $n+1$ is formulated as the derivation of the estimate that maximizes the Bayes error between the reference distribution of the target model and the distribution of the candidate model. For the similarity measure the discrete formulation of the Bhattacharya coefficient is choosen since we have discrete color distributions on the one hand and the Bhattacharya coefficient is nearly optimal and imposes a metric structure on the other hand. The Bhattacharya coefficient and the distance between the two color distributions of target and candidate model are defined as follows

$$\rho[\mathbf{p}(\hat{\mathbf{x}}_{new}), \mathbf{q}(\hat{\mathbf{x}})] = \sum_{u=1}^{m} \sqrt{p_u(\hat{\mathbf{x}}_{new})q_u(\hat{\mathbf{x}})} \quad (5)$$

$$d(\hat{\mathbf{x}}_{new}) = \sqrt{1 - \rho[\mathbf{p}(\hat{\mathbf{x}}_{new}), \mathbf{q}(\hat{\mathbf{x}})]}. \quad (6)$$

The aim is to minimize the distance (6) as a function of $\hat{\mathbf{x}}_{new}$ in the neighborhood of a given position $\hat{\mathbf{x}}_0$ by using the mean shift algorithm. Starting with the Taylor expansion around $p_u(\hat{\mathbf{x}}_0)$ the Bhattacharya coefficient is approximated as

$$\rho[\hat{\mathbf{p}}(\hat{\mathbf{x}}_{new}), \hat{\mathbf{q}}(\hat{\mathbf{x}})] \approx \frac{1}{2}\sum_{u=1}^{m}\sqrt{p_u(\hat{\mathbf{x}}_0)q_u(\hat{\mathbf{x}})} + \frac{C}{2}\sum_{i=1}^{N}\omega(\mathbf{x}_i)k\left(\left\|\frac{\mathbf{x}_i - \hat{\mathbf{x}}_{new}}{h}\right\|^2\right) \quad (7)$$

In equation (7) only the second term is dependent on $\hat{\mathbf{x}}_{new}$. Hence, for minimizing the distance it is sufficient to maximize the second term of (7). This term corresponds to the density estimate computed with kernel profile k at location $\hat{\mathbf{x}}_{new}$ in frame $n+1$, whereas the data is weighted with $\omega(\mathbf{x}_i)$. The maximization can be achieved using the mean shift algorithm. By running this algorithm the kernel is recursively moved from $\hat{\mathbf{x}}_0$ to $\hat{\mathbf{x}}_1$ according to the mean shift vector.

3. SHAPE ADAPTIVE MEAN SHIFT TRACKING

3.1 Asymmetric kernel selection

Traditional mean shift tracking is working with a symmetric kernel. But an object shape can not be described properly by a symmetric kernel. Therefore, the use of isotropic or symmetric kernels will always cause an influence of background information on the target model, which can even lead to tracking errors. To overcome these difficulties we are using an asymmetric and anisotropic kernel.

As the mean shift tracker cannot initialize the object by itself, it either requires some user input or the result from a detection process which provides an object mask like [9]. Based on such an object mask our asymmetric kernel is constructed by estimating for each pixel inside the mask $\mathbf{x}_i = (x,y)$ its normalized distance to the object boundary: $K(\mathbf{x}_i) = \mathbf{x}_i\_distance\_from\_boundary / max\_distance\_from\_boundary$, where the distance from boundary is estimated using morphological operations. In Figure 1 an object, its mask and the mask based asymmetric kernel are shown.

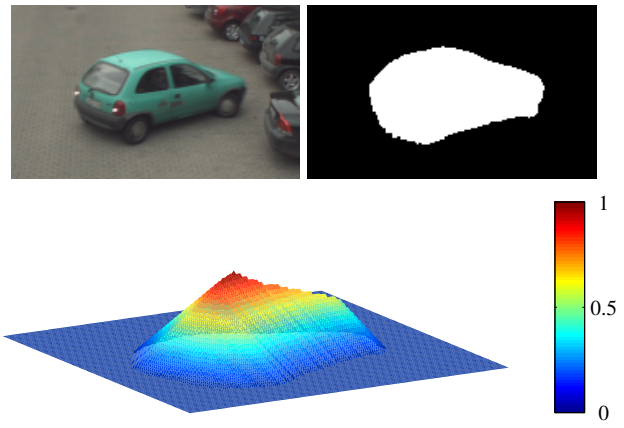

Figure 1: Object in image (top left), object mask (top right) and asymmetric object kernel retrieved from object mask (bottom)

3.2 Definition of the Scale Dimension

To adapt the scale of the kernel using the mean shift iterations, a scale dimension needs to be added to the search space. Instead of running the algorithm only in the local space the mean shift iterations are performed in the extended search space $\Omega = (x,y,\sigma)$ consisting of the image coordinates (x,y) and a scale dimension σ. Thus, the object's changes in position and scale can be evaluated through the mean shift iterations simultaneously.

Given that an object can be represented by a closed curve, the image coordinates of an object pixel $\mathbf{x}_i$ can be easily transformed to the scale dimension

$$\sigma_i = \frac{\delta(\mathbf{x}_i)}{r(\theta_i)} = \frac{\|\mathbf{x}_i - \hat{\mathbf{x}}\|}{r(\theta_i)} \quad (8)$$

where $\delta(\mathbf{x}_i)$ is the distance between an object pixel $\mathbf{x}_i$ and the object centroid $\hat{\mathbf{x}}$, $r(\theta_i)$ the kernel bandwith at angle θ_i and σ_i the scale of the object pixel.

An important property when running the mean shift iteration in the spatial-scale-space is the constancy of the scale mean. This property means that the two sums of pixel scales on both sides of the scale mean are equal:

$$\sigma_i = \int_0^{2\pi} \int_0^{\hat{\sigma} r(\alpha)} \frac{\delta}{r(\alpha)} d\delta d\alpha = \int_0^{2\pi} \int_{\hat{\sigma} r(\alpha)}^{r(\alpha)} \frac{\delta}{r(\alpha)} d\delta d\alpha \quad (9)$$

After integrating the equation $2\hat{\sigma}^2 - 1 = 0$ is obtained and can be rearranged to determine the sample mean as $\hat{\sigma} = \frac{1}{\sqrt{2}}$. Thus, the scale mean is a constant and therewith independent from the objects shape. Using the mean shift iterations a scale update is estimated and the new scale is set to $\hat{\sigma} + \Delta\sigma$. To take advantage of the scale update a relation between the scale and the bandwith has to be considered. In [8] this relation is defined over the bandwith update factor $d = 1 + \sqrt{(2)}\Delta\sigma$ which is used to compute the new bandwith $r_{new}(\alpha) = dr(\alpha)$. Given the new scale the bandwith update factor d and therewith the new bandwith can be calculated.

3.3 Mean Shift Tracking in Spatial-Scale-Space

To run the mean shift iterations in the joint search space a 3D kernel consisting of the product of the spatial object based kernel from Section 3.1 and a kernel for the scale dimension

$$K(x, y, \sigma_i) = K(x, y)K(\sigma) \quad (10)$$

is defined. The kernel for the scale dimension is a 1D Epanechnikov kernel with the kernel profile $k(z) = 1 - |z|$ if $|z| < 1$ and 0 otherwise, where $z = (\sigma_i - \hat{\sigma})/h_\sigma$. The mean shift vector given in equation 3 can now be computed in the joint space as

$$\Delta\Omega = \frac{\sum_i K(\Omega_i - \hat{\Omega})\omega(\mathbf{x}_i)(\Omega_i - \hat{\Omega})}{\sum_i K(\Omega_i - \hat{\Omega})\omega(\mathbf{x}_i)} \quad (11)$$

with $\Delta\Omega = (\Delta x, \Delta y, \Delta\sigma)$.

Given the object mask for the initial frame the object centroid $\hat{\mathbf{x}}$ and the target model are computed. To make the target model more robust the histogram of a specified neighborhood of the object is also estimated and bins of the neighborhood histogram are set to zero in the target histogram to eliminate the influence of colors which are contained in the object as well as in the background. In case of an object mask with a slightly different shape than the object shape too many object colors might be supressed in the target model, if the direct neighbored pixels are considered. Therefore, the directly neighbored pixels are not included in the considered neighborhood.

Taking the distribution $\{q_u(\hat{\mathbf{x}})\}_{u=1\ldots m}$ of the target model at location $\hat{\mathbf{x}}$ in frame n the algorithm iterates as follows:

1. Initialize the location of the candidate model in frame $n+1$ with $\hat{\mathbf{x}}_0 = \hat{\mathbf{x}}$ and set $d_0 = 1$.
2. Subsequently compute the distribution $\mathbf{p}(\hat{\mathbf{x}}_0) = \{p_u(\hat{\mathbf{x}}_0)\}_{u=1\ldots m}$ and $\rho[\mathbf{p}(\hat{\mathbf{x}}_0), \mathbf{q}(\hat{\mathbf{x}})] = \sum_{u=1}^{m} \sqrt{\hat{p}_u(\hat{\mathbf{x}}_0)\hat{q}_u(\hat{\mathbf{x}})}$.
3. Compute the weights $\omega(\mathbf{x}_i)$ according to equation (4).
4. According to the mean shift vector (11) estimate
   - the new position of the candidate model $\hat{\mathbf{x}}_1 = \hat{\mathbf{x}}_0 + \Delta\mathbf{x}$
   - the bandwith update factor $d_1 = d_0(1 + \sqrt{(2)}\Delta\sigma)$
   - $\{p_u(\hat{\mathbf{x}}_1)_{u=1\ldots m}\}$
   - $\rho[\mathbf{p}(\hat{\mathbf{x}}_1), \mathbf{q}(\hat{\mathbf{x}})]$.
5. If $\|\hat{\mathbf{x}}_1 - \hat{\mathbf{x}}_0\| < \varepsilon$ stop, else $\hat{\mathbf{x}}_0 \leftarrow \hat{\mathbf{x}}_1$, $d_0 \leftarrow d_1$ and go to step 2.

The algorithm uses the mean shift vector in step 4 to maximize the Bhattacharya coefficient. The termination threshhold ε in step 5 implies that the vectors $\hat{\mathbf{x}}_0$ and $\hat{\mathbf{x}}_1$ point at the same pixel in image coordinates. Therefore, the algorithm terminates for one thing if the same or a larger value for the Bhattacharya coefficient is found and for the other thing if the candidate model doesn't change its position in two subsequent iterations.

3.4 Final shape estimation

After the mean shift iterations have been converged the final shape of the object is evaluated from the first estimate of the scaled object shape. Therefore, the image is segmented. Segmentation is done using the mean shift method according to [2]. For each segment inside and in the close neighborhood of the found object we have to decide if it still belongs to the object shape or to the background. Segments which are fully included in the mask are assigned as object segments. For each segment being only partly included in the mask its color histogram is compared to the target model. If at least 50% of the segments color are existing in the target model the segment is assigned as an object segment, otherwise the segment is considered to belong to the background.

These decisions work well for segments being either totally included in the initial mask or containing object colors which are not eliminated from the target model to make the mean shift tracking more robust to similar background colors. But possible object segments which are more than 50% included in the initial mask and contain a lot of object color information which has been eliminated from the target model are neglected by these decisions.

In order to avoid a loss of these object segments a geometric constraint is also considered. If more than 50% of a segment area is included in the initial mask the segment is assigned as an object segment as well. In Figure 2 (middle) the three different types of object segments are shown: segments which are completely included in the initial mask are shown in blue, segments which are partly included and are containing color information of the target model are marked in yellow and the green segments are the ones assigned as object segments using the geometric constraint. Red segments are background segments.

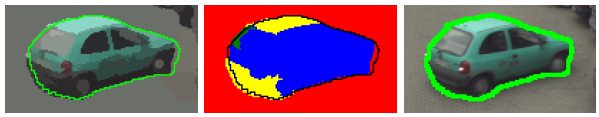

Figure 2: The initial object mask retrieved from the mean shift iterations in spatial-scale-space is shown above the segmented object (left). The segments are classified either as one of the three possible object segment types (blue, yellow and green segments) or as background segments (red segments). According to the object segments the contour of the final object mask is estimated and displayed on the object being tracked (right).

The next object based kernel can now be obtained from the final shape and the next mean shift iterations can be initialized.

4. EXPERIMENTS

After the first object mask is determined by a motion detection algorithm, the object centroid and the mask based asymmetric kernel are computed. The masked based kernel

Figure 3: Tracking results using the traditional mean shift tracker combined with the $\pm 10\%$ method (red box) and the proposed method (green contour) for the sequence *parking_lot_1* shown from top left to bottom right.

is then used for computing the histogram in the RGB space with 32x32x32 bins. For the scale dimension the Epanechnikov kernel with a bandwith of $h_\sigma = 0.4$ is used. For mean shift segmentation a multivariate kernel defined according to equation (35) in [2] as the product of two Epanechnikov kernels, one for the spatial domain (pixel coordinates) and one for the range domain (color), is used. The bandwith of the Epanechnikov kernel in range domain was set to $h_r = 4$, and the bandwith of the one in spatial domain to $h_s = 5$. The minimal segment size was set to 5 pixels.

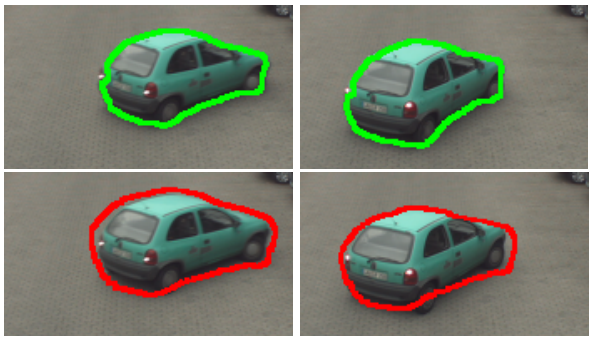

Figure 4: Object tracking using the proposed method (top) and the scale and orientation adaptive method (bottom)

The proposed algorithm has been tested on several video sequences. In Figure 4 the tracking results of the proposed shape adaptive mean shift tracker (green contour) are compared to the method which only adapts the scale and orientation of the initial object mask (red contour). At the beginning of the tracking process only very little difference between both methods is noticeable, but as soon as the car starts turning the shape adaptive tracking technique outperforms the method which is only able to adapt to in-plane rotations.

In Figure 3 the proposed method is compared with the traditional mean shift tracking using the $\pm 10\%$ method. The traditional method is only partly able to adapt to the changing size and position of the white van, while location and contour of the van are well tracked by the scale and shape adaptive mean shift tracker even when the van turns. Only a small part of the front of the van is not included in the tracked object shape due to some segmentation errors. The results of tracking a racing car with the scale and shape adaptive mean shift tracker are shown in the first two rows of Figure 5. Even if the racing car is moving with high velocity which leads to fast changes in size and shape of the object, the proposed method is able to detect the position as well as the shape of the racing car. In rows 3 and 4 of Figure 5 the results of tracking the green car are shown.

To further evaluate the tracking performance frame-based detection rate R_D and false alarm rate for the false positives R_{FP} and the false negatives R_{FN} were calculated, and then averaged over image frames. R_D, R_{FP} and R_{FN} of sequence *parking_lot_2* are shown in Table 1. All rates were computed by comparing the tracking result with the object area of a manually labeled ground truth. R_D is defined as the pixels from the tracked moving object that fall into the true object area. The false positives are defined as the number of pixels that are wrongly tracked as moving object pixels while the false negatives are the number of missing object pixels.

sequence	frames	R_D (%)	R_{FP} (%)	R_{FN} (%)
parking_lot_2	35	98.54	8.25	1.45

Table 1: Average detection rate R_D, average false positives rate R_{FP} and average false negatives rate R_{FN}.

5. CONCLUSIONS

The proposed method extends the traditional mean shift algorithm to track objects with changing size and shape. This is achieved by using an object mask based kernel to track the object in a three dimensional search space to update the location of the object as well as its scale. To obtain the object shape more precisely the mean shift iterations are followed by a segmentation process. Thus, the object shape is obtained very well even if the object is performing 3D rotations. In the case of similar object and background colors the scale

Figure 5: Tracking results using the proposed method. Rows 1-2: sequence *formel_1*. Rows 3-4: sequence *parking_lot_2*

and shape adaptive tracker has to deal with errors, since segmentation errors can occur and to many object colors may be deleted from the target model. In future we will work on this problem as well as on an extension for multi-object tracking.

REFERENCES

[1] Y. Cheng, "Mean Shift, Mode Seeking, and Clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, pp. 790–799, Aug. 1995.

[2] D. Comaniciu, P Meer, "Mean Shift: A Robust Approach Toward Feature Space Analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 603–619, May 2002.

[3] D. Comaniciu, V. Ramesh, P Meer, "Real-time tracking of non-rigid objects using mean shift," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2000, pp. 142–149.

[4] G.R. Bradski, "Computer vision face tracking for use in a perceptual user interface," *Intel Technology Journal*, vol. 2, pp. 12–21, 1998.

[5] D. Comaniciu, V. Ramesh, P. Meer, "Kernel-Based Object Tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, pp. 564–575, May 2003.

[6] R. T. Collins, "Mean-shift blob tracking through scale space," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, 2003, pp. 234–240.

[7] Q. Qifeng, D. Zhang, Y. Peng, "An Adaptive Selection of the Scale and Orientation in Kernel Based Tracking," in *Proc. of the Third International IEEE Conference on Signal-Image Technologies and Internet-Based System*, 2007, pp. 659–664.

[8] A. Yilmaz, "Object Tracking by Asymmetric Kernel Mean Shift with Automatic Scale and Orientation Selection," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, June 2007, pp. 1–6.

[9] A. Elgammal, D. Harwood, L. Davis, "Non-parametric Model for Background Subtraction," in *Proc. of the 6th European Conference on Computer Vision*, Jun./Jul. 2000, pp. 751–767.