Track-clustering Error Evaluation for Track-based Multi-Camera Tracking System Employing Human Re-identification

Chih-Wei Wu[1], Meng-Ting Zhong[1], Yu Tsao[2], Shao-Wen Yang[3], Yen-Kuang Chen[3], and Shao-Yi Chien[1]

[1]National Taiwan University, Taiwan
[2]Academia Sinica, Taiwan
[3]Intel Cooperation, USA

Abstract

In this study, we present a set of new evaluation measures for the track-based multi-camera tracking (T-MCT) task leveraging the clustering measurements. We demonstrate that the proposed evaluation measures provide notable advantages over previous ones. Moreover, a distributed and online T-MCT framework is proposed, where re-identification (Re-id) is embedded in T-MCT, to confirm the validity of the proposed evaluation measures. Experimental results reveal that with the proposed evaluation measures, the performance of T-MCT can be accurately measured, which is highly correlated to the performance of Re-id. Furthermore, it is also noted that our T-MCT framework achieves competitive score on the DukeMTMC dataset when compared to the previous work that used global optimization algorithms. Both the evaluation measures and the inter-camera tracking framework are proven to be the stepping stone for multi-camera tracking.

1. Introduction

Multi-camera tracking (MCT), also known as multi-target multi-camera tracking (MTMC), is a task of tracking multiple objects through multiple cameras, and is the ultimate goal for intelligent camera networks. It is of great importance for surveillance systems to learn to track objects automatically, since video data collected by end-point cameras is overwhelming for both human understanding and data transferring, especially in the era of Internet of things (IoT). A complete MCT system usually consists of two major components: single camera tracking (SCT) and track-based multi-camera tracking (T-MCT) [2], as shown in Figure 1. The SCT is first applied to associate detections into tracks, followed by T-MCT to re-identify each track and to form trajectories for identities. Please note that we use the term T-MCT instead of "inter-camera tracking" in this paper, because T-MCT does not only associate tracks of "different cameras". The most challenging and unexplored part of MCT lies within T-MCT. Illumination changes, view angle variation and object appearance inconsistency all contribute to making T-MCT much more difficult than SCT, a problem considerably improved over the past few years in the MOT challenge [12].

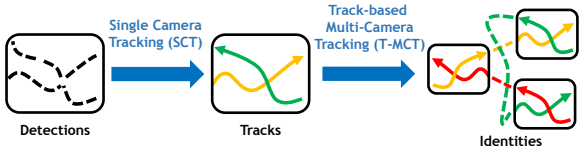


Figure 1: A typical multi-camera tracking (MCT) pipeline.

According to Zheng *et al.* [21], around ten years ago, a pioneering work [3] first framed T-MCT as a visual matching problem and introduced an independent research field—human re-identification (Re-id). Re-id, which focuses on identifying individuals from gallery of images shot by different cameras, is often viewed as an image retrieval problem rather than a tracking problem. Early datasets of Re-id, for example, ViPER[4], PRID[5], and iLVDS-VID[15], provide images of the same identity between only two cameras. More recently, some algorithms and benchmarks have been proposed to address Re-id on larger galleries, which comprised images from more than two cameras [9, 20]. Moreover, there are some researches about the implementation of Re-id on multi-shot dataset [21, 19]. By solving Re-id on a larger scale, multi-shot dataset, we are one step closer to leveraging Re-id algorithms on T-MCT. An urge to design an evaluation measure and a system framework suited for bridging Re-id and T-MCT is on the horizon.

T-MCT is a long-standing problem, while well-constructed evaluation measures have not been introduced

until recently. There are three main challenges that hinder us from fairly evaluating T-MCT. First, different types of error are valued differently depending on how we frame the T-MCT task. For example, the track-order based Frag/IDS measurement is adopted in USC[8, 1] and CamNeT[18]. Meanwhile, the detection-based evaluation measure, which considers the tracking length of each identity, is developed by DukeMTMC[13]. Later in subsection 3.1, we will detail these two types of measures. Second, the T-MCT evaluation measures need to isolate the errors in the SCT stage. When considering the typical MCT pipeline (in Figure 1), the SCT associates detections into tracks, and then T-MCT associates tracks into identities. Intuitively, the number of associations needs to make SCT much larger than that at T-MCT. Thus, the previous MCT evaluation measures, which evaluate MCT as a whole, may not be indicative for T-MCT. Finally, the evaluation measures should be able to deal with tricky issues in T-MCT, such as jump links, as illustrated in Figure 4(b), and lost forever issue, as illustrated in Figure 4(c). We will discuss these issues in Section 3.

In this paper, motivated by the need to bridge the gap between Re-id and MCT, we aim to propose a set of new evaluation measures that is suitable for measuring T-MCT. We incorporate the following properties when designing the evaluation measures:

- *Isolate the T-MCT errors from the SCT errors.* We intend to only consider the T-MCT errors to indicate the performance of T-MCT. The SCT errors should be calculated separately from the T-MCT errors using other relevent measures, such as MOTA and MOTP [6].

- *Robust against various tracking scenarios.* The measures can handle different situations in T-MCT with objective index and should be consistent with human intuition.

- *Robust against different camera settings.* The measures can be employed with different camera settings, such as non-overlapping cameras, overlapping cameras, and heterogeneous cameras.

To demonstrate our idea, in Section 2, we first introduce our track-clustering based evaluation measures specifically designed for interpreting Re-id based T-MCT systems. We then compare the proposed evaluation measures with related works on a T-MCT task in Section 3 through different scenarios. Next, in Section 4, we proposed an online distributed T-MCT framework to demonstrate the scheme for embedding various Re-id algorithms in T-MCT. Finally, in Section 5, the T-MCT evaluation results as well as Re-id evaluation results are presented to confirm the validity of the proposed measures and test the ability of our Re-id based T-MCT framework.

2. Proposed Track-clustering Based Evaluation Measures

2.1. The Proposed Evaluation Measures

In this paper, we propose a set of track-clustering based evaluation measures to test T-MCT performance. The basic concept of the track-clustering based evaluation measures is to treat the T-MCT task as a clustering problem and measure the correctness of correspondences predicted by the T-MCT system. Correspondence is defined as the relation between a pair of tracks in algorithm results (AR), where tracks are a series of consecutive detections generated in SCT. Let us denote two AR tracks as T_i and T_j, both of which belong to SCT-generated AR track set τ, then the overall correspondence Φ is defined as:

$$\Phi = \{(T_i,\ T_j) \mid i \neq j \wedge T_i, T_j \in \tau\} \tag{1}$$

As illustrated in Figure 2, we frame AR tracks as data samples in the clustering problem, whereas the correspondence is the relation between data sample pairs. Here, we make an assumption that each AR track only tracks one ground truth (GT) identity. Detailed process of how to achieve this assumption will be discussed in subsection 2.2. Each track has two attributes: an identity obtained by AR, and a GT identity that it actually tracks. For each track T_k, we define the identity given by AR as the class of data sample, denoted as A_k. Class is the predicted cluster for a data sample, represented by dotted circle in Figure 2. As for the GT identity T_k actually tracks, we define it as the label of data sample, denoted as G_k. Label is the actual object type of a data sample, represented by different sample shapes in Figure 2. In this manner, we can then apply typical external clustering measurement on T-MCT, namely the F-measure.

The new evaluation measures are defined based on counting correspondences between AR tracks. Figure 2 demonstrates the definition of true positive (TP), false positive (FP), and false negative (FN) correspondences, which follows the definition in clustering. Each data sample T_k has a class and a label, corresponding to A_k and G_k for a

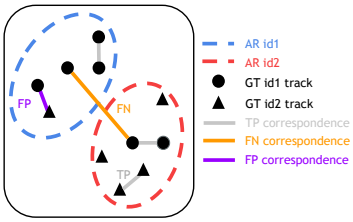


Figure 2: Our proposed track-clustering based evaluation measures. Only part of TP/FP/FN correspondences are displayed for a cleaner presentation. Each data sample represents a track generated by an SCT tracker.

track. TP, FP, and FN correspondences are defined as

$$TP = |\{(T_i,\ T_j)|A_i = A_j \wedge G_i = G_j \wedge (T_i, T_j) \in \Phi\}|, \tag{2}$$

$$FP = |\{(T_i,\ T_j)|A_i = A_j \wedge G_i \neq G_j \wedge (T_i, T_j) \in \Phi\}|, \tag{3}$$

$$FN = |\{(T_i,\ T_j)|A_i \neq A_j \wedge G_i = G_j \wedge (T_i, T_j) \in \Phi\}|. \tag{4}$$

Finally, the evaluation measures are defined as:

$$ClustP = \frac{TP}{TP + FP}, \tag{5}$$

$$ClustR = \frac{TP}{TP + FN}, \tag{6}$$

$$ClustF1 = \frac{2 \times ClustP \times ClustR}{ClustP + ClustR}, \tag{7}$$

where TP, FN, and FP are defined in Eq. 2,3,4. ClustP and ClustR represent precision and recall of correspondences, respectively, and ClustF1, the harmonic mean of ClustP and ClustR, serves as an overall score.

For easier understanding of how to calculate the new indexes, let us consider the example shown in Figure 2. In this example, TP indicates correspondences between track pairs of same label and class, which is $TP = \binom{4}{2} + \binom{2}{2} + \binom{4}{2} = 13$. To simplify the calculation, $TP + FP$ can be obtained by $TP + FP = P$, where P indicates correspondences between track pairs with same class. That is, $P = \binom{5}{2} + \binom{6}{2} = 25$. Similarly, $TP + FN$ can be obtained by $TP + FN = T$, where T indicates correspondences between track pairs with same label, namely $T = \binom{6}{2} + \binom{5}{2} = 25$. The final score would be $ClustP = ClustR = 13/25 = 52\%$ and $ClustF1 = (2 \times 0.52 \times 0.52)/(0.52 + 0.52) = 52\%$.

Calculating the track-based F-measure empowers our evaluation measures with desired properties for evaluating T-MCT. Since we model SCT tracks as data samples, we eliminate SCT errors and consider those errors only made in T-MCT. Meanwhile, measuring correspondences between pairs is intuitive for evaluating image retrieval algorithms, such as Re-id algorithms, the algorithms we embed in our T-MCT framework. We can even evaluate T-MCT for general camera settings. The procedure for general camera setting is the same as the one we introduced here.

2.2. Overall Evaluation Process

The overall evaluation process consists of three steps. First, we break down cross-camera trajectory of each AR identity into AR tracks. This operation can be done by simply cutting out tracks when a trajectory's detection is disconnected in time. In the second step, we determine which GT identity each track is tracking through track identification. After that, track-clustering based evaluation measures can be computed as described in previous subsection.

Track identification, which is the second step of overall evaluation process, is composed of two stages: per-frame matching and max occurrence ID pooling. Per-frame matching assigns the most matched GT identity to each AR detection box in every video frame. To do so, we perform a bipartite matching between GT boxes and AR detection boxes within the same frame, followed by thresholding intersection-over-union ratio (IoU) of every match. Let us denote the i-th GT box in a frame as $d_t^{GT}(i)$ and the j-th AR detection box in the same frame as $d_t^{AR}(j)$. A cost matrix C is then constructed, where its element $c_{i,j}$ is assigned with IoU of $d_t^{GT}(i)$ and $d_t^{AR}(j)$. By performing Hungarian algorithm on cost matrix C, each $d_t^{AR}(j)$ is matched to a $d_t^{GT}(i)$ in the bipartite matching process. We then threshold IoU of matched pairs to assure the correctness of assignment. In this study, the threshold is set to 0.5. By denoting GT identity of $d_t^{GT}(i)$ as $g_t^{GT}(i)$, and the best matched GT identity of $d_t^{AR}(j)$ as $g_t^{AR}(j)$, we have:

$$g_t^{AR}(j) = \begin{cases} g_t^{GT}(i), & \text{if } c_{i,j} > 0.5 \\ false\ alarm\ flag, & \text{otherwise} \end{cases} \tag{8}$$

where (i, j) are assignments made by Hungarian algorithm on the cost matrix C.

After obtaining the best matched GT identity of all AR detection boxes, max occurrence ID pooling is performed to vote for the most matched GT identity of an AR track. Please note that we denote an AR track as T in the previous subsection. An AR track is composed of a set of AR detection boxes, denoted as $T = \{d^{AR}\}$. Each d^{AR} has a corresponding best matched GT identity $d^{AR} \rightarrow g^{AR}$. We determine the best matched GT identity G for AR track T, based on:

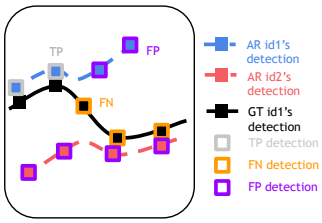
$$\gamma = max\ occurrence\ g^{AR}, \tag{9}$$

$$G = \begin{cases} \gamma, \text{ if } \frac{|\gamma|}{|T|} > 0.5 \wedge \gamma \neq false\ alarm\ flag \\ false\ alarm\ id, \text{ otherwise} \end{cases} \tag{10}$$

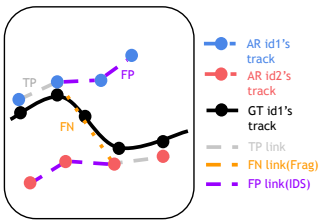
If the number of votes, which is the occurrence number of γ, exceeds half the length of T, we determine that T is tracking GT identity G. Otherwise, a false alarm id is assigned to G to represent a false alarm track. It is also worth noting that each false alarm track has different false alarm id ,since it is not reasonable to form correspondence between two false alarm tracks.

3. Related Works

In this section, we first review two major evaluation measures for MCT, the detection-based and track-order based measures, and then compared them with the proposed measures through three toy examples.



(a) Detection-based evaluation measure



(b) Track-order based evaluation measure

Figure 3: Two previous MCT evaluation measures. Data samples in (a) respresent AR detection boxes, whereas data samples in (b) represent AR tracks generated from SCT.

3.1. Detection- and Track-order Based Evaluation Measure

Detection-based. A detection-based evaluation measure has been proposed by DukeMTMC [13] recently, known as $IDmeasure$. Basically, it measures tracking length of each matched GT identity and matched AR identity, as illustrated in Figure 3(a). When a GT identity and an AR identity are matched to each other, TP are the matched detections whereas FP and FN are mismatch detections in trajectory of AR identity and the GT identity, respectively. IDP, IDR, and $IDF1$ are the evaluation indexes, directly computed from TP, FP, and FN. Its overall procedure mainly measures MCT system as a whole, not T-MCT only. Consequently, we also consider the $IDmeasure$ difference between SCT and MCT, suggested by the original paper that it can capture ICT performance. For further detail, please refer to the corresponding paper [13]. To be specific, we will visualize $IDP(MCT), IDR(MCT), IDF1(MCT)$ and $IDF1(ICT)$ in this work.

Track-order based. In early MCT contests and works[18, 8, 1], a track-order based evaluation measure is usually adopted. It derives from the evaluation measure of multi-object tracking (MOT), ID switch (IDS) and Fragmentation (Frag). However, none of the previous works clarifies the detail of employing IDS and Frag to MCT. Hence we demonstrate our definition of IDS and Frag for T-MCT in Figure 3(b). IDS and Frag correspond to FP and FN links in the figure respectively. IDS is defined as a link between AR tracks having the same AR identity A but different GT identity G, and Frag is a link between AR tracks having different A but same G. Notice that the difference between track-order based and our track-clustering based evaluation measures is that the former only counts links between successive AR tracks strictly in time order, while the later computes correspondences between track pairs regardless of time. That is, if an AR identity consists of three tracks with time order of T_1, T_2, T_3, links only exists between (T_1, T_2) and (T_2, T_3), but not (T_1, T_3). From another point of view, the track-order based evaluation measure considers T-MCT as a graph problem. IDS equally measures how many edges should be cut, and Frag measures how many edges should be made to restore the correct graph. From the definition stated above, we defined the Link Precision (LinkP) and Link Recall (LinkR) as:

$$LinkP = 1 - IDS\ Rate = \frac{TP}{TP + FP}, \quad (11)$$

$$LinkR = 1 - Frag\ Rate = \frac{TP}{TP + FN}, \quad (12)$$

where TP, FN, and FP are defined as Figure 3(b).

3.2. Toy Examples

Three toy examples are presented to demonstrate the properties of different evaluation measures. All of them are common scenarios in MCT.

Isolate SCT error. In Figure 4(a), this example demonstrates the difference between detection-based and track-clustering based evaluation measures. In the example, there are two GT identities (gray dotted line) and only one AR identity (blue line). The two blocks represent two different camera views. We can notice that T-MCT is totally wrong since the only correspondence between track in camera 1 and track in camera 2 should not be associated. SCT within each camera is also imperfect since the tracks alternates between two GT identities. The results for detection-based evaluation measure are $TP = 10$, $FP = 10$, $FN = 30$, and thus $IDP(MCT) = 50\%$, $IDR(MCT) = 25\%$, $IDF1(MCT) = 16.67\%$. As for SCT, $TP = 12$, $FP = 8$, $FN = 28$ resulting in $IDF1(SCT) = 20\%$, hence we have $IDF1(ICT) = 20\% - 16.67\% = 3.3\%$. On the other hand, for track-clustering based evaluation measure, $TP = 0$, $FP = \binom{2}{2} = 1$, $FN = 0$ resulting in $ClustF1 = 0\%$. Since this is a T-MCT fail scenario, we expect the result to be 0%, which $ClustF1$ reports correctly and neither $IDF1, IDP, IDR$ for MCT or $IDF1$ for ICT does. It shows that our evaluation measure has the ability to isolate T-MCT error from SCT error whereas detection-based evaluation measure does not.

Handle jump links. Figure 4(b) shows an example comprising two scenarios for comparing track-order based and track-clustering based evaluation measures. Both scenario consists of two GT identities (gray dotted line) and two AR

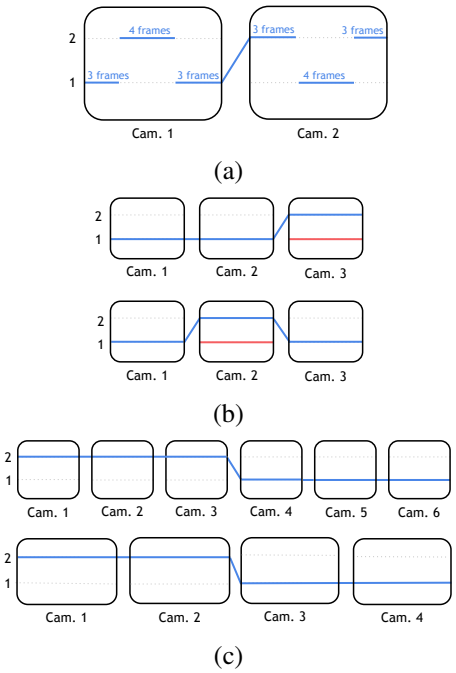


Figure 4: Three toy examples for comparing the track-clustering based, detection-based, and track-order based evaluation measures.

identities (red/blue line). The difference between the two is that situations in camera 2 and 3 are switched. In this example, T-MCT performance should be similar in both scenarios. The track-order based evaluation measure assigns both $LinkP$ and $LinkR$ with $1/(1+1) = 50\%$ in upper scenario and $0/(0+2) = 0\%$ in lower scenario. Radical changes from 50% to 0% are observed in both index due to the fact that track-order based evaluation measure considers time order. On the other hand, track-clustering based evaluation measure, which calculates correspondences between all pairs, yields $ClustP = ClustR = 1/3 = 33\%$ for both scenarios ($TP = \binom{2}{2} = 1$, $FP = \binom{3}{2} + 0 = 3$, $FN = \binom{3}{2} + 0 = 3$). We can conclude that track-clustering based evaluation measure is more robust in this example.

Handle lost forever issue. The last example emphasizes the ability of penalizing lost forever issue with track-clustering based evaluation measure over track-order based evaluation measure. As illustrated in Figure 4(c), the example has two GT identities but only one AR identity. The system mistakens tracks of two different identities as the same one. This type of error is undesirable for practical applications since identity 2 is lost forever after the system makes a wrong association. We expect a low evaluation score in this example, even as camera amount grows. In this situation, the upper scenario reports $4/(4+1) = 80\%$ for $LinkP$, while the track-clustering based evaluation measure reports $(\binom{3}{2} + \binom{3}{2})/\binom{6}{2} = 6/15 = 40\%$ for $ClustP$. In the bottom scenario, track-order based evaluation index reports $2/(2+1) = 66\%$ for $LinkP$, while track-clustering based evaluation index reports $(\binom{2}{2} + \binom{2}{2})/\binom{4}{2} = 2/6 = 33\%$ for $ClustP$. We can note that $LinkP$ is increased from 66% to 80%, and $ClustP$ is increased from 33% to 40% when the track amount increases; the former grows faster toward 100% while the later grows slower towards 50%; we also confirm that the tendencies toward 100% and 50%, respectively in our supplementary material. In summary, we can conclude that one of the advantages of the track-clustering based evaluation measure over the track-order based counterpart is that it penalizes the lost forever issue more adequately.

From the above three toy examples, we confirm that our proposed evaluation measures have the desired properties over previous ones: they can isolate T-MCT errors from the SCT errors and are robust against various tracking scenarios such as jump links and lost forever issue.

4. Framework for Re-id Based T-MCT

To prove the effectiveness of the proposed evaluation measures, we construct a T-MCT framework employing human Re-id algorithms. Motivated by applications in IoT, we incorporate the framework with distributed and online properties. To realize such properties, track association is done by Re-id algorithms within each camera, eliminating the need for powerful central server and high capacity communication. The property of online comes along with this kind of system setup, meaning that tracking results can be obtained within a short time window even if the identity has not completed its path in the camera network. Our framework is quite different from the previous work in T-MCT or MCT tasks, such as CamNeT [18], DukeMTMC [13], NLPR MCT [2], USC [8], and GMMCP [2], where they primarily focus on improving global optimization algorithm after all tracks are collected. The difficulty of achieving distributed and online T-MCT is much higher. The success in Re-id in recent years enables us to create such unique framework.

Figure 5 shows the Re-id based T-MCT scheme. Once an identity leaves the field of view of camera A, its feature vector extracted by Re-id algorithms is broadcast to adjacent cameras B and C, and stored within their own identity buffers (Figures 5(a)(b)). When a new identity is detected in one of camera B or C after a while, a traveling time model is applied to filter unlikely candidates in the identify buffer. The traveling time model is constructed by modeling traveling time between each camera pair or returning time of the same camera with a Gaussian distribution. When identifying different tracks, the candidates with traveling time outside the range of $[\mu - 2\sigma, \mu + 2\sigma]$ are filtered out, where

μ and σ are the mean and standard deviation of the Gaussian. The Re-id distance metrics is then computed between the feature extracted from the new identity and those valid identities (Figure 5(c)). Non-maximum suppression along with thresholding are used to determine whether the two feature vectors belong to the same identity. If they belong to the same identity, we recognize the leaving identity in camera A and the arriving identity in camera B/C (in this case camera C) as the same identity.

A variety of Re-id algorithms has been employed in our framework for comparison. For feature extraction, we used LOMO[11], BoW[20], HistLBP[17], and DGD-CNN[16] to encapsulate the information of a identity into a vector instead of large raw image data. In order to represent a track with a feature vector reliably, we perform average pooling or max pooling across N vectors belonging to N images sampled from the track evenly. Distance metrics including XQDA[11], KISSME[7], Mahalanobis distance, and Euclidean distance are adopted to estimate the affinity of two tracks. Note that all learning based algorithms are first trained on training set and then tested on validation or testing set. The training and testing setups will be described with more detail in next section. At last, we filter out identities with short tracking length, which might indicate false alarm tracks.

5. Experiments and Results

In this section, we present the experimental setup and provide discussions on the experimental results.

5.1. Experiment Setup

To evaluate our framework using the proposed evaluation measures, we use the multi-camera human tracking dataset (DukeMTMC [13]), which comprises videos from eight non-overlapping cameras recorded simultaneously, spanning 50 minutes for labeled sequence, "trainval." A total of 1,812 identities appear throughout the entire sequence. We conducted experiments on two single camera tracking outcomes, one was the GT tracks provided by the dataset, and the other was generated by DukeMTMC's baseline system [13, 14]. Furthermore, we divided the "trainval" set into two subsets: "train" and "validation & test," spanning 25 minutes for each. Among them, we used "train" as our training set. "Validation & test" was a bit more complicated: we used GT tracks in "validation & test" as our validation set and baseline system tracks of [13, 14] as the testing set.

The experiments are organized as follows. First, we compared the performance of each Re-id algorithm with the classic Re-id evaluation methods, namely cumulated matching characteristics (CMC) curve and mean average precision (mAP), on validation set. Next, the top ranked Re-id algorithms for each feature extractor were embedded into our T-MCT framework and tested on validation set. The result of this step aims to (1) validate our proposed evaluation measure (2) depict the effectiveness of Re-id algorithms when there are no distractors from SCT. Finally, the selected Re-id algorithms were applied to SCT tracker generated tracks, i.e. testing set. We compared the results with the proposed track-clustering based measure and the existing detection-based $IDmeasure$ recommended by DukeMTMC. We set $N = 15$ to pull out N images from a track for feature extraction, as mentioned in Section 4. As a result, $\{47430, 28740, 29865\}$ images are gathered for $\{$training, validation, testing$\}$ sets, respectively, which are more than enough to construct a large Re-id dataset, such as CUHK [9], Market-1501 [20], and MARS [19].

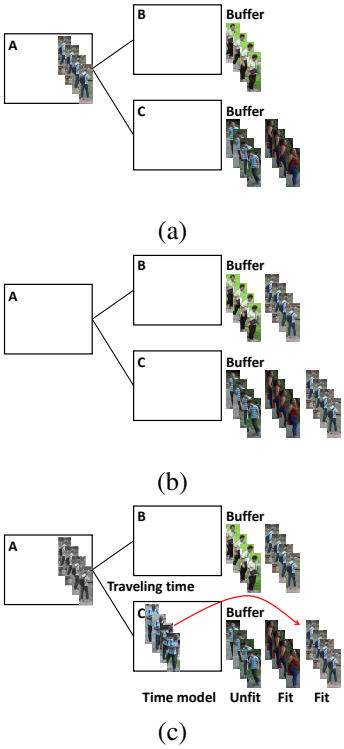


Figure 5: (a) A pedestrian object is leaving the field of the view of camera A. (b) The feature information of the object is transmitted to the neighboring cameras (B and C). (c) The re-identification can be done when the pedestrian object is entering the field of view of camera C.

5.2. Comparison Using Re-id Evaluation

We conducted comparisons of different human Re-id algorithms based on the toolkit provided by MARS [19]. Results are represented by CMC and mAP in Figure 6. We observe that LOMO and BoW feature extractors outperform others by achieving top score of 42.74% and 44.97% respectively in terms of mAP, and 45.77% and 46.92% in

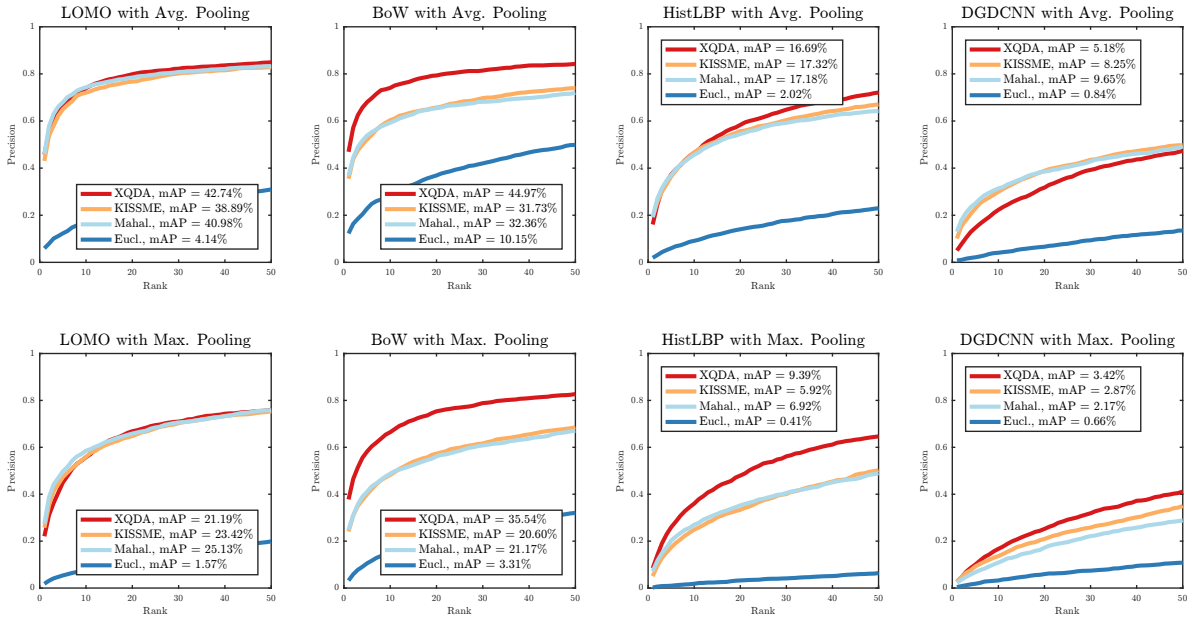


Figure 6: Results of different Re-id algorithms using typical Re-id evaluation measures: CMC and mAP. The results in the same column were obtained using the same Re-id feature extractor. The results in the same row were obtained using the same pooling techniques. The mAP score was also presented in the legend of each figure.

Table 1: The T-MCT results of different Re-id algorithms based on ground truth single camera tracking, along with the Re-id evaluation results in Section 5.2. All results were obtained using the average pooling to generate feature for tracks. The results are shown in percentage. The best result of each column is highlighted with boldface.

	Re-id Evaluation				Tracking Evaluation						
Tracking System (Re-id Algorithm)	**r=1**	**r=10**	**r=20**	**mAP**	**ClustP (T-MCT)**	**ClustR (T-MCT)**	**ClustF1 (T-MCT)**	**IDP (MCT)**	**IDR (MCT)**	**IDF1 (MCT)**	**IDF1 (ICT)**
LOMO-XQDA	45.77	73.96	**79.85**	42.74	58.76	**46.40**	**51.85**	**70.76**	**70.53**	**70.65**	**17.09**
BoW-XQDA	**46.92**	**74.06**	79.33	**44.97**	**62.65**	35.71	46.26	66.07	65.86	65.97	24.76
HistLBP-KISSME	19.26	45.67	54.49	17.32	36.97	10.94	16.88	54.14	53.97	54.05	33.33
DGDCNN-Mahal.	13.00	31.37	38.52	9.65	15.88	5.55	8.23	51.24	51.08	51.16	34.92

terms of rank-1 precision, while other features could not provide comparable results. However, there is no clear evidence of which distance learning metrics is the best. Only the naïve euclidean distance approach falls behind the others by a large margin. Another interesting fact is that average pooling for combining track features performs quite well with all Re-id algorithms, but max pooling varies quite a lot. To name a few, LOMO feature extractor seems to favor average pooling rather than max pooling. LOMO with XQDA obtains mAP of 42.74% when using average pooling, while LOMO with Mahalanobis distance obtains only 25.13% when using max pooling. Another fact should be noted is that deep learning feature extration method, i.e. DGDCNN, falls behind all the others by a large margin, may due to lack of fine-tuning model on this dataset. Although the author of DGDCNN claimed that deep learning model trained with various dataset is capable of extracting generalized feature for human, our results showed the best mAP for DGDCNN is still 35.32% behind the best performer. Fine-tuning DGDCNN for the dataset is left for future work. The best combination of each feature extraction method would proceed to the next stage of our experiment, i.e. track-based multi-camera tracking.

5.3. Comparison Using T-MCT Evaluation

Recall that in Section 1, we introduced that Re-id is essentially a soft identity assignment problem derived from T-MCT. As a consequence, we expect performance measure of T-MCT to correlate to Re-id evaluation measure, especially rank 1 accuracy and mAP. To verify how well our new evaluation measures corresponds with Re-id results, we tested our Re-id based T-MCT framework on ground truth tracks, namely the validation set. Table 1 shows the results of both Re-id and T-MCT evaluations. For T-MCT evalution, we choose to visualize track-clustering based ($ClustP$, $ClustR$, $ClustF1$) and detection-based evalu-

Table 2: The T-MCT results of different Re-id algorithms based on single camera tracking result of the baseline provided by DukeMTMC. All results were obtained using the average pooling to generate feature for tracks. The results are shown in percentage. The best result of each column is highlighted with boldface.

Tracking System (Re-id Algorithm)	**ClustP (T-MCT)**	**ClustR (T-MCT)**	**ClustF1 (T-MCT)**	**IDP (MCT)**	**IDR (MCT)**	**IDF1 (MCT)**	**IDF1 (ICT)**
DukeMTMC*	39.40	32.77	35.78	62.04	49.00	54.76	19.43
LOMO-XQDA	**40.76**	**37.97**	**39.32**	**64.17**	**49.29**	**55.76**	**13.34**
BoW-XQDA	40.66	31.03	35.20	59.87	45.99	52.02	17.82
HistLBP-KISSME	23.40	7.94	11.86	49.31	37.88	42.85	26.97
DGDCNN-Mahal.	14.70	5.24	7.73	48.09	36.94	41.78	27.88

$*$ The baseline MCT system provided by DukeMTMC does not utilize Re-id algorithm for T-MCT.

ation index ($IDP(MCT)$, $IDR(MCT)$, $IDF1(MCT)$, $IDF1(ICT)$), as mentioned in subsection 3.1. From the table, we first notice that both the proposed evaluation measures and the detection-based counterpart correlates to Re-id performance mostly. That is, T-MCT systems with more accurate Re-id algorithms perform better under T-MCT evaluation measure. However, we can easily figured out that only $ClustP$ has the same ranking order of rank 1 accuracy and mAP. In particular, all T-MCT evaluation measures grade BOW-XQDA lower than LOMO-XQDA except $ClustP$, where for the Re-id evaluation measure the order is vice versa. We conclude that our proposed evaluation measures indeed correlate to Re-id performance better than existing measurements. It is also noted that detection-based evaluation measure reports the gap between different Re-id algorithms smaller than the proposed evaluation measure. For instance, rank 1 accuracy difference between BoW-XQDA and DGDCNN-Mahal. is 33.92%, roughly same as $ClustF1$ difference 38.03%, but much larger than $IDF1(MCT)$ difference 11.92% and $IDF1(ICT)$ difference 8.57%. This can be owing to that detection-based evaluation measure also accounts SCT error for total error. In short, we have proven that the track-clustering based evaluation measures are more representative than the detection-based counterpart for the Re-id based T-MCT system.

While the previous experiments have verified the effectiveness of the proposed evaluation measures on ground truth SCT tracks, the real challenges lie within using realistic SCT tracks from other trackers. At this stage, we aim to study how our T-MCT framework performs on wild data. Results of systems using baseline tracks released by DukeMTMC [13, 14], namely the testing set, are shown in Table 2. The whole MCT system of DukeMTMC [13, 14], which is not a Re-id based tracking system, is also compared in the table. Results highlight that our T-MCT framework using LOMO-XQDA Re-id method achieved comparable results in all of the evaluation measures, even slightly better than DukeMTMC [13, 14], where a global optimization algorithm is employed. However, it is noted that DukeMTMC utilizes simple color feature and histogram, whereas LOMO extracts complex appearance information. BoW-XQDA performs slightly worse than LOMO-XQDA employed to T-MCT, although it reaches a higher score in Re-id. From the result, we gained some insight on how Re-id algorithms empower T-MCT in a realistic scenario.

6. Conclusion

In this paper, we proposed a set of novel track-clustering based evaluation measures for the T-MCT task. The proposed evaluation measures were presented to test the T-MCT performances by validating correspondences between tracks. Performance comparisons with detection-based evaluation measures on three carefully-planned experiments proved that the proposed evaluation measures can present the performance of T-MCT much more faithfully. We also showed that the proposed evaluation measures are robust to be used in different scenarios where track-order based evaluation measures failed. Furthermore, we designed an online and distributed T-MCT framework. Experimental results not only proved the correctness of our evaluation measures, but also showed that our Re-id based T-MCT framework can achieve similar performance to the global optimized MCT system. The best performer of our T-MCT framework employs LOMO as the feature extractor and XQDA as the metrics learning algorithm, achieving 55.76% in $IDF1(MCT)$, 13.34% in $IDF1(ICT)$ and 39.32% in $ClustF1$. Some Re-id algorithms are also compared under Re-id and T-MCT evaluation measures, which can be useful references for future works. Our proposed T-MCT framework is also a good platform for designing Re-id T-MCT scheme, with the properties suitable for IoT applications.

Acknowledgements

This research was supported in part by the Ministry of Science and Technology of Taiwan (MOST 106-2633-E-002-001), National Taiwan University (NTU-106R104045), and Intel Corporation.

References

[1] Y. Cai and G. Medioni. Exploring context information for inter-camera multiple target tracking. In *IEEE Winter Conference on Applications of Computer Vision*, pages 761–768. IEEE, 2014.

[2] L. Cao, W. Chen, X. Chen, S. Zheng, and K. Huang. An equalised global graphical model-based approach for multi-camera object tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 2014.

[3] N. Gheissari, T. B. Sebastian, and R. Hartley. Person reidentification using spatiotemporal appearance. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 1528–1535. IEEE, 2006.

[4] D. Gray and H. Tao. Viewpoint invariant pedestrian recognition with an ensemble of localized features. In *European conference on computer vision*, pages 262–275. Springer, 2008.

[5] M. Hirzer, C. Beleznai, P. M. Roth, and H. Bischof. Person re-identification by descriptive and discriminative classification. In *Proc. Scandinavian Conference on Image Analysis (SCIA)*. Springer, 2011.

[6] R. Kasturi, D. Goldgof, P. Soundararajan, V. Manohar, J. Garofolo, R. Bowers, M. Boonstra, V. Korzhova, and J. Zhang. Framework for performance evaluation of face, text, and vehicle detection and tracking in video: Data, metrics, and protocol. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(2):319–336, 2009.

[7] M. Koestinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof. Large scale metric learning from equivalence constraints. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2288–2295. IEEE, 2012.

[8] C.-H. Kuo, C. Huang, and R. Nevatia. Inter-camera association of multi-target tracks by on-line learned appearance affinity models. In *European Conference on Computer Vision*, pages 383–396. Springer, 2010.

[9] W. Li, R. Zhao, T. Xiao, and X. Wang. Deepreid: Deep filter pairing neural network for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 152–159, 2014.

[10] Y. Li, C. Huang, and R. Nevatia. Learning to associate: Hybridboosted multi-target tracker for crowded scene. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2953–2960. IEEE, 2009.

[11] S. Liao, Y. Hu, X. Zhu, and S. Z. Li. Person re-identification by local maximal occurrence representation and metric learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2197–2206, 2015.

[12] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, and K. Schindler. Mot16: A benchmark for multi-object tracking. *arXiv preprint arXiv:1603.00831*, 2016.

[13] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *European Conference on Computer Vision*, pages 17–35. Springer, 2016.

[14] E. Ristani and C. Tomasi. Tracking multiple people online and in real time. In *Asian Conference on Computer Vision*, pages 444–459. Springer, 2014.

[15] T. Wang, S. Gong, X. Zhu, and S. Wang. Person re-identification by video ranking. In *European Conference on Computer Vision*, pages 688–703. Springer, 2014.

[16] T. Xiao, H. Li, W. Ouyang, and X. Wang. Learning deep feature representations with domain guided dropout for person re-identification. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[17] F. Xiong, M. Gou, O. Camps, and M. Sznaier. Person re-identification using kernel-based metric learning methods. In *European conference on computer vision*, pages 1–16. Springer, 2014.

[18] S. Zhang, E. Staudt, T. Faltemier, and A. K. Roy-Chowdhury. A camera network tracking (camnet) dataset and performance baseline. In *2015 IEEE Winter Conference on Applications of Computer Vision*, pages 365–372. IEEE, 2015.

[19] L. Zheng, Z. Bie, Y. Sun, J. Wang, C. Su, S. Wang, and Q. Tian. Mars: A video benchmark for large-scale person re-identification. In *European Conference on Computer Vision*, pages 868–884. Springer, 2016.

[20] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian. Scalable person re-identification: A benchmark. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1116–1124, 2015.

[21] L. Zheng, Y. Yang, and A. G. Hauptmann. Person re-identification: Past, present and future. *arXiv preprint arXiv:1610.02984*, 2016.