Computer Vision and Image Understanding 114 (2010) 400–408

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Mean Shift tracking with multiple reference color histograms

Ido Leichter *, Michael Lindenbaum, Ehud Rivlin

Computer Science Department, Technion – Israel Institute of Technology, Haifa 32000, Israel

ARTICLE INFO

Article history:
Received 21 January 2009
Accepted 9 December 2009
Available online 4 January 2010

Keywords:
Visual tracking
Mean Shift
Multiple references

ABSTRACT

The Mean Shift tracker is a widely used tool for robustly and quickly tracking the location of an object in an image sequence using the object's color histogram. The reference histogram is typically set to that in the target region in the frame where the tracking is initiated. Often, however, no single view suffices to produce a reference histogram appropriate for tracking the target. In contexts where multiple views of the target are available prior to the tracking, this paper enhances the Mean Shift tracker to use multiple reference histograms obtained from these different target views. This is done while preserving both the convergence and the speed properties of the original tracker. We first suggest a simple method to use multiple reference histograms for producing a single histogram that is more appropriate for tracking the target. Then, to enhance the tracking further, we propose an extension to the Mean Shift tracker where the convex hull of these histograms is used as the target model. Many experimental results demonstrate the successful tracking of targets whose visible colors change drastically and rapidly during the sequence, where the basic Mean Shift tracker obviously fails.

1. Introduction

The target's color histogram is widely used for visual tracking (e.g. [1–5]) and, as was shown by Comaniciu et al. [3,6], tracking using this feature may be performed very quickly via the Mean Shift procedure [7]. This paper extends Comaniciu et al.'s tracker in [3,6], which will be referred to in this paper by its common name *Mean Shift tracker*.

The Mean Shift tracker works by searching in each frame for the location of an image region whose color histogram is closest to the reference color histogram of the target. The distance between two histograms is measured using their Bhattacharyya coefficient, and the search is performed by seeking the target location via Mean Shift iterations beginning from the target location estimated in the previous frame (the tracker is outlined in Section 2).

In the Mean Shift tracker, as well as in the other trackers cited previously, the reference color histogram is approximated according to a single view of the target, typically as it appears in the first frame of the sequence. Although using this method for obtaining the reference histogram proved to be very robust in many scenarios, it produces, in many cases, a poor representation of the target, which might result in poor tracking. More seriously, the support of a reference histogram obtained by this method may become non-overlapping with the support of the target's histogram as it appears in the sequence, usually resulting in target loss. Indeed, for many objects, *any* viewing direction may be replaced with a different viewing direction where all the object's colors apparent in the latter view differ from those in the former. An (unscrambled) Rubik Cube is an extreme example of such an object; each side is a different color, and three sides at most are visible from any viewing direction. Major changes in the apparent colors of a target may also result from changes in the actual target's colors, as when a person puts on or removes a piece of clothing, or as in the case of an alternating street advertisement.

Often, several different views of the target are available prior to the tracking, either from images that were previously acquired (e.g. [8–11]) or when performing off-line tracking (e.g. [12–15]). In these contexts, this paper extends the Mean Shift tracker to using multiple reference color histograms. At first we suggest a simple method to combine these histograms into a single histogram that is more appropriate for tracking the target. In order to enhance the tracking further, we then propose an extension to the Mean Shift tracker, where the convex hull of these histograms is used as the target model. That is, rather than searching for the image region whose color histogram is closest to a single reference histogram, we search for the image region by minimizing the distance of its color histogram from the convex hull of several reference histograms.

Time-varying histograms of colors (e.g. [5,16]) or of other features such as filter responses (e.g. [17]) have been used for target modeling before, and many trackers have modeled the target's 2D appearance as being time-varying within a subspace (e.g. [8,9,18–20]). In the latter group of trackers, the search for the target (and possibly for additional transformation parameters) is

* Corresponding author.
E-mail addresses: idol@cs.technion.ac.il (I. Leichter), mic@cs.technion.ac.il (M. Lindenbaum), ehudr@cs.technion.ac.il (E. Rivlin).

1077-3142/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.cviu.2009.12.006

performed by minimizing the distance of its appearance in the current frame from that subspace. This approach is applied here by modeling the target's color histogram as being a time-varying linear combination of several reference histograms, under the restriction that the mixture coefficients are nonnegative and sum to unity (so that the linear combination will be a histogram mixture).

Section 2 outlines the original Mean Shift tracker [3]. A simple method for combining multiple reference histograms into one is proposed in Section 3. Section 4 describes the proposed extension of the Mean Shift tracker to use the convex hull-based target model. Experimental results are described in Section 5, Section 6 includes a discussion, and a paper summary is provided in Section 7.

2. The Mean Shift tracker

In this section we outline the Mean Shift tracker described in [3]. The notations used here are similar to those in [3], with minor modifications to suit the subsequent sections.

2.1. Histograms, their approximation and their distance

Let $\hat{\mathbf{q}} = \{\hat{q}_u\}_{u=1,\ldots,m}$ $(\sum_{u=1}^{m}\hat{q}_u = 1)$ be an m-bin reference color histogram of the target. The center point $\mathbf{y}$ of the image region in which the color histogram $\hat{\mathbf{p}}(\mathbf{y}) = \{\hat{p}_u(\mathbf{y})\}_{u=1,\ldots,m}$ $(\sum_{u=1}^{m}\hat{p}_u = 1)$ is closest to $\hat{\mathbf{q}}$ is sought in each frame of the image sequence. The metric used in [3] for measuring the distance between the histograms is

$$d_{\hat{\mathbf{q}}}(\mathbf{y}) = \sqrt{1 - \rho[\hat{\mathbf{p}}(\mathbf{y}), \hat{\mathbf{q}}]}, \tag{1}$$

where $\rho[\hat{\mathbf{p}}(\mathbf{y}), \hat{\mathbf{q}}]$ is their Bhattacharyya coefficient:

$$\hat{\rho}(\mathbf{y}) \equiv \rho[\hat{\mathbf{p}}(\mathbf{y}), \hat{\mathbf{q}}] = \sum_{u=1}^{m}\sqrt{\hat{p}_u(\mathbf{y})\hat{q}_u}. \tag{2}$$

The image region used for calculating the color histogram is elliptical.[1] The elliptical region is specified in one (typically the first) frame in the sequence, and the reference color histogram is calculated using this region. Then, with the ellipse motion approximated as translational, the center of this elliptical region is tracked. Note that approximating the motion of the ellipse (which encloses the tracked image region) as translational does not imply that the motion of the region itself is only translational. A simple mechanism for adapting also to isotropic scale changes of the region was proposed in [3]. This scale adaptation is carried out by running the Mean Shift tracker for three different scales – the scale estimated in the previous frame, an enlarged scale, and a reduced scale – and then choosing the one that produced the target location in which the color histogram is closest to the reference histogram. The chosen scale is then input into an IIR filter for producing the final scale. This mechanism operates on top of the Mean Shift tracker and it may be applied exactly in the same manner using the algorithms proposed here. Therefore, for simplicity, the target scale in this paper is assumed to be fixed.

The pixel locations are normalized such that the specified reference ellipse will become the unit circle centered at the origin. (This is achieved by rotating, non-isotropically scaling, and translating the coordinate system of the image.) The normalized locations of the pixels inside the reference ellipse are denoted $\{\mathbf{x}_i^{\star}\}_{i=1,\ldots,n}$, and the reference color histogram is computed as

$$\hat{q}_u = C\sum_{i=1}^{n} k\left(\left\|\mathbf{x}_i^{\star}\right\|^2\right)\delta[b(\mathbf{x}_i^{\star}) - u], \tag{3}$$

where $b(\mathbf{x})$ is the bin number $(1,\ldots,m)$ associated with the color at the pixel of normalized location $\mathbf{x}$, δ is the Kronecker delta function, $k(x)$ is a kernel profile that assigns smaller weights to pixels farther from the circle center, and C is a constant that normalizes the histogram to be of unit sum. Similarly, the normalized locations of the pixels inside an ellipse centered at a candidate normalized location $\mathbf{y}$ are denoted $\{\mathbf{x}_i\}_{i=1,\ldots,n}$, and the color histogram in the corresponding region is thus

$$\hat{p}_u(\mathbf{y}) = C\sum_{i=1}^{n} k(\|\mathbf{y} - \mathbf{x}_i\|^2)\delta[b(\mathbf{x}_i) - u]. \tag{4}$$

It is assumed here that both elements of $\mathbf{y}$ correspond to integer numbers of pixels. Otherwise, n and C may fluctuate according to the exact inter-pixel location of the ellipse center.

The Epanechnikov kernel was used in [3], and we use it here as well. The profile of this kernel is

$$k(x) \propto \begin{cases} 1 - x, & 0 \leqslant x \leqslant 1, \\ 0, & x > 1. \end{cases} \tag{5}$$

2.2. Target localization via Mean Shift

The goal in each frame of the sequence is to estimate the target translation $\hat{\mathbf{y}}$ that minimizes the distance $d_{\hat{\mathbf{q}}}(\hat{\mathbf{y}})$ between its corresponding histogram $\hat{\mathbf{p}}(\hat{\mathbf{y}})$ and the reference histogram $\hat{\mathbf{q}}$. This is equivalent to estimating the target translation $\hat{\mathbf{y}}$ that maximizes the Bhattacharyya coefficient (2) between these two histograms. Denote by $\hat{\mathbf{y}}_0$ the estimated target location in the previous frame. Approximating the Bhattacharyya coefficient (2) in the current frame by its first-order Taylor expansion around the values $\hat{p}_u(\hat{\mathbf{y}}_0)$ and substituting (4) for $\hat{\mathbf{p}}(\mathbf{y})$ results in

$$\rho(\mathbf{y}) \approx C_{\hat{\mathbf{q}},\hat{\mathbf{y}}_0} + \frac{C}{2}\sum_{i=1}^{n} w_i k(\|\mathbf{y} - \mathbf{x}_i\|^2), \tag{6}$$

where

$$w_i = \sum_{u=1}^{m}\sqrt{\frac{\hat{q}_u}{\hat{p}_u(\hat{\mathbf{y}}_0)}}\delta[b(\mathbf{x}_i) - u] \tag{7}$$

and $C_{\hat{\mathbf{q}},\hat{\mathbf{y}}_0}$ is independent of $\mathbf{y}$.

The search for $\hat{\mathbf{y}}$ in the current frame starts at the estimated target location in the previous frame, i.e., initially $\hat{\mathbf{y}} \leftarrow \hat{\mathbf{y}}_0$. Now, the $\mathbf{y}$-dependent term in the right-hand side of (6) may be viewed as a kernel density estimate computed with kernel profile $k(x)$ at $\mathbf{y}$, with the samples $\mathbf{x}_i$ being weighted by w_i (7). As was proven in [7], if the weights w_i are nonnegative and the kernel profile $k(x)$ is monotonically non-increasing and convex (all these requirements are fulfilled (5,7)), then a higher density value is reached by *shifting* $\hat{\mathbf{y}}$ from $\hat{\mathbf{y}}_0$ to the *mean* of the sample, which is weighted by the kernel whose profile is $g(x) = -k'(x)$ and which is centered at $\hat{\mathbf{y}}_0$,

$$\hat{\mathbf{y}} \leftarrow \frac{\sum_{i=1}^{n}\mathbf{x}_i w_i g(\|\hat{\mathbf{y}}_0 - \mathbf{x}_i\|^2)}{\sum_{i=1}^{n} w_i g(\|\hat{\mathbf{y}}_0 - \mathbf{x}_i\|^2)}. \tag{8}$$

(An exception is a shift by zero, indicating that the estimated location is already at a density mode.) Moreover, iteratively repeating the Mean Shift (8), each time replacing $\hat{\mathbf{y}}_0$ by $\hat{\mathbf{y}}$ calculated at the previous iteration, will result in the convergence of $\hat{\mathbf{y}}$ to a density mode. (Although rarely, it might happen that an application of (8) will decrease the true $\rho(\mathbf{y})$, since (6) is only an approximation of it.)

[1] The ellipse in [3] had axes parallel to the image border, but of course generalizing to account for rotated ellipses is performed by simply rotating the coordinate system of the image.

Fig. 1. Results of the Mean Shift tracker for Sequence I, where the reference color histogram was set in the first frame.

Note that the weights w_i (7) should be recomputed after each Mean Shift iteration (8), since each iteration causes the color histogram $\hat{\mathbf{p}}(\hat{\mathbf{y}})$ to change. Observe that for the Epanechnikov profile (5), the Mean Shift iteration (8) reduces to a simple weighted average.

3. Combining multiple histograms into one

Sometimes no view of the target yields a reasonable approximation of its circumferential color histogram. An extreme example is presented in Fig. 1. This figure shows the results of the Mean Shift tracker for Sequence I, where a Rubik Cube is tracked. The reference color histogram was set in the first frame, where the visible colors of the cube are orange,[2] blue and yellow. As the cube rotates, different sets of colors become visible, causing the large deviations of the estimated locations from the target center. At Frame 475 none of the originally visible colors of the target remain visible, and so the tracking fails. Each side of the cube is a different color, and three sides at most may be visible from any viewing direction. Therefore, this problem could not be generally solved by using a different view of the target to model its reference color histogram.

One possible solution to the above problem is to stay with a single reference histogram, but set it using multiple color histograms obtained from different target views. Assume we are given M views of the target and denote the color histograms visible from these views by $\hat{\mathbf{q}}^v = \{\hat{q}_u^v\}_{u=1,\ldots,m}$, $v = 1,\ldots,M$. Since any of these target views may appear during the tracking process, the maximal reference histogram's distance from all the given histograms should be as small as possible. To this end, the following minimax optimization problem has to be solved for $\hat{\mathbf{q}}$:

$$\begin{aligned} &\text{minimize} \quad \max_v \sqrt{1-\rho[\hat{\mathbf{q}},\hat{\mathbf{q}}^v]}, \\ &\text{subject to} \quad \hat{q}_u \geqslant 0, \quad u = 1,\ldots,m, \\ &\qquad\qquad \sum_{u=1}^{m} \hat{q}_u = 1. \end{aligned} \tag{9}$$

The solution to this problem coincides with to that of the problem:

$$\begin{aligned} &\text{maximize} \quad \min_v \sum_{u=1}^{m} \sqrt{\hat{q}_u \hat{q}_u^v}, \\ &\text{subject to} \quad \hat{q}_u \geqslant 0, \quad u = 1,\ldots,m, \\ &\qquad\qquad \sum_{u=1}^{m} \hat{q}_u = 1. \end{aligned} \tag{10}$$

The feasible set of this maximization problem is convex. Likewise, each term in the sum in the objective function is concave, and concavity of functions is preserved under addition and pointwise minimization, so that the whole objective function is concave. Therefore (10) is a convex optimization problem, and thus can be solved efficiently [21].

[2] For interpretation of color in Figs. 1–11, the reader is referred to the web version of this article.

To test the above method for setting the reference color histogram, the Mean Shift tracker was tested again on Sequence I, this time using the reference color histogram (9) obtained from two different views of the target ($M = 2$). One view is the one used in the previous experiment (Frame 1 in the sequence), and the other view is from the opposite direction. These two views and an illustration of the computed final histogram $\hat{\mathbf{q}}$ are shown in Fig. 2. The tracking results are presented in Fig. 3. The improved accuracy with respect to Fig. 1 is clearly evident. In particular, the tracking failure at Frame 475, where all the initially visible target colors were occluded, was resolved here.

Despite the higher accuracy and robustness achieved by this method in the sequence, a final reference model consisting of a *single* histogram is still a mediocre representation of the true, time-varying histogram of the target. An example of this may be seen in Frame 500 in Fig. 3. In this frame, an image region consisting of three different colors, all present in the final reference histogram $\hat{\mathbf{q}}$, has a color histogram closer to $\hat{\mathbf{q}}$ than the histogram at the correct image region, which consists of only two colors, both present in $\hat{\mathbf{q}}$.

The above method of setting the reference histogram has an additional, minor drawback: the need to solve (although only once) the optimization problem (10) before the tracking begins. In this work we used the RGB color space, where each color band was equally divided into eight bins. This yielded $m = 8^3 = 512$ color bins in all. Solving the convex optimization problem (10) with this number of variables took several seconds using `cvx` [22]. We stress that this computation is done offline, once per target and prior to the tracking. Therefore, it does not affect the speed of the on-line tracking, which is carried out by the original Mean Shift tracker.

4. Convex hull-based target model

As different sides of the target face the camera, the target's histogram changes. To accommodate for a time-varying target histogram, we propose to extend the reference target model used by the Mean Shift tracker to include the convex hull of multiple reference histograms obtained from different target views. That is, the target model is approximated as the mixture of M reference histograms

$$\hat{\mathbf{q}}(\boldsymbol{\alpha}) = \sum_{v=1}^{M} \alpha_v \hat{\mathbf{q}}^v, \quad \forall v \ \alpha_v \geqslant 0, \quad \sum_{v=1}^{M} \alpha_v = 1, \tag{11}$$

where the mixture proportions $\boldsymbol{\alpha} = \{\alpha_v\}_{v=1,\ldots,M}$ vary with time.

Thus, the tracking process now consists of finding in each frame the target translation $\hat{\mathbf{y}}$ that minimizes the minimal distance

$$\min_{\boldsymbol{\alpha}} d_{\hat{\mathbf{q}}(\boldsymbol{\alpha})}(\hat{\mathbf{y}})$$

between $\hat{\mathbf{p}}(\hat{\mathbf{y}})$ and the set of all histogram mixtures $\hat{\mathbf{q}}(\hat{\boldsymbol{\alpha}}) = \{\hat{q}_u(\hat{\boldsymbol{\alpha}})\}_{u=1,\ldots,m}$. Although the previous Bhattacharyya-based distance (1)–(2) may be minimized here as well, tracking using the convex hull-based target model provided better experimental results by minimizing the sum of absolute difference (SAD),

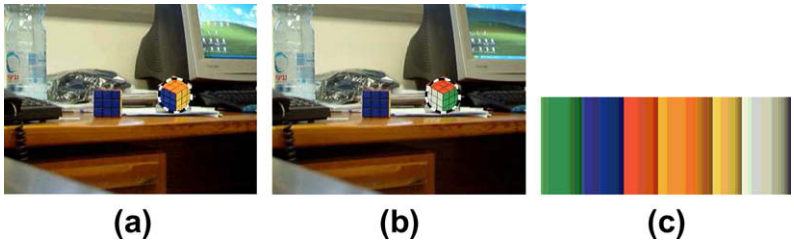

Fig. 2. The two target views used for approximating the two reference histograms: (a) $\hat{\mathbf{q}}^1$ and (b) $\hat{\mathbf{q}}^2$. These two reference histograms were used to set the final reference histogram $\hat{\mathbf{q}}$ according to (9). The latter histogram is illustrated in (c) by an image with an identical color histogram.

Fig. 3. Results of the Mean Shift tracker for Sequence I, where the reference color histogram was set using the two target views shown in Fig. 2.

$$d_{\hat{\mathbf{q}}(\hat{\boldsymbol{\alpha}})}(\mathbf{y}) = \sum_{u=1}^{m} |\hat{p}_u(\mathbf{y}) - \hat{q}_u(\hat{\boldsymbol{\alpha}})|. \tag{12}$$

As in (1)–(2), the measure (12) imposes a metric structure as well.

Like the search for the current $\hat{\mathbf{y}}$ in the original Mean Shift tracker (Section 2.2), the search for the current target location is performed here by the iterative reduction of (12) via the iterative modification of the target location and mixture proportions $(\hat{\mathbf{y}}, \hat{\boldsymbol{\alpha}})$, beginning from the target location and mixture proportions estimated in the previous frame, i.e., initially $(\hat{\mathbf{y}}, \hat{\boldsymbol{\alpha}}) \leftarrow (\hat{\mathbf{y}}_0, \hat{\boldsymbol{\alpha}}_0)$. The minimization of (12) is performed by successive repetitions of the following two steps:

1. Shifting the estimated target location $\hat{\mathbf{y}}$ such that (12) is reduced, while keeping the estimated mixture proportions $\hat{\boldsymbol{\alpha}}$ fixed.
2. Minimizing (12) with respect to the mixture proportions $\hat{\boldsymbol{\alpha}}$, while keeping the estimated target location $\hat{\mathbf{y}}$ fixed.

As in the reduction of the distance (1) in the original Mean Shift tracker, the reduction of the distance (12) in Step 1 above is performed via a Mean Shift iteration. Because the minimization problem in Step 2 is convex, it may be performed rapidly as well. These two steps are described in detail in the following two subsections.

Since (12) is bounded from below and is reduced in each of the two steps, the sequence of distances obtained by the successive application of these steps is guaranteed to converge. In practice, these two steps are repeated until the shift of the estimated target location is smaller than half a pixel (this criterion is very similar to the one in the original Mean Shift implementation [3]) and each of the mixture proportions is modified by less than 0.01. As in the original Mean Shift implementation, the above pair of steps usually has to be performed only a few times per frame until these two criteria are met.

4.1. Step 1: shifting $\hat{\mathbf{y}}$

In this step, the distance $d_{\hat{\mathbf{q}}(\hat{\boldsymbol{\alpha}})}(\mathbf{y})$ in (12) has to be reduced by shifting the estimated target location $\hat{\mathbf{y}}$ from the location obtained after the previous application of Step 1 (or that obtained in the previous frame, if this is the first application of Step 1 in the current frame), while keeping the estimated mixture proportions $\hat{\boldsymbol{\alpha}}$ unchanged. As $\hat{\boldsymbol{\alpha}}$ is fixed in this step, the histogram mixture $\hat{\mathbf{q}}(\hat{\boldsymbol{\alpha}})$ will be denoted here in shorthand by $\hat{\mathbf{q}}$. For notational compatibility with the previous Mean Shift derivation (Section 2.2), let us denote the estimated target location obtained after the previous application of Step 1 by $\hat{\mathbf{y}}_0$.

Since the metric used here (Eq. (12)) is different from that used in [3], the minimization procedure should be suitably adapted. Reducing $d_{\hat{\mathbf{q}}}(\mathbf{y})$ is equivalent to increasing $-d_{\hat{\mathbf{q}}}(\mathbf{y})$. Approximating $-d_{\hat{\mathbf{q}}}(\mathbf{y})$ by its first-order Taylor expansion around the values $\hat{p}_u(\hat{\mathbf{y}}_0)$ yields[3]

$$\begin{aligned} -d_{\hat{\mathbf{q}}}(\mathbf{y}) &\approx -d_{\hat{\mathbf{q}}}(\hat{\mathbf{y}}_0) + \begin{pmatrix} \operatorname{sgn}(\hat{q}_1 - \hat{p}_1(\hat{\mathbf{y}}_0)) \\ \vdots \\ \operatorname{sgn}(\hat{q}_1 - \hat{p}_1(\hat{\mathbf{y}}_0)) \end{pmatrix}^T \begin{pmatrix} \hat{p}_1(\mathbf{y}) - \hat{p}_1(\hat{\mathbf{y}}_0) \\ \vdots \\ \hat{p}_m(\mathbf{y}) - \hat{p}_m(\hat{\mathbf{y}}_0) \end{pmatrix} \\ &= C'_{\hat{\mathbf{q}},\hat{\mathbf{y}}_0} + \sum_{u=1}^{m} \hat{p}_u(\mathbf{y}) \operatorname{sgn}(\hat{q}_u - \hat{p}_u(\hat{\mathbf{y}}_0)), \end{aligned} \tag{13}$$

where $C'_{\hat{\mathbf{q}},\hat{\mathbf{y}}_0}$ is independent of $\mathbf{y}$ and $\operatorname{sgn}(\cdot)$ is the sign function. Substituting (4) for each $\hat{p}_u(\mathbf{y})$ results in

$$-d_{\hat{\mathbf{q}}}(\mathbf{y}) \approx C'_{\hat{\mathbf{q}},\hat{\mathbf{y}}_0} + C \sum_{i=1}^{n} w_i k(\|\mathbf{y} - \mathbf{x}_i\|^2), \tag{14}$$

where

$$w_i = \sum_{u=1}^{m} \operatorname{sgn}(\hat{q}_u - \hat{p}_u(\hat{\mathbf{y}}_0)) \delta[b(\mathbf{x}_i) - u]. \tag{15}$$

The $\mathbf{y}$-dependent term in this approximation to the objective function $-d_{\hat{\mathbf{q}}}(\mathbf{y})$ is of similar form as that in (6), but with different pixel weights w_i. Since now the weights may be negative, $\hat{\mathbf{y}}$ cannot be updated via the regular Mean Shift iteration (8), as it was in the maximization of the Bhattacharyya coefficient. However, shifting $\hat{\mathbf{y}}$ according to

$$\hat{\mathbf{y}} \leftarrow \hat{\mathbf{y}}_0 + \frac{\sum_{i=1}^{n} (\mathbf{x}_i - \hat{\mathbf{y}}_0) w_i g(\|\hat{\mathbf{y}}_0 - \mathbf{x}_i\|^2)}{\sum_{i=1}^{n} |w_i g(\|\hat{\mathbf{y}}_0 - \mathbf{x}_i\|^2)|} \tag{16}$$

[3] Since the absolute value function is not differentiable at 0, its derivative there is replaced by the mean of its derivatives from each side, which equals 0. As may be seen in the obtained expression for the objective function, terms resulting from differentiating the absolute value function at 0 are independent of $\mathbf{y}$, and thus do not affect its optimization.

will increase the kernel density even when there are negative weights (shown by Collins [2]).

4.2. Step 2: optimizing $\hat{\boldsymbol{\alpha}}$

In this step, the distance $d_{\hat{\mathbf{q}}(\hat{\boldsymbol{\alpha}})}(\mathbf{y})$ in (12) has to be minimized with respect to the mixture proportions $\hat{\boldsymbol{\alpha}}$, while keeping the estimated target location $\hat{\mathbf{y}}$ unchanged. As $\hat{\mathbf{y}}$ is fixed in this step, the histogram at the estimated target location $\hat{\mathbf{p}}(\hat{\mathbf{y}})$ will be denoted here in shorthand by $\hat{\mathbf{p}}$.

In order to minimize $d_{\hat{\mathbf{q}}(\hat{\boldsymbol{\alpha}})}(\hat{\mathbf{y}})$ with respect to $\hat{\boldsymbol{\alpha}}$, the following problem has to be solved for $\hat{\boldsymbol{\alpha}}$:

$$\begin{aligned} &\text{minimize} \quad \sum_{u=1}^{m}\left|\hat{p}_u - \sum_{v=1}^{M}\hat{\alpha}_v\hat{q}_u^v\right|, \\ &\text{subject to} \quad \hat{\alpha}_v \geqslant 0, \quad v = 1,\ldots,M, \\ &\qquad\qquad \sum_{v=1}^{M}\hat{\alpha}_v = 1. \end{aligned} \tag{17}$$

The feasible set of this minimization problem is convex. Likewise, it is easy to see that each term in the first sum in the objective function is convex, and since convexity of functions is preserved under addition, the whole objective function is convex as well. Therefore (17) is a convex optimization problem, which may be solved quickly, especially since only a few mixture proportions need to be optimized (2–4 in the experiments) in practice.

4.3. Running time

The proposed method consists of alternately performing Steps 1 and 2 above until convergence. As mentioned, the number of repetitions of these two steps until convergence is low and similar to the number of Mean Shift iterations in the original Mean Shift tracker (an average of about five iterations per frame). As was shown in Section 4.1, Step 1 is in fact a Mean Shift calculation, and it is identical to that performed in a single Mean Shift iteration of the original Mean Shift tracker (only that the pixel weights are different and may be negative). Therefore, the only difference from the computational load in the original Mean Shift tracker, which already performs Step 1 iterations, is the time required for performing Step 2 iterations. The average time-per-frame of the original Mean Shift tracker (i.e., only Step 1 iterations) with constant scale, as measured on a standard modern Personal Computer (PC), is about 1 ms [24]. As explained (Section 4.2), the execution of Step 2 merely consists of solving a convex optimization problem of $M-1$ variables: one variable per each of the M mixture coefficients, minus one degree of freedom due to their unit sum. For $M = 4$ (the largest used in the experiments), a simple gradient descent implementation in MATLAB required on average about 1.5 ms in total for all Step 2 repetitions in one frame on a standard modern PC. This 1.5 ms period, and shorter for smaller M, is the difference in time-per-frame from the original Mean Shift tracker. We conclude that the proposed tracker maintains the real-time capability of the original Mean Shift tracker. Table 1 summarizes the runtime for $M \leqslant 4$.

Table 1
Average time per frame, in milliseconds, with $M = 1,\ldots,4$ reference histograms (without scale adaptation). The per-frame runtime for the Step 2 iterations (mixture coefficients optimizations) increases with M, whereas Steps 1 (Mean Shift iterations) do not depend on M. When scale adaptation [3] is performed, all these figures should be multiplied by 3.

M	Step 1 iterations	Step 2 iterations	Total
1	1	–	1
2	1	0.4	1.4
3	1	1.2	2.2
4	1	1.5	2.5

5. Experimental results

Results of testing the Mean Shift tracker with the convex hull-based target model are presented for seven sequences. All the targets tracked in the experiments were such that their color histogram could not be reasonably modeled from a single view. In all experiments the RGB color space was used. Each color band was equally divided into eight bins, except for Sequence III, where each color band had to be divided into 32 bins because the target's colors were very similar to the background's.

The target locations in the first frame and in the reference images were manually marked. Choosing the reference views was easy in practice. In each sequence, we simply selected two arbitrary frames where the target colors seemed to be very distinct, and this proved to be sufficient in most cases. In some cases (e.g., Sequences IV and VI), it was revealed experimentally that the two selected reference views were not enough, so a third reference view was selected according to the failing point. It may happen that further reference views would be required to be selected in the same way, as was the case for the fourth reference view in Sequence IV.

Video files of all presented results are given as Supplementary material.

Sequence I. The Mean Shift tracker with the convex hull-based target model was first tested on Sequence I. This sequence was previously used to test the regular Mean Shift tracker with one target view and with two target views (Fig. 2a and b). These two target views were used for the convex hull-based target model here. Tracking results, along with the estimated mixture proportions (rounded to the thousandth), are presented in Fig. 4. The improvement over the fixed target model (Figs. 1 and 3) is evident.

Sequence II. Tracking the Rubik Cube in a different, longer image sequence was successful. The two target views used for approximating the reference histograms are shown in Fig. 5, and the results are shown in Fig. 6.

Sequence III. This sequence consisted of a woman exiting an apartment. The woman, wearing light-colored clothes, puts on a black coat before she exits the apartment. Thus, her appearance changes drastically. Two target views were used for approximating the reference histograms (Fig. 5). As may be seen in Fig. 7, the tracking was successful.

Sequence IV. In this experiment a greeting card, differently colored on each side, was tracked. Although the card is planar, four target views were used (Fig. 5) to account for the extreme lighting changes. The results are shown in Fig. 8.

Sequence V. In this sequence a rotating street advertisement was tracked. Since different sides of the advertisement had different colors, two reference views had to be used (Fig. 5). The tracking was successful (Fig. 9).

Sequence VI. This sequence contains a street advertisement screen that alternates between several different advertisements. The tracking results using three reference views (Fig. 5) are shown in Fig. 10.

Sequence VII. This sequence was filmed by camera switching between color and IR (Infrared) modes. One color reference view and one IR reference view were used (Fig. 5). The tracking results are shown in Fig. 11.

As mentioned, the proposed tracker is intended for the particular context where the target colors seen by the camera change in time, no single view of the target is sufficient for reasonably modeling the target colors, and where multiple reference views of the target are available prior to the tracking. Trackers are typically intended to be used in a context where the target appearance is provided to them in one frame only – usually that where the tracking

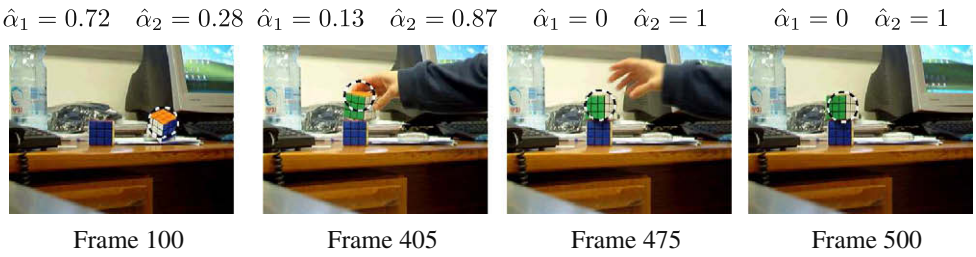

Fig. 4. Tracking results for Sequence I using the convex hull-based target model. The reference color histograms were set using the two target views shown in Fig. 2.

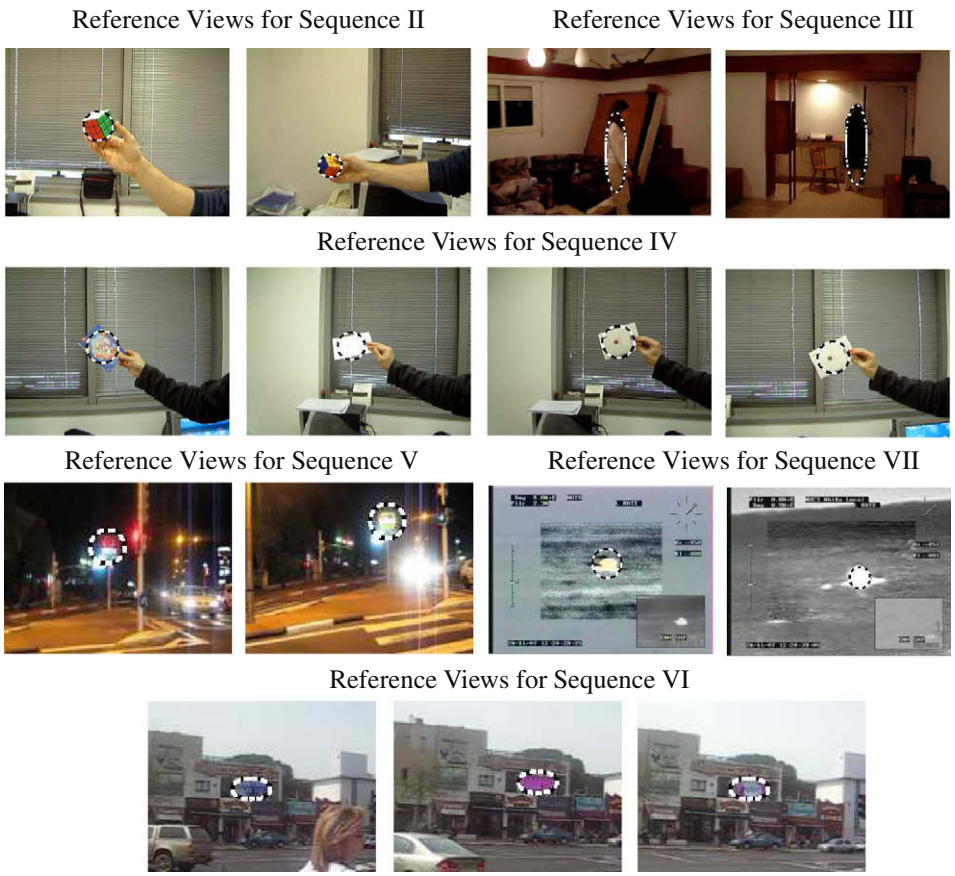

Fig. 5. The target views used for approximating the reference histograms $\hat{\mathbf{q}}^1, \ldots, \hat{\mathbf{q}}^M$, from left to right, in Sequences II–VII.

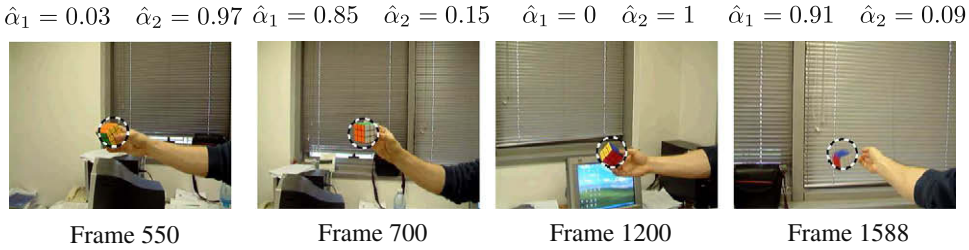

Fig. 6. Tracking results for Sequence II using the convex hull-based target model. The reference color histograms were set using the two target views shown in Fig. 5.

is initiated. All the experiments were chosen to demonstrate the proposed tracker in its particular context, for which "regular" trackers are inappropriate and will likely lose the target when its colors change. As expected, experiments showed that the original Mean Shift tracker lost the target in all sequences due to the change in the target colors. We also tested the tracker by Leichter et al. [23] on these sequences. Due to its target model adaptation scheme, this tracker occasionally overcame the changes in target colors, but it still lost the target due to these changes in more than half of the sequences (III, V–VII).

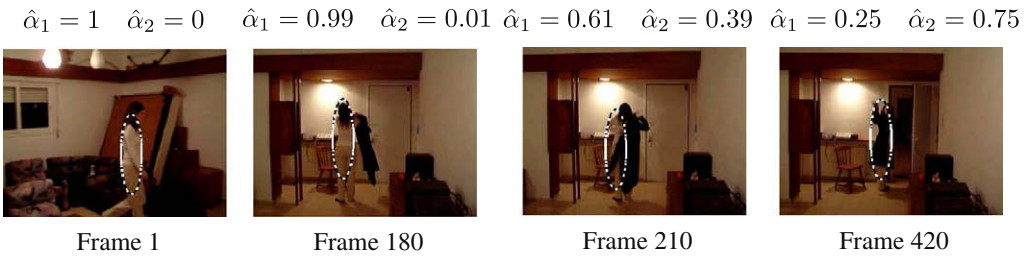

Fig. 7. Tracking results for Sequence III using the convex hull-based target model. The reference color histograms were set using the two target views shown in Fig. 5.

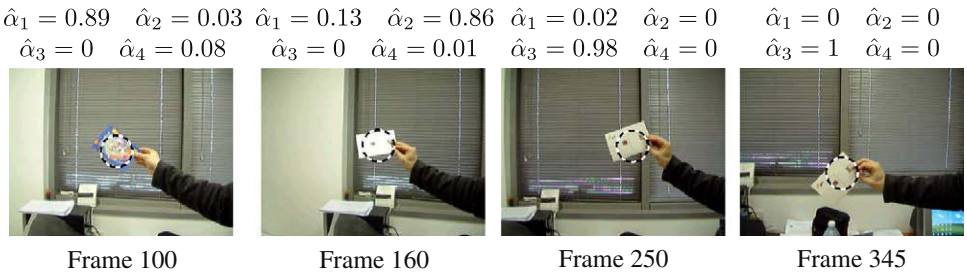

Fig. 8. Tracking results for Sequence IV using the convex hull-based target model. The reference color histograms were set using the four target views shown in Fig. 5.

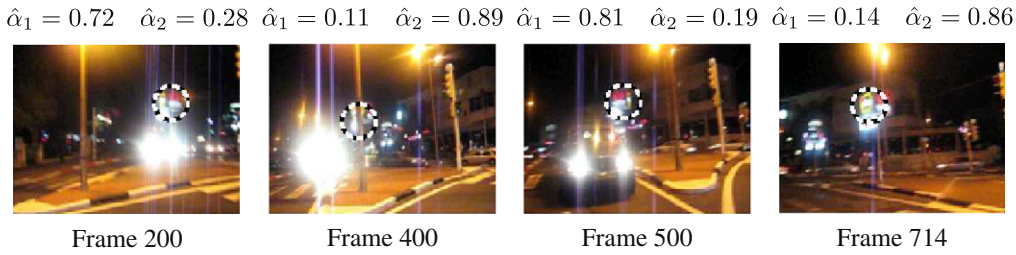

Fig. 9. Tracking results for Sequence V using the convex hull-based target model. The reference color histograms were set using the two target views shown in Fig. 5.

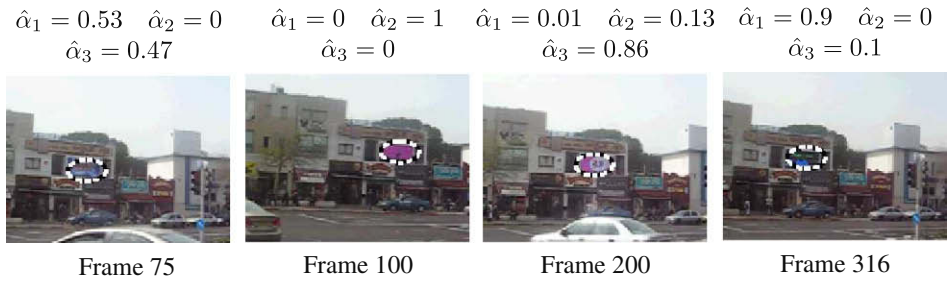

Fig. 10. Tracking results for Sequence VI using the convex hull-based target model. The reference color histograms were set using the three target views shown in Fig. 5.

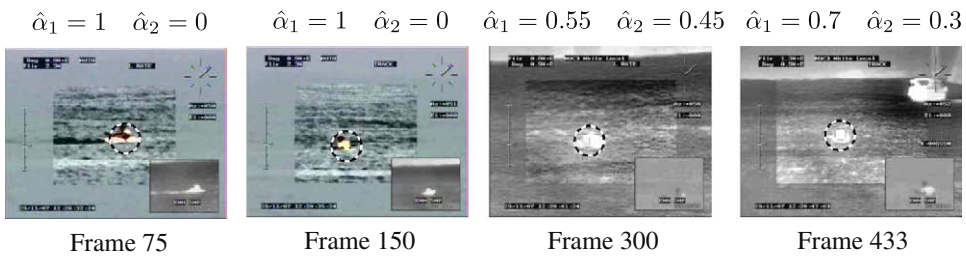

Fig. 11. Tracking results for Sequence VII using the convex hull-based target model. The reference color histograms were set using the two target views shown in Fig. 5.

6. Discussion

There appears to be a resemblance between the problem dealt with in this work and those in the papers by Bajramovic et al. [24] and by Maggio and Cavallaro [25], which also enhance the tracking by employing multiple reference histograms. However, the problems are distinct. Here, we are concerned with the problem of temporal changes in the target's features (e.g., due to rotations in space), whereas [24] deals with the problem of fusing different types of features in the tracking process. These two different problems have led to two very different solutions: in our solution (Section 4) the target is modeled as a linear (convex) combination of histograms (of the same feature type) and the location in the image with a feature histogram of smallest distance from this linear model is sought. In [24], on the other hand, the sought location in the image is that where the linear combination of distances from a small set of different histograms is minimal. We stress the point that, here the linear combination is applied on the reference histograms whereas in [24] it is applied on the distances. Therefore, in the proposed algorithm the reference histogram is allowed to change, whereas in [24] there is a small set of fixed reference histograms (two in the experiments). To illustrate that the framework in [24] is unsuited for the context considered in the paper, consider the experiment related to Sequence V. The target in this experiment is a rotating advertisement cube with two advertisements; each side of the cube is one of these advertisements. Two reference views of the target are provided – each one consists of one advertisement (Fig. 5). Consider now a frame where the cube is viewed from a direction where the two advertisements are substantially visible (e.g., Frame 200 in Fig. 9). In the proposed tracker the target's histogram in this frame is naturally approximated as a weighted average of the two reference histograms, which is an appropriate representation of the target's histogram in this frame. In [24] only the two fixed reference histograms would be used for the target model, and both are less appropriate than their combination in this frame. There is no apparent reason for using a linear combination of the two (large) distances here. However, we would like to note that the framework in [24] may be used to fuse the proposed tracker with trackers that use other types of features.

The tracker by Maggio and Cavallaro [25] is in fact within the framework in [24], but with all feature weights a priori fixed and equal. Therefore, the target's reference histograms are fixed, that is, the target model is not allowed to change in time. Moreover, the target model is composed of several color histograms, each related to a different region of the target in the image plane. This makes the tracker unsuitable for cases where the target rotates in space and different sides of it are colored differently, or when the target colors change due to lighting changes. These make the tracker in [25] unsuited for the considered context as well.

We note that an extension to the Mean Shift tracker of similar sort was proposed by Tu et al. [11], where the target model employed by the Mean Shift tracker was extended to use mixtures of histogram pairs. The tracking there was performed by alternating between target location optimization using one measure (the Bhattacharyya coefficient) and model parameter optimization using a second measure (an observation likelihood). Thus, unlike the work here (and the original Mean Shift tracker), no definite measure was minimized during the tracking, and therefore convergence was not guaranteed. In the extension proposed in this paper, both the convergence and the speed properties of the original tracker are preserved.

7. Conclusion

While the commonly used, Mean Shift tracker [3] proved to be robust in many tracking scenarios, there are cases where no single view suffices to produce a reference color histogram appropriate for tracking the target.

This paper presented a method for immunizing the Mean Shift tracker against the above problem by using multiple reference color histograms. These histograms are obtained from different target views or for different target states. A simple method for combining these histograms into a single histogram that is more appropriate for tracking the target was suggested. In order to enhance the tracking further, an extension to the Mean Shift tracker, where the convex hull of these histograms is used as the target model, was proposed. Unlike the Mean Shift tracker extension of similar sort proposed in [11], in the extension proposed here both the convergence and the speed properties of the original tracker are preserved.

The extended Mean Shift tracker was experimentally verified in many scenarios where the visible target colors changed drastically and rapidly during the sequence. For these scenarios, the original Mean Shift tracker is obviously inappropriate.

Finally, although both methods – that of using several target views to obtain a single reference histogram and that of using the convex hull – were proposed in the context of the Mean Shift tracker, these may also be accommodated in other tracking frameworks such as CONDENSATION [26].

Appendix A. Supplementary material

Supplementary data associated with this article can be found, in the online version, at doi:10.1016/j.cviu.2009.12.006.

References

[1] S. Birchfield, Elliptical head tracking using intensity gradients and color histograms, in: Proceedings of the 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 232–237.
[2] R.T. Collins, Mean-shift blob tracking through scale space, in: Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, 2003, pp. 234–240.
[3] D. Comaniciu, V. Ramesh, P. Meer, Kernel-based object tracking, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (5) (2003) 564–577.
[4] P. Pérez, C. Hue, J. Vermaak, M. Gangnet, Color-based probabilistic tracking, in: Proceedings of the 7th European Conference on Computer Vision, vol. 1, 2002, pp. 661–675.
[5] N.S. Peng, J. Yang, Z. Liu, Mean shift blob tracking with kernel histogram filtering and hypothesis testing, Pattern Recognition Letters 26 (2005) 605–614.
[6] D. Comaniciu, V. Ramesh, P. Meer, Real-time tracking of non-rigid objects using mean shift, in: Proceedings of the 2000 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, 2000, pp. 142–149.
[7] D. Comaniciu, P. Meer, Mean shift: a robust approach toward feature space analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (5) (2002) 603–619.
[8] M.J. Black, A.D. Jepson, EigenTracking: robust matching and tracking of articulated objects using a view-based representation, International Journal of Computer Vision 26 (1) (1998) 63–84.
[9] F. De la Torre, C.J.G. Rubio, E. Martinez, Subspace eyetracking for driver warning, in: Proceedings of the 2003 International Conference on Image Processing, vol. 3, 2003, pp. 329–332.
[10] S. Avidan, Support vector tracking, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (8) (2004) 1064–1072.
[11] J. Tu, H. Tao, T. Huang, Online updating appearance generative mixture model for meanshift tracking, in: Proceedings of the 7th Asian Conference on Computer Vision, vol. 1, 2006, pp. 694–703.
[12] J. Sun, W. Zhang, X. Tang, H.-Y. Shum, Bi-directional tracking using trajectory segment analysis, in: Proceedings of the 10th IEEE International Conference on Computer Vision, vol. 1, 2005, pp. 717–724,.
[13] A. Buchanan, A. Fitzgibbon, Interactive feature tracking using K-D trees and dynamic programming, in: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, 2006, pp. 626–633.
[14] F. Tang, S. Brennan, Q. Zhao, H. Tao, Co-tracking using semi-supervised support vector machines, in: Proceedings of the 11th IEEE International Conference on Computer Vision, 2007.
[15] Y. Wei, J. Sun, X. Tang, H.-Y. Shum, Interactive affine tracking for color objects, in: Proceedings of the 11th IEEE International Conference on Computer Vision, 2007.

[16] S.J. McKenna, Y. Raja, S. Gong, Tracking colour objects using adaptive mixture models, Image and Vision Computing 17 (1999) 225–231.
[17] A.D. Jepson, D.J. Fleet, T.F. El-Maraghi, Robust online appearance models for visual tracking, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (10) (2003) 1296–1311.
[18] G.D. Hager, P.N. Belhumeur, Efficient region tracking with parametric models of geometry and illumination, IEEE Transactions on Pattern Analysis and Machine Intelligence 20 (10) (1998) 1025–1039.
[19] G.J. Edwards, T.F. Cootes, C.J. Taylor, Face recognition using active appearance models, in: Proceedings of the 5th European Conference on Computer Vision, vol. 2, 1998, pp. 581–595.
[20] J. Ho, K.-C. Lee, M.-H. Yang, D. Kriegman, Visual tracking using learned linear subspaces, in: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, 2004, pp. 782–789.
[21] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, 2004.
[22] M. Grant, S. Boyd, Y. Ye. CVX: Matlab software for disciplined convex programming, version 1.0RC3. February 2007. <http://www.stanford.edu/~boyd/cvx/>.
[23] I. Leichter, M. Lindenbaum, E. Rivlin, Tracking by affine kernel transformations using color and boundary cues, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (1) (2009) 164–171.
[24] F. Bajramovic, B. Deutsch, Ch. Gräßl, J. Denzler, Efficient adaptive combination of histograms for real-time tracking, EURASIP Journal on Image and Video Processing (2008) (Article ID 528297).
[25] E. Maggio, A. Cavallaro, Multi-part target representation for color tracking, in: Proceedings of the IEEE International Conference on Image Processing, vol. 1, 2005, pp. 729–732.
[26] M. Isard, A. Blake, CONDENSATION – conditional density propagation for visual tracking, International Journal of Computer Vision 29 (1) (1998) 5–28.