Inter-camera Association of Multi-target Tracks by On-Line Learned Appearance Affinity Models

Cheng-Hao Kuo, Chang Huang, and Ram Nevatia

University of Southern California, Los Angeles, CA 90089, USA
{chenghak,huangcha,nevatia}@usc.edu

Abstract. We propose a novel system for associating multi-target tracks across multiple non-overlapping cameras by an on-line learned discriminative appearance affinity model. Collecting reliable training samples is a major challenge in on-line learning since supervised correspondence is not available at runtime. To alleviate the inevitable ambiguities in these samples, Multiple Instance Learning (MIL) is applied to learn an appearance affinity model which effectively combines three complementary image descriptors and their corresponding similarity measurements. Based on the spatial-temporal information and the proposed appearance affinity model, we present an improved inter-camera track association framework to solve the "target handover" problem across cameras. Our evaluations indicate that our method have higher discrimination between different targets than previous methods.

1 Introduction

Multi-target tracking is an important problem in computer vision, especially for applications such as visual surveillance systems. In many scenarios, multiple cameras are required to monitor a large area. The goal is to locate targets, track their trajectories, and maintain their identities when they travel within or across cameras. Such a system consists of two main parts: 1) intra-camera tracking, *i.e.* tracking multiple targets within a camera; 2) inter-camera association, *i.e.* "handover" of tracked targets from one camera to another. Although there have been significant improvements in intra-camera tracking , inter-camera track association when cameras have non-overlapping fields of views (FOVs) remains a less explored topic, which is the problem we focus on in this paper.

An illustration for inter-camera association of multiple tracks is shown in Figure 1. Compared to intra-camera tracking , inter-camera association is more challenging because 1) the appearance of a target in different cameras may not be consistent due to different sensor characteristics, lighting conditions, and viewpoints; 2) the spatio-temporal information of tracked objects between cameras becomes much less reliable. Besides, the open blind area significantly increases the complexity of the inter-camera track association problem.

Associating multiple tracks in different cameras can be formulated as a correspondence problem. Given the observations of tracked targets, the goal is to find the associated pairs of tracks which maximizes a joint linking probability,

K. Daniilidis, P. Maragos, N. Paragios (Eds.): ECCV 2010, Part I, LNCS 6311, pp. 383–396, 2010.

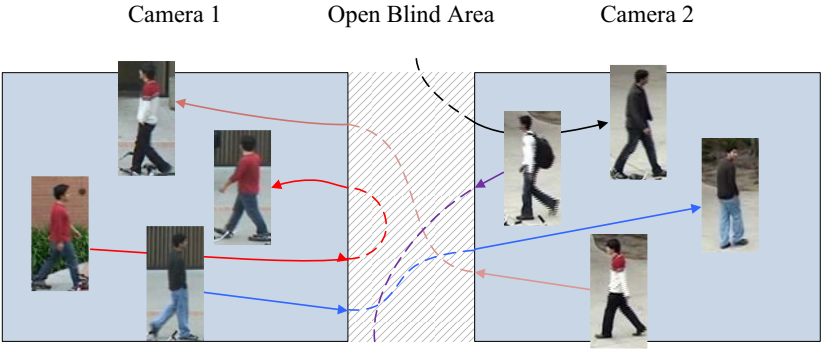

Fig. 1. Illustration of inter-camera association between two non-overlapping cameras. Given tracked targets in each camera, our goal is to find the optimal correspondence between them, such that the associated pairs belong to the same object. A target may walk across the two cameras, return to the original one, or exit in the blind area. Also, a target entering Camera 2 from blind area is not necessarily from Camera 1, but may be from somewhere else. Such open blind areas significantly increase the difficulty of the inter-camera track association problem.

in which the key component is the affinity between tracks. For the affinity score, there are generally two main cues to be considered: the spatio-temporal information and appearance relationships between two non-overlapping cameras. Compared to spatial-temporal information, the appearance cues are more reliable for distinguishing different targets especially in cases where FOVs are disjoint. However, such cues are also more challenging to design since the appearances of targets are complex and dynamic in general. A robust appearance model should be adaptive to the current targets and environments.

A desired appearance model should incorporate discriminative properties between correct matches and wrong ones. Between a set of tracks among two non-overlapping cameras, the aim of the affinity model is to distinguish the tracks which belong to the same target from those which belong to different targets. Previous methods [1,2,3] mostly focused on learning the appearance models or mapping functions based on the correct matches, but no negative information is considered in their learning procedure. To the best of our knowledge, online learning of a discriminative appearance affinity model across cameras has not been utilized.

Collecting positive and negative training samples on-line is difficult since no hand-labelled correspondence is available at runtime. Hence, traditional learning algorithms may not apply. However, by observing spatio-temporal constraints of tracks between two cameras, some potentially associated pairs of tracks and some impossible pairs are formed as "weakly labelled samples". We propose to adopt the Multiple Instance Learning (MIL) [4,5,6] to accommodate the ambiguity of labelling during the model learning process. Then the learned discriminative appearance affinity model is combined with spatio-temporal information to

compute the crucial affinities in the track association framework, achieving a robust inter-camera association system. It can be incorporated with any intra-camera tracking method to solve the problem of multi-object tracking across non-overlapping cameras.

The rest of the paper is organized as follows. Related work is reviewed in Section 2. The overview of our approach is given in Section 3. The framework of track association between two cameras is described in Section 4. The method of learning a discriminative appearance affinity model using multiple instance learning is discussed in Section 5. The experimental results are shown in Section 6. The conclusion is given in Section 7.

2 Related Work

There is large amount of work, *e.g.* [7,8,9], for multi-camera tracking with overlapping field of views. These methods usually require camera calibration and environmental models to track targets. However, the assumption that cameras have overlapping fields of view is not always practical due to the large number of cameras required and the physical constraints upon their placement.

In the literature, [10,11,12] represent some early work for multi-camera tracking with non-overlapping field of views. To establish correspondence between objects in different cameras, the spatio-temporal information and appearance relationship are two important cues. For the spatio-temporal cue, Javed *et al.* [13] proposed a method to learn the camera topology and path probabilities of objects using Parzen windows. Dick and Brooks [14] used a stochastic transition matrix to describe people's observed patterns of motion both within and between fields of view. Makris *et al.* [15] investigated the unsupervised learning of a model of activity from a large set of observations without hand-labeled correspondence.

For the appearance cue, Porikli [1] derived a non-parametric function to model color distortion for pair-wise camera combinations using correlation matrix analysis and dynamic programming. Javed *et al.* [2] showed that the brightness transfer functions(BTFs) from a given camera to another camera lie in a low dimensional subspace and demonstrated that this subspace can be used to compute appearance similarity. Gilbert and Bowden [16] learned the BTFs incrementally based on Consensus-Color Conversion of Munsell color space [17].

Besides, there is some work addressing the optimization framework of multiple targets correspondence. Kettnaker and Zabih [12] used a Bayesian formulation to reconstruct the paths of targets across multiple cameras. Javed *et al.* [13] dealt with this problem by maximizing the a posteriori probability using a graph-theoretic framework. Song and Roy-Chowdhury [18] proposed a multi-objective optimization framework by combining short-term feature correspondences across the cameras with long-term feature dependency models.

Learning a discriminative appearance affinity model across non-overlapping cameras at runtime makes our approach different from the existing ones. Most previous methods did not incorporate any discriminative information to distinguish different targets, which is important for inter-camera track association especially when the scene contains multiple similar targets.

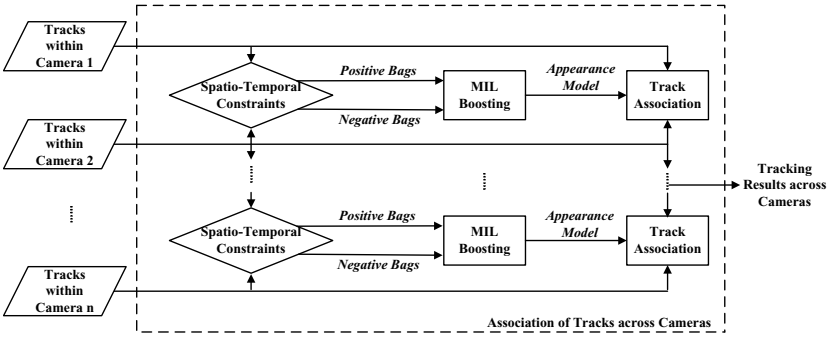

Fig. 2. The block diagram of our system for associating multiple tracked targets from multiple non-overlapping cameras

3 Overview of our Approach

Our system contains three main components: the method of collecting online training samples, the discriminative appearance affinity model, and track association framework. We use a time sliding window method to process video sequences. The learned appearance affinity models are updated in each time window. The system block diagram of our method is shown in Figure 2.

The collection of online training samples is obtained by observing the spatio-temporal constraints in a time sliding window. Assuming that the multi-object tracking is finished in each camera, a training sample is defined as a pair of tracks from two cameras respectively. Negative samples are collected by extracting pairs of tracks in two cameras which overlap in time. It is based on the assumption that one object can not appear in two non-overlapping cameras at the same time. Positive samples could be collected by similar spatio-temporal information. However, it is difficult to label the positive training sample in an online manner since it is indeed the correspondence problem that we want to solve. Instead of labelling each sample, several potentially linked pairs of tracks constitute one positive "bag", which is suitable for the Multiple Instance Learning (MIL) algorithm.

The learning of appearance affinity model is to determine whether two tracks from different cameras belong to the same target or not according to their appearance descriptors and similarity measurements. Instead of using only color information as in previous work, appearance descriptors consisting of the color histogram, the covariance matrix, and the HOG feature, are computed at multiple locations to increase the power of description. Similarity measurements based on those features among the training samples establish the feature pool. Once the training samples are collected in a time sliding window, a MIL boosting algorithm is applied to select discriminative features from this pool and their corresponding weighted coefficients, and combines them into a strong classifier in the same time sliding window so that the learned models are adapted to the

current scenario. The prediction confidence output by this classifier is transformed to a probability space, which cooperates with other cues (e.g. spatial correspondence and time interval) to compute the affinity between tracks for association.

The association of tracks in two cameras is formulated as a standard assignment problem. A correspondence matrix is defined where the pairwise association probabilities are computed by spatio-temporal cues and appearance information. This matrix is designed to consider all possible scenarios in two non-overlapping cameras. The Hungarian algorithm is applied to solve this problem efficiently.

4 Track Association between Cameras

To perform track association across multiple cameras, we firstly focus on the track association between two cameras and then extend it to the case of multiple cameras. Previous methods often model it as an MAP problem to find the optimal solution via Bayes Theorem [12,3], a graph theoretic approach [13], and expected weighted similarity [19]. We present an efficient yet effective approach which maximizes the joint linking probability. Assuming that the task of single camera tracking has been already solved; there are m tracks in camera C^a denoted by $\mathcal{T}^a = \{T_1^a, \cdots, T_m^a\}$ and n tracks in camera C^b denoted by $\mathcal{T}^b = \{T_1^b, \cdots, T_n^b\}$ respectively. We may simply create a m by n matrix and find the optimal correspondence between $\mathcal{T}^a$ and $\mathcal{T}^b$. However, in the case of non-overlapping cameras, there exist "blind" areas where objects are invisible. For example, an object which leaves C^a does not necessarily enter C^b as it may either go to the exit in the blind area or return to C^a. We define an extended correspondence matrix of size $(2m + 2n) \times (2m + 2n)$ as follows:

$$\mathbf{H} = \left[\begin{array}{cc|cc} \mathbf{A}_{m\times m} & \mathbf{B}_{m\times n} & \mathbf{F}_{m\times m} & -\infty_{m\times n} \\ \mathbf{D}_{n\times m} & \mathbf{E}_{n\times n} & -\infty_{n\times m} & \mathbf{G}_{n\times n} \\ \hline \mathbf{J}_{m\times m} & -\infty_{m\times n} & \multicolumn{2}{c}{\multirow{2}{*}{$\mathbf{0}_{(m+n)\times(m+n)}$}} \\ -\infty_{n\times m} & \mathbf{K}_{n\times n} & & \end{array}\right] \tag{1}$$

This formulation is inspired by [20], but we made the necessary modification to accommodate all situation which could happen between the tracks of two non-overlapping cameras. The components of each matrix are defined as follows: $B_{ij} = \log P_{link}(T_i^a \rightarrow T_j^b)$ is the linking score of that the tail of T_i^a links to the head of T_j^b. It models the situation that a target leaves C^a and then enters C^b; a similar description is applied to $D_{ij} = \log P_{link}(T_j^a \rightarrow T_i^b)$. $A_{ij} = \log P_{link}(T_i^a \rightarrow T_j^a)$ if $i \neq j$ is the linking score of that the tail of T_i^a links to the head of T_j^a. It models the situation that a target leaves C^a and then re-enters camera a without travelling to camera C^b; a similar description is also applied to $E_{ij} = \log P_{link}(T_i^b \rightarrow T_j^b)$ if $i \neq j$. F_{ij} or G_{ij}if $i = j$ is the score of the T_i^a or T_j^b is terminated. It models the situation that the head of target can not be linked to the tail of any tracks. J_{ij} and K_{ij} if $i = j$ is the score of that the T_i^a or T_j^b is initialized. It models the situation that the tail of target can not link to the head of any track. By applying the Hungarian algorithm to $\mathbf{H}$, the optimal

Table 1. A short summary of the elements in each sub-matrix in **H**, which models all possible situations between the tracks of two non-overlapping cameras. The optimal assignment is solved by Hungarian algorithm.

matrix	description	element
A	the target leaves and returns to C^a	$A_{ij} = -\infty$ if $i = j$
B	the target leaves C^a and enters C^b	B_{ij} is a full matrix
D	the target leaves C^b and enters C^a	D_{ij} is a full matrix
E	the target leaves and returns to C^b	$E_{ij} = -\infty$ if $i = j$
F	the target terminates in C^a	$F_{ij} = -\infty$ if $i \neq j$
G	the target terminates in C^b	$G_{ij} = -\infty$ if $i \neq j$
J	the target is initialized in C^a	$J_{ij} = -\infty$ if $i \neq j$
K	the target is initialized in C^b	$K_{ij} = -\infty$ if $i \neq j$

assignment of association is obtained efficiently. A summary of each sub-matrix in **H** is given in Table 1.

The linking probability, *i.e.* affinity between two tracks T_i and T_j is defined as the product of three important cues(appearance, space, time):

$$P_{link}(T_i \rightarrow T_j) = P_a(T_i, T_j) \cdot P_s\big(e(T_i), e(T_j)\big) \cdot P_t\big(T_i \rightarrow T_j | e(T_i), e(T_j)\big) \quad (2)$$

where $e(T_i)$ denotes the exit/entry region of T_i. Each of three components measures the likelihood of T_i and T_j being the same object. The latter two terms P_s and P_t are spatio-temporal information which can be learned automatically by the methods proposed in [15,3]. We focus on the first term P_a and propose a novel framework of online learning a discriminative appearance affinity model.

5 Discriminative Appearance Affinity Models with Multiple Instance Learning

Our goal is to learn a discriminative appearance affinity model across the cameras at runtime. However, how to choose positive and negative training samples is a major challenge since exact hand-labelled correspondence is not available while learning online. Based on the spatio-temporal constraints, we are able to only exclude some impossible links and retain several possible links, which are called "weakly labelled training examples".

Recent work [5,6] presents promising results on face detection and visual tracking respectively using Multiple Instance Learning (MIL). Compared to traditional discriminative learning, MIL describes that samples are presented in "bags", and the labels are provided for the bags instead of individual samples. A positive "bag" means it contains at at least one positive sample; a negative bag means all samples in this bag are negative. Since some flexibility is allowed for the labelling process, we may use the "weakly labelled training examples" by spatio-temporal constraints and apply a MIL boosting algorithm to learn the discriminative appearance affinity model.

5.1 Collecting Training Samples

We propose a method to collect weakly labelled training samples using spatio-temporal constraints. To learn an appearance affinity model between cameras, a training sample is defined as a pair of tracks from two cameras respectively. Based on the tracks generated by a robust single camera multi-target tracker, we make a conservative assumption: any two tracks from two non-overlapping cameras which overlap in time represent different targets. It is based on the observation that one target can not appear at different locations at the same time. Positive samples are more difficult to obtain since there is no supervised information to indicate which two tracks among two cameras represent the same objects. In other words, the label of "+1" can not be assigned to individual training samples. To deal with the challenging on-line labelling problem, we collect possible pairs of tracks by examining spatio-temporal constraints and put them into a "bag" which is labelled "+1". The MIL boosting is applied to learn the desired discriminative appearance affinity model.

In our implementation, there are two set to be formed for each track: a set of "similar" tracks and a set of "discriminative" tracks. For a certain track T_j^a in camera C^a, each element in its "discriminative" set $\mathcal{D}_j^b$ indicates a target T_k^b in camera C^b which is impossible to be the same target with T_j^a; each element in the "similar" set $\mathcal{S}_j^b$ represents a possible target T_k^b in C^b which might be the same target with T_j^a. These cases are described as:

$$\begin{aligned} T_k^b \in \mathcal{S}_j^b \text{ if } P_s(T_j^a \to T_k^b) \cdot P_t(e(T_j^a), e(T_k^b)) > \theta \\ T_k^b \in \mathcal{D}_j^b \text{ if } P_s(T_j^a \to T_k^b) \cdot P_t(e(T_j^a), e(T_k^b)) = 0 \end{aligned} \tag{3}$$

The threshold θ is adaptively chosen to maintain a moderate number of instances included in each positive bag. The training sample set $\mathcal{B} = \mathcal{B}^+ \cup \mathcal{B}^-$ can be denoted by

$$\begin{aligned} \mathcal{B}^+ &= \left\{ x_i : \{T_j^a, T_k^b\}, \forall T_k^b \in \mathcal{S}_j^b; y_i : +1 \right\} \\ \mathcal{B}^- &= \left\{ x_i : (T_j^a, T_k^b), \text{if } T_k^b \in \mathcal{D}_j^b; y_i : -1 \right\} \end{aligned} \tag{4}$$

where each training sample x_i may contain multiple pairs of tracks which represents a bag. A label is given to a bag.

5.2 Representation of Appearance Model and Similarity Measurement

To build a strong appearance model, we begin by computing several local features to describe a tracked target. In our design, three complementary features: color histograms, covariance matrices, and histogram of gradients (HOG) constitute the feature pool. Given a tracked target, features are extracted at different locations and different scales from the head and tail part to increase the descriptive ability.

We use RGB color histograms to represent the color appearance of a local image patch. Histograms have the advantage of being easy to implement and having well studied similarity measures. Single channel histograms are concatenated to form a vector $\mathbf{f}_{RGB_i}$, but any other suitable color space can be used. In our implementation, we use 8 bins for each channel to form a 24-element vector. To describe the image texture, we use a descriptor based on covariance matrices of image features proposed in [21]. It has been shown to give good performance for texture classification and object categorization. To capture shape information, we choose the Histogram of Gradients (HOG) Feature proposed in [22]. In our design, a 32D HOG feature $\mathbf{f}_{HOG_i}$ is extracted over the region R; it is formed by concatenating 8 orientations bins in 2×2 cells over R.

In summary, the appearance descriptor of a track T_i can be written as:

$$\mathcal{A}_i = (\{\mathbf{f}^l_{RGB_i}\}, \{\mathbf{C}^l_i\}, \{\mathbf{f}^l_{HOG_i}\}) \tag{5}$$

where $\mathbf{f}^l_{RGB_i}$ is the feature vector for color histogram, $\mathbf{C}^l_i$ is the covariance matrix, and $\mathbf{f}^l_{HOG_i}$ is the 32D HOG feature vector. The superscript l means that the features are evaluated over region R^l.

Given the appearance descriptors, we can compute similarity between two patches. The color histogram and HOG feature are histogram-based features so standard measurements, such as χ^2 distance, Bhattacharyya distance, and correlation coefficient can be used. In our implementation, correlation coefficient is chosen for simplicity. The distance measurement of covariance matrices is determined by solving a generalized eigenvalues problem, which is described in [21].

After computing the appearance model and the similarity between appearance descriptors at different regions, we form a feature vector by concatenating the similarity measurements with different appearance descriptors at multiple locations. This feature vector gives us a feature pool that we can use an appropriate boosting algorithm to construct a strong classifier.

5.3 Multiple Instance Learning

Our goal is to design a discriminative appearance model which determines the affinity score of appearance between two objects in two different cameras. Again, a sample is defined as a pair of targets from two cameras respectively. The affinity model takes a pair of objects as input and returns a score of real value by a linear combination of weak classifiers. The larger the affinity score, the more likely that two objects in one sample represent the same target. We adopt the MIL Boosting framework proposed in [5] to select the weak classifiers and their corresponding weighted coefficients. Compared to conventional discriminative boosting learning, training samples are not labelled individually in MIL; they form "bags" and the label is given to each bag, not to each sample. Each sample is denoted by x_{ij}, where i is the index for the bag and j is the index for the sample within the bag. The label of each bag is represented by y_i where $y_i \in \{0, 1\}$.

Although the known labels are given to bags instead of samples, the goal is to learn the the instance classifier which takes the following form:

$$H(x_{ij}) = \sum_{t=1}^{T} \alpha_t h_t(x_{ij}) \tag{6}$$

In our framework, the weak hypothesis is from the feature pool obtained by Section 5.2. We adjust the sign and normalize $h(x)$ to be in the restricted range $[-1, +1]$. The sign of $h(x)$ is interpreted as the predicted label and the magnitude $|h(x)|$ as the confidence in this prediction.

The probability of a sample x_{ij} being positive is defined as the standard logistic function,

$$p_{ij} = \sigma(y_{ij}) = \frac{1}{1 + \exp(-y_{ij})} \tag{7}$$

where $y_{ij} = H(x_{ij})$. The probability of a bag being positive is defined by the "noisy OR" model:

$$p_i = 1 - \prod_j (1 - p_{ij}) \tag{8}$$

If one of the samples in a bag has a high probability p_{ij}, the bag probability p_i will be high as well. This property is appropriate to model that a bag is labelled as positive if there is at least one positive sample in this bag. MIL boosting uses the gradient boosting framework to train a boosting classifier that maximizes the log likelihood of bags:

$$\log L(H) = \sum_i y_i \log p_i + (1 - y_i) \log(1 - p_i) \tag{9}$$

The weight of each sample is given as the derivative of the loss function $\log L(H)$ with respect to the score of that sample y_{ij}:

$$w_{ij} = \frac{\partial \log L(H)}{\partial y_{ij}} = \frac{y_i - p_i}{p_i} p_{ij} \tag{10}$$

Our goal is to find $H(x)$ which maximizes (9), where $H(x)$ can be obtained by sequentially adding new weak classifiers. In the t-th boosting round, we aim at learning the optimal weak classifier h_t and weighted coefficient α_t to optimize the loss function:

$$(\alpha_t, h_t) = \arg\min_{h,\alpha} \log L(H_{t-1} + \alpha h) \tag{11}$$

To find to the optimal (α_t,h_t), we follow the framework used in [23,5] which views boosting as a gradient descent process, each round it searches for a weak classifier h_t to maximize the gradient of the loss function. Then the weighted coefficient α_t is determined by a linear search to maximize $\log L(H + \alpha_t h_t)$. The learning procedure is summarized in Algorithm 1.

Algorithm 1. Multiple Instance Learning Boosting

Input: $\mathcal{B}^+ = \{(\{x_{i1}, x_{i2}, ...\}, +1)\}$: Positive bags
$\mathcal{B}^- = \{(\{x_{i1}, x_{i2}, ...\}, -1)\}$: Negative bags
$\mathcal{F} = \{\mathbf{h}(x_{ij})\}$: Feature pools

Initialize $H = 0$
for $t = 1$ to T **do**
  for $k = 1$ to K **do**
    $p_{ij}^k = \sigma(H + h_k(x_{ij}))$
    $p_i^k = 1 - \prod_j (1 - p_{ij}^k)$
    $w_{ij}^k = \frac{y_i - p_i^k}{p_i^k} p_{ij}^k$
  end for
  Choose $k^* = \arg\max_k \sum_{ij} w_{ij}^k h_k(x_{ij})$
  Set $h_t = h_{k^*}$
  Find $\alpha^* = \arg\max_\alpha \log L(H + \alpha h_t)$ by linear search
  Set $\alpha_t = \alpha^*$
  Update $H \leftarrow H + \alpha_t h_t$
end for

Output: $H(x) = \sum_{t=1}^{T} \alpha_t h_t(x)$

6 Experimental Results

The experiments are conducted on a three-camera setting with disjoint FOVs. First, we evaluate the effectiveness of our proposed on-line learned discriminative appearance affinity model by formulating the correspondence problem as a binary classification problem. Second, for a real scenario of multiple non-overlapping cameras, the evaluation metric is defined, and the tracking results using our proposed system are presented. It is shown that our method achieves good performance in a crowded scene. Some graphical examples are also provided.

6.1 Comparison of Discriminative Power

We first evaluate the discriminative ability of our appearance affinity model, independent of the tracking framework that it will be embedded in. Given the tracks in each camera, we manually label the pairs of tracks that should be associated to from the ground truth. Affinity scores are computed among every possible pair in a time sliding window by four methods: (1) the correlation coefficients of two color histogram; (2) the model proposed in Section 5 but without MIL learning, *i.e.* with equal coefficients α_t; (3) off-line MIL learning, *i.e.* learning is done on another time sliding window; (4) MIL learning on the same time

Table 2. The comparison of the Equal Error Rate using different appearance affinity models. It shows that the on-line learning method has the most discriminative power.

Camera pair	color only	no learning	off-line learning	on-line learning
C^1,C^2	0.231	0.156	0.137	**0.094**
C^2,C^3	0.381	0.222	0.217	**0.159**

window. In a three-camera setting, the experiments are done in two camera pairs (C^1,C^2) and (C^2,C^3); equal error rate in two tasks is the metric to evaluate the performance. In (C^1,C^2), the number of evaluated pairs is 434 and the number of positive pairs is 35. In (C^2,C^3), the number of evaluated pairs is 148 and the number of positive pairs is 18. The length of time sliding window is 5000. The experimental results are shown in Table 2. In each camera pair, the model using online MIL learning achieves the lowest equal error rate compared to the other three methods.

6.2 Evaluation Metrics

In previous work, quantitative evaluation of multi-target tracking across multiple cameras is barely mentioned or simply a single number *e.g.* tracking accuracy is used. It is defined as the ratio of the number of objects tracked correctly to the total number of objects that passed through the scene in [2,3]. However, it may not be a suitable metric to measure the performance of a system fairly, especially in a crowded scene where targets have complicated interactions. For example, if two tracked targets exchange their identities twice while travelling across a series of three cameras should be worse than if they exchange only once. Nevertheless, these two situations are both counted as incorrect tracked objects in the metric of "tracking accuracy". We need a more complete metric to evaluate the performance of inter-camera track association.

In the case of tracking within a single camera, fragments and ID switches are two commonly used metrics. We adopt the definitions used in [24] and apply it to the case of tracking across cameras. Assuming that multiple targets tracking in a single camera is obtained, we only focus on the fragments and ID switches which are not defined within cameras. Given the tracks in two cameras C^a and C^b: $\mathcal{T}^a = \{T_1^a, \cdots, T_m^a\}$ and $\mathcal{T}^b = \{T_1^b, \cdots, T_n^b\}$, the metrics in tracking evaluation are:

- Crossing Fragments(X-Frag): The total number of times that there is a link between T_i^a and T_j^b in the ground truth, but missing in the tracking result.
- Crossing ID switches(X-IDS): The total number of times that there is no link between T_i^a and T_j^b in the ground truth, but existing in the tracking result.
- Returning Fragments(R-Frag): The total number of times that there is link between T_i^a and T_j^a which represents a target leaving and returning to C^a in ground truth, but missing in the tracking result.

Table 3. Tracking results using different appearance models with our proposed metrics. The lower the numbers, the better performance it is. It shows that our on-line learned appearance affinity models achieve the best results.

Method	X-Frag	X-IDS	R-Frag	R-IDS
(a)input tracks	206	0	15	0
(b)color only	9	18	12	8
(c)off-line learning	6	15	11	7
(d)on-line learning	**4**	**12**	**10**	**6**

• Returning ID switches(R-IDS): The total number of times that there is no link between T_i^a and T_j^a which means they represent different targets in ground truth, but existing in the tracking result.

For example, there are T_1^a, T_2^a in C^a, and T_1^b, T_2^b in C^b. In the ground truth, (T_1^a, T_1^b) and (T_2^a, T_2^b) are the linked pairs. If they switch their identities in the tracking result, *i.e.* (T_1^a, T_2^b) and (T_2^a, T_1^b) are the linked pairs, that is considered as 2 X-frag and 2 X-IDS. This metric is more strict but well-defined than the traditional definition of fragments and ID switches. Similar descriptions apply to R-Frag and R-IDS. The lower these four metrics, the better is the tracking performance.

6.3 Tracking Results

The videos used in our evaluation are captured by three cameras in a campus environment with frame size of 852×480 and length of 25 minutes. It is more challenging than the dataset used in the previous works in the literature since this dataset features a more crowded scene (2 to 10 people per frame in each camera). There are many inter-object occlusions and interactions and people walking across cameras occurs often. The multi-target tracker within a camera we use is based on [24], which is a detection-based tracking algorithm with hierarchical association.

We compare our approach with different appearance models. The results are also shown in Table 3. The result of (a) represents the input, *i.e.* no linking between any tracks in each camera. The result of (b) uses only color histogram is used as the appearance model. In the result of (c), our proposed appearance model is used but learned in an off-line environment, which means the coefficients α_t are fixed. The result of (d) uses our proposed appearance models. It shows that our proposed on-line learning method outperforms these two appearance models. This comparison justifies that our stronger appearance model with on-line learning improves the tracking performance. Some association results are shown in Figure 3. It shows that our method finds the correct association among multiple targets in a complex scenen, *e.g.* people with IDs of 74, 75, and 76 when they travel from camera 2 to camera 1.

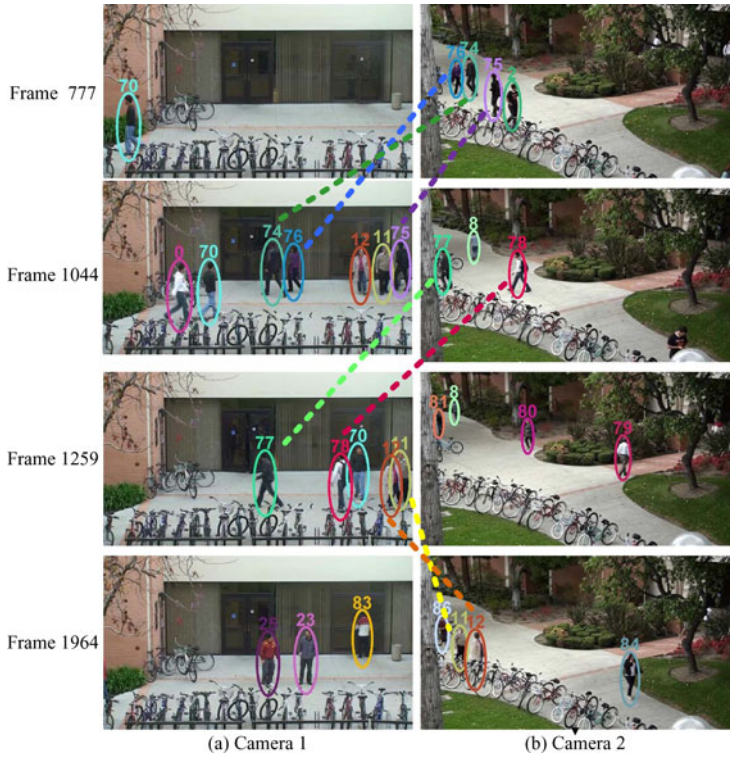

Fig. 3. Sample tracking results on our dataset. Some tracked people travelling through the cameras are linked by dotted lines. For example, the targets with IDs of 74, 75, and 76 leave Camera 2 around the same time, our method finds the correct association when they enter Camera 1. This figure is best viewed in color.

7 Conclusion

We describe a novel system for associating multi-target tracks across multiple non-overlapping cameras. The contribution of this paper focuses on learning a discriminative appearance affinity model at runtime. To solve the ambiguous labelling problem, we adopt Multiple Instance Learning boosting algorithm to learn the desired discriminative appearance models. An effective multi-object correspondence optimization framework for intra-camera track association problem is also presented. Experimental results on a challenging dataset show the robust performance by our proposed system.

Acknowledgments. This paper is based upon work supported in part by, the U.S. Army Research Laboratory and the U.S. Army Research Office under contract number W911NF-08-C-0068 and, in part, by Office of Naval Research under grant number N00014-10-1-0517.

References

1. Porikli, F.: Inter-camera color calibration by correlation model function. In: ICIP (2003)
2. Javed, O., Shafique, K., Shah, M.: Appearance modeling for tracking in multiple non-overlapping cameras. In: CVPR (2005)
3. Chen, K.W., Lai, C.C., Hung, Y.P., Chen, C.S.: An adaptive learning method for target tracking across multiple cameras. In: CVPR (2008)
4. Dietterich, T.G., Lathrop, R.H., Lozano-Perez, T., Pharmaceutical, A.: Solving the multiple-instance problem with axis-parallel rectangles. Artificial Intelligence 89, 31–71 (1997)
5. Viola, P., Platt, J.C., Zhang, C.: Multiple instance boosting for object detection. In: NIPS (2005)
6. Babenko, B., Yang, M.H., Belongie, S.: Visual Tracking with Online Multiple Instance Learning. In: CVPR (2009)
7. Cai, Q., Aggarwal, J.: Tracking human motion in structured environments using a distributed-camera system. IEEE Tran. on PAMI 21, 1241–1247 (1999)
8. Collins, R., Lipton, A., Fujiyoshi, H., Kanade, T.: Algorithms for cooperative multisensor surveillance. Proceedings of the IEEE 89, 1456–1477 (2001)
9. Khan, S., Shah, M.: Consistent labeling of tracked objects in multiple cameras with overlapping fields of view. IEEE Tran. on PAMI 25, 1355–1360 (2003)
10. Huang, T., Russell, S.: Object identification in a bayesian context. In: IJCAI (1997)
11. Pasula, H., Russell, S., Ostl, M., Ritov, Y.: Tracking many objects with many sensors. In: IJCAI (1999)
12. Kettnaker, V., Zabih, R.: Bayesian multi-camera surveillance. In: CVPR (1999)
13. Javed, O., Rasheed, Z., Shafique, K., Shah, M.: Tracking across multiple cameraswith disjoint views. In: ICCV (2003)
14. Dick, A.R., Brooks, M.J.: A stochastic approach to tracking objects across multiple cameras. In: Australian Conference on Artificial Intelligence (2004)
15. Makris, D., Ellis, T., Black, J.: Bridging the gaps between cameras. In: CVPR (2004)
16. Gilbert, A., Bowden, R.: Tracking objects across cameras by incrementally learning inter-camera colour calibration and patterns of activity. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3952, pp. 125–136. Springer, Heidelberg (2006)
17. Sturges, J., Whitfield, T.: Locating basic colour in the munsell space. Color Research and Application 20, 364–376 (1995)
18. Song, B., Roy-Chowdhury, A.: Robust tracking in a camera network: A multi-objective optimization framework. IEEE Journal of Selected Topics in Signal Processing 2, 582–596 (2008)
19. Song, B., Roy-Chowdhury, A.K.: Stochastic adaptive tracking in a camera network. In: ICCV (2007)
20. Huang, C., Wu, B., Nevatia, R.: Robust object tracking by hierarchical association of detection responses. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 788–801. Springer, Heidelberg (2008)
21. Tuzel, O., Porikli, F., Meer, P.: Region covariance: A fast descriptor for detection and classification. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3952, pp. 589–600. Springer, Heidelberg (2006)
22. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR (2005)
23. Mason, L., Baxter, J., Bartlett, P., Frean, M.: Boosting algorithms as gradient descent in function space (1999)
24. Kuo, C.H., Huang, C., Nevatia, R.: Multi-target tracking by on-line learned discriminative appearance models. In: CVPR (2010)