Christopher Madden – Eric Dahai Cheng - Massimo Piccardi*

Tracking People across Disjoint Camera Views by an Illumination-Tolerant Appearance Representation

* Faculty of Information Technology, University of Technology, Sydney (UTS),
PO Box 123, Broadway NSW 2007, Australia
Phone: (+61 2) 9514 7942, Fax: (+61 2) 9514 4535
Correspondence e-mail address: massimo@it.uts.edu.au

Abstract Tracking single individuals as they move across disjoint camera views is a challenging task since their appearance may vary significantly between views. Major changes in appearance are due to different and varying illumination conditions and the deformable geometry of people. These effects are hard to estimate and take into account in real-life applications. Thus, in this paper we propose an illumination-tolerant appearance representation, which is capable of coping with the typical illumination changes occurring in surveillance scenarios. The appearance representation is based on an online *k*-means colour clustering algorithm, a data-adaptive intensity transformation and the incremental use of frames. A similarity measurement is also introduced to compare the appearance representations of any two arbitrary individuals. Post-matching integration of the matching decision along the individuals' tracks is performed in order to improve reliability and robustness of matching. Once matching is provided for any two views of a single individual, its tracking across disjoint cameras derives straightforwardly. Experimental results presented in this paper from a real surveillance camera network show the effectiveness of the proposed method.

Keywords Tracking - Disjoint camera views – Colour histograms - Online *k*-means clustering – Object similarity measurement

I. INTRODUCTION

People tracking is a fundamental function of any video surveillance system as it provides the basis for important surveillance operations, such as behavioural analysis, activity recognition and detection of events of interest. The basic problem at the foundation of tracking is that of correctly associating single physical individuals with their footprint in each frame of a surveillance video. This problem, generally known as probabilistic data association (PDA), was clearly identified in the radar and sonar literature well before video-based tracking became widespread [1]. In video-based tracking, the PDA problem has been approached based on combinations of features such as motion, appearance and shape, which have to undergo some coherency model along the time [2-9]. If the available camera views of single individuals are significantly disjoint in time and/or space, such coherency cannot be easily assessed. This is an increasingly important point for real-life video surveillance applications as disjoint views are dominant in existing, manned CCTV (Closed-Circuit Television) systems. This occurs because human operators do not need to continuously view a person to track it. If automated systems proved capable of effectively tracking across disjoint views, re-use of costly camera infrastructure would be possible and adoption of video surveillance solutions could be improved. In this paper we propose a new approach for matching single individuals from disjoint camera views in typical video surveillance scenarios. Our simplifying assumption is that each person is correctly segmented and tracked within a single camera view and its relevant information (object's mask and pixel values in each frame) is stored into the record of a "track". Our goal is that of finding correspondences between such tracks.

Various papers in the literature have addressed the problem of tracking across multiple, possibly disjoint, cameras ([10-15] and others). Amongst them, the main references for

our work are the recent papers from Javed *et al*. [13, 14]. Their approach proposes an algorithm to compensate for the different illumination conditions by estimating intensity transfer functions between each camera pair during an initial training phase. Such functions are estimated by displaying common targets to the two cameras under a significant range of illumination conditions, and modelling correspondences in the targets' colour component histograms. However, the authors' assumptions in [13, 14] that objects are planar, radiance diffuse and illumination uniform throughout the whole field of view do not generally hold in real applications. Illumination is actually different at each pixel location and has first-order effects on appearance. In [16], Weiss proposed an effective method to estimate illumination in each frame of a sequence and extract a pure reflectance image of the scene. In [17], Matsushita *et al*. have extended the method to deal with time-varying reflectance images. Such methods work well for static scenes; however they cannot accurately estimate the illumination over 3D moving and deformable targets such as people. Actually, accurately estimating illumination over 3D moving targets would require detailed knowledge of the position and parameters of sources of lights, reflections and shadowing and geometry of the target's surfaces. Natural light sources are also time-varying and much hard to predict. We propose an approach that does not rely on either training or estimation. It is based on an appearance representation and a data-adaptive intensity transformation, which can tolerate the illumination variations occurring in typical surveillance scenarios.

The main steps of our approach are as follows: First, we define an appearance representation to be used for each segmented object in a frame. We call this appearance representation the Major Colour Spectrum Histogram Representation (MCSHR) since it aims to describe the object's main colours. An online *k*-means clustering algorithm is proposed here to obtain an accurate MCSHR. Later in the paper, we introduce an *incremental* MCSHR (IMCSHR) to be computed over a short window of successive

frames (typically, three to five) to compensate for small, short–term changes in the object's pose. The problem of varying illumination across disjoint views is mitigated by using an intensity re-mapping of the object's R, G, B components occurring prior to the computation of its IMCSRH. Given two arbitrary segmented objects from two disjoint sequences, a similarity measurement between their IMCSHRs is then proposed to compare their appearance for matching. To increase the reliability of the matching results, post-matching integration is performed along the whole available tracks. Experimental results from real footage under diverse operational conditions have proved this an effective solution to the problem of tracking people across disjoint views.

The rest of the paper is organized as follows: Section II describes the colour histogram representation and the online *k*-means algorithm. Section III presents the algorithm used to reduce the illumination effects on moving objects in the disjoint camera environment. Section IV formally describes the similarity measure between two objects and its extension to two tracks. Section V presents and discusses the experimental results. Conclusions summarise the main results from this work.

II. APPEARANCE REPRESENTATION

In this section, we introduce our definition of colour distance between any two colour pixels based on a normalized geometric distance in the RGB space. Such a colour distance is defined as:

$$d(C_1, C_2) = \frac{\| C_1 - C_2 \|}{\| C_1 \| + \| C_2 \|} = \frac{\sqrt{(R_1 - R_2)^2 + (G_1 - G_2)^2 + (B_1 - B_2)^2}}{\sqrt{R_1^2 + G_1^2 + B_1^2} + \sqrt{R_2^2 + G_2^2 + B_2^2}} \qquad (1)$$

where C_1 and C_2 represent the colour vectors for the two RGB pixels. (1) defines a "normalized" distance in that the Euclidean distance between the two RGB colours is divided by the sum of their magnitudes. This choice is similar to the colour distance

developed by Li *et. al*. [18], which was shown to be robust to illumination changes and noise.

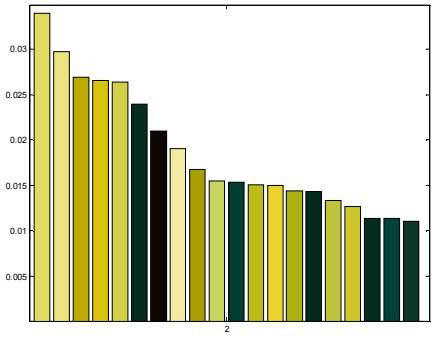
2.1 The proposed Major Colour Spectrum Histogram Representation

Given the concept of colour distance, it is possible to cluster the colours of a segmented object without losing significant accuracy in representing its appearance. Since we aim at real-time application, clustering speed is also a main requirement. Several colour clustering methods are available from the literature [11, 19-24]. In particular, in [11], a method for clustering colours of moving objects was proposed based on a mixture of Gaussians. Each Gaussian component in the mixture is associated with a cluster and the number, relative weights, means and covariances of the Gaussian components are optimised with an Expectation-Maximisation algorithm. In this work, instead, we chose to use a relatively large number of simple spherical clusters which all have the same radius under the normalized distance given in (1). This choice aims to reduce the number of parameters and proves an accurate representation even in the frequent case of data.that do not clearly separate into clusters. The number of the clusters is chosen with a simple heuristic and their positions are optimised with a *k*-means algorithm. Despite the large number of clusters, clustering speed is preserved thanks to the smaller number of parameters to be estimated. We call the set of these clustered colours the Major Colour Spectrum Histogram Representation (MCSHR). In order to efficiently compute the MCSHR, we use the algorithm described in the following.

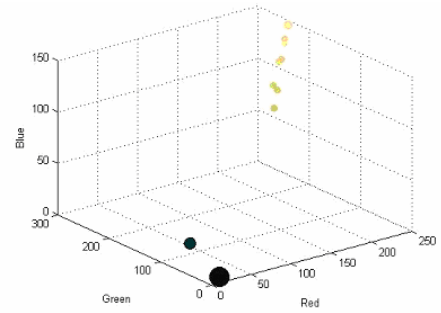
The first step of our algorithm creates the initial set of clusters on which to later apply the *k*-means algorithm. The object's pixels are scanned in row-major order. As the first pixel appears, its colour is set as the centre of the first cluster. If each following pixel is within a threshold under the normalised RGB distance from an existing cluster's centre, the pixel count for that cluster is increased by one; otherwise, a new cluster is created,

centred on that pixel. In the normalised space, this is equivalent to having clusters with a common radius and uniformly spaced. In RGB, instead, clusters are denser at lower magnitudes. This procedure is similar to that proposed by Li *et al*. to calculate their principal colours [25]. Figure 1 shows a picture of a flower containing several tones of yellow and green and the MCSHR outcome of this first step. The original image is depicted in (a) and has 115,537 different colours. The MCSHR outcome contains 839 clusters. In (b), the twenty colours of MCSHR with the highest count are displayed by coloured bars with height proportional to the colour's count. In (c), the ten colours with highest count in the MCSHR are displayed by small coloured spheres with size proportional to the colour's count.

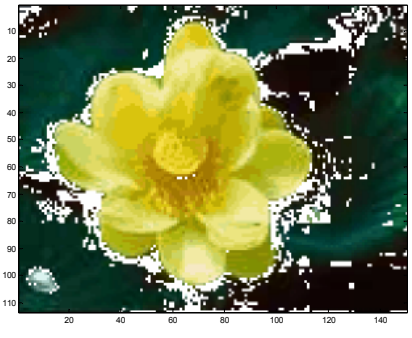
(a) Original 'tn_flower' image



(b) First 20 MCSHR clusters for 'tn_flower' with colour distance threshold of 0.05



(c) MCSHR outcome after the first step for 'tn_flower' with color distance threshold of 0.01 – sphere representation



(d) Back-projection of 297 major colours representing 90% of the pixels onto 'tn_flower'.

Fig. 1 Major colour representation of 'tn_flower'

The colour representation is further simplified by only storing the most common clusters representing 90 percent of the pixels for later use. This reduces the number of clusters from 839 to 297 whilst still representing the majority of pixels in the image. Even in this case where the image has a rich diversity of colours, MCSHR can still provide an adequate representation as demonstrated by the re-projected image in (d) where each original colour has been replaced by the corresponding cluster's colour.

2.2 The online k-means clustering algorithm

Because we have used a simple initial cluster creation procedure, a cluster's centre may be significantly displaced with respect to the cluster's centroid i.e. the average position of its member pixels. In our experiments, we found that this may affect the comparisons between object representations. Thus, we use a *k*-means algorithm to refine the clusters' centroids [26]. The *k*-means algorithm is an expectation-maximisation technique iteratively alternating membership calculation and centroid adjustment. Such algorithms are notoriously sensitive to the initial choice of parameters as they converge to local optima. In our application, however, the heuristic for the initial step allows the *k*-means algorithm to start from reasonable initial values, thus the local optima prove to be adequate solutions in general. Our online *k*-means major colour clustering algorithm works as follows: the objects' pixels are scanned in row-major order. For the current pixel, the closest cluster centre is computed and the pixel assigned to it. Then, the centre of this cluster is updated as:

$$\begin{cases} R_c(i) = w(i)R(i) + (1 - w(i))R_c(i - 1) \\ G_c(i) = w(i)G(i) + (1 - w(i))G_c(i - 1) \\ B_c(i) = w(i)B(i) + (1 - w(i))B_c(i - 1) \end{cases} \quad (2)$$

where i is the current number of pixels in the cluster, $R(i)$, $G(i)$, $B(i)$ are the RGB components of the i-th (current) pixel, $R_c(i)$, $G_c(i)$, $B_c(i)$ are those of the cluster's centre after the i-th pixel has been processed, and $w(i) = 1/i$ the current weighting coefficient.

One can see that with the increase in the number of pixels falling into a cluster, the weighting coefficient decreases. Changes in the centroid position tend to gradually slow down. Since cluster centres are moving, iterations are necessary until all pixel assignments and cluster centres stabilise. In our experiments, between 80% and 90% of pixels are usually already member of their final cluster after the first iteration. Figure 2 shows the picture of another flower (an Ore Gold rose) again rich in tones and nuances. Figure 3 shows its MCSHR calculated by using the described online *k*-means clustering algorithm.

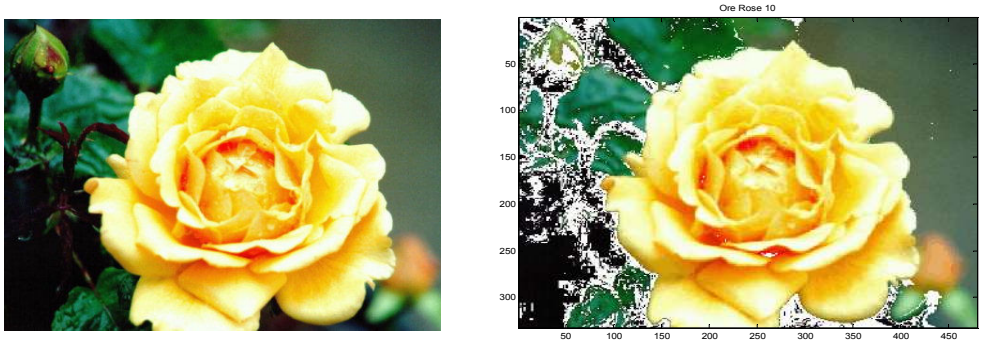
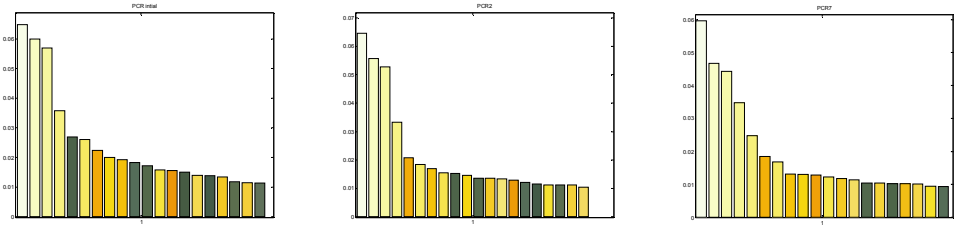


Fig. 2 The original Ore Gold rose image (left) and the back-projection based on the 90% most common pixel clusters after 7 *k*-means iterations (right)

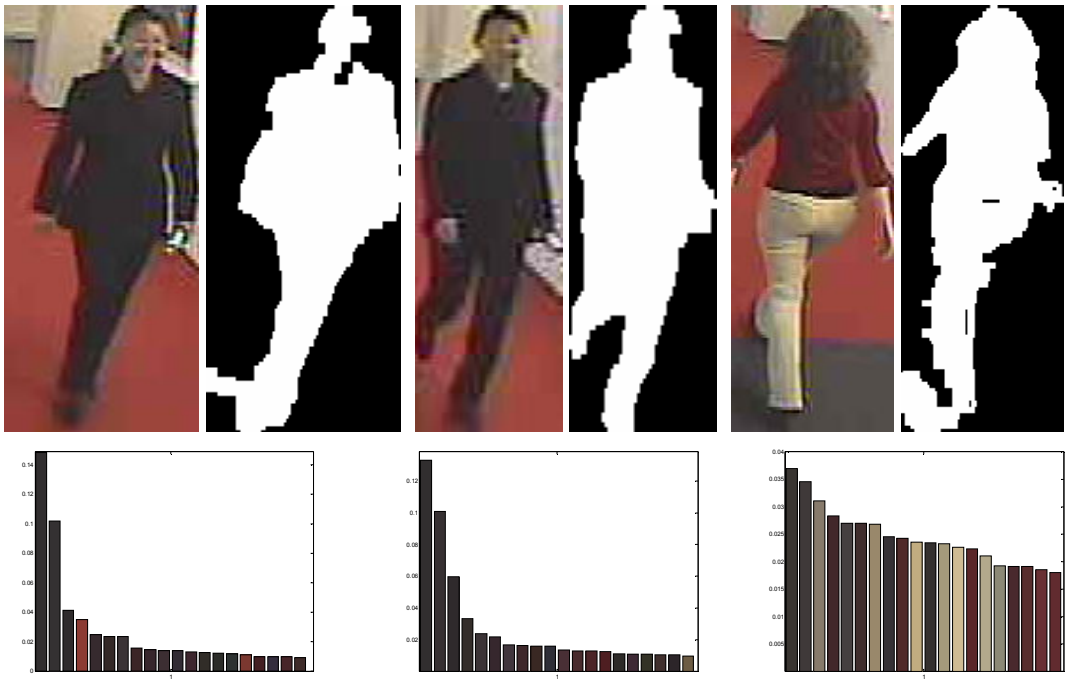


(a) Iter 0, 20 most major colours (b) Iter 2, 20 most major colours (c) Iter 7, 20 most major colours

Fig. 3 MCSHR with the proposed online *k*-means clustering algorithm

Figures 2 and 3 show that the MCSHR computed using the online *k*-means clustering algorithm is visually accurate. Figure 3 shows that no major improvement was made by increasing the number of iterations from two (b) to seven (c), yet the increase in computation time is significant, especially for larger images. Figure 4 shows the MCSHR from objects automatically segmented from three different frames from a single camera. The similarity between the MCSHR for frames 775 and 1297

demonstrates the ability of this representation to capture the dominant colours, which also appear similar in the frames. The MCSHR is clearly distinct in frame 997 where a different person is observed in the same area.



(a) Person A, frame 775 (b) Person A, frame 1297 (c) Person B, frame 997

Fig. 4 Images, masks, and MCSHR from three automatically detected people as seen from the 5_corridor camera

III. COMPENSATING FOR VARYING ILLUMINATION ACROSS DISJOINT VIEWS

The greatest challenge for matching moving objects from disjoint camera views is in the different and varying illumination, which can cause significant differences in appearances. This occurs even under the assumption of either artificial white or natural light sources that typically occur within building environments. The geometry of the objects are also subject to deformation and self-shadowing. As explained in section one, the computation of an exact transformation justifying the changes in appearance can prove unfeasible or impractical. For this reason, we propose to use a fixed, data-adaptive intensity transformation we call ‘controlled equalisation’. It is based upon a modified cumulative colour histogram computed locally to each object.

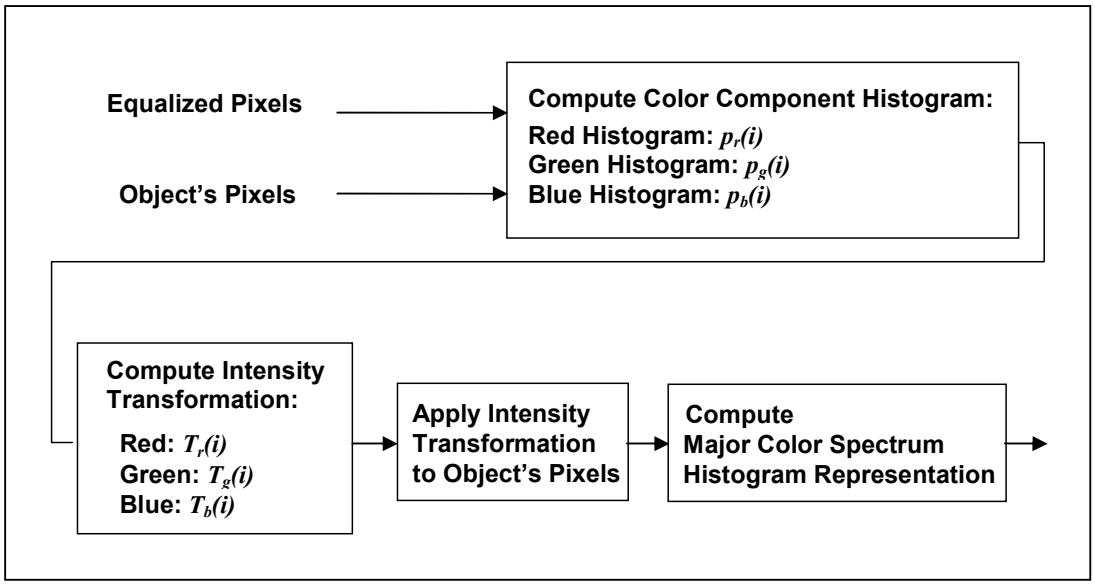


Fig. 5 Compensating for varying illumination across disjoint views: overview

3.1 The intensity transformation

Let us call A the set of the N pixels in a generic object. Let us also call B a second set of N pixels, perfectly equalized in their R, G, B components. On their union, $A \cup B$, the histograms of the R, G, B components, $p_r(i)$, $p_g(i)$, $p_b(i)$, $i = 0...255$, are computed. Then, a cumulative histogram transformation (or equalization) is derived from each component's histogram as shown in (3-5).

$$T_r(i) = \frac{255}{2N} \sum_{j=0}^{i} p_r(j) \tag{3}$$

$$T_g(i) = \frac{255}{2N} \sum_{j=0}^{i} p_g(j) \tag{4}$$

$$T_b(i) = \frac{255}{2N} \sum_{j=0}^{i} p_b(j) \tag{5}$$

The three resulting intensity transformations are then applied to re-map the R, G, B components in the moving object's pixels. Thus, the "controlled equalisation" can be applied to any object to compensate for local illumination without requiring either training or other assumed scene knowledge. Figure 4 shows the application of the

proposed approach. An example of application is shown in Fig. 5. The object, its mask and MCSHR are shown in (a-c). The object and its MCSHR after the intensity transformation are shown in (d and e). The original histogram for the R component is shown in (f). After the proposed intensity transformation (g), the histogram qualitatively retains its original shape (h), but extends over the available spectrum, thus compensating for local illumination variations. The effect of the proposed transformation is not to be confused with that of full equalization, which would result in a flat histogram. Some analogy may instead be seen with histogram stretching algorithms. However, such algorithms are very sensitive to the stretching parameters (original starting and ending bins and/or position and number of modes) while the proposed transformation does not rely on explicit parameters.

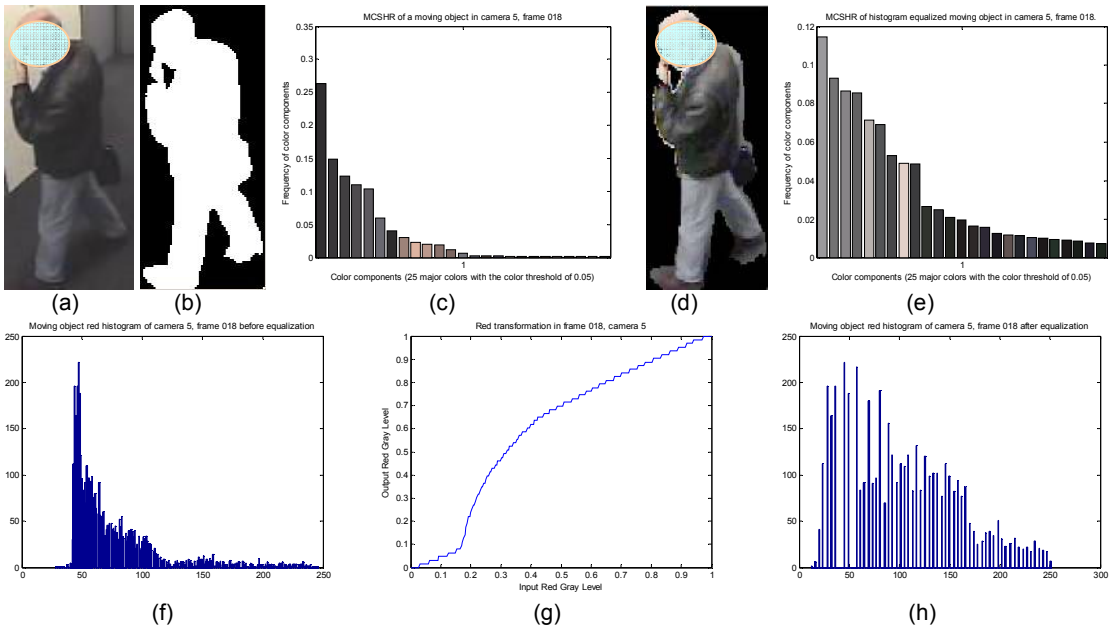


Fig. 6 Effects of the proposed intensity transformation: (a) original object; (b) its mask and (c) major colours; (d) the object after intensity transformation and (e) its major colours; Red histograms: (f) before and (h) after the transformation, and the applied transformation (g). (h) makes better use of the available spectrum than (f), and "normalises" the histogram with respect to local illumination

3.2 The incremental MCSHR

After computing the object's MCSHR for each frame in its track, we integrate each MCHSR over the window of the last K frames, with K a small value. The optimum

window size differs for different camera speeds and gait periods, with ideally one step or half a gait period. This aims to keep the window short, yet provide maximum information about objects appearance under pose variation along the track. For our data obtained at approximately six frames per second, this value was three frames. Indeed experimental results indicated that very marginal improvements were made with a larger window size. We call this augmented representation the incremental MCHSR (IMCSHR). Given the MCSHR of an object, *A*, at frame *q* represented as

$$\boldsymbol{MCSHR}(A_q) = \{C_1(A_q), C_2(A_q), \cdots, C_M(A_q)\} \tag{6}$$

where $C_{i_i}, i = 1,2,\cdots,M$ are the major colours' centres and

$$p(A_q) = \{p_1(A_q), p_2(A_q), \cdots, p_{M_q}(A_q)\} \tag{7}$$

their bin counts, the IMCSHR of the object at the *q*-th frame can be represented as:

$$\boldsymbol{IMCSHR}(A_q) = \sum_{k=-(K-1)}^{0} MCSHR(A_{q+k}). \tag{8}$$

$$p_{\text{IMCSHR}}(A_q) = \sum_{k=-(K-1)}^{0} p(A_{q+k}). \tag{9}$$

The $\sum$ sign in (8-9) is used here to mean a special "summation", i.e. the merging accumulation of the MCSHR's of frames $(q-(K-1)),..,q$ based on the colour threshold. *K* was set to three in all experiments reported in this paper.

The combination of the IMCSHR representation and the proposed intensity transformation proved tolerant to illumination variations of typical surveillance scenarios. In the remainder of the paper, the acronym MCSHR is used also to indicate this incremental representation for the sake of concise notation.

IV. TRACK MATCHING

After determining the appearance representation, a similarity measurement is needed to quantify the overall similarity between any two MCSHRs. For this purpose, we could use a standard distribution distance such as the Kullback-Leibler divergence to compute the distance between the two MCSHRs and use its reciprocal as the similarity [27]. However, we prefer to compute the similarity measurement directly. In this section, we present a method based on a most-similar colour searching algorithm. Later, the matching is extended along the tracks using a simple post-matching integration algorithm.

4.1 Similarity measurement

We assume that there exist *M* major colours in object *A* which can be represented as:

$$\boldsymbol{MCSHR}(A) = \{C_{A_1}, C_{A_2}, \cdots, C_{A_i}, \cdots, C_{A_M}\} \tag{10}$$

with their bin frequencies represented as:

$$\mathrm{p}(A) = \{p(A_1), p(A_2), \cdots, p(A_i), \cdots p(A_M)\}\,. \tag{11}$$

Object B can be represented similarly over *N* colours by the ***MCSHR****(B)* and ***p****(B)* vectors. In order to define the similarity between two objects, a subset of ***MCSHR****(B)* is firstly defined as:

$$\boldsymbol{MCSHR'}(B \mid C_{A_i}, \sigma) = \{C_{B_1'}, C_{B_2'}, \cdots, C_{B_L'}\} \tag{12}$$

where the distance between $C_{B_j'}, j = 1,2,\cdots,L$ and C_{A_i} is less than a given threshold, σ. This subset represents the colours clusters that are considered to be close enough to C_{A_i} to be potential matches. $C_{B_j|A_i}$ is defined as the most similar colour to C_{A_i} in subset ***MCSHR****'(B)* satisfying:

$$C_{B_j|A_i} : j = \underset{k=1,\cdots,L}{arg\,min}\{d(C_{B_k'}, C_{A_i})\} \tag{13}$$

We define the similarity of colours C_{A_i} and $C_{B_j|A_i}$ as:

$$Sim(C_{A_i}, C_{B_j|A_i}) = \min\{p(A_i),\ p^{[A_i]}(B_j)\} \quad (14)$$

where $p^{[A_i]}(B_j)$ is the frequency of $C_{B_j|A_i}$. The min operator in (14) is used to retain the "common part" of $p(A_i)$ and $p^{[A_i]}(B_j)$ as the similarity between the two colours. It is possible to note that their "different part", or absolute difference, $\left|p(A_i) - p^{[A_i]}(B_j)\right|$, is the well-known Kolmogorov distance under equal priors [27]. In this sense, the similarity measurement presented here is analogous to the complement of the Kolmogorov distance. The similarity between the whole objects, *A* and *B*, in the direction from *A* to *B* is then given by:

$$Sim(A,B) = \sum_{i=1}^{M} Sim(C_{A_i}, C_{B_j|A_i}) \quad (15)$$

The similarity between *A* and *B* in the direction from *B* to *A*, *Sim*(*B*,*A*), is defined in a similar way. Note that *Sim*(*B*,*A*) generally differs from *Sim*(*A*,*B*) as for any given $C_{B_j|A_i}$ and $C_{A_k|B_j}$, $i \neq k$. To derive a symmetric similarity measurement we first take their minimum and maximum:

$$Sim_{min}(A,B) = min\{Sim(A,B), Sim(B,A)\} \quad (16)$$

$$Sim_{max}(A,B) = max\{Sim(A,B), Sim(B,A)\} \quad (17)$$

and eventually combine them into a single final value, *Similarity(A,B)*, defined as follows: if $Sim_{min}(A,B)$ is less than a given discrimination threshold, $\eta_{discrim}$, the similarity of objects *A* and *B* is defined as:

$$Similarity(A,B) = Sim_{\min}(A,B) \quad (18)$$

The rationale in this case is that *Sim*(*A*,*B*) and *Sim*(*B*,*A*) are either very asymmetric or both low and for this reason we decide to bound *Similarity*(*A*,*B*) by their lowest value. Instead, if $Sim_{min}(A,B)$ is $\geq \eta_{discrim}$, we define:

$$Similarity(A,B) = 1 - \frac{Sim_{max}(A,B) - Sim_{min}(A,B)}{Sim_{max}(A,B) + Sim_{min}(A,B)} \quad (19)$$

In this case, we are confident that the two visual objects are possibly a same physical one. As a further verification, we choose to check the difference between the maximum and minimum similarities in a ratio form. In (21), a large difference between the maximum and minimum similarity leads to a lower similarity value. The definition of *Similarity(A,B)* in (20, 21) aims to prevent asymmetric, partial matches between two objects and let us set a final similarity threshold for matching assessment more easily. In practice, all the measurements above are computed over IMCHSR values.

4.2 Post-matching integration

In order to evaluate the matching between the two tracks of objects A and B over a sequence of N frames, two basic alternatives are possible: i) either extending the object representation to cover whole tracks and performing a single, overall matching operation, ii) or repeatedly comparing pairs of IMCSHR from the two tracks and integrating the results. We believe that the latter is intrinsically more robust to large segmentation errors which may occur occasionally at the frame level. Thus, in our approach we compare IMCSRH pairs in frame order before making a binary decision on their matching. These decisions are integrated along a minimum number of N frames. Two tracks are considered to be matching if the percentage of successfully matched IMCSRH pairs is above a threshold. The choice of comparing frame pairs in frame order is arbitrary and is aimed at keeping a linear computational complexity, $O(N)$, for the algorithm. Linear computational complexity in the number of frames is the minimum reasonable complexity for matching over a frame sequence and allows the algorithm to meet real-time constraints. It also makes an *on-line* version of the post-matching integration possible: given the surveillance application scenario, the two tracks cannot be acquired from a single individual at the same time.

The track matching algorithm

Given two tracks, A, B:

A preparatory loop for the first K-1 frames:

For each i = 1:K-1

1. Compensate illumination of frame A_i as per Section III;
2. Compute MCSHR(A_i) as per Section II;
3. Repeat steps 1-2 for B_i

The actual loop along the tracks. K is typically set to 3. N is the common length of the two tracks:

For each i = K:N

1. TotalSimilarity = 0;
2. Compensate illumination of frame A_i;
3. Compute MCSHR(A_i);
4. Compute IMCSHR(A_i) as:
   IMCSHR(A_i) = 0;
   For each k = K-1:0
   IMCSHR(A_i) = IMCSHR(A_i) + MCSHR(A_{i-k});
5. Repeat steps 1-3 for B_i ;
6. Compute the similarity between IMCSHR(A_i) and IMCSHR(B_i) as per Section IV;
7. If similarity >= .80, S = 1; else S = 0;
8. TotalSimilarity = TotalSimilarity + S;

The final decision:

If TotalSimilarity >= .80, objects in tracks A and B are matched; else, unmatched

The MCSHR(A_i) algorithm

The initial step:

cluster list = {empty};
For each pixel p in A_i

1. Find the closest cluster in the cluster list;
2. If cluster more distant than threshold, create new cluster;
   else, increment cluster count;

The k-means iterations:

For j = 1:2
For each pixel p in A_i

1. Find the closest cluster in the cluster list;
2. If cluster different than current cluster,
   increment new cluster count and adjust cluster centre;
   decrement old cluster count and adjust cluster centre;

The final step:

Sort clusters in descending count order and return the first up to 90% of the number of pixels

Algorithm Panel: The track matching and MCSHR algorithms

Assuming one track has already been recorded in the system and the other is forming, matching can be stated as soon as *N* frames from the forming track become available. Figure 6 shows a temporal display of this post-matching integration algorithm.

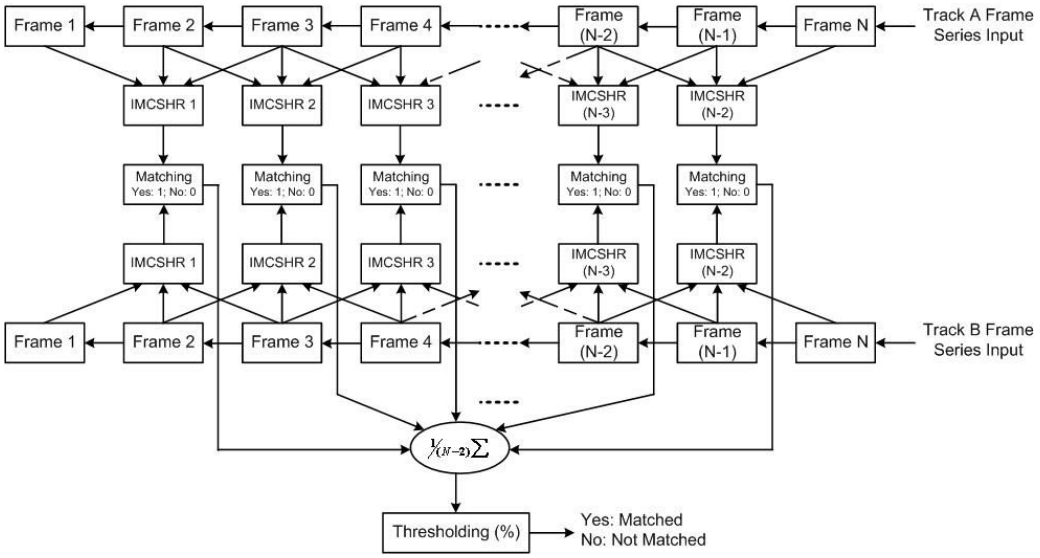


Fig. 7 IMCSHR matching and post-matching integration

V. EXPERIMENTAL RESULTS AND ANALYSIS

In this section, we report results from four disjoint video surveillance cameras installed in the Faculty of Information Technology main building at the University of Technology, Sydney. The cameras are operated daily for surveillance purposes by the university's security services and have not been installed or chosen to ease the performance of automated video surveillance tasks. In order to properly evaluate the proposed approach, the results presented in sections 5.1, 5.2, and 5.3 were obtained from objects that have been tracked and segmented with manual intervention. Results presented in section 5.4 are based upon automatically tracked and segmented objects, but otherwise utilise the same procedure and parameters. In this way, we are able to evaluate performance of the proposed appearance representation against substantial illumination variations and the deformable geometry of people.

5.1 Matching of a same moving person in disjoint camera views

This section reports on manually segmented data from the same person recorded from two video surveillance cameras (camera *3a*, frames 001-019, and camera *5*, frames 300-318). The two cameras are significantly disjoint in both space and time, as shown in Figure 4, and the person's appearance in the two tracks could not be trivially matched. Moreover, illumination varies significantly with the object's position within each camera view. The results given in Table 1 show that the IMCSHR matching and post-matching integration are capable of coping with such variations in appearance, and the person is reliably matched.

Fig. 8 Moving objects from camera *3a*, frames 001-009 and camera *5*, frames 300-308

Table 1 Results of IMCSHR Matching - same person

Test Case	Frame No	Camera	Similarity	Matching Results
1	001-005	*3a*	0.9817	1 (Yes)
	300-304	*5*		
2	003-007	*3a*	0.9758	1 (Yes)
	302-006	*5*		
3	005-009	*3a*	0.9772	1 (Yes)
	304-308	*5*		
4	007-011	*3a*	0.9856	1 (Yes)
	306-310	*5*		
5	009-013	*3a*	0.9452	1 (Yes)
	308-312	*5*		
Integration	001-019	*3a*		100% (Match)
	300-318	*5*		

Notes: with 90% major colours cut off, colour threshold = similarity colour threshold = 0.05, discrimination threshold = 0.4, IMCSHR matching threshold = 0.8, and final integration matching threshold = 80%.

5.2 Matching of two different people from disjoint camera views

This section reports on the manually segmented results from two different people recorded from the same camera pair as in the previous example (camera *3a*, frames 001-019, and camera *5*, frames 010-022), with some of the frames shown in Fig. 8. Table 2 reports the IMCSHR matching and post-matching integration results that demonstrate the two moving objects are correctly discriminated. The integrated matching rate is only 40%, which is significantly lower than our 80% threshold.

Fig. 9 Moving objects from camera *3a*, frames 001-009, and camera *5*, frames 010-018

Table 2 Results of IMCSHR Matching - two different people

Test Case	Frame No	Camera	Similarity	Matching Results
1	001-005	*3a*	0.3538	0 (No)
	010-014	*5*		
2	003-007	*3a*	0.7588	0 (No)
	012-016	*5*		
3	005-009	*3a*	0.7224	0 (No)
	014-018	*5*		
4	007-011	*3a*	0.8348	1 (Yes)
	016-020	*5*		
5	009-013	*3a*	0.8075	1 (Yes)
	018-022	*5*		
Integration	001-019	*3a*		40% (No match)
	010-022	*5*		

Note: The parameters used are the same as those given for Table 1.

5.3 Comprehensive matching test on manually segmented object tracks

Five sets of additional track matching results from camera views disjoint either in space or time, or both are reported in Table 3. These results show that the proposed method is

capable of both correct matching, and discriminating between different individuals. The differences in matching rates for same and different individuals are significant and allow easy discrimination by thresholding (we use an 80% threshold).

Table 3 Comprehensive Results of IMCSHR Matching

Test Case	Frame No	Camera	Typical frame similarity (IMCSHR)	Integrated matching rates
1 (Same object, time disjoint)	282-294	*3_0*	0.9785	80% (4 out of 5 matched)
	001-013	*3a*		
2 (Same object, space disjoint)	001-013	*3a*	0.9817	100% (5 out of 5 matched)
	300-312	*5*		
3 (Different objects, time and space disjoint)	050-062	*4*	0.3696	20% (4 out 5 discriminated)
	010-022	*5*		
4 (Same object, time and space disjoint)	282-294	*3_0*	0.8410	100% (5 out of 5 matched)
	300-312	*5*		
5 (Different objects, space disjoint)	050-062	*4*	0.3696	20% (4 out 5 discriminated)
	010-022	*5*		

In test case 1, the same person is viewed under a same camera in the morning and the afternoon. In the morning view there is a significant amount of natural light in the right part of the scene (with resemblances to a typical outdoor view) while artificial illumination is predominant in the left and central parts. In the afternoon the whole view is dominated by artificial illumination, with slight changes in chromaticity. Variations in the intensity of the *R*, *G*, *B* components for the moving object across and between such views are in the order of 25-30% and would not allow trivial colour histogram matching. The object is successfully matched using the method we have proposed with an integrated matching rate of 80%. The other test cases cover a variety of disjointedness in time and space. Cases 2 and 4 show the same object successfully matched with an integrated matching rate of 100%. In test cases 3 and 5, different objects are successfully discriminated because of an integrated matching rate of 20%.

Fig. 10 Typical frames used for test cases 1-5

5.4 Results using automatically tracked and segmented objects

The test cases presented in sections 5.1, 5.2, and 5.3 demonstrate that the method works reliably with manually selected and segmented objects. This section presents results obtained by automatically tracking and segmenting objects from two cameras. The cameras are named 5_corridor and 5_lifts and provide views with object movement being restricted in different directions, and significant areas of different background colour. Cameras were chosen to ensure that lighting conditions and background colour of the areas of interest are different throughout the majority of the scene, as shown in Figure 11. Within these cameras four people of interest are studied wearing different coloured clothing. Their appearance and typical segmentation masks are shown in Figure 12. The clothing was selected as they are typical to indoor environments and are not intended to be of high contrast to the background for easy segmentation.

Fig. 11 Background views from the two cameras: 5_Corridor left, 5_lifts right

Fig 12 Four people of interest (Persons A, B, C, D from left) and typical automatically segmented masks (from frames 775, 1095, 1542, 2044)

Four main areas of interest are considered in the automated results presented. Case 1 presents the results of matching the same individual from different tracks obtained within the same camera. Case 2 examines matching tracks from the same individual

between different cameras. Case 3 examines the differentiation of the tracks of two individuals within the same camera. Case 4 examines the differentiation of the tracks of two individuals between different cameras. Results from the matching of the tracks of an individual are also presented separately, where one track of that individual has regions where the segmentation is cluttered by other individuals in the scene.

The automatic segmentation of individuals is important as significant amount of oversegmentation or undersegmentation can alter the levels of colours represented in the IMCSHR. The results presented in this paper are based upon background subtraction, where the background model is created using a running Gaussian average [2]. A 2 pixel controlled dilation is used the amount of noise, especially in the 5_corridor scene where natural illumination is strong. Colour region growing is also used around identified foreground pixels. Typical segmentation results are shown in Figure 12, indicating that whilst dark colours are segmented well, light colours, and even some degree of facial skin are hard to detect. Such segmentation errors are currently typical within areas where the background tends to be of lighter colour.

The results given in Table 4 demonstrate that even though our original assumption of correct segmentation is broken, correct matching of individuals remains high, and discrimination between two individuals is largely maintained, even without ad-hoc tuning of the parameters. Particular cases, such as where Person D's legs are not segmented create false impressions of largely homogenous dark colour. This can then be incorrectly matched with Person A, who is actually of a similar, but truly homogenous dark colour. This case accounts for the majority of cases where two individuals are incorrectly matched both within the same camera, and across cameras. This also indicates that occlusions of objects may be likely to lead to incorrect results, and thus need to be identified for removal from the IMCSHR process. Cluttered scenes also lead to significant segmentation errors with individuals incorrectly joined together and need

to be identified as a source of possible errors. Two cases are shown in in Figure 13 and reported in Table 4 as Cases 5 and 6. These cluttered scenes were transient and only polluted a small number of the frames within the track (4 frames within each case).

Fig 13 Poor segmentation in cluttered frames a) frame 2140 b) frame 2937

Table 4 Comprehensive Results of Automated IMCSHR Matching

Test Case	Number of cases	Cameras Used	Typical similarity (IMCSHR)	Matched Cases	Unmatched Cases
1 Same Person Same Camera	10	*5_corridor or 5_lift*	0.9436	10	0
2 Same Person Disjoint Camera	13	*5_corridor and 5_lift*	0.8214	9	4
3 Different People Same Camera	8	*5_corridor or 5_lift*	0.3726	0	8
4 Different People Disjoint Camera	10	*5_corridor and 5_lift*	0.3913	4	6
5 Matching Person A in a Cluttered Track	1	*5_corridor*	0.9187	1	0
6 Matching Person B in a Cluttered Track	1	*5_corridor*	0.9408	1	0

The results based upon automated segmentation show that within the same camera, disjoint tracks of an individual tend to be correctly matched. When tracks are compared with other tracks in the second camera, the matching rate is diminished. This is probably due to a different chromatic response in the two cameras that could be compensated for with an initial calibration stage. In addition, segmentation errors in the two cameras are different for the lighter colours as the background colours are different. In general, objects in 5_lifts camera tended to be undersegmented more significantly than 5_corridor camera.

Despite the segmentation errors, correct matching of individuals is achieved in a large majority of the studied cases; however, it is obviously impacted by such errors. As

mentioned earlier, Person A and Person D, shown in Figure 12, are sometimes incorrectly matched due to significant segmentation errors. Thus, people who are wearing colours that are similar in part, for instance the same colour top, are more difficult to discriminate between where segmentation errors are large.

5.5 Discussion

From the experimental results, we can state that the proposed approach has proven effective for the matching of single individuals from disjoint camera views in a real-life video surveillance scenario. Its limitations in discrimination capability are inherent to the features chosen. Any two people of sufficiently similar appearance will tend to be matched as similar under normal conditions. This could certainly be the case, for instance, of personnel or school children wearing uniforms. Scenario analysis will have to be conducted to assess the level of threat carried by such individuals (such as the possibility of security controls on all uniformed personnel allowed to be on the premises). Also people who change their clothing will no longer be matched to their original disjoint track within the system. A new track will be generated for the duration of the session. Extensions to the feature set could be naturally integrated in the method proposed in this paper that may help to mitigate these problems. Our current work is implementing the following extensions:

1. Adding an invariant shape feature. The feature currently selected is the moving person's average height over a sequence of frames longer than the estimated gait period [28].
2. Making the system capable of detecting the main pose related views of a person, such as front, back, and side. The colour histogram matching procedure could then be parametric in the detected pose.
3. Making the system capable of detecting obstructed views of a person in order to prevent using such views for matching.

4. Evaluating different fusion scenarios, including using the features in a hierarchy. This would start from those features that offer the best trade off between discrimination and computational costs and proceed in diminishing order so as to limit the average computational costs at no significant expense for accuracy.

VI. CONCLUSIONS

In this paper, we have proposed a method for tracking people across disjoint camera views based on an illumination-tolerant appearance representation. Disjoint views are challenging because illumination conditions can be very different between views, significantly varying the appearance of objects. Computing an exact transformation to compensate for such appearance changes is impractical since appearance of moving objects depends on a number of illumination parameters which cannot be fully retrieved from videos, even with an initial training stage. The main contributions of this paper are: an accurate appearance representation capable of dealing with the small pose changes of a moving person, a modified cumulative histogram transformation compensating for the varying illumination conditions "called controlled equalisation", and a matching strategy that extends along the whole available track. Results from experiments reported in this paper can be summarised as:

1. The modified cumulative histogram transformation makes the appearance of a single object reasonably invariant across disjoint camera views while different objects remain easy to discriminate;
2. The proposed *k*-means online clustering algorithm has proved an accurate and efficient appearance representation to the purpose of matching;
3. The incremental major colour spectrum histogram representation (IMCSHR) copes with the small view changes occurring over a window of successive frames;

4. The IMCSHR can tolerate small levels of segmentation errors, and recover from large segmentation errors in a small percentage of the frames within a track;
5. Post-matching integration of the frame-level decision along the objects' tracks for a minimum number of frames improves the reliability of overall matching;
6. The proposed overall matching algorithm has linear computational complexity in the number of frames, *N*, used for matching;

The proposed matching procedure can provide video surveillance applications with the ability of tracking single, moving objects across disjoint camera views which are predominant in existing surveillance camera networks. Such an ability is potentially useful for several surveillance applications such as tracking of assigned individuals from entry to exit of a building in real time ("watch list"), or as a forensic tool to automatically back-track movements of people from an assigned point in time and space related to an event of interest.

ACKNOWLEDGMENT

This research is supported by the Australian Research Council under the ARC Discovery Project Grant Scheme 2004 - DP0452657.

REFERENCES

1. Bar-Shalom, Y. and Jaffer, A. G., "Adaptive nonlinear filtering for tracking with measurements of uncertain origin," IEEE Conf. Decision and Control, New Orleans, 243-247 (1972).
2. C. Wren, A. Azarbayejani, T. Darrell and A. Pentland, "Pfinder: Real-Time Tracking of The Human Body," IEEE Trans. on Pattern Anal. Mach. Intell., **19(**7), 780-785 (1997).

3. I. Haritaoglu, D. Harwood and L.S. Davis, "W4: Who? When? Where? What? A Real Time System for Detection and Tracking People", IEEE Conf. Automatic Face and Gesture Recognition, 222-227, (1998).
4. A. Lipton, H. Fujiyoshi, and R. Patil, “Moving target classification and tracking from real-time video,” Proc. IEEE Image Understanding Workshop, 129-136 (1998).
5. X. Varona, J. Gonzalez, F.X. Roca, J.J. Villanueva, "Track: Image-based Probabilistic Tracking of People", International Conf. on Pattern Recognition, **3**, 1110-1113 (2000).
6. S. McKenna, Y. Raja, and S. Gong, “Tracking color objects using adaptive mixture models,” Image and Vision Computing, 17, 225-231 (1999).
7. L.M. Fuentes and S.A. Velastin, "People Tracking in Surveillance Applications", In: Proc. IEEE Workshop on Performance Evaluation of Tracking and Surveillance (PETS2001), (2001).
8. H. Tao, H. S. Sawhney, R. Kumar, “Object Tracking with Bayesian Estimation of Dynamic Layer Representations,” IEEE Trans. on Pattern Analysis and Machine Intelligence, **24**(1), 75-89 (2002).
9. T. Zhao, R. Nevatia, “Tracking Multiple Humans in Complex Situations,” IEEE Trans. on Pattern Analysis and Machine Intelligence, **26**(9), 1208-1221 (2004).
10. T. Huang, S. J. Russell, “Object Identification in a Bayesian Context,” In: Proceedings of IJCAI 1997, 1276-1283, (1997).
11. J. Orwell, P. Remagnino, G.A. Jones, "Multi-Camera Colour Tracking", In: Proc. IEEE International Workshop on Visual Surveillance, June 26, Fort Collins, Colorado, 14-21 (1999).
12. T.H. Chang and Gong, "Tracking Multiple People with a Multi-Camera System", In: Proc. IEEE Workshop on Multi-Object Tracking, 19-26, (2001).

13. O. Javed, Z. Rasheed, K. Shafique, M. Shah, “Tracking Across Multiple Cameras With Disjoint Views,” IEEE Int. Conf. on Computer Vision, **2**, 952-957, (2003).
14. O. Javed, K. Shafique, M. Shah, “Appearance Modeling for Tracking in Multiple Non-overlapping Cameras,” IEEE CS Conf. Computer Vision and Pattern Recognition, **2**, 26-33 (2005).
15. M. Piccardi and E. D. Cheng, “Multi-Frame Moving Objects Track Matching Based On An Incremental Major Color Spectrum Histogram Matching Algorithm”, IEEE International Workshop on Object Tracking and Classification in and Beyond the Visible Spectrum (OTCBVS’05), San Diego, CA, USA, June 20, (2005).
16. Y. Weiss, “Deriving intrinsic images from image sequences,” IEEE Conf. on Computer Vision, **2**, 68-75 (2001).
17. Y. Matsushita, K. Nishino, K. Ikeuchi, M. Sakauchi, “Illumination normalization with time-dependent intrinsic images for video surveillance,” IEEE Trans. on Pattern Anal. and Mach. Intell., **26**(10), 1336-1347, (2004).
18. L. Li, M. K. H. Leung, “Robust change detection by fusing intensity and texture differences”, in Proc. Of CVPR 2001, **1**, 777-784, (2001).
19. H. J. Zhang, J. Wu, D. Zhong and S. W. Smoliar, “An integrated system for content-based video retrieval and browsing,” Pattern Recognition, **30**(4), 643-658 (1997).
20. Y. Rubner, C. Tomasi, L. J. Guibas, “The Earth Mover's Distance as a Metric for Image Retrieval,” International Journal of Computer Vision, **40**(2), 99-121, (2000).
21. J. Hu and A. Mojsolovic, “Optimum color composition matching of images,” IEEE Conf. on Pattern Recognition, **4**, 47-51 (2000).
22. W. Lu and Y. P. Tan, “A Color Histogram Based People Tracking System”, IEEE International Symp. Circuits and Systems, **2**, 137-140, (2001).
23. A. Senior, A. Hampapur, Y.-L. Tian, L. Brown, S. Pankanti, and R. Bolle, “Appearance Models for Occlusion Handling”, PETS (2001).

24. Zoran Zivkovic and Ben Krose, “An EM-like algorithm for color-histogram-based object tracking,” IEEE Conf. Comp. Vision and Pattern Recognition (2004).
25. L. Li, W. Huang, I.Y.H. Gu, K. Leman, Q. Tian, “Principal Color representation for Tracking Persons,” In: Proceedings of SMC 2003, **1**, 1007-1012 (2003).
26. S. P. Lloyd, “Least Squares Quantization in PCM,” IEEE Trans. Information Theory, **28**, 129-137 (1982).
27. S.K. Zhou, R. Chellappa, “From sample similarity to ensemble similarity: probabilistic distance measures in reproducing kernel Hilbert space,” IEEE Trans. on Pattern Anal. And Machine Intell., **28**(6), 917-929 (2006)
28. C. Madden, M. Piccardi, “Height Measurement as a Session-based Biometric for People Matching Across Disjoint Camera Views”, IEEE Conf. Image and Vision Computing New Zealand, Dunedin, New Zealand, 282-286 (2005).