IEEE Workshop on Motion and Video Computing (WACV/MOTION'05)-Volume 2
January 05-07, 2005

Persistent Objects Tracking Across Multiple Non Overlapping Cameras

Jinman Kang, Isaac Cohen and Gérard Medioni
IRIS, Computer Vision Group, University of Southern California
Los Angeles, CA 90089-0273
`{jinmanka|icohen|medioni}@iris.usc.edu`

Abstract

We present an approach for persistent tracking of moving objects observed by non-overlapping and moving cameras. Our approach robustly recovers the geometry of non-overlapping views using a moving camera that pans across the scene. We address the tracking problem by modeling the appearance and motion of the moving regions. The appearance of the detected blobs is described by multiple spatial distributions models of blobs' colors and edges. This representation is invariant to 2D rigid and scale transformation. It provides a rich description of the detected regions, and produces an efficient blob similarity measure for tracking. The motion model is obtained using a Kalman Filter (KF) process, which predicts the position of the moving objects while taking into account the camera motion. Tracking is performed by the maximization of a joint probability model combining objects' appearance and motion. The novelty of our approach consists in defining a spatio-temporal Joint Probability Data Association Filter (JPDAF) for integrating multiple cues. The proposed method tracks a large number of moving people with partial and total occlusions and provides automatic handoff of tracked objects. We demonstrate the performance of the system on several real video surveillance sequences.

1. Introduction

Video surveillance systems rely on an increasing number of cameras distributed over large areas. Understanding human activity in such scenes requires the integration of video cues acquired by a set of cameras as the activity usually unfolds across multiple cameras. Multiple stationary cameras are commonly used for monitoring large regions of interest. However these cameras do not necessarily overlap, and the integration of the information gathered by the large number of sensor is not straightforward. In this paper, we consider the scenario where a scene is monitored by set of stationary cameras with no spatial overlap, and a pan tilt zoom (PTZ) camera used by an operator to monitor a scene or track suspicious behaviour. The problems addressed in this paper are: the inference of the relative geometry of the stationary cameras using the PTZ camera and the persistent tracking of moving objects across these cameras.

The crucial element for tracking across multiple cameras is the registration of views and trajectories. The registration step is a challenging task, as it requires space and time registration of the trajectories recovered from each camera [7]. The registration of non-overlapping views is an even more challenging task as we are not able to directly extract the geometric constraint since there are no overlapping regions.

In the following sections, we review related work and present an overview of the proposed approach. In Section 2, we introduce the stochastic models used for tracking moving objects. Section 3 presents the registration of non overlapping cameras and the tracking of moving objects across this network of cameras. Obtained results are presented and discussed in Section 4. Finally, Section 5 concludes our paper with a discussion on future work.

1.1. Previous Work

Tracking moving objects from multiple cameras is usually based on a prior registration of the cameras using common scene features or tracked moving objects. Image or video registrations have been extensively studied and one can find a summary in [10] and [12] respectively. The major limitation consists in identifying a common part of the scene for registering these multiple cameras. More recently, authors in [4], [7] and [13] have proposed a spatio-temporal registration method based on temporal coherences across cameras. Registration of the detected trajectories permits to infer the relative locations of the cameras. While an automatic detection of the common regions of the scene is achievable, it still requires a fair amount of overlap between the views to derive an accurate registration of the cameras. If the cameras' relative position and orientation is known, the problem is naturally simplified. In [14], the authors proposed a homography-based approach for registering non-overlapping views obtained from a calibrated rig. Similarly, in [9], by integrating detection of moving objects across views the relative positions of the cameras are approximated. These methods are effective, but restricted to a network of cameras of a fixed configuration. 3D trajectories of moving objects have been also proposed for registering multiple cameras. In [2], the authors propose to use 3D trajectories, derived using the epipolar geometry estimation, for establishing temporal correspondences between two video streams. The proposed method assumes a single moving region across cameras and ignores the complexity of data association. The performance of the proposed approach is very sensitive to the quality of the recovered 3D trajectories. When multiple moving objects are present, the proposed geometric registration relies only on the temporal coherences between trajectories and the inferred 3D trajectories may be corrupted leading to wrong cameras registration. Similar trajectories-based approaches are described in [11] and [3]. In [11], authors propose to use trajectories of pedestrians across multiple cameras for learning a dominant trajectories' spatio-temporal transition probability by matching target features. The dominant trajectories are manually selected, and only a small number of trajectories are used for obtaining the spatiotemporal transition probability across cameras. In [3], an unsupervised learning method is proposed for calculating the spatio-temporal transition probability across cameras. Simple entering and exiting activity in pre-specified regions is detected and used for inferring the topology of the cameras. To ensure statistical stability, the method requires a large amount of data (i.e. 24 hours of video data), and a dominant peak (a dominant spatio-temporal transition probability) between each link. These proposed methods are limited to stationary cameras and are sensitive to the quality of tracking results.

1.2. Overview of the Proposed Approach

In this paper, we propose a novel approach for tracking moving objects based on a spatio temporal Joint Probability Data Association Filter (JPDAF). Persistent object tracking across multiple cameras is formulated as a joint probability model encoding objects' appearance and motion. The spatio temporal JPDAF allows to extract optimal trajectories by collecting discriminating evidences in a given buffer while a classical JPDAF only considers a pair of frames.

The appearance of detected moving blobs from stationary or moving cameras [8] is encoded using a radial distribution invariant to 2D rigid transformation and to scale changes. This distribution encodes color and edge properties of the blob. An appearance probability model is defined based on a similarity function measuring the likelihood between two distributions.

The motion probability model is inferred from a Kalman Filter (KF). This model is calculated by a Gaussian distribution between the predicted bounding box position and the bounding box position of the observed blobs.

A spatio-temporal joint probability model combining the appearance and motion model of moving objects is proposed. It permits tracking moving objects efficiently by taking advantage of the relative position and orientation of the cameras. The proposed model is used for tracking persistently moving objects across non-overlapping cameras.

2. Tracking using a Joint Probability Model

Tracking is expressed as the maximization of a joint probability model. Two probabilistic models are defined and represent objects' appearance and velocity. The appearance model encodes both color and edge information, and measures the similarity of blobs accurately. The motion model is derived from a first order KF modeling the kinematics of the moving objects. The main contribution of this paper, beside the appearance model, is the use of a sliding buffer in which the optimal tracking decision is made by the JPDAF. The use of this spatio-temporal JPDAF allows tracking moving objects in cases where the tracking decision provided by classical JPDAF may not be optimal due to the lack of observations (i.e. occlusions) or incomplete measurements (partial detections). The newly defined JPDAF provides a better tracking by combining evidence collected in all the frames within a specific buffer. Also, the reacquisition of the optimal path provides smoother paths, compared to traditional JPDAF.

2.1. Appearance Model

Various color-based methods have been proposed in the literature to solve the tracking problem. Many of them use only one color histogram model per object. Multiple color models and their relative localization were also considered for object tracking. In [1], a multiple color model is proposed for human appearance's description, but requires a segmentation of the detected blob into the head, torso, and legs regions.

Appearance changes are expected while tracking moving people in the scene. Indeed, limbs motion will creates shape variations and self-occlusions. Therefore, object appearance models have to be continuous in the sense that a small localized change of the object color and shape should create a small localized variation in its signature. The object description should also be invariant to 2D rigid transformation and scale change in order to accommodate for change of perspective.

We propose in this paper an appearance model that is invariant to 2D rigid transformation and scale change. This model, defined by a polar distribution, provides a description of object's colors and shape properties. This 2D distribution model is used for measuring the similarity of tracked objects and for reacquiring objects after occlusions of short and long durations.

The color distribution model is obtained by mapping the blob into a polar representation. Several shape or color distribution models using a polar representation have been proposed [5][6][8]. In [5], the proposed approach is focused on the object's shape description (edge) instead of their appearance (color), and it is only limited to representing local shape properties. In [8], the proposed model measures color distribution using a similar polar representation, but focuses on characterizing a global appearance signature of the object. The model is not 2D rotation-invariant and we propose here to use the shape description model proposed in [6] for guaranteeing invariance to 2D rigid transformation and scale change (see Figure 1).

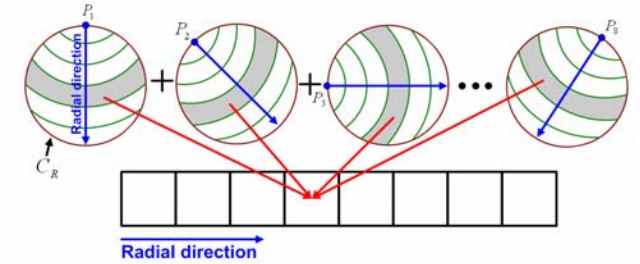


Figure 1. *Mapping of the proposed 2D rigid motion invariant polar representation*

Given a detected moving blob, we define a reference circle C_R defined by the smallest circle containing the blob. This circle is uniformly sampled into a set of control points P_i. For each control point P_i a set of concentric circles of various radii are used for defining the bins of the appearance model. Inside each bin, a Gaussian color model is computed for modeling the color properties of the overlapping pixels of the detected blob. Therefore, for a given control point P_i we have a one-dimensional distribution $\gamma_i(P_i)$. The normalized combination of the distributions obtained from each control point P_i defines the appearance model of the detected blob: $\Lambda = \sum \gamma_i(P_i)$.

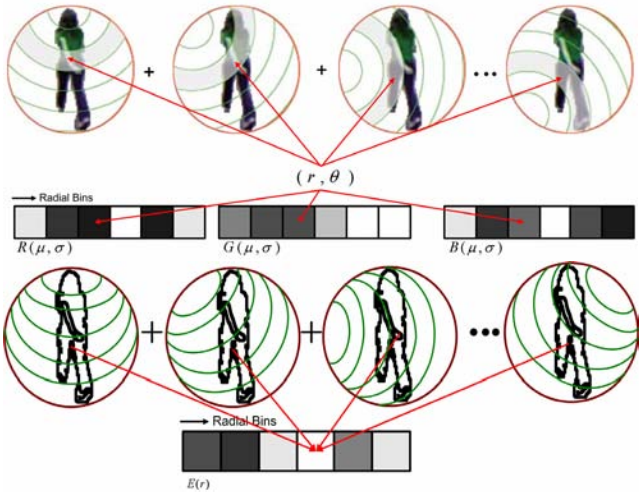


Figure 2. *Computation of the color and shape based appearance model of detected moving blobs*

We illustrate this process in Figure 2 where we sampled the reference circle with 8 control points. The defined model is translation invariant. Rotation invariance is obtained by taking a larger number of control points along the reference circle. Finally, normalizing the reference circle to unit circle guarantees scale invariance.

We derive two types of appearance descriptors using this radial tessellation of the detected blob: A color and edge descriptor. The color descriptor is modeled by a Gaussian in each bin and each channel. The shape-based description of the blob is modeled by counting in each bin the number of edge pixels belonging to the moving blob. The 2D shape description is obtained by collecting and normalizing corresponding edge points for each bin as follows.

$$E(r) = \frac{1}{\max_r \left(\sum_i E_r(P_i)\right)} \sum_i E_r(P_i) \tag{1}$$

where, $E(r)$ is edge distribution for each radial bin.

The appearance probability model is defined as a similarity measure among detected blobs in successive frames. The appearance model describes blobs thru a distribution function. We use the Kullback-Leibler distance for measuring the similarity of the computed appearance models. Due to the different distribution models considered, Gaussian distribution for the color model and uniform distribution for the shape model, the similarity measures are computed separately.

The similarity function of the color model can be expressed in terms of the mean and variance of the Gaussian model in each bin. Given the appearance descriptors of each blob, we define the following likelihood ratio:

$$\Gamma_{Color} = \frac{1}{2N_{rgb}} \sum_{i=0 \dots N_{rgb}} \left\{ (\mu_{i,t} - \mu_{i,t+1})^2 \cdot \left(\frac{1}{\sigma^2_{i,t+1}} + \frac{1}{\sigma^2_{i,t}} \right) + \frac{\sigma^2_{i,t}}{\sigma^2_{i,t+1}} + \frac{\sigma^2_{i,t+1}}{\sigma^2_{i,t}} \right\} \tag{2}$$

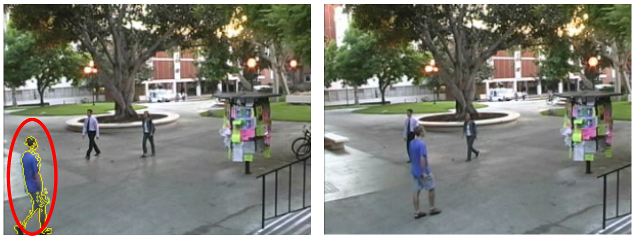
where, $\mu_{i,t}$ and $\sigma_{i,t}$ are respectively the mean and the variance of the color component in the *ith* bin at time *t*, and N_{rgb} is total number of bins of the color component. We have $1 \le \Gamma_{color} \le \infty$. The shape's similarity of detected blobs is measured by the KL distance:

$$Dist_{Shape} = \frac{1}{2} \sum (E(r)_t - E(r)_{t+1}) \log \frac{E(r)_t}{E(r)_{t+1}} \tag{3}$$

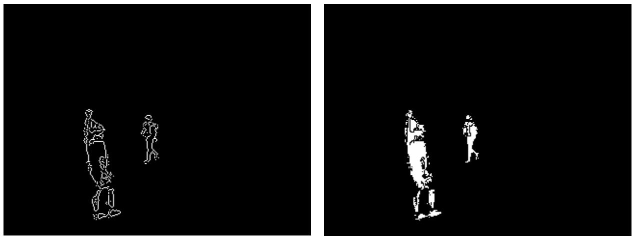
The color likelihood ratio and shape similarity are combined for defining the appearance probability model of detected regions. This probability is defined by:

$$P_{appearance} = 1 \Big/ \sqrt{(\Gamma_{Color})^2 + (Dist_{Shape})^2} \tag{4}$$

where a normalization factor is added after collecting the observations within the sliding buffer. In the following we illustrate the contribution of color and edge information for defining a good similarity measure. Figure 3 shows the two frames considered for this experiment. In the following we will focus only on the moving object outlined by the red ellipse in Figure 3.a. We also present the edge of the moving object by yellow color. The proposed similarity measures are compared and evaluated using the target frame presented in Figure 3.b by searching for the optimal location in the image maximizing the two similarity measures. Figure 3.c and d present edge map and the masked target frame focusing on moving blob. The target frame considered contains all detected moving blobs. Note that the detected moving regions in the target frame contain multiple persons merged into a single blob.



(a) Reference frame *(b) Target frame*



(c) Edge of target frame *(d) Mask of target frame*

Figure 3*. Frames use for the experiment (red ellipse: target object)*

In Figure 4, we illustrate the contribution of the edge information to the appearance model. The proposed likelihood ratio and similarity measure locate the reference moving object amongst the detected moving blobs as illustrated by Figure 4.a and b. One can observe a bias in the localization of the target object when only color information is used, due to a dynamic occlusion.

(a) No edge *(b) With edge*

Figure 4*. The contribution of edge information*

2.2. Motion Model

The motion model of the moving object is obtained by the KF framework proposed in [8]. The proposed motion model simultaneously handles stationary and moving cameras by compensating for the camera motion using the estimated affine transformation. Moreover, a single KF model is used for tracking objects across multiple cameras. It permits automatic cameras hand-off and tracking of objects occluded for short periods of time. The detailed description of the proposed motion model is addressed in [8].

The KF framework is used for defining the motion probability model of detected moving objects. The state vector considered in this paper is defined by:

$$x_i = (x^i_{top}, y^i_{top}, v^i_{top}, u^i_{tip}, x^i_{bottom}, y^i_{bottom}, v^i_{bottom}, u^i_{bottom}) \qquad (5)$$

where, (x^i_{top}, y^i_{top}) is the top-left corner of the detected bounding box, $(x^i_{bottom}, y^i_{bottom})$ is the bottom-right corner of the detected bounding box, (v^i_{top}, u^i_{top}) is the 2D velocity of (x^i_{top}, y^i_{top}), and $(v^i_{bottom}, u^i_{bottom})$ is the 2D velocity of $(x^i_{bottom}, y^i_{bottom})$.

The motion probability model P_{motion} is derived from the KF estimates. A Gaussian (normal) model is used and is defined by:

$$P_{motion} = \frac{1}{((2\pi)^{N_s} \bullet \det(P_t))^{1/2}} e^{-\frac{(x_t - \hat{x}_t)^T \bullet P_t^{-1} \bullet (x_t - \hat{x}_t)}{2}} \qquad (6)$$

where, N_s is the number of variables in the object's state vector, $\hat{x}_t$ is the mean obtained from KF, x_t is the observed position, and P_t is the covariance matrix obtained from the KF.

In Figure 5, the predicted position of the detected bounding box of a moving object is shown. As one can see in Figure 5.a, when the object is occluded by the pole (circled in red in the figure), the bounding box is correctly estimated and no delay or oscillations are noticed when the object reappears. In Figure 5.b we show an example with a moving camera, where two moving objects in the scene are occluding each other. The proposed model resolves the occlusions by efficiently predicting the position and bounding box of the moving objects.

(a) Example of a stationary occlusion

(b) Example of a dynamic occlusion

Figure 5. *Predicting the position of moving objects in a non stationary camera using the proposed Kalman Filter (red circle: occlusion)*

2.3. Spatio-temporal Joint Probability Model

In [8], JPDAF-based tracking approach was proposed; it formulated the tracking problem as characterizing the position (X^*) of the moving object that maximizes appearance and motion models. The optimal position at each time step depends on the current observations, as well as the motion estimation obtained at the previous optimal positions. A joint probability of the state vector at time *t* is given by the equation:

$$
\begin{aligned}
P_{total}(X_t) &= P(A_t, X_t, A_{t-1}, X^*_{t-1}, \cdots A_1, X^*_1, A_0, X^*_0) \\
&= P(A_t \mid X_t) P(X_t \mid X^*_{t-1}, \cdots, X^*_1, X^*_0) P_{total}(X^*_{t-1}) \quad (7) \\
&= P_{appearance}(X_t) P_{motion}(X_t) P_{total}(X^*_{t-1})
\end{aligned}
$$

where, X_t and A_t denotes respectively the position and appearance at time *t*.

The proposed appearance model encodes both color and edge information, and therefore $P_{appearance}(X_t) = P(A_t \mid X_t)$ can be rewritten as: $P(A_t \mid X_t) = P(C_t, E_t \mid X_t)$ where C_t denotes color observation, and E_t denotes edge observation at time *t*.

In order to avoid the accumulation of products of probability, we consider the *log* of the probabilities as the joint probability. Furthermore, for ensuring a stable calculation, we discard old measurements from the estimation process. This shortens the memory of the KF and allows variations in speed and color similarities.

The classical JPDAF-based tracking approach produces local optimal solution since the decision made at time *t* is based only on current measurement and previous solution at time *t-1*. If a wrong estimation of the position is selected at time *t*, due to occlusions or to a bad detection, the tracking will not be able to recover the right solution later on.

In this section we propose to extend the dependency of the current classical JPDAF based solution to a set of past observations and consider these observations for choosing the optimal solution at time *t*. Our method propagates a set of possible solutions within a temporal sliding window, and characterizes the optimal solution by extracting the optimal path among these past observations. This uses a set of previous solutions for characterizing the optimal solution rather than relying solely on the previous estimation. In Figure 6, we illustrate this process. For each moving object at time *t*, all possible solutions (depicted by the ellipses) within the sliding buffer are considered. At a given time T, we collect discriminating evidences from the set of solutions stored in the buffer by calculating the joint probability from time *T* to *T-W* as illustrated in Figure 6 (depicted in red arrow). The motion and appearance probability models are directly computed from the stored solutions. The refined path is compared with the initial path, and the path which has the best score (*i.e.* joint probability defined by Eq.(7)) is selected.

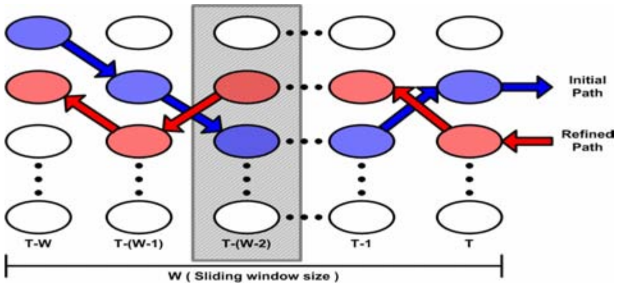


Figure 6. *Illustration of optimal path refining process*

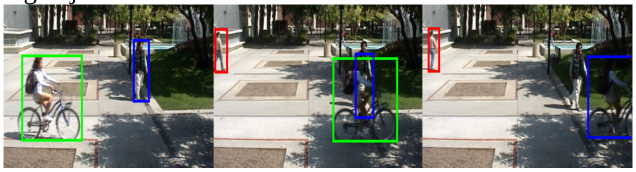
The proposed approach is based on the following observation: if the initial path obtained from the classical JPDAF based approach (depicted by the blue arrow in Figure 6) contains non optimal nodes (depicted by grey box), the joint probability $P_{total}(X_t)$ will not be optimal. This may occur when motion and/or appearance model probabilities at time *t-1* are incorrectly estimated due to occlusions or wrong detections. In the proposed JPDAF method, the optimal solution corresponds to the path which has the best score within the selected buffer. The optimal path is computed using an exhaustive search of all possible paths within the buffer. This process guarantees the selection of a path with a higher joint probability $P_{total}(X_t)$ than the one provided by the classical JPDAF. We currently use a brute force search for selecting the optimal path, and the complexity of the method is exponential. To reduce the algorithmic complexity of the search of the optimal path, we only consider nodes for which the bounding box overlaps with the initial path. The number of possible candidates is bounded by *(2 * the size of the chosen buffer)*.

The selection of an optimal path is guided by the following equation:

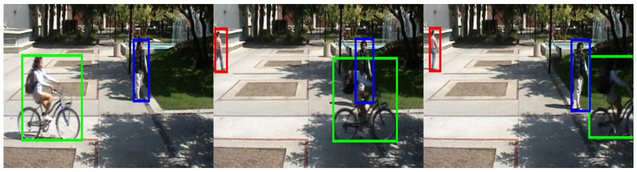
$$
\xi^* = \max \left\{ \begin{array}{c} \alpha P_{total}(X_t) \\ (1-\alpha)\sum_{\Gamma} \log(\tilde{P}_{appearance}(\xi)) + \log(\tilde{P}_{motion}(\xi)) \end{array} \right. \quad (8)
$$

where, $0 < \alpha < 1$ is a weight factor used for increasing the confidence level of the refined path, ξ^* is the refined optimal path, ξ is the selected possible sub-optimal path, $\tilde{P}_{appearance}(\xi)$ and $\tilde{P}_{motion}(\xi)$ correspond respectively to the appearance and motion probability along the path ξ. The value of α is manually selected based on the complexity of the dynamics of moving objects in the scene. The typical range of values considered for α is $0.45 < \alpha < 0.75$.

If the initial path provided by the classical JPDAF contains only optimal nodes, the confidence level of the initial path should be distinctively higher than other possible solutions regardless of the weight α. Otherwise, the dominant sub-optimal path is considered and the optimal solution is updated. The use of proposed formalism corrects wrong decisions made by the classical JPDAF, and it permits to infer spatio-temporally smooth path. Figure 7 illustrates the proposed approach by showing an example where a more accurate tracking was achieved. In Figure 7.a, a classical JPDAF was used: the moving blob depicted by the blue bounding box cannot be tracked correctly after the object was occluded by the moving bike. Figure 7.b shows the obtained tracking using the proposed spatio-temporal JPDAF. Moving objects are persistently tracked across cameras even in cases of occlusions by other moving objects.



(a) Tracking result using classical JPDAF



(b) Tracking result using the proposed JPDAF method

Figure 7. *Tracking improvement using the proposed approach*

3. Registration of Non-Overlapping Views

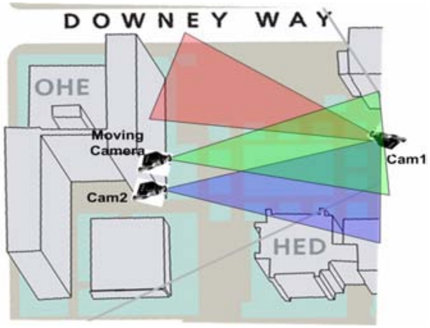
In deployed video surveillance systems the location of the cameras are chosen according to areas of interests while trying to maximize the coverage of the scene with the minimal number of cameras. Overlapping parts of the scene across cameras are very infrequent. In this paper we consider two stationary cameras with no overlap and a panning camera used by the operator for monitoring the scene or tracking suspicious behaviour. The problems addressed here are the inference of the topology of the stationary cameras using the PTZ camera and the persistent tracking of moving objects across these views.

In Figure 8.a and b, we show the two non overlapping stationary views. These cameras are looking in opposite directions. As we can observe in Figure 8.c, the mosaic computed from the moving camera spans across the two stationary cameras. We can therefore register the two non-overlapping cameras using a homography by registering each stationary camera to the computed mosaic. The relative position and orientation of the two cameras is inferred by combining the estimated affine motion to the estimated homographies.

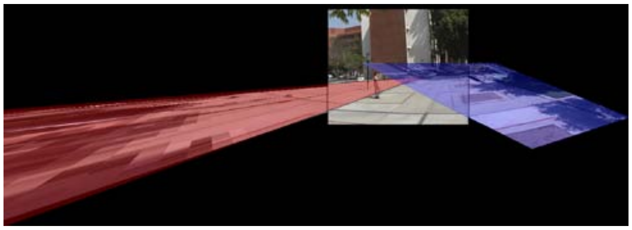
The combination of the affine transform and the perspective transform registers the moving and stationary cameras by a series of concatenated homographies [8]. Figure 8.e presents an example of registering the non-overlapping views by incorporating the moving camera motion. Although the proposed method allows registering non-overlapping views, accumulated registration errors are common and the topology of the cameras may be incorrect. The registration errors can be reduced either by incorporating other moving cameras, which move between the non-overlapping views, or by refining the topology of the cameras using multiple passes between the non-overlapping views as the operator is scanning the scene.

(a) Stationary View 1 *(b) Stationary View 2*

(c) Mosaic of the moving camera that we use for registration



(d) Synthetic top-down view depicting the relative position of the camera



(e) Registration of three views viewing from moving camera view point. FOV of other cameras are overlapped with different colors on top of the reference moving frame

Figure 8. *Example of non-overlapping views registered using a moving camera*

The knowledge of the relative location and orientation of the cameras guides the data association in the JPDAF tracking. Using the relative positions of the cameras, we can characterize the FOV of other cameras for at each time t and identify overlapping regions in the scene. In these regions we register the foot positions (or lower part of the detected bounding box) of detected moving blobs across all overlapping views. When a correct matching of these positions is obtained the JPDAF appearance model is refined using information provided from overlapping cameras. Moreover, the registration of multiple views guides the search during the data association and reduces the tracking ambiguities. Similarly, estimated blob's motion is refined from overlapping views and provides a more robust data association mechanism in the JPDAF framework. Finally, the dynamic registration of multiple views provides an efficient cameras hand-off mechanism.

4. Experimental results

We present in this section some results obtained on real sequences for illustrating the registration of multiple non-overlapping views, and object tracking across these cameras. In Figure 9.a and b, the field of view of each camera is overlaid on top of the other views using cameras' registration. The homography transformation is obtained from a set of corresponding ground plane points, and the affine transformation stabilizing the moving camera panning between the non-overlapping views. The accuracy of the registration process can be observed from the boundary of each FOV. Figure 9.c presents an example of simultaneous tracking of an object across multiple views. Color tagging is used in these experiments for characterizing the same moving objects across cameras. The continuous tracking of multiple objects across non-overlapping views is illustrated in Figure 9.d. We have used a color coding scheme for displaying the FOV of the cameras. The FOV of the stationary cameras 1 and 2 is red, and blue respectively. The FOV of the moving camera is shown in green. At each frame we re-project these FOV onto the other cameras to illustrate the overlapping areas in the scene, as well as the cameras hand-off process. In Figure 9.e, we show an example of continuous tracking of moving objects across views using the motion estimation and the relative positions of the cameras. As one can observe from these figures, the moving object (depicted by the green box) is only visible from one or two views at each time instance, and disappear as it moves outside the FOV of the moving camera. The motion estimation obtained from KF and the recovered camera configuration resolves such ambiguities and provides a persistent tracking of moving objects in the scene.

Acknowledgements

This research was partially funded by the Advanced Research and Development Activity of the U.S. Government under contract MDA-904-03-C1786.

5. Conclusion

We have presented a novel approach for persistent tracking of moving objects across non-overlapping views. The relative locations of non-overlapping stationary cameras are recovered by the dynamic registration of the moving and stationary cameras. In each view, the combination of the appearance model and the motion model, obtained by the stochastic approach presented in Section 2, allows us to track the moving objects efficiently. The appearance model encodes both color and edge information of the detected blob. It is used to measure accurately the similarity of objects appearance. The use of a spatio-temporal JPDAF allowed tracking moving objects in cases where the tracking provided by classical JPDAF was not optimal due to the lack of discriminating evidences. The newly defined JPDAF provides an optimal tracking by combining evidences collected in the frames within a chosen buffer.

The proposed approach does not make use of 3D information (e.g. 3D ground plane or 3D trajectory) and we believe that integrating this 3D information will improve tracking performances. The computational complexity of the spatio-temporal JPDAF depends on the number of tracked moving objects. Reducing the computational complexity for real-time performances will be the focus of our future work.

References

[1] A. Elgammal and L. S. Davis, "Probabilistic Framework for Segmenting People Under Occlusion", *In IEEE ICCV*, 2001.

[2] C. Rao. A. Gritai, M. Shah and Tsyeda-Mahmood, "View-invariant Alignment and Matching of Video Sequence", *In IEEE ICCV*, 2003.

[3] D. Makris, T. Ellis and J. Black, "Bridging the Gaps between Cameras", *In IEEE CVPR*, 2004.

[4] G. Stein, "Tracking from Multiple View Points: Self-calibration of Space and Time", *In IEEE CVPR*, pp. 521-527, 1999.

[5] H. Zhang and J. Malik, "Learning a discriminative classifier using shape context distance", *In IEEE CVPR*, Vol. 1, pp. 242-247, 2003.

[6] I. Cohen and H. Li, "Inference of Human Postures by Classification of 3D Human Body Shape", *In IEEE IWAMFG*, 2003.

[7] J. Kang, I. Cohen and G. Medioni, "Continuous Multi-Views Tracking using Tensor Voting", *In IEEE WMVC*, 2002.

[8] J. Kang, I. Cohen and G. Medioni, "Continuous Tracking Within and Across Camera Streams", *In IEEE CVPR*, 2003.

[9] J. Rasheed, Z. Shafique and M. Shah, "Tracking Across Multiple Cameras With Disjoint Views", *In IEEE ICCV*, 2003.

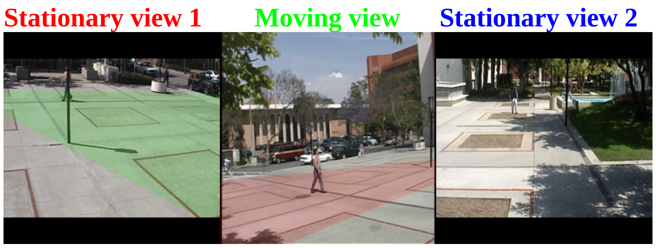
[10] M. Irani and P. Anandan, "About direct methods", *In Vision Algorithm Workshop*, pp. 267-277, Corfu, 1999.

[11] O. Javed, Z. Rasheed, K. Shalfique and M. Shah, "Tracking Across Multiple Cameras With Disjoint Views", *In IEEE ICCV*, Nice, France, 2003.

[12] Q. Cai and J. Aggarwal, "Automatic Tracking of Human Motion in Indoor Scenes Across Multiple Synchronized video Streams", *In IEEE ICCV*, 1998.

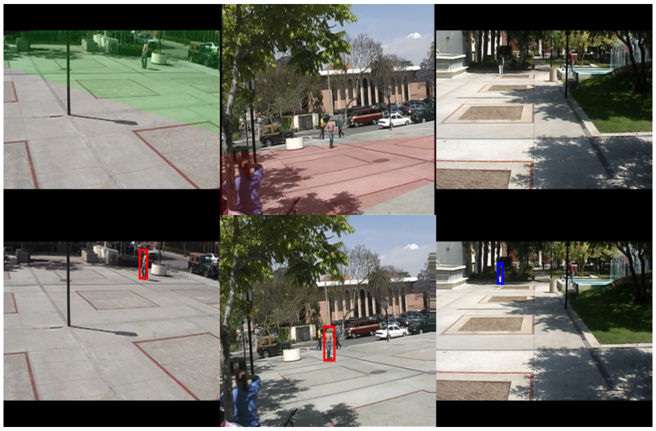
[13] Y. Caspi and M. Irani, "A step towards sequence-to-sequence alignment", *In IEEE CVPR*, June, 2000.

[14] Y. Caspi and M. Irani, "Alignment of non-overlapping sequences", *In IEEE ICCV*, 2001.



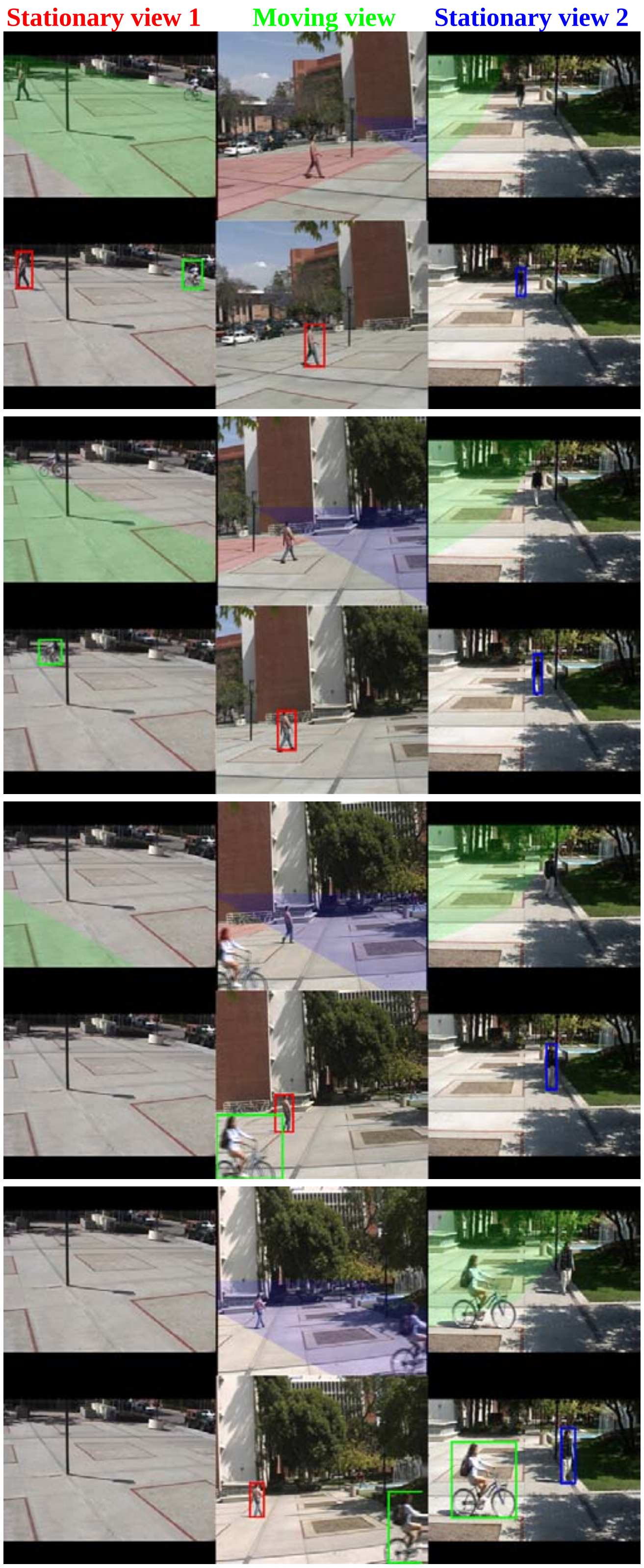
(a) Overlapping FOV across the stationary camera 1 and the moving camera

(b) Overlapping FOV across the stationary camera 2 and the moving camera



(c) Simultaneous tracking of object 2 (red box) from both stationary view 1 and moving camera (blue box : object 1)

(d) Simultaneous tracking of different objects across different views



(e) Continuous tracking of object 3 (green box) across cameras

Figure 9. *Tracking of moving objects across multiple non-overlapping stationary and moving cameras.*