# Evaluation of Multi-Part Models for Mean-Shift Tracking

Darren Caulfield*, Kenneth Dawson-Howe
*Graphics, Vision and Visualisation Group*
*School of Computer Science and Statistics*
*Trinity College Dublin, Ireland*
{Darren.Caulfield, Kenneth.Dawson-Howe}@cs.tcd.ie

## Abstract

*Mean-shift tracking is a data-driven technique for tracking objects through a video sequence. We propose an innovation to mean-shift tracking that combines the background exclusion constraint with multi-part appearance models. The former constraint prevents the tracker from moving to regions where no foreground objects are present, while the multi-part nature of the models enforces a spatial structure on the tracked object. We also use a simple formula to determine the scale of the object in each video frame, and note the importance of setting an appropriate convergence condition. An evaluation of our proposed tracker and several existing trackers is performed using a ground truth dataset. We demonstrate that our innovation yields more accurate tracking than existing mean-shift techniques.*

## 1. Introduction

Since its introduction in 2000 mean-shift tracking has attracted much attention in the computer vision community [2, 4, 5]. As a bottom-up, or data-driven, technique it permits regions of an image to be tracked over time without the need to specify complex motion or appearance models. A simple colour histogram is used to encode the appearance of the object to be tracked, while a spatial kernel enforces a degree of structure on the histogram.

Although mean-shift tracking is popular due to its relative simplicity and computational efficiency, it suffers from a number of weaknesses: it is prone to distraction by other objects similar to the one being tracked; it does not cope well with changes in the scale of the object; and it lacks a mechanism for encoding the spatial layout of the colours the object. To date various researchers have attempted to address these problems. Zhao *et al.* [12] have used the *background exclusion* constraint to make the tracker favour regions that are dissimilar to the background. Collins [1] has developed a method for selecting the appropriate scale for the tracker in each frame when the object's size is changing. In order to enforce a particular spatial structure on the object various multi-part models have been proposed [6, 8, 11].

We have developed a mean-shift-based tracker that utilises both the background exclusion constraint and multi-part appearance models. We have also performed an evaluation of various trackers against a ground truth dataset, which demonstrates that our proposed innovation yields more accurate tracking of its target object. In order to deal with the changing size of the objects as they move through the scene we use a simple formula that relates an object's size to its position in the image. We also note that one of the parameters used in mean-shift tracking – the convergence condition – critically affects a tracker's performance.

The remainder of this paper is organised as follows: section 2 describes previous research into mean-shift tracking, including multi-part models and background exclusion. In section 3 we present our method of combining these two elements in a single tracker, while section 4 details the formula used to select the object's scale at each frame. Section 5 describes the various trackers whose performance we assess and the metrics used in the evaluation. Finally, section 6 presents the results and draws conclusions about the trackers.

## 2. Related research

### 2.1. Basic mean-shift tracking

Mean-shift, or kernel-based, tracking tries to find the area of a video frame that is both (a) most similar to a previously initialised model and (b) close to the tracker's location in the previous frame. By applying the technique to each video frame in sequence a region can be tracked over time. The method was first presented by Comaniciu in 2000

*This work was in part supported by a grant from the Irish Research Council for Science, Engineering and Technology: funded by the National Development Plan.

978-0-7695-3332-2/08 $25.00 © 2008 IEEE
DOI 10.1109/IMVIP.2008.14

[2, 3]. The tracking begins with an object model being created from the region in the first frame. The probability density function (pdf) of the region to be tracked (the *model*) is represented by a histogram $\mathbf{q} = \{q_u\}_{u=1...m}$ where, for each histogram bin $u$,

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

In this equation $k$ is a kernel function that gives more weight to pixels at the centre of the model, and $C$ is a normalising constant that ensures that all of the elements of the histogram sum to 1 (there are $n$ pixels in the model). The function $\delta$ is the Kronecker delta and $b$ is a histogram binning function for each pixel location $\mathbf{x}_i$. Similarly, the pdf of the candidate region $\mathbf{p}(\mathbf{y}) = \{p_u(\mathbf{y})\}_{u=1...m}$ at location $\mathbf{y}$ is given by

$$p_u(\mathbf{y}) = C_h \sum_{i=1}^{n_h} k\left(\left\|\frac{\mathbf{y} - \mathbf{x}_i}{h}\right\|^2\right) \delta\left[b\left(\mathbf{x}_i\right) - u\right] \tag{2}$$

where $h$ is the kernel bandwidth, which determines the size of the candidate region. It is useful to think of $u$ as a colour, but the histograms could actually represent any feature space, e.g edge magnitudes or oriented gradients. The index $i$ ranges over each pixel in the tracked region.

Central to the operation of mean-shift is the weighting $w_i$ for each pixel:

$$w_i = \sum_{u=1}^{m} \sqrt{\frac{q_u}{p_u(\mathbf{y_0})}} \delta\left[b\left(\mathbf{x}_i\right) - u\right] \tag{3}$$

which is derived from the Bhattacharyya similarity measure. ($\mathbf{y_0}$ is the location of the candidate region in the previous frame.) As in [3] we use an Epanechnikov kernel for $k$, so that the new location $\mathbf{y_1}$ for the candidate region is found simply as

$$\mathbf{y}_1 = \frac{\sum_{i=1}^{n_h} \mathbf{x}_i w_i}{\sum_{i=1}^{n_h} w_i} \tag{4}$$

Mean-shift is an iterative procedure, so the above formula must be applied until convergence. The tracker is considered to have converged if the (x,y) locations returned by two successive iterations are separated by less than a particular threshold.

## 2.2. Multi-part models

Both Maggio *et al.* [6] and Dong *et al.* [11] have developed multi-part models that retain much of the structure of the basic histogram (equation 2). Dong divides the region to be tracked into concentric ellipses (figure 1) and adds an extra dimension to the histogram to encode this spatial structure:

$$p_{u,v}(\mathbf{y}) = C_p \sum_{i=1}^{n_p} k\left(\left\|\frac{\mathbf{y}-\mathbf{x}_i}{h_p}\right\|^2\right) \delta\left[b_u\left(\mathbf{x}_i\right) - u\right] \delta\left[b_v\left(\mathbf{x}_i\right) - v\right] \tag{5}$$

where $u = 1...m$ are the colours (as before) and $v = 1...n$ are the ellipses. This results in a modification of equation 3 for calculating the weights:

$$w_i = \sum_{u=1}^{m} \sum_{v=1}^{n} \sqrt{\frac{q_{u,v}}{p_{u,v}(\mathbf{y_0})}} \delta\left[b_u\left(\mathbf{x}_i\right) - u\right] \delta\left[b_v\left(\mathbf{x}_i\right) - v\right] \tag{6}$$

while the optimisation formula (equation 4) remains unchanged.

Although Maggio does not derive the equations explicitly, his approach is very similar to Dong's. However, Maggio's tracker can be used to create models with regions that overlap.

Parameswaran's multi-part model [8] has a different formulation to the others: a separate kernel is associated with each region of the model. For comparison we have included this tracker in our evaluation (section 5).

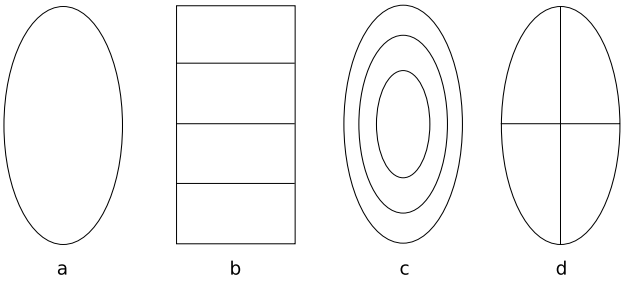


**Figure 1. Spatial division of models for each tracker: (a) basic tracker (b) Parameswaran's (c) Dong's (d) our implementation of Maggio's tracker**

## 2.3. Background exclusion

Various authors have attempted to exploit background models of the scene to improve the performance of mean-shift tracking. Zhao *et al.* [12] and Porikli *et al.* [10] have both modified the pixel weights (equation 3) to take account of the appearance of the background:

$$w_i = \lambda_f w_i^f - \lambda_b w_i^b \tag{7}$$

where the first term represents "object attraction" and the second represents "background exclusion". The idea is that a pixel to be tracked should be similar to the corresponding pixel in the object model but dissimilar to the corresponding

pixel in the background model. The foreground weight $w_i^f$ is calculated as in equation 3, but the background weight $w_i^b$ takes on a more complex form:

$$w_i^b = \sum_{u=1}^{m} \left( \sqrt{\frac{d_u(\mathbf{y_0})}{p_u(\mathbf{y_0})}} \delta \left[ b_f \left( \mathbf{x}_i \right) - u \right] + \sqrt{\frac{p_u(\mathbf{y_0})}{d_u(\mathbf{y_0})}} \delta \left[ b_b \left( \mathbf{x}_i \right) - u \right] \right) \tag{8}$$

where $\mathbf{d}(\mathbf{y}) = \{d_u(\mathbf{y})\}_{u=1\ldots m}$ is the colour histogram of the corresponding region in the background model.

The incorporation of the background exclusion term makes the tracking appreciably more robust and improves its localisation (its positioning on an object).

## 3. Combining background exclusion with multi-part models

We propose an enhancement of the basic mean-shift tracker that is analogous to the work of Pérez *et al.* in the area of particle filters [9]. The goal is to create a mean-shift tracker the has both of the following properties:

- multi-part appearance model: the model to be tracked should be represented by a number of histograms, so that some element of the spatial layout of the object to be tracked is recorded
- background exclusion:. the tracked region should look similar to the model but different to the corresponding region in the empty background scene

We achieve the above aims by combining the background exclusion tracker of Zhao with the multi-part models of Maggio and Dong.

In Pérez's work the likelihood of a candidate region $\mathbf{p}(\mathbf{y})$ at location $\mathbf{y}$ given the model $\mathbf{q}$ is found using the expression

$$\exp -\lambda D^2[\mathbf{q}, \mathbf{p}(\mathbf{y})] \tag{9}$$

where $D$ is the Bhattacharyya distance. For a multi-part model with $J$ regions the expression becomes

$$\exp -\lambda \sum_{j=1}^{J} D^2[\mathbf{q}_j, \mathbf{p}_j(\mathbf{y})] \tag{10}$$

and when a background model $\mathbf{d}(\mathbf{y})$ is available Pérez uses

$$\exp -\lambda \left( D^2[\mathbf{q}, \mathbf{p}(\mathbf{y})] - D^2[\mathbf{d}(\mathbf{y}), \mathbf{p}(\mathbf{y})] \right) \tag{11}$$

Expression 11 has a very similar form to that used by Zhao et al. [12] to exploit "background exclusion" in mean-shift tracking (section 2.3):

$$L(y) = \lambda_f D(\mathbf{q}, \mathbf{p}(\mathbf{y})) - \lambda_b D(\mathbf{d}(\mathbf{y}), \mathbf{p}(\mathbf{y})) \tag{12}$$

We modify Zhao's formula to allow the object, candidate and background models to have $J$ regions:

$$L(y) = \lambda_f \sum_{j=1}^{J} D(\mathbf{q}_j, \mathbf{p}_j(\mathbf{y})) - \lambda_b \sum_{j=1}^{J} D(\mathbf{d}_j(\mathbf{y}), \mathbf{p}_j(\mathbf{y})) \tag{13}$$

The weight $w_i$ for the entire expression on the right-hand side is still given by equation 7. However, $w_i^f$ is now calculated as the sum over $J$ regions:

$$w_i^f = \sum_{j=1}^{J} C_j w_{i,j}^f \delta \left[ b_j \left( \mathbf{x}_i \right) - j \right] \tag{14}$$

where $C_j$ is a normalising constant and $w_{i,j}^f$ is found according to equation 6. Note that equation 14 accommodates overlapping regions in the multi-part model. The new expression for $w_i^b$ with multi-part models can be derived in a similar manner.

## 4. Scale adaptation for tracked objects

There is currently no known application-independent way to adapt the scale of the trackers to accommodate changes in the size of the object being tracked [1]. For this reason we have decided to exploit the application-specific constraints available to us in order to provide an explicit scale for the trackers at every frame. Because the videos we process are recorded by a camera located some distance above the ground, every object's apparent height is a linear function of its lowest image row (see figure 2). For an object whose initial image row is $y_1$ and whose initial size is assumed to be 1, its size $s$ at image row $y$ is found simply according to

$$s = 1 + \frac{(y - y_1) * \text{scale\_adaption\_factor}}{\text{image\_height}} \tag{15}$$

where scale_adaption_factor $< 1$ is the factor by which the object shrinks as it moves away from the camera (from the bottom of the image to the top). This factor is a constant across all objects for a given camera setup. By adapting the scale in this deterministic manner we can avoid the problems that other techniques introduce.

## 5. Evaluation

### 5.1. Trackers

In total we have implemented seven different mean-shift trackers whose performance we will assess. The details of each tracker are as follows (see also figure 1):

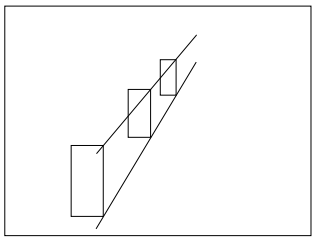

**Figure 2. Scale of an object can be determined as a linear function of its lowest image row**

**Figure 3. Frames of various trackers in operation. The red rectangle shows the ground truth bounding box and the ellipses denote the positions of the trackers (red: basic; green: BG exclusion; blue: Parameswaran).**

- Basic tracker: Comaniciu's original mean-shift tracker (section 2.1) [3]
- BG exclusion: Zhao's background exclusion tracker (section 2.3) [12]
- Parameswaran multi-part [8]
- Dong 3-circle multi-part (section 2.2) [11]
- Maggio 4-quadrant multi-part
- Combined 3-circle: our proposed innovation – combining background exclusion with Dong's multi-part model (section 3)
- Combined 4-quadrant: another version of our proposed innovation – combining background exclusion with Maggio's multi-part model

### 5.2. Convergence condition

We have discovered that the value of the threshold used in the convergence condition of the mean-shift tracker can have a critical effect on its performance. To date, this has not been discussed in the literature. We have found that the default value of 1 pixel (as recommended by Comaniciu in [3]) is suitable for the basic tracker, but causes the multi-part models to perform very poorly. It appears that the steps taken by these trackers at each iteration are smaller than this value, and so a 1-pixel threshold causes the tracker to cease "hill-climbing" prematurely. A value of 0.25 pixels improves the performance significantly.

### 5.3. Metrics for single runs

We have evaluated each tracker's performance with respect to the ground truth bounding boxes of the CAVIAR dataset[1]. Sample frames from the videos with ground truth data and the trackers' positions overlaid are shown in figure 3. We have used a metric from the *Video Performance Evaluation Resource* (ViPER) system [7] called the "dice coefficient" to evaluate the frame-by-frame degree of overlap between the ground truth bounding box and each tracker's bounding box (figure 4). The dice coefficient ($2 * \text{shared\_area}/\text{area\_sum}$) is a symmetric measure and so is less skewed by excessively large tracker bounding boxes than simple overlap measures.

A second metric that we have used measures each tracker's positional accuracy on a frame-by-frame basis (figure 5). It is simply the distance of the tracker centroid from the ground truth centroid. The values of these metrics can be plotted for each frame in a single run of the tracker.

### 5.4. Aggregate metrics over multiple runs

In order to more thoroughly explore the performance of the trackers we have aggregated the above frame-by-frame metrics into two single numbers for each run of the tracker. This allows us to evaluate each tracker's performance over multiple runs, where each run is initialised with a different model.

We display metrics of this kind as box plots, e.g., figure 6, with each column representing a different tracker. The "boxes and whiskers" show the distribution of the data over multiple runs. The top and bottom of the blue box mark the upper and lower quartiles, respectively, while the whiskers extend for 1.5 quartiles in each direction. The median of the data is marked by a red line, and outliers are shown as red crosses. The "notch" in each blue box delimits the 95% confidence interval. If the notches for two trackers do not overlap we can assert that there is a statistically significant performance difference between them.

## 6. Results and conclusions

Figures 4 and 5 show the performance of each tracker

[1]The datasets come from the EC Funded CAVIAR project/IST 2001 37540, found at `http://homepages.inf.ed.ac.uk/rbf/CAVIAR/`

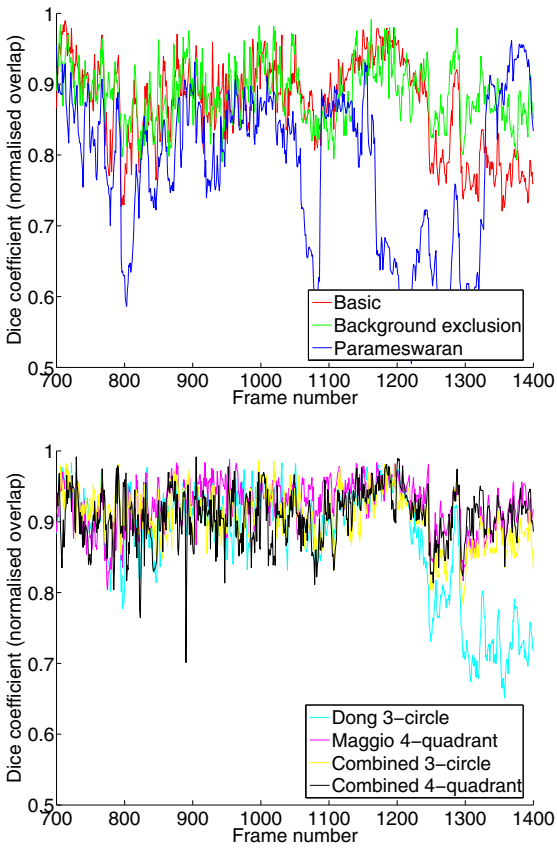


**Figure 4. Dice coefficient for a single run of all seven trackers**

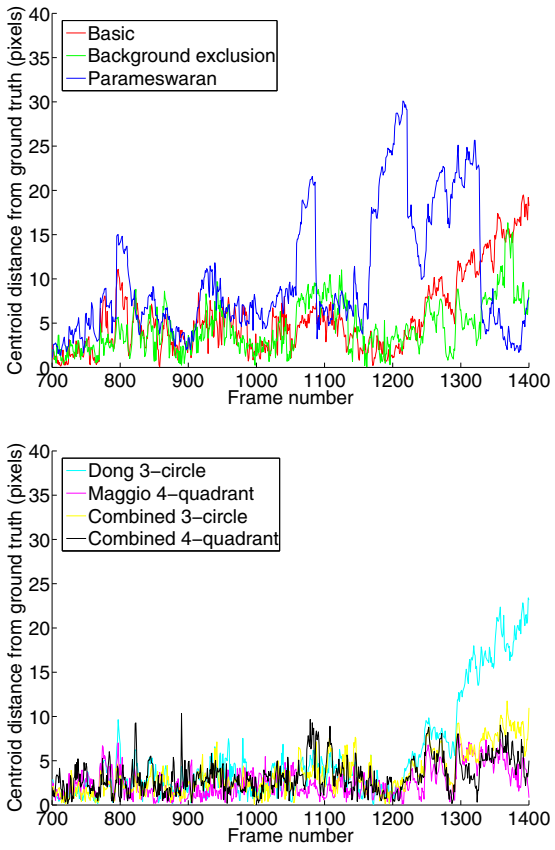


**Figure 5. Centroid distance for a single run of all seven trackers**

over a single run where the woman being tracked is never occluded. Qualitatively, the performance of the the basic, background exclusion and Parameswaran trackers is below that of the others.

We have run the trackers multiple times with different models to track. (In each run the same object is being tracked; we have simply used different frames to initialise the model.) The multiple runs allow us to extract aggregate statistics, which are shown in figure 6. We can see that the two trackers that implement our proposed innovation ("combined 3-circle" and "combined 4-quadrant") display a small but statistically significant improvement over all other trackers.

Table 1 presents the lost-track performance of the trackers for a variety of scenarios: "unoccluded"; "quarter-off target" and "half-off target", where we deliberately initialise the tracker inaccurately (not centred on the person); and three scenarios featuring occlusion (see figure 7). Looking at the first three rows of the table it is clear that only those trackers that use background exclusion can reliably cope with poor initialisation. In the presence of occlusions (last three rows) the results are inconsistent: the BG exclusion and combined 3-circle trackers appear to have comparable lost-track performance, as do Parameswaran and combined 4-quadrant. However, even in those scenarios where Parameswaran's lost track percentage is much lower than for other trackers, its centroid distance and dice coefficient metrics are much worse than the "good" trackers (those based on background exclusion). We will investigate this paradoxical situation in our future work.

## References

[1] R. T. Collins. Mean-shift blob tracking through scale space. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 234–240, 2003.

[2] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean shift. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 142–149, 2000.

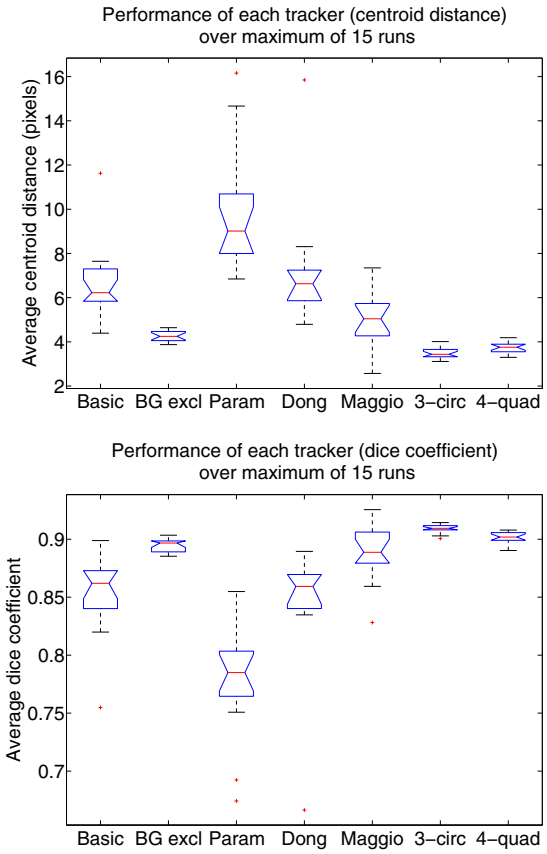


**Figure 6. Aggregate performance of all seven trackers over 15 runs. Our trackers are labelled "3-circ" and 4-quad".**

| Scenario/Tracker | Basic | BG exclusion | Parameswaran multi-part | Dong 3-circle | Maggio 4-quadrant | Combined 3-circle | Combined 4-quadrant |
|---|---|---|---|---|---|---|---|
| Unoccluded | 0 | 0 | 0 | 40 | 7 | 0 | 0 |
| Quarter-off target | 0 | 0 | 20 | 20 | 47 | 0 | 0 |
| Half-off target | 73 | 0 | 93 | 87 | 100 | 0 | 0 |
| Modest occlusion | 17 | 47 | 13 | 27 | 10 | 53 | 10 |
| Severe occl. 1 | 45 | 69 | 72 | 69 | 93 | 55 | 72 |
| Severe occl. 2 | 91 | 76 | 47 | 68 | 94 | 74 | 38 |

**Table 1. Percentage of tracker runs that ended in a lost track for various scenarios**

**Figure 7. Frames from the "modest occlusion" scenario (top row), and one frame from each of the two "severe occlusion" scenarios (bottom row).**


[3] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based object tracking. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(5):564–577, 2003.

[4] Z. Fan, M. Yang, Y. Wu, G. Hua, and T. Yu. Efficient Optimal Kernel Placement for Reliable Visual Tracking. *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Volume 1*, pages 658–665, 2006.

[5] G. Hager, M. Dewan, and C. Stewart. Multiple kernel tracking with SSD. *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, 1:790–797, 2004.

[6] E. Maggio and A. Cavallaro. Multi-Part Target Representation for Color Tracking. *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, 1:729–732, 2005.

[7] V. Mariano, J. Min, J. Park, R. Kasturi, D. Mihalcik, H. Li, D. Doermann, and T. Drayer. Performance Evaluation of Object Detection Algorithms. *International Conference on Pattern Recognition, ICPR02*, pages 965–969, 2002.

[8] V. Parameswaran, V. Ramesh, and I. Zoghlami. Tunable Kernels for Tracking. *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Volume 2*, pages 2179–2186, 2006.

[9] P. Perez, C. Hue, J. Vermaak, and M. Gangnet. Color-based probabilistic tracking. *European Conference on Computer Vision*, 1:661–675, 2002.

[10] F. Porikli and O. Tuzel. Multi-Kernel Object Tracking. *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*, pages 1234–1237, 2005.

[11] D. Xu, Y. Wang, and J. An. Applying a New Spatial Color Histogram in Mean-Shift Based Tracking Algorithm. *Image and Vision Computing New Zealand*, 2005.

[12] T. Zhao and R. Nevatia. Tracking multiple humans in crowded environment. *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, 2:406–413, 2004.