2008 10th Intl. Conf. on Control, Automation, Robotics and Vision
Hanoi, Vietnam, 17–20 December 2008

Multiple Target Tracking with an Efficient Compact Colour Correlogram

Pankaj Kumar, Anthony Dick, Michael J. Brooks
School of Computer Science
University of Adelaide
Adelaide, Australia
pankaj.kumar@adelaide.edu.au, anthony.dick@adelaide.edu.au, michael.brooks@adelaide.edu.au

***Abstract*—A robust approach to detection and tracking of multiple moving targets from a moving camera is presented. The main novelty of this approach is that objects are represented using efficient compact form of the colour correlogram. Like previous correlograms, this encodes both spatial pattern and appearance information about the target. However it is less complex to compute, making it applicable to real time target tracking. The correlogram representation is incorporated into a particle filter tracking framework. Robustness to camera motion is obtained by identifying homographies linking adjacent frames, and using them to align corresponding image areas to a common local reference frame. We demonstrate successful detection and tracking in real life image sequences.**

***Index Terms*—Video Surveillance, Moving Cameras, Detection, Tracking, Particle Filter, Multiple Cue, Correlogram**

I. Introduction

Tracking multiple interacting targets in video generated from a moving camera is a difficult, challenging, and important problem in computer vision. Tracking in complex environments requires object detection, a discriminative object representation, and a dynamic estimation framework that does not constrain camera or object motion. In this paper we address these three aspects of tracking in order to achieve a robust and complete tracking algorithm for tracking multiple targets.

Automatic detection of independent moving targets has been shown to be useful not only for automatic initialization of tracking but also for recovery from errors due to drifting to background regions of the frames [1], [2]. Jinman Kang *et. al.* [3] proposed calculation of pixel statistics by tracking pixel's location in a window of several frames using the affine transform. Further more in [4] Kang *et. al* applied extra constraints of epipolar geometry and structure consistency to the above detection, to more precisely detect independently moving pixels when there is strong parallax in the scene due to the translation of camera center and lack of planarity in the scene.

We propose a general homography transform to align pixels to the common location corresponding to the reference frame in a window of frames. The statistics of each pixel in the reference frame is modelled with Gaussian distributions and then statistically compared to detect foreground pixels in the reference frame. This method is shown to be robust for detection of targets in videos from PTZ cameras and for cases where the background is planar. Simultaneously is it simple enough for real time implementation.

Histograms are commonly used for target representation in tracking applications. Histograms have proven to be robust to partial occlusions, and changes in size, shape and rotation because they discard spatial information. However the loss of spatial information reduces specificity in the model, thus increasing the susceptibility to distraction by the background or other objects. It is desirable for robustness to include spatial information in the representation such that shape and size invariance is still maintained. Thus there have been several approaches to incorporate some spatial information in the target representation. Birchfield and Rangarajan [5], [6] came up with spatiograms, which are histograms augmented with spatial mean and covariances to capture richer description of the target. More recently Zhao and Tao [7], [8] used correlogram based representation in a mean-shift framework for tracking pose and position of an object simultaneously. The computation and representation of correlograms is more expensive than histograms; thus it will be difficult to expect them to work in real time for multiple target tracking applications. We propose a reduced version of the full correlogram which is discriminative in target representation and can be computed as efficiently as a histogram. We then apply the correlogram based representation in a multiple cue particle filter framework, which is well suited to multiple target tracking and can integrate multiple feature cues for estimation of the state vector of the target.

The contributions of this paper are:

1) An algorithm to detect independently moving objects in image sequences captured from a moving camera (Section II).
2) A novel colour correlogram object representation which is efficient to compute (Section III).
3) Use of proposed correlogram in a particle filter frame work for tracking multiple targets (Section IV).

II. Detection of Moving Objects

The usual paradigm for detection of pixels on a moving object in frames from a stationary camera, is to observe pixels for a period of time. Based on these observations a statistical model of each pixel in the frame is built. If a pixel's observation differs significantly from the learnt statistical properties of that pixel then the pixel is considered to be from a moving object [9], [10]. The underlying assumption of this approach is that the colour of a pixel at a particular location in an image sequence is generated from the same 3D location of the scene, unless occupied by a moving object. This fundamental assumption is violated when the camera used for observations is non-stationary. To align frames from a moving camera we use a homography transform $\mathbf{H}$ computed automatically between frames using feature correspondences and RANSAC estimation. The homography transform [11] can align frames when the camera center is fixed or when the scene being observed is planar. These conditions are usually true of videos generated in surveillance systems which use fixed mounted PTZ cameras or cameras on Unmaned Aerial Vehicles (UAV).

For alignment of frames and generation of statistical models of pixels we consider frames in a sliding window of width W around the reference frame I_t, that is $I_{t-W/2}, I_{t+1-W/2}, ..., I_t, ..., I_{t+W/2-1}, I_{t+W/2}$. The Harris corner detector is applied to each frame in the window and features are matched across frames using cross correlation. Matched features are then used to estimate the homographies $\mathbf{H}_{t-W/2}, ..., \mathbf{H}_{t-1}, \mathbf{H}_{t+1}, ..., \mathbf{H}_{t+W/2}$ using the RANSAC method. Each frame $I_{t-W/2}, I_{t+1-W/2}, ..., I_{t-1}, I_{t+1} ..., I_{t+W/2-1}, I_{t+W/2}$ is generated by applying the inverse transform $\mathbf{H}^{-1}$ with linear interpolation.

Thus for a pixel in the reference frame we have W corresponding pixels from $W/2$ past and $W/2$ future frames. This means that processing for detection of moving objects has to be done at a time lag of $W/2$ frames. To build a statistical model of a pixel in reference frame I_t at location (i, j) a histogram $H_{ij}(u)$ of the W aligned pixels is built, where u is the bin index. A window of size $2 \times \omega$ around the mode of the histogram is used to compute the parameters of the Gaussian distribution used to model the pixel.

$$\mu_{ij} = \frac{1}{\sum_{u=(u_{ij}^{max}-\omega)}^{(u_{ij}^{max}+\omega)} H_{ij}(u)} \sum_{u=(u_{ij}^{max}-\omega)}^{(u_{ij}^{max}+\omega)} u \times H_{ij}(u) \quad (1)$$

$$\sigma_{ij} = \frac{1}{\sum_{u=(u_{ij}^{max}-\omega)}^{(u_{ij}^{max}+\omega)} H_{ij}(u)} \sum_{u=(u_{ij}^{max}-\omega)}^{(u_{ij}^{max}+\omega)} (u - \mu_{ij})^2 \times H_{ij}(u) \quad (2)$$

The process is repeated for each channel under the assumption that channels are independent and covariance is a diagonal matrix with cross-covariance terms equal to zero. In our computations we use $W = 50$ and $\omega = 10$.

If the background has motion due to moving leaves, reflective surfaces etc. then these can be modelled by adopting a mixture of Gaussians to model each pixel.

For detection of foreground pixels in the reference frame each pixel in the reference frame is compared with the corresponding pixels statistics:

$$|p_{ij_t} - \mu_{ij_t}| > \beta \times \sigma_{ij_t} \quad (3)$$

where p_{ij_t} is the intensity of one colour channel at pixel i, j in the reference frame, μ_{ij_t} is the mean value of the pixel at i, j, and σ_{ij_t} is the variance of the pixel values. β is an

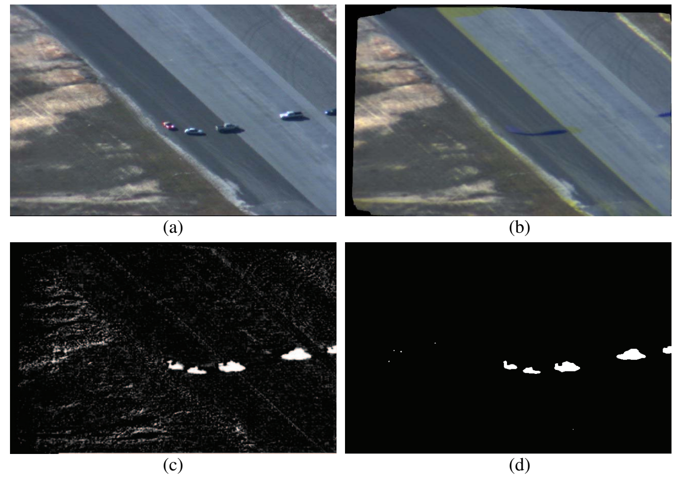

Fig. 1. (a) is a reference frame 411 from PETS2005 dataset sequence EgTest02 (b) is the mean background model of the image sequence with the proposed methods. (c) and (d) are the raw detection results of moving targets and the morphologically filtered detection results, respectively.

empirical threshold parameter. If the above equation is satisfied for two or more colour channels then the pixel is classified as a pixel on a moving object. Pixels detected this way are morphologically filtered and connected into blobs using eight connected component analysis. Each connected component whose pixel count is greater than a threshold is enclosed by a bounding box whose length, breadth and position is measured for data association with current targets being tracked. Figure 1 shows a reference frame and the background model for this reference frame and detection results.

III. Colour Correlogram based Target Representation

A. Overview and Definition

Correlograms are tables showing how the autocorrelation of a signal changes with distance. Spatial analysts adapted this idea to spatial distance [12], and Huang [13] adapted the idea to the spatial separation of pixels in an image to arrive at colour correlograms. The colour correlogram (henceforth correlogram) of an image encodes both colour distribution and spatial distribution of pixels in an image. The correlograms were originally used for indexing and retrieval of images based on a query. As the correlogram of an image I is a table indexed by colour pairs and distance, the size of a correlogram is $m^2 d$ where m is the number of bins for colour quantization and d the number of quantized distances between pixels in the image I. Because $m^2 d$ is large for computation and storage, a subset, the autocorrelogram, was considered for encoding images. Autocorrelogram is a correlogram that includes only the correlation of identical colours; therefore its size is md. Zhao and Tao [7] came up with another simplified version of the correlogram where only fixed distances along major and minor axes were considered. The size of this correlogram was of the order $O(m^2)$. The correlogram was employed for tracking single objects in a 3D mean-shift based algorithm. The advantage of the use of the correlogram in tracking is that

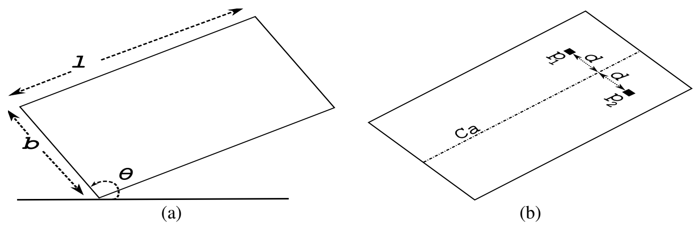

Fig. 2. (a) shows the pictorial representation of an object defined by l, b, and θ in image space. (b) shows a pixel pair around central axis and how they are considered for computing the proposed Colour Correlogram.

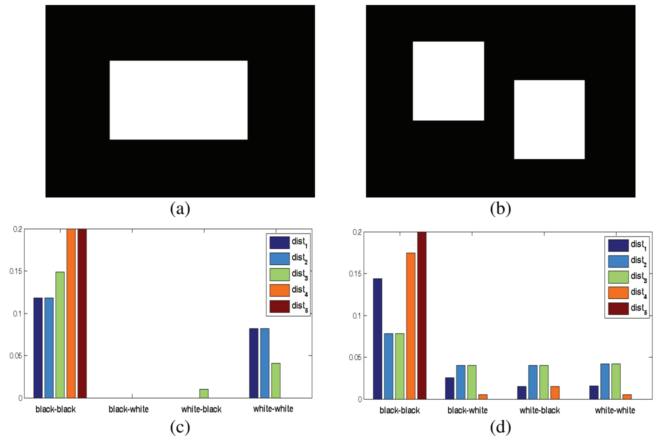

Fig. 3. (a) and (b) are two bilevel images with identical histogram but different spatial arrangements of pixels. (c) and (d) are their correlogram, respectively. In plots (c) and (d) y-axis is the bin score and on x-axis are different bin pairs with different distance bins. The four possible pixels pairs for are: black-black, black-white, white-black, and white-white. Distance amongst pixel pairs are quantized to five different levels namely, $dist_1, dist_2, dist_3, dist_4$, and $dist_5$. The correlogram of images (a) and (b) are substantially different although their histograms are identical.

both translation and rotational movement of object is tracked simultaneously. The disadvantage of the use of correlogram is its high computational requirement.

We present a new correlogram representation which is much faster to compute and can be applied straightforwardly to a particle filter algorithm for tracking multiple targets. The advantage of the proposed correlogram over previous representations is that it is as efficient to compute as a colour histogram. As it is developed for colour pixels, it is more discriminative than gray scale correlograms.

B. Reduced and Simplified Version of Correlogram

An object in an image frame I is defined by its length l, breadth b and orientation θ. θ is the angle between the horizontal axis of the image and one side of the object, as shown in Figure 2 (a). For the generation of correlogram we consider pairs of pixels which are equidistant from the central axis of the object, as shown in Figure 2 (b).

The colour channels in image I are each quantized into m bins $c_1, ..., c_m$ and the distance between pixel pairs p_1, p_2 and the central axis C_a are quantized into d bins $dist_1, ..., dist_d$. Then the correlogram is defined as

$$\gamma_{c_i,c_j}^{dist_k} \triangleq \; prob[p_1 \in I(c_i), p_2 \in I(c_j), \\ |p_1 - C_a| = |p_2 - C_a| \in I(dist_k)].$$

where $I(c_i)$ and $I(c_j)$ is the colour range for bins c_i and c_j respectively and $I(dist_k)$ is the distance range for bin $dist_k$. Note that the index i, j used here are different from those used in equations (1) and (2).

The proposed correlogram is a subset of the full correlogram but it is adequate to differentiate between images which have almost identical histograms. Figure 3 (a) (b) shows two bilevel images whose histograms are identical while their proposed correlograms are significantly different 3 (c) (d). We chose a bilevel image for demonstration purposes because the colour pairs are limited to black-black, black-white, white-black, and white-white and distances are quantized to five bins, so the graphical representation of correlogram is simplified.

The advantage of the proposed correlogram is its computational efficiency. If n is the total number of pixels in the region defined by b, l, and θ then the order of computation for computing correlogram by our method is only $O(n)$ where as for previous methods it is at least $O(nd)$ where d is the number of distance quanta considered.

Input: Image patch $\{I(l, b, \theta)\}$
Output: Correlogram γ
Rotate $I(l, b, \theta)$ by θ to obtain new image I_x
$[\gamma_x,\ sum_x] =$ Correlogram(I_x, l, b)
Rotate $I(l, b, \theta)$ by $\pi/2 - \theta$ to obtain new image I_y
$[\gamma_y,\ sum_y] =$Correlogram(I_y, b, l)
$\gamma = (\gamma_x + \gamma_y) / (sum_x + sum_y)$
End

function Correlogram(I, l, b) returns γ, sum
$\gamma =$ zeros$(c_m, c_m, dist_d)$;
$sum = 0$;
$for\ \ i = 0 : b/2$
$for\ \ j = 0 : l$
$c_i = Q(I(i, j), (256/c_m))$; $Q(a, b) = a/b$
$c_j = Q(I((b - i), j)/(256/c_m))$; %index for pixel pair across C_a
$dist_k = Q(((b - 2i) \times dist_d), b)$; %index for distance of pixel from C_a
$weight = (w(p(i, j) + w(p((b - i), j))/2$; % $w(p(i, j))$ is pixel weight [2]
$sum = sum + weight$;
$\gamma_{c_i,c_j}^{dist_k} = \gamma_{c_i,c_j}^{dist_k} + weight$;
return γ

Fig. 4. Pseudo code for computation of colour correlogram of an image region.

For objects symmetrical along axis C_a and having identical histograms but differing in arrangement of patterns around C_a as shown in Figure 5 (a) (b), this version of the correlogram fails to differentiate images, as shown in Figure 5 (c), (d). In such a case considering pixel pairs along axis perpendicular to C_a solves the problem and results in different correlogram as shown in Figure 5(e) (f). The time complexity of computing this correlogram is still $O(n)$. Pseudo-code for computing the correlogram of a single channel image region defined by l, b, θ is shown in Figure 4.

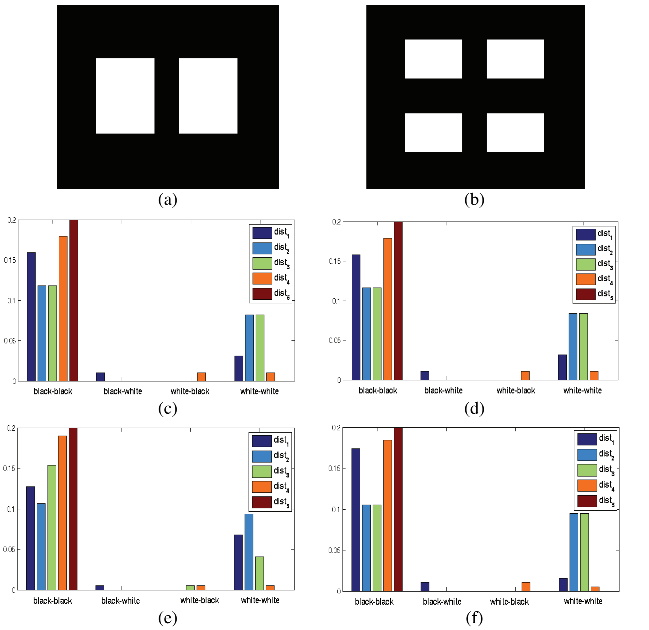

Fig. 5. (a) and (b) are bilevel images with identical histogram but different spatial arrangements of pixels. (c) and (d) are correlogram for distance across axis C_a. The values on x-axis and y-axis are same as those for Figure 3. (e) and (f) are correlogram for distance across axis C_a and axis perpendicular to C_a. Correlograms (c) and (d) are identical but considering distance across axis perpendicular to C_a results in discriminative correlogram (e) and (f).

IV. Integration into Particle Filter based Tracking

Zhao and Tao [7] integrated their colour correlogram in a mean-shift tracking algorithm for tracking a single target in complex scenarios. Our motivation is to be able to track multiple interacting objects in complex situations. Therefore we integrate the correlogram into a particle filter based tracking method. Particle filter based trackers maintain multiple hypotheses of targets being tracked and therefore are more robust in recovering from errors which usually occur when multiple targets undergo occlusions and cross overs. Additionally, the mean-shift tracking algorithm requires a differentiable similarity measure where as for particle filter a non-differentiable likelihood measure will do. We use the L_1 norm for computing the match measure between target correlogram model and candidate correlogram.

A. Target State and Representation

A target is represented by its state X_t and its correlogram model. The state of a target is expressed as $X_t = [x_c, y_c, l, b, \theta]^T$ where x_c, y_c are the centroid co-ordinates and l, b are the length and breadth of the object and θ is its orientation in the image frame. Correlogram of a target is a set of three probability distributions of the three colour channels RGB,

$$X_{\gamma} = \{[\gamma^{dist_k}_{c_i,c_j}]_R, [\gamma^{dist_k}_{c_i,c_j}]_G, [\gamma^{dist_k}_{c_i,c_j}]_B\}. \quad (4)$$

During computation of the probability distribution $[\gamma^{dist_k}_{c_i,c_j}]$ the pixels of the image region are weighed by the distance transform weight $w(p^i)$ [2], which has been shown to be more effective than the usual Epanechnikov kernel weight.

B. Hypothesis Generation and Evaluation

We use the Sequential Importance Resampling (SIR) algorithm for computing the approximation of the probability density function (pdf) of the state vector X_t. In this algorithm the hypothesis/particles are redistributed with equal weights according to the posterior pdf of the state vector in the last iteration. We use a random walk motion model to generate hypothesis/particles for the next iteration. $X^s_{t+1} = X^s_t + v_t$ where v_t is independently distributed zero mean Gaussian noise.

Each hypothesis of the target state is evaluated in the current observation Z_t using correlogram model X_{γ} and segmentation information. A likelihood function is used to compute the weight of each hypothesis integrating correlogram and segmentation cues.

$$\mathcal{L}(Z_t|X^s_t) = \mathcal{L}_{corr}(Z_{CM,t}|X^s_t) \times \mathcal{L}_{seg}(Z_{seg,t}|X^s_t), \quad (5)$$

where

$$\mathcal{L}_{seg}(Z_{seg,t}|X^s_t) = \exp\left(-D_{seg}(X^s_t, Z^q_{seg_t})/\sigma_{z_{seg}}\right), \quad (6)$$

$Z^q_{seg_t}$ is the measurement $(x^q_c, y^q_c, l^q, b^q, \theta^q)$ of the blob from the current frame which matches the target being tracked. $D_{seg}(X^s_t, Z^q_t) = \sqrt{(1 - \exp(-\lambda)}$ and

$$\lambda = [\frac{||X^s_t - Z^q_t||^2}{l^s \times b^s}] \quad (7)$$

In case of merge, where two or more targets merge to give rise to one measurement λ is obtained as follows

$$\lambda = [\frac{(x^s_c - x^{mq})^2 + (y^s_c - y^{mq}_c)^2}{l^s \times b^s}] \quad (8)$$

where x^{mq}_c, y^{mq}_c is the weighted centroid of the foreground pixels in the region defined by hypothesis X^s_t in the current detection result.

1) Correlogram based Hypothesis Evaluation: Previous methods of colour based appearance model for tracking has used the Bhattacharyya coefficient for computing the likelihood [2], [14], [15]. The proposed correlogram is sparse as most of its elements are zero therefore the Bhattacharyya coefficient is not good for differentiating between similar and dissimilar appearance models. Therefore we propose to use the L_1 norm for computing the match measure between the target correlogram model T_{γ} and the correlogram model of the hypothesis X^s_{γ} in the current frame. Thus the likelihood

$$\mathcal{L}_{corr}(Z_{\gamma,t}|X^s_t) = exp(\frac{-D_{corr}(T_{t,\gamma}, X^s_{t,\gamma})}{\sigma_{Z_{corr}}}) \quad (9)$$

and

$$D_{corr}(T_{\gamma}, \quad X^s_{\gamma}) = \sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{k=1}^{d}\Big\{|T_{[\gamma^{dist_k}_{c_i,c_j}]_R} - X_{[\gamma^{dist_k}_{c_i,c_j}]_R}| + |T_{[\gamma^{dist_k}_{c_i,c_j}]_G} - X_{[\gamma^{dist_k}_{c_i,c_j}]_G}| + |T_{[\gamma^{dist_k}_{c_i,c_j}]_B} - X_{[\gamma^{dist_k}_{c_i,c_j}]_B}|\Big\}$$

here the time subscript t has been dropped for simplicity of representation.

C. Target Model Update

To handle the appearance change of the object due to variation in illumination, pose, distance from the camera, etc., the object model is updated using the auto-regressive learning process

$$T_{t+1,\gamma} = (1-\alpha)T_{t,\gamma} + \alpha X^{est}_{t,\gamma}. \tag{10}$$

Here $X^{est}_{t,\gamma}$ is the correlogram of the region defined by the mode of the particles used in tracking the object, and α is the learning rate. The higher the value of α the faster the object model will be updated to the new region. The model update is applied when there is one to one matching between the target and measurements from the current reference frame. In case of merge or split the target model is not updated.

V. Results

For quantitative evaluation of tracking based on the proposed correlogram object model against a colour histogram based object model we use a synthetic image sequence with a static background. The object to be tracked is an overlaid image of a car which rotates clockwise and shrinks in dimension and then expands in dimension. We use this image sequence because ground truth is easily generated by simple frame differencing, using the background image. Figure 6 (a), (b), and (c) shows the tracking results using the colour histogram and 6 (d), (e), and (f) shows the tracking results by the correlogram based object model. We compute and plot two errors for tracking of this sequence: e_{pos} is the position localisation error by the tracker and is computed as

$$e_{pos} = |x_{c_t} - x_{c_{gt}}| + |y_{c_t} - y_{c_{gt}}| \tag{11}$$

and e_θ is the orientation tracking error computed as

$$e_\theta = |\theta_t - \theta_{gt}| \tag{12}$$

Plot 7 shows the quantitative evaluation of tracking. Errors e_{pos} and e_θ are plotted on y axis against frame numbers on the x-axis. The correlogram based tracker has lower overall errors for both position and orientation tracking as compared to the colour histogram based tracking. The average position tracking error (e_{pos}) per frame by histogram model is 5.85 pixels where as by the correlogram model is 4.66 pixels. The average orientation tracking error (e_θ) by the histogram model is 2.54 degrees where as by the correlogram model is only 0.88 degree. This is due to the incorporation of spatial information in the correlogram based object model, where as there is no spatial information in the histogram based object model. In all our experiments we use as little as 50 particles per target to track them. Better results can be obtained by using more particles.

Figure 8 shows the tracking results on sequence *egtest02* from PETS2005 dataset. The tracking results are shown by different colours and patterns of bounding boxs on the vehicles. We choose not to show the trace because the camera is moving. The vehicles have been correctly tracked for position and orientation in spite of their small size, a moving camera and poor quality of the video. Here the tracker failed during cross

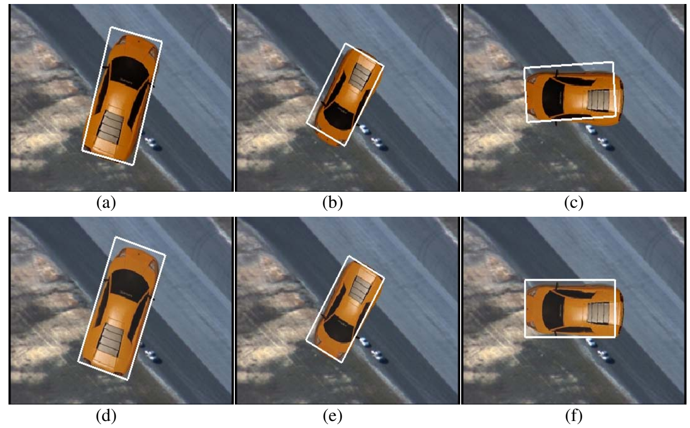

Fig. 6. (a), (b), and (c) shows the tracking results using the colour histogram model and (d), (e), and (f) shows the tracking results using the proposed correlogram model. The tracking of position and orientation by the correlogram object model is better than by the colour histogram model.

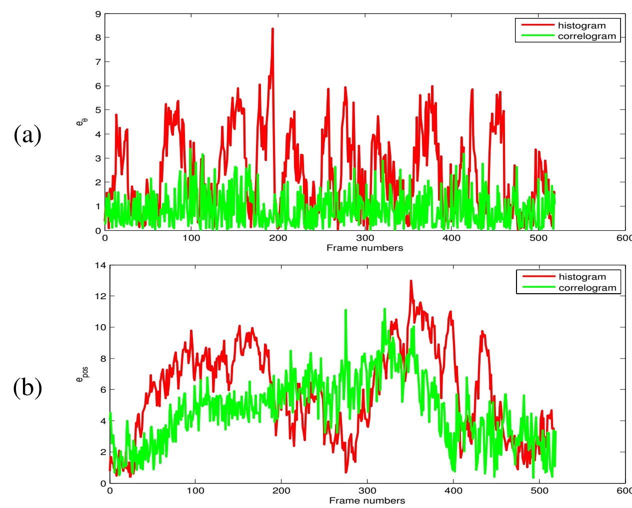

Fig. 7. (a)plot of orientation tracking errors for each frame of the image sequence. (b) plot of position tracking errors for each frame of the sequence. The errors by the correlogram tracking is less than the errors by the colour histogram tracking.

overs. This is because the colours and patterns of the vehicles are very similar, they are small in size, and the quality of the video is poor. Figure 9 shows tracking of people in an image sequence from a moving camera. Here there are no changes in orientation but because of combining colour and segmentation cues in a particle filter frame work the targets has been correctly tracked in spite of almost complete and partial occlusions. From design prespective there is no constraint on the number of objects that can be simultaneously tracked. But as the number of objects increases the computation load also increases.

VI. Conclusions

An enhanced scheme for tracking multiple objects in video has been proposed and demonstrated. Novel contributions of this work include detection of moving objects in image sequence from a moving camera, a new efficient correlogram

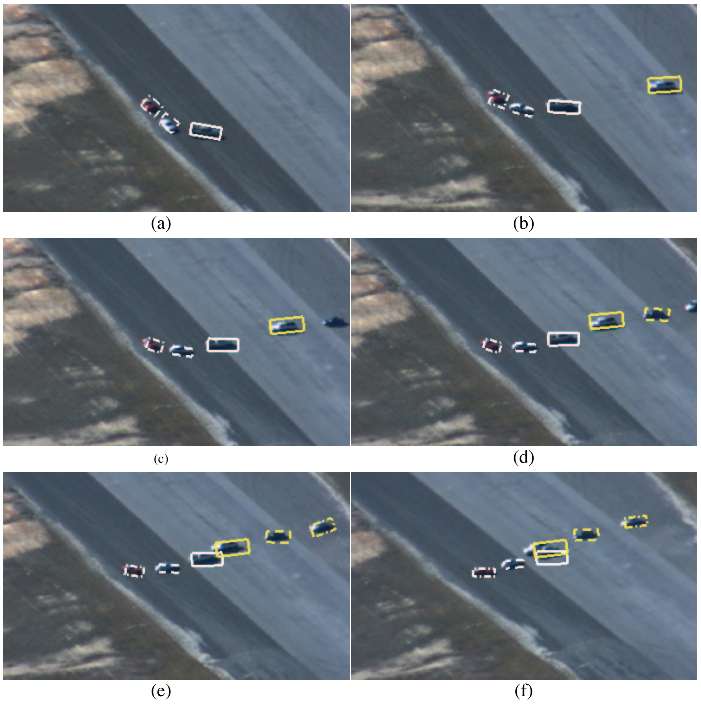

Fig. 8. (a), (b), (c), (d), (e), and (f)shows the tracking results on the *egtest02* sequence. The corresponding frame numbers in the sequence are 301, 385, 420, 440, 460, and 500. The tracking results are shown by different colours and patterns of bounding box on each target. In spite of the small size of the targets and a moving camera the position and orientation of the vehicles have been correctly tracked.

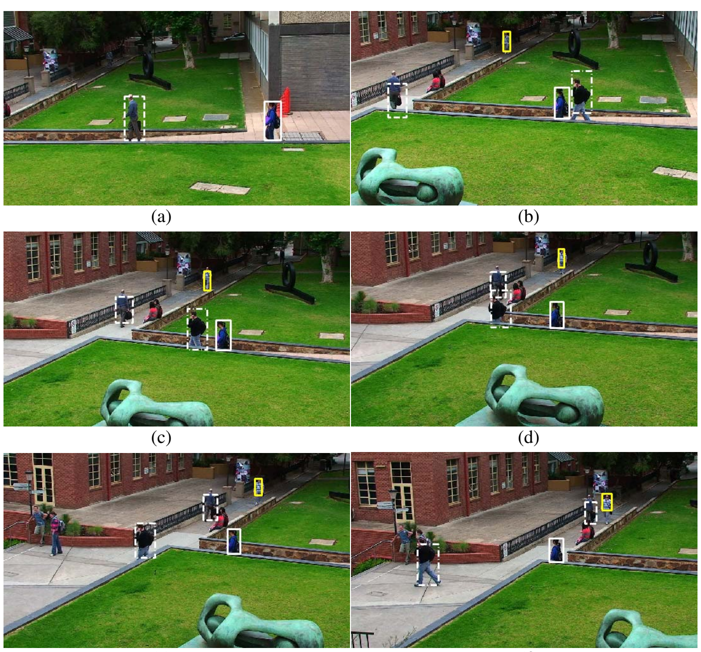

Fig. 9. (a), (b), (c), (d), (e), and (f) shows the tracking results on image sequence of people moving. In spite of targets being almost completely and partially occluded and cross overs the people have been correctly tracked.

model to simultaneously track the orientation and position of targets, and integration of correlogram and segmentation cues in a particle filter frame work for robust tracking of objects. It has been quantitatively demonstrated that the proposed correlogram model based tracking is more accurate than the histogram model based tracking. The measurement obtained from moving object detection is integrated with the correlogram cue to achieve better localisation of the object. The integration of two cues improves handling of objects model undergoing change, rendering the system less susceptible to the drift problem. Furthermore the tracker can follow a target with as few as 50 particles. But when the objects being tracked are similar in colour and patterns then the proposed approach will be unable to disambiguate them during cross overs. In cases where the background doesnot satisfy the planarity condition then more complex detection algorithm [4] can be used.

REFERENCES

[1] P. Kumar, S. Ranganath, K. Sengupta, and H. Weimin, "Co-operative multi-target tracking and classification.," in *ECCV (1)*, pp. 376–389, 2004.

[2] P. Kumar, M. J. Brooks, and A. R. Dick, "Adaptive multiple object tracking using colour and segmentation cues," in *Asian Conference on Computer Vision*, pp. 853–863, 2007.

[3] J. Kang, I. Cohen, and G. Medioni, "Continuous tracking within and across camera streams," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 267–272, 2003.

[4] J. Kang, I. Cohen, G. Medioni, and C. Yuan, "Detection and tracking of moving objects from a moving platform in presence of strong parallax," in *Proceedings of the Tenth International Conference on Computer Vision*, vol. 1, (Beijing, China), pp. 10–17, 2005.

[5] S. T. Birchfield and S. Rangarajan, "Spatiograms versus histograms for region-based tracking," in *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2*, (Washington, DC, USA), pp. 1158–1163, IEEE Computer Society, 2005.

[6] S. T. Birchfield and S. Rangarajan, "Spatial histograms for region-based tracking," *ETRI*, vol. 29, pp. 697–699, october 2007.

[7] Q. Zhao and H. Tao, "Object tracking using color correlogram," in *IEEE Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS) in conjunction with ICCV*, pp. 263–270, 2005.

[8] Q. Zhao and H. Tao, "Motion observability analysis of the simplified color correlogram for visual tracking," in *Asian Conference on Computer Vision*, pp. I: 345–354, 2007.

[9] C. Stauffer and W. Grimson, "Adaptive background mixture models for real time tracking," in *Proceedings of Computer Vision and Pattern Recognition*, pp. 246–252, June 1999.

[10] P. Kumar, S. Ranganath, and W. Huang, "Queue based fast background modelling and fast hysteresis thresholding for better foreground segmentation," in *Proceedings of the ICICS-PCM 2003*, vol. 2, (Singapore), pp. 743– 747, 2003.

[11] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[12] G. J. G. Upton and B. Fingleton, *Spatial data analysis by example*, vol. 1. John Wiley & Sons, 1985.

[13] J. Huang, *Color Spatial Image Indexing and Application*. PhD thesis, August 1998. available at www.cs.cornell.edu/home/huang/thesis.pdf.

[14] C. Dorin, R. Visvanathan, and P. Meer, "Kernel-based object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 564–577, 2003.

[15] K. Nummiaro, E. Koller-Meier, and L. J. V. Gool, "Object tracking with an adaptive color-based particle filter," in *Proceedings of the 24th DAGM Symposium on Pattern Recognition*, (London, UK), pp. 353–360, Springer-Verlag, 2002.