Simultaneous Clustering and Tracklet Linking for Multi-Face Tracking in Videos

Baoyuan Wu
National Laboratory of Pattern Recognition, CASIA
Beijing 100190, China
wubaoyuan1987@gmail.com

Siwei Lyu
Computer Science, Univ. at Albany, SUNY
Albany, NY 12222, USA
lsw@cs.albany.edu

Bao-Gang Hu
National Laboratory of Pattern Recognition, CASIA
Beijing 100190, China
hubg@nlpr.ia.ac.cn

Qiang Ji
ECSE, Rensselaer Polytechnic Institute
Troy, NY 12180, USA
qji@ecse.rpi.edu

Abstract

We describe a novel method that simultaneously *clusters and associates short sequences of detected faces (termed as* face tracklets*) in videos. The rationale of our method is that face tracklet clustering and linking are related problems that can benefit from the solutions of each other. Our method is based on a hidden Markov random field model that represents the joint dependencies of cluster labels and tracklet linking associations . We provide an efficient algorithm based on constrained clustering and optimal matching for the simultaneous inference of cluster labels and tracklet associations. We demonstrate significant improvements on the state-of-the-art results in face tracking and clustering performances on several video datasets.*

1. Introduction

Reliably tracking and clustering of faces in unconstrained videos is a challenging problem, which is complicated by drastic changes in backgrounds, illuminations, view points, camera movements and occlusions that frequently occur in actual videos. Most previous works treat these two problems separately, as either a *face tracking* problem where sequences of face images are associated [14, 18] or a *face clustering* problem where face images are partitioned into different clusters [3, 22, 23].

We address in this paper the problem of *simultaneously* clustering and linking of short sequences of detected faces (termed as *face tracklets*) in videos. We argue that it is advantageous to solve the two problems simultaneously – they provide useful information and constraints to each other, thus can bootstrap and improve the performances of each other. Figure 1 exemplifies the benefits of simultaneous face clustering and tracklet linking: in the first case (top row), linking tracklets without considering their cluster labels leads to incorrect association tracklets from cluster 1 and 2 together. In the second case (bottom row), incorrect clustering of faces (in this case, separation of faces of the same person into two clusters 1 and 2) can be avoided with the knowledge that there is a high likelihood that they are in the same long track.

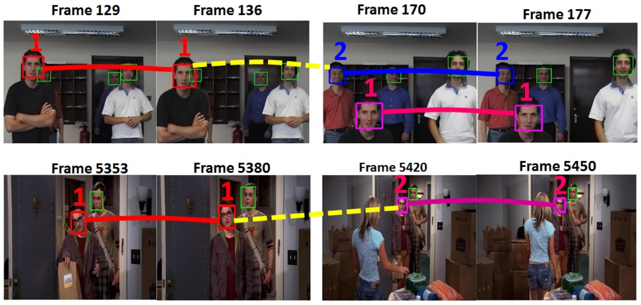

Figure 1. *The benefit of simultaneous face clustering and tracklet linking. Detected face tracklets are indicated by bounding boxes connected with solid lines (we only highlight a few detected tracklets for the sake of presentation). Linkings and cluster labels of tracklets are indicated by the dashed curves and numbers over the bounding boxes, respectively.* **(Top)** *Without considering clustering labels, tracklets of different clusters are linked incorrectly.* **(Bottom)** *Without considering tracklet linking, tracklets in the same track are incorrectly partitioned into different clusters.*

The basis of our method is a novel hidden Markov random field (HMRF) [1] model [8] that jointly models the face cluster labels and face tracklet associations. We formulate the problem of simultaneously clustering and linking of tracklets as a Bayesian inference problem based on this model, and provide an efficient coordinate-descent solution. Specifically, from detected face tracklets with similar

[1] As the constraints used in our problem are un-directed correlations, HMRF is a better choice than the directed models such as HMM or DBN.

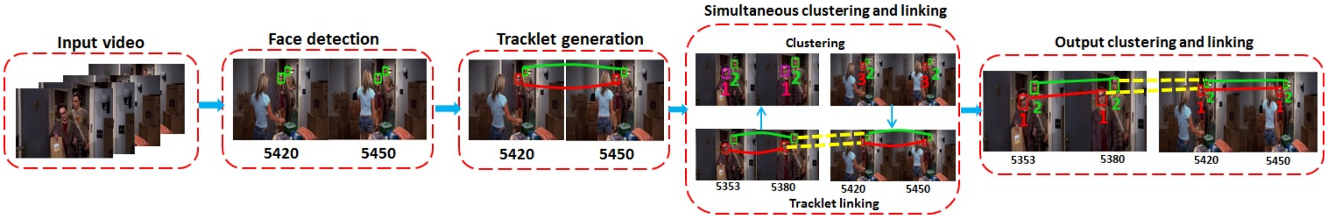

Figure 2. *Overall workflow of our method for simultaneous face clustering and tracklet linking. With an input video, we first detect all faces in each frame, and then form face tracklets from adjacent frames. The face tracklets are iteratively clustered and linked into longer tracks in a bootstrapping manner, with the final output of the algorithm being the complete long tracks of detected faces with cluster labels.*

appearances and adjacent spatial locations in consecutive frames, our method iterates between two steps:

- finding long tracks of faces that are consistent in motion and appearance using the intermediate tracklet clustering as an important cue,
- recovering face cluster labels using constrained clustering with constraints given by the intermediate face tracklet linking step.

This procedure is illustrated in Figure 2. The contributions of this work are thus three-fold:

- We present a systematic approach of jointly solving two related tasks, i.e., face clustering and face tracklet linking, taking advantage of the additional information from each task to boost their overall performances.
- We formulate the problem as a unified hidden Markov random field model and provide an efficient algorithm.
- Our face tracking and clustering performances improves on the state-of-the-art results for three data sets.

The rest of the paper is organized as follows. After reviewing relevant previous works in Section 2, we describe the HMRF model and its optimization in Section 3, and provide further implementation details in Section 4. Experimental evaluations and comparisons of our method are reported in Section 5 and Section 6 concludes the paper with discussion and future works.

2. Related Works

Face clustering in videos has been the topic of many previous works, e.g., [1, 4, 5, 6, 19]. These methods commonly treat faces from different video frames as a set of still images, and apply conventional clustering algorithms based on similarities in appearance and poses. A few recent works also study the use of pairwise constraints that are distinct for detected faces in videos [3, 22, 23]: *faces in the same tracklet should be must-linked, while faces from a pair of overlapped tracklets should be cannot-linked*. However, these methods still suffer from the frequent changes in pose and illumination, and the intrinsic low resolution of video frames. As shown in the bottom row of Figure 1, because of the pose change and low resolution, two tracklets of the same person are easily to be grouped into different clusters. If given a pair of non-overlapped tracklets belongs to the same track, many more pairwise constraints can be obtained, which are expected to further enhance the performance of constrained face clustering.

Recently, many works in video tracking explore the methodology based on short consecutive tracked locations (known as tracklets) that can be reliably obtained (e.g., [10, 15, 24, 17]). These methods then proceed by linking the tracklets into longer tracks, which is achieved by first constructing pairwise similarities between tracklets (based on appearance as in [15, 10], motion smoothness [24] or entry/exit maps [24]) followed by optimally matching (using the Hungarian algorithm [9] for optimal matching of bipartite graph [7, 17] or as a Bayesian inference [15]). Compared to traditional monolithic tracking solutions, the tracklet based methods are more robust and suitable for tracking multiple objects in heavily occluded scenes. Furthermore, the tracklet-based methodology has also been employed for face tracking in [14, 18]. Though bootstrapping clustering and tracklet linking has been discussed in the context of simplified context with fixed camera (e.g., [20, 17]), to our best knowledge, no previous work has explore this approach in a more challenging scenario with many camera motions and occlusions for the clustering and tracking of faces.

3. Method

3.1. Problem Formulation

We assume that a long video has been processed to obtain a set of face tracklets $U = (\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_n)$ (details given in Section 5). Each tracklet $\mathbf{u}_i$ is represented as a list of triples collected from a sequence of n_i continuous video frames, as $\mathbf{u}_i = \{t_j^{(i)}, \mathbf{x}_j^{(i)}, \mathbf{l}_j^{(i)}\}_{j=1}^{n_i}$, where $t_j^{(i)}$ is the frame index, $\mathbf{x}_j^{(i)}$ corresponds to appearance features, and $\mathbf{l}_j^{(i)}$ represents the location and scale of the bounding box of each detected face, respectively. We use $\mathbf{t}^{(i)}$, $X^{(i)}$ and $L^{(i)}$ to represent the ensemble of $t_j^{(i)}$, $\mathbf{x}_j^{(i)}$ and $\mathbf{l}_j^{(i)}$ of tracklet $\mathbf{u}_i$, respectively. We also compute similarities between every pair of tracklets (details to be described in Section 4.2), and save them in an $n \times n$ matrix M.

Our goal is to *simultaneously* link the tracklet into longer

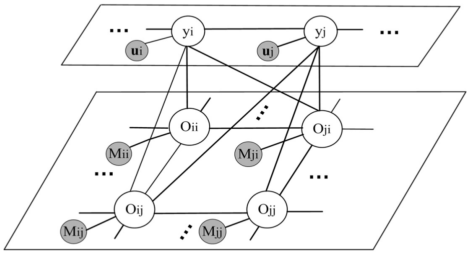

Figure 3. *The graphical model of the hidden Markov random field (HMRF) model. The open and closed circles represent latent and observable variables, respectively. The top layer represents the cluster label variables, while the bottom layer denotes the linking variables. The lines between nodes represent their relations. Note that the linking variables in the same row/column are fully connected, and the links between two non-adjacent cluster label variables are omitted for clarity.*

Symbols	Descriptions
$U = (\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_n)$	a set of face tracklets
$\mathbf{u}_i = \{t_j^{(i)}, \mathbf{x}_j^{(i)}, \mathbf{l}_j^{(i)}\}_{j=1}^{n_i}$	one tracklet of n_i faces
$\mathbf{t}^{(i)} = (t_1^{(i)}, t_2^{(i)}, \cdots, t_{n_i}^{(i)})'$	frame indexes of the bounding boxes
$X^{(i)} = (\mathbf{x}_1^{(i)}, \mathbf{x}_2^{(i)}, \cdots, \mathbf{x}_{n_i}^{(i)})$	appearance of the bounding boxes
$L^{(i)} = (\mathbf{l}_1^{(i)}, \mathbf{l}_2^{(i)}, \cdots, \mathbf{l}_{n_i}^{(i)})$	locations and scales of the bounding boxes
$\mathbf{y} = (y_1, y_2, \cdots, y_n)$	the cluster labels of tracklets
$y_i \in \{1, 2, \ldots, K\}$	K is the number of clusters
$V \in \{-1, 0, 1\}^{n \times n}$	the relations between $\mathbf{y}$
$O \in \{0, 1\}^{n \times n}$	the linking matrix
$M = \{M_{ij}\} \in \mathrm{R}^{n \times n}$	the observation matrix for O
$M_{ij} = f(\mathbf{u}_i, \mathbf{u}_j)$	the similarity between $\mathbf{u}_i$ and $\mathbf{u}_j$

Table 1. *Main notations used in this paper.*

tracks and partition the face images into distinct clusters, based on cues from face appearances and motion trajectories. For simplicity, we assume in this work that the total cluster number of K is known a priori[2]. We denote the cluster labels of the tracklets as a vector $\mathbf{y} = (y_1, y_2, \cdots, y_n)$ with each $y_i \in \{1, 2, \ldots, K\}$. The linking relations of tracklets are represented with a matrix $O \in \{0,1\}^{n \times n}$ where $O_{ij} = 1$ if and only if tracklets $\mathbf{u}_i$ and $\mathbf{u}_j$ are adjacent in a track with $\mathbf{u}_i$ precedes $\mathbf{u}_j$, and $O_{ii} = 1$ if and only if tracklet $\mathbf{u}_i$ is the last tracklet in a long track. Table 1 summarizes all major notations used in this work.

We model the probabilistic inter-dependencies between U, M, $\mathbf{y}$ and O with a hidden Markov random field (HMRF) [8], in which $\mathbf{y}$ and O are latent variables and U and M are observable variables. A brief illustration of the HMRF is shown in Figure 3. Their joint probability distribution, along with model parameters $\theta = (\{\mu_i\}_{i=1}^K, \{\Sigma_i\}_{i=1}^K, \beta)$ (to be specified in the following), is defined as:

$$P(U, M, \mathbf{y}, O; \theta) = P(U|\mathbf{y}; \theta) P(M|O) P(\mathbf{y}, O), \quad (1)$$

where we assume conditional independence between U and M given the latent variables. Specifically, we model the likelihood of the appearances of the face images in the tracklets given their cluster labels with a simple Gaussian model,

$$P(U|\mathbf{y}; \theta) = \prod_{i=1}^{n} \prod_{j=1}^{n_i} \mathcal{N}(\mathbf{x}_j^{(i)} | \mu_{y_i}, \Sigma_{y_i}), \quad (2)$$

where parameters μ_{y_i} and Σ_{y_i} correspond to the cluster-specific means and covariance matrices, respectively and are estimated during the inference of $\mathbf{y}$ (described in Section 4.1). The other likelihood term, $P(M|O)$, which captures the relation beween tracklet similarities and their linking relations in long tracks, is defined as:

$$P(M|O) = \frac{1}{Z_1} \prod_{i=1}^{n} \prod_{j=1}^{n} \exp(\lambda_1 O_{ij} M_{ij}), \quad (3)$$

with $Z_1 = \sum_M \prod_{i=1}^n \prod_{j=1}^n \exp(\lambda_1 O_{ij} M_{ij})$ being the partition function. The tuning of parameter λ_1 will be presented in Section 4.2. Term $P(\mathbf{y}, O)$ represents the dependencies between $\mathbf{y}$ and O, and is defined as:

$$P(\mathbf{y}, O; \theta) = \frac{1}{Z_2} \Bigg(\prod_{i=1}^{n} \psi_1(y_i, y_j) \prod_{i=1}^{n} \prod_{j \neq i} \psi_2(y_i, y_j, O_{ij}) \prod_{i=1}^{n} \psi_3(\sum_{j=1}^{n} O_{ij}, \sum_{j \neq i} O_{ji}) \Bigg)^{\beta}, \quad (4)$$

with Z_2 being the partition function. The model parameter β controls the weight between $\log P(\mathbf{y}, O; \theta)$ and $\log(P(U|\mathbf{y}; \theta) P(M|O))$. It will be learned automatically during the optimization (described in Section 3.2.1). The first potential function ψ_1 is defined as

$$\psi_1(y_i, y_j) = e^{\left[\sum_{V_{ij} \geq 0} V_{ij}(I(y_i = y_j) - 1) + \sum_{V_{ij} < 0} V_{ij} I(y_i = y_j)\right]}, \quad (5)$$

where I is the indicator function, whose value is 1 if the argument is true, and 0 otherwise. The pre-computed matrix $V = \{V_{ij} | i, j = 1, \ldots, n\} \in \mathbf{R}^{n \times n}$ embeds the relations among the cluster labels $\mathbf{y}$: $V_{ij} > 0$ indicates positive correlation between y_i and y_j, i.e., they are more likely to take the same value; $V_{ij} < 0$ indicates negative correlation between y_i and y_j, i.e., they are more likely to take different values; $V_{ij} = 0$ represents the cluster labels being independent. More details about the computation of V will be described in Section 4.1. Potential function ψ_2 is defined as

$$\psi_2(y_i, y_j, O_{ij}) = \exp\big(\lambda_2 O_{ij} (I(y_i = y_j) - 1)\big), \quad (6)$$

and corresponds to the following constraint between $\mathbf{y}$ and O: *if $y_i \neq y_j$, then the configuration $O_{ij} = 1$ will be discouraged, and vice versa*. The tuning of λ_2 will be dis-

[2] For example, for videos from TV episodes, K can be determined using the number of major cast.

Algorithm 1 *Overall algorithm for simultaneous face clustering and tracklet linking.*

```
Input: tracklets $U$, their similarity $M$, number of clusters $K$
Output: cluster labels $\mathbf{y}$ and tracklet linking relation $O$
  Initialize $O$ based on $M$, using Hungarian algorithm;
  while not converge do
     optimizing $\mathbf{y}$ and $\theta$ with fixed $O$ (Section 3.2.1);
     optimizing $O$ with fixed $\mathbf{y}$ (Section 3.2.2);
  end while
  return  $\mathbf{y}^\star$ and $O^\star$
```

cussed in Section 4.1. Potential function ψ_3 models the dependencies among components of O and is defined as

$$\psi_3(\sum_{j=1}^{n} O_{ij}, \sum_{j\neq i}^{n} O_{ji}) = I(\sum_{j=1}^{n} O_{ij} = 1) I(\sum_{j\neq i}^{n} O_{ji} < 1). \tag{7}$$

3.2. Optimization

With the HMRF formulation, our task of simultaneous clustering and tracklet linking of face images can be formulated as a Bayesian inference problem, where we seek for

$$\max_{\mathbf{y},O,\theta} \quad \log P(\mathbf{y}, O|U, M; \theta) \equiv \tag{8}$$
$$\max_{\mathbf{y},O,\theta} \quad \log P(U|\mathbf{y}; \theta) + \log P(M|O) + \log P(\mathbf{y}, O; \theta),$$

where θ corresponds to the model parameters $(\mu_{y_i}, \Sigma_{y_i}, \beta)$. In this work, we solve this optimization problem with coordinate-descent, and the overall algorithm is summarized in Algorithm 1.

3.2.1 Optimizing $\mathbf{y}$ and θ with Fixed O

The objective function of this step is

$$(\mathbf{y}^\star, \theta^\star) = \underset{\mathbf{y},\theta}{\operatorname{argmax}} \left(\log P(U|\mathbf{y}; \theta) + \log P(\mathbf{y}, O; \theta)\right),$$

where the objective function can be further simplified by dropping constant terms to obtain

$$\sum_{i=1}^{n}\sum_{j=1}^{n_i} \log \mathcal{N}(\mathbf{x}_j^{(i)}|\mu_{y_i}, \Sigma_{y_i}) + \beta \sum_{i=1}^{n} \Bigg(\sum_{V_{ij}<0} V_{ij} I(y_i = y_j) +$$
$$\sum_{V_{ij}>0} V_{ij}[I(y_i = y_j) - 1] + \lambda_2 \sum_{j\neq i} O_{ij}[I(y_i = y_j) - 1] \Bigg). \tag{9}$$

With O fixed, this reduces to a constrained clustering problem, where the first term encourages tightness within each cluster, and the remaining terms specify constraints over $\mathbf{y}$ that are provided in V and O. Specifically, the second term incorporates the cannot-link constraint: given $V_{ij} < 0$, the result $y_i = y_j$ will be penalized, because the objective function (9) will decrease by $-\beta|V_{ij}|$ in this case. The third term embeds the must-link constraint: given $V_{ij} > 0$, the result $y_i \neq y_j$ will be penalized, because Equation (9) will also decrease by $-\beta|V_{ij}|$. Similarly, the last term also serves as the must-link constraint: given $O_{ij} = 1$, the result $y_i \neq y_j$ will be penalized, because (9) will decrease by $-\beta\lambda_2|O_{ij}|$. Note that these constraints are *softly* incorporated in our model, since the penalty degrees of violating these constraints are finite values, i.e., $\beta|V_{ij}|$ or $\beta\lambda_2|O_{ij}|$[3]. This soft manner can be understood as: it is possible that some cluster labels will violate the given constraints in V and O, and it allows the possible errors in some constraints, which is often the case in practice.

This constrained clustering problem can be directly solved by the simulated filed algorithm [2, 23], in which the inference of $\mathbf{y}$ and the learning of parameters $\theta = (\mu, \Sigma, \beta)$ are performed alternatively. More details on obtaining V and the tuning of λ_2 will be described in Section 4.1.

3.2.2 Optimizing O with Fixed $\mathbf{y}$

This step is achieved with the following objective function:

$$O^\star = \underset{O\in\{0,1\}^{n\times n}}{\operatorname{argmax}} \log P(M|O) + \log P(\mathbf{y}, O; \beta),$$

which is further simplified by dropping constant terms to yield

$$\sum_{i=1}^{n}\sum_{j=1}^{n} O_{ij}[\lambda_1 M_{ij} + \beta\lambda_2(I(y_i = y_j) - 1)]$$
$$+ \beta \sum_{i=1}^{n} \log[I(\sum_{j=1}^{n} O_{ij} = 1) I(\sum_{j\neq i}^{n} O_{ji} \leq 1)] \tag{10}$$

Equation (10) can be considered as a matching problem of a weighted bipartite graph, which can be solved by the Hungarian algorithm [9]. Specifically, $[\lambda_1 M_{ij} + \beta\lambda_2(I(y_i = y_j) - 1)]$ denotes the edge weight, and if this edge is selected, then $O_{ij} = 1$. Obviously in the setting of bipartite matching, the constraints embedded in $I(\sum_{j=1}^{n} O_{ij} = 1)$ and $I(\sum_{j\neq i}^{n} O_{ji} \leq 1)$ are satisfied automatically.

Note that in linking the tracklets, we also incorporate constraints from the clustering results with the term $\beta\lambda_2(I(y_i = y_j) - 1)$ in Equation (10): if $y_i \neq y_j$, then the similarity between $\mathbf{u}_i$ and $\mathbf{u}_j$ will be reduced by $\beta\lambda_2$; consequently, the probability of $O_{ij} = 1$ will decrease. Details of this algorithm will be described in Section 4.2.

4. Implementation

4.1. Constrained Clustering

Here we describe more details about the optimization of Equation (9). We augment the HMRF-based constrained clustering proposed in [23] by incorporating additional constraints obtained from tracklet linking results. The pairwise constraints used in the original HMRF-based clustering include two types: (1) faces in one tracklet should belong to the same cluster; (2) faces from two overlapped tracklets (some faces of them co-exist in the same frame) should belong to different clusters. As mentioned in Section 3.2.1,

[3] Note that if we set $|V_{ij}| = \infty$ and $\lambda_2 = \infty$, then they become *hard* constraints.

the constraints are softly embedded when V_{ij} is set as a finite value. Without loss of generality, we set $|V_{ij}| = 1$ when a constraint exists. Specifically, $V_{ij} = 1$ means a soft must-link constraint; $V_{ij} = -1$ represents a soft cannot-link constraint; $V_{ij} = 0$ indicates no constraint.

The additional constraints are derived from tracklet linking: if two tracklets $\mathbf{u}_i$ and $\mathbf{u}_j$ are linked after tracklet linking, faces from them should grouped into the same cluster. Besides, considering the transitivity of must-link constraints, additional cannot-link constraints can also be obtained. For example, given $\mathbf{u}_i$ and $\mathbf{u}_j$ are linked, if $\mathbf{u}_j$ is overlapped with another tracklet $\mathbf{u}_k$, then faces from $\mathbf{u}_i$ and $\mathbf{u}_k$ should also be cannot-linked.

Such obtained constraints V and O are usually very sparse with many zero entries. Following [23], the initial constraints are propagated based on the constraint-level smoothness [12, 13]: *given a must-link (cannot-link) between two faces* $\mathbf{x}_1$ *and* $\mathbf{x}_2$*, if* $\mathbf{x}_3$ *is close to* $\mathbf{x}_2$ *in appearance, then it is assumed that there is also a must-link (cannot-link) between* $\mathbf{x}_1$ *and* $\mathbf{x}_3$. The propagated constraints are formulated as follows [13]:

$$\overline{V} = (1-\alpha)^2(I-\alpha L)^{-1}V(I-\alpha L)^{-1}, \quad (11)$$

$$\overline{O} = (1-\alpha)^2(I-\alpha L)^{-1}O(I-\alpha L)^{-1}, \quad (12)$$

$$L = D^{-\frac{1}{2}}AD^{-\frac{1}{2}}, A_{ij} = \exp^{-d_a^2(\mathbf{x}_i,\mathbf{x}_j)/\sigma_i\sigma_j}, \quad (13)$$

where A is the affinity matrix, L is the normalized affinity matrix, and $\alpha \in (0,1)$ is a pre-determined. The degree matrix D is diagonal, and $D_{ii} = \sum_{j=1}^{n} A_{ij}$. The local kernel sizes σ_i and σ_j are determined following the strategy described in [16]. In each iteration, we replace V and O by the propagated matrices $\overline{V}$ and $\overline{O}$, which are then adopted in the simulated filed algorithm (see Section 3.2.1).

To reduce the computation of clustering, we adopt the approximation framework in [23], which is based on the observation that faces in adjacent frames of the same tracklet are very similar. Its main process is summarized as follows: (1) randomly sampling a fixed number of faces from each tracklet to obtain a subset of n_s faces; (2) running constrained clustering on this subset; (3) determining the labels of all faces based on the labels of faces in the subset. As a simple example, if 5 faces are sampled from one tracklet, and their labels are $(3,2,3,3,1)$ after clustering, then the label of this tracklet is determined as the mode value 3. All faces in this tracklet are also relabeled as 3. The cluster labels of tracklets are called as *tracklet-level* clustering, while labels of all faces are referred to as *face-level* clustering. Since the number of frames in each tracklet is not equal, the clustering accuracies of these two levels may be different.

Last, as β is learned during the optimization, the only parameter we need to adjust is the ratio λ_2, which controls the relative weight between the constraints in V and O. Since the constraints in V are always correct, while the constraints in O that from tracklet linking may have some errors, we set $\lambda_2 < 1$. Besides, the tracklet linking results are believed to become more accurate as the iteration proceeds, leading to more reliable constraints in O. So we gradually increase λ_2 during the whole optimization process.

4.2. Tracklet Linking

A key component for tracklet linking is the tracklet similarity represented in matrix M. As reported in some previous works [24, 17], the tracklet similarity takes account of three aspects, including the temporal adjacency, appearance affinity and motion smoothness. The overall similarity measure is given as follows:

$$M_{ij} = \begin{cases} e^{-d_t(\mathbf{t}^{(i)},\mathbf{t}^{(j)})-\eta_1 d_a(\mathbf{x}^{(i)},\mathbf{x}^{(j)})-\eta_2 d_m(\mathbf{l}^{(i)},\mathbf{l}^{(j)})} & i \neq j \\ c & i = j \end{cases} \quad (14)$$

with constants η_1 and η_2 in Equation(14) are two trade-off parameters, which will be determined by cross-validation.

Specifically, distance d_t enforces the temporal constraint: *if* $\mathbf{u}_j$ *occurs before* $\mathbf{u}_i$ *or they are overlapped in some frames, then* $O_{ij} = 0$. Here $t^{(i)} = (t_{i1}, \ldots, t_{in_i})'$ is a column vector containing the frame indices of all faces in tracklet $\mathbf{u}_i$. Similar to the work in [17], we define d_t as:

$$d_t(\mathbf{t}^{(i)},\mathbf{t}^{(j)}) = \begin{cases} 0, & \text{if } 0 < \triangle t_{ij} < t_0, \\ \infty, & \text{otherwise,} \end{cases}$$

where $\triangle t_{ij} = \mathbf{t}_1^{(j)} - \mathbf{t}_{n_i}^{(i)}$ indicates the temporal difference between $\mathbf{u}_i$ and $\mathbf{u}_j$ and t_0 is a pre-defined threshold to avoid linking two tracklets with a large frame gap. Distance d_a measures appearance affinity. Following [17], the appearance of each detected face is represented as a 24-dimensional vector by concatenating 8 bins of each HSV channel. A tracklet is further represented by the average vector of the included faces. Then the Bhattacharyya distance is computed as $d_a(\mathbf{x}^{(i)},\mathbf{x}^{(j)})$. Last, distance d_m reflects motion smoothness. In particular, denote $\mathbf{l}_j^{(i)} \in \mathrm{R}^{4\times 1}$ as the location and scale of the j^{th} bounding box in tracklet $\mathbf{u}_i$, represented by the horizontal and vertical coordinate of the central pixel and the width and height of the box. Treating each face $\mathbf{l}_j^{(i)}$ as a point, then one tracklet can be seen as a sequence of discrete points in a 4-dimensional space. We fit this sequence through the polynomial curve fitting and the fitted curve $\mathbf{s}^{(i)}(\cdot)$ can be used to predict the bounding boxes of the other tracklet $\mathbf{u}_j$. The difference between the predicted bounding box and the true box is utilized to define d_m, as follows:

$$\begin{aligned} d_m(\mathbf{l}^{(i)},\mathbf{l}^{(j)}) &= \sum_{r\in\{1,2,3\}} ||\mathbf{s}^{(i)}(\mathbf{t}_r^{(j)}) - \mathbf{l}_r^{(j)}|| \quad (15) \\ &+ \sum_{r\in\{n_i-2,n_i-1,n_i\}} ||\mathbf{s}^{(j)}(\mathbf{t}_r^{(i)}) - \mathbf{l}_r^{(i)}||, \end{aligned}$$

where $\mathbf{s}^{(i)}(\mathbf{t}_r^{(j)})$ denotes the predicted bounding box at frame $\mathbf{t}_r^{(j)}$ on the curve $\mathbf{s}^{(i)}$ (see Figure 4).

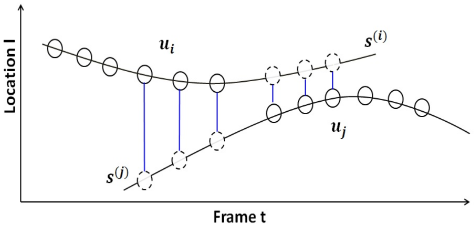

Figure 4. *Definition of d_m. The two central curves correspond to $\mathbf{s}^{(i)}$ and $\mathbf{s}^{(j)}$. The solid circles correspond to detected faces in one tracklet, while the dashed circles are those predicted by the fitted trajectory. We match detected faces that are highlighted in blue color. For simplicity, here we only show one dimension of $\mathbf{l}_j^{(i)}$, and the computations for the other three dimensions are similar.*

The diagonal value M_{ii} is set as a constant value to indicate the end of a track, as shown in Equation(14). The value c will influence the final number of tracks and the lengths of tracks: if c is large, then many short tracks will be obtained; otherwise fewer but longer tracks will be presented. In practice, c can be adjusted according to the user's demand. In our experiments, c is determined as the 3 times of the mode value among the off-diagonal values in M, excluding the infinite values. The ratio $\frac{\beta\lambda_2}{\lambda_1}$ is adjusted to control the relative weight between the constraints and tracklet similarities. In our experiments, we initially set $\frac{\beta\lambda_2}{\lambda_1} = 0.1c$. Since the clustering results are expected to become more accurate as the iteration proceeds, we gradually increase $\frac{\beta\lambda_2}{\lambda_1}$.

5. Experiments

Our experiments are based on three publicly available videos: Frontal and Turning data sets in [14], and BBT01 data set in [18]. These datasets correspond to different challenges for face tracking and clustering. The first two short videos are recorded with a fixed camera view point. In Frontal, there are frequent occlusions and fast movements that make tracklet linking difficult. The Turning video has frequent occlusions and many profile faces that make clustering challenging. The BBT01 video is a whole episode from the TV sitcom Big Bang Theory (Ep. 01-01). This video is significantly longer (23 mins) with many small faces due to the full scene shots. Another difficulty represented in this video is the frequent changes of camera view and scene. Some statistics of datasets are shown in Table 2.

5.1. Tracklet Generation

With a given input video, we extract reliable face tracklets by first applying the Viola-Jones face detector [21] in each frame. The detected faces in adjacent frames are then linked, based on similarities in their appearances, locations and scales of the bounding boxes. To further avoid identity switches, we set a threshold on the matching score of detected faces. Specifically, if the similarity between two faces is smaller than the threshold, then they will not be linked. The small tracklets which include less than 10 faces are deleted in Frontal and Turning, while tracklets including less than 20 faces are deleted in BBT01.

Data	time	frame	person	track	tracklet	face
Front [14]	51s	1277	4	9	43	4267
Turning [14]	40s	1007	4	4	50	2799
BBT01 [18]	1373s	32977	5	73	182	11525

Table 2. *Basic statistics of the data sets used in our experiments.*

5.2. Performance Evaluation Criteria

For face clustering, we use accuracies computed from the confusion matrix between the predicted labels and the ground-truth labels. In particular, we evaluate clustering performances at two levels: the face-level clustering gives the cluster label for each detected face; the tracklet-level clustering outputs the cluster label for each tracklet, which is determined based on the face-level clustering, as mentioned in Section 4.1.

For tracklet linking, we adopt the following metrics used in [11]: the number of predicted tracks (PT, i.e., the long tracks after linking), mostly tracked tracks (MT, larger is better), Fragments (Frag, smaller is better) and ID switch (IDS, smaller is better). Since we just focus on the performance of tracklet linking, rather than tracking, the metric of the mostly lost tracks (ML) is not used here. Besides, the ground-truth tracks (GT) is predefined based on a threshold of the frame gap t_0: firstly, for the same person, we link all tracklets based on temporal correlations; then, if the frame gap between two adjacent tracklets is larger than t_0, they will be cut to different tracks. In this paper we set $t_0 = 150$.

5.3. Comparisons

For face clustering, we compare the proposed method with HMRF-pc [4] [23], which corresponds to only using the clustering step in our framework without the constraints from tracklet linking. For tracklet linking, we compare a baseline tracklet linking method (denoted as Basic-Linking), corresponding to only running the tracklet linking step in our framework without constraints from clustering results. Besides, we also compare with the state-of-the-art method in the literature of face tracklet linking [18][5].

[4] To highlight the benefits of the additional constraints from tracklet linking, in our experiments the label-level local-smoothness used in [23] is not considered. So it is denoted as HMRF-pc, rather than HMRF-com.

[5] The original method of [18] uses cues from face poses. However, because the code of this method is not directly available to us and our purpose here is to highlight the difference between the subsequent clustering and tracklet linking steps, our implementation of [18] is simplified to use only appearance and motion cues consistent with other methods compared. Another related work [24] considers human body tracking in the fixed scene or the scene with only camera motion. It is not suitable for the dynamic scene, so we did not compare with it.

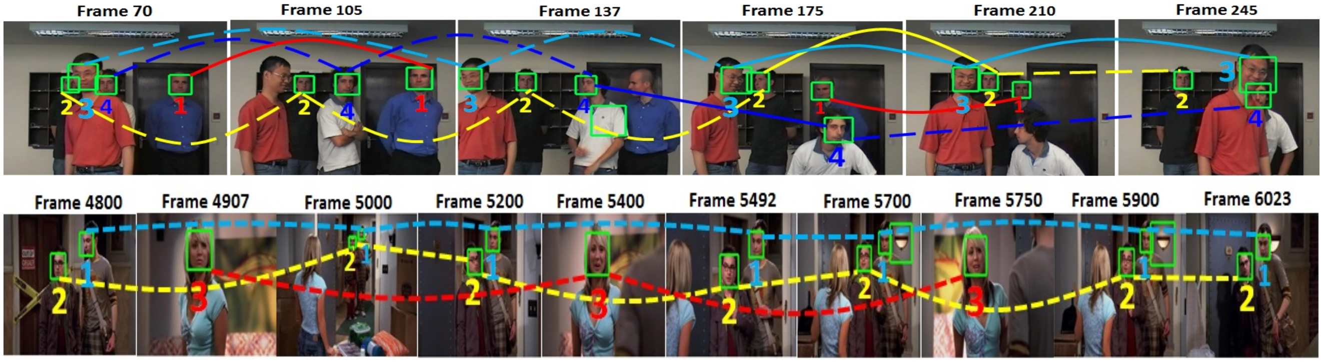

Figure 5. *Illustration of the output face clustering and tracklet linking from our method on* **(Top)** *Turning and* **(Bottom)** *BBT01. The colored numbers indicate the different cluster labels. The solid curves connect detected faces of the same tracklet. The dashed curves represent linking of tracklets.*

5.4. Results

The experiment results on face tracklet linking and clustering are summarized in Table 3 and 4.As shown in Table 3, HMRF-pc performs well on the Frontal video due to the discriminative appearance information of frontal faces. Since the room to improve is small, the tracklet linking fails to help further enhance the clustering result based on HMRF-pc. On the Turning video, because the existence of non-frontal faces, HMRF-pc shows poor performance. On the other hand, including the tracklet linking improves the clustering accuracies significantly by 22% and 25% in tracklet-level and face-level respectively. On BBT01, the clustering results on both levels are also improved over HMRF-pc.

For tracklet linking, Roth *et al.*[18] and the Basic-Linking show the same performance on the Frontal video (Table 4). This may be due to the fact that the classifiers trained on local appearance models from each pair of overlapped tracklets in [18] become less effective, as the appearances of frontal faces are sufficiently distinct in this video. However, constraints originated from the clustering results aids the linking step and reduces errors due to the fast movement. Our method shows improvements on MT and IDS, with a small cost of Frag. On the Turning video, Roth *et al.* [18] performs better than Basic-Linking, because the classifier cue can overcome the difficulty of pose change. With the help of clustering results, good improvements in all metrics are gained over the Basic-Linking. On BBT01, both Roth *et al.* [18] and our method show better performance than Basic-Linking. Roth *et al.* performs better in Frag, while our model gives much smaller IDS.

Some results of the proposed method over Turning and BBT01 are shown in Figure 5. The presented frames show different challenges for clustering and tracklet linking, including: changes in poses, shots, backgrounds, camera movements and occlusions. These results demonstrate the robustness of our method.

The computational cost of the proposed method consists of two parts, including constrained clustering and tracklet linking. For one iteration of clustering, the computational complexity is $O(KN^2)$, where $N = \sum_{i=1}^{n} n_i$ denotes the total number of detected faces. However, utilizing the approximation algorithm mentioned in Section 4.1, it can be significantly reduced. Specifically, the empirical time of constrained clustering is 4.62s, 6.28s and 58.5s, for Frontal, Turning and BBT01 respectively. For one iteration of tracklet linking, the main cost is the Hungarian algorithm. As such, its cost is similar with other linking methods. The proposed method oftentimes converges in less than 10 iterations between clustering and linking in our experiments.

6. Conclusions and Discussions

We describe a novel method that simultaneously clusters and associates faces of distinct humans in long video sequences for identity maintenance. Our method is based on a hidden Markov random field model that represents the joint dependencies of cluster labels and tracklet linking relations. We provide an efficient algorithm, based on constrained clustering with the simulated field algorithm and optimal matching for the simultaneous inference of cluster labels and tracklet associations. We show improvements on the state-of-the-art results in face tracking and clustering performances on several challenging video datasets.

There are a few future directions we would like to further explore. As shown in the top row of Figure 5, a missing and a false positive detection occur in frame 137, due to the pose change. Consequently, the two tracklets in red color fail to be linked. It demonstrates that the performance of the face clustering and tracklet linking can benefit with more sophisticated face detection methods that robust to pose, orientation or illumination changes. Furthermore, we will also investigate more efficient optimization procedures of the constrained clustering and matching problems, and incorporating the simultaneous face clustering and linking into an overall system for video summarization.

	Frontal [14]		Turning [14]		BBT01 [18]	
	tracklet-level	face-level	tracklet-level	face-level	tracklet-level	face-level
HMRF-pc [23]	**90.70**	**94.95**	68.40	67.83	62.64	59.61
Ours	**90.70**	**94.95**	**90.00**	**92.57**	**66.48**	**66.77**

Table 3. *Clustering accuracies on three videos. Tracklet-level clustering means the cluster labels of each tracklet, while face-level clustering represents the cluster labels of each face.*

	Frontal [14]				Turning [14]				BBT01 [18]			
	PT	MT	Frag	IDS	PT	MT	Frag	IDS	PT	MT	Frag	IDS
Roth *et al.* [18]	11	4	**24**	13	5	2	**8**	**4**	72	**68**	**81**	10
Basic-Linking	11	4	**24**	13	6	3	11	7	74	64	82	9
Ours	15	**5**	25	**10**	5	**4**	**8**	5	79	**68**	83	**4**

Table 4. *Experiment results of tracklet linking on three videos. PT: number of predicted tracks. MT: mostly tracked tracks (larger is better). Frag: number of fragments (smaller is better). IDS: number of ID switch (smaller is better).*

7. Acknowledgements

The work was completed when the first author was a visiting student at Rensselaer Polytechnic Institute (RPI), supported by a scholarship from China Scholarship Council (CSC). We thank CSC and RPI for their support. Qiang Ji's involvement in this work is supported in part by a grant from the US National Science Foundation (NSF, No. 1145152). Siwei Lyu is supported by an U.S. NSF CAREER Award (IIS-0953373). Bao-Gang Hu and Baoyuan Wu are supported by the National Natural Science Foundation of China (NSFC, No. 61075051). We thank the reviewers for their constructive comments.

References

[1] O. Arandjelovic and R. Cipolla. Automatic cast listing in feature-length films with anisotropic manifold space. In *CVPR*, 2006.

[2] G. Celeux, F. Forbes, and N. Peyrard. Em procedures using mean field-like approximations for markov model-based image segmentation. *Pattern Recognition*, 36(1):131–144, 2003.

[3] R. Cinbis, J. Verbeek, and C. Schmid. Unsupervised metric learning for face identification in tv video. In *ICCV*, 2011.

[4] A. Fitzgibbon and A. Zisserman. On affine invariant clustering and automatic cast listing in movies. In *ECCV*, 2002.

[5] A. Fitzgibbon and A. Zisserman. Joint manifold distance: a new approach to appearance based clustering. In *CVPR*, 2003.

[6] Y. Hu, A. Mian, and R. Owens. Sparse approximated nearest points for image set classification. In *CVPR*, 2011.

[7] C. Huang, B. Wu, and R. Nevatia. Robust object tracking by hierarchical association of detection responses. In *ECCV*, pages 788–801. Springer, 2008.

[8] D. Koller and N. Friedman, editors. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, Cambridge, MA, 2009.

[9] H. W. Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[10] C.-H. Kuo, C. Huang, and R. Nevatia. Multi-target tracking by on-line learned discriminative appearance models. In *CVPR*, 2010.

[11] Y. Li, C. Huang, and R. Nevatia. Learning to associate: Hybridboosted multi-target tracker for crowded scene. In *CVPR*, 2009.

[12] Z. Li, J. Liu, and X. Tang. Pairwise constraint propagation by semidefinite programming for semi-supervised classification. In *ICML*, pages 576–583, 2008.

[13] Z. Lu and H. H. S. Ip. Constrained spectral clustering via exhaustive and efficient constraint propagation. In *ECCV*, pages 1–14, 2010.

[14] E. Maggio, E. Piccardo, C. Regazzoni, and A. Cavallaro. Particle phd filtering for multi-target visual tracking. In *ICASSP*, 2007.

[15] P. Nillius, J. Sullivan, and S. Carlsson. Multi-target tracking-linking identities using bayesian network inference. In *CVPR*, 2006.

[16] P. Perona and L. Zelnik-Manor. Self-tuning spectral clustering. *Advances in neural information processing systems*, 17:1601–1608, 2004.

[17] Z. Qin and C. R. Shelton. Improving multi-target tracking via social grouping. In *CVPR*, 2012.

[18] M. Roth, M. Bauml, R. Nevatia, and R. Stiefelhagen. Robust multi-pose face tracking by multi-stage tracklet association. In *ICPR*, 2012.

[19] R.Wang, S. Shan, X. Chen, and W. Gao. Manifold-manifold distance with application to face recognition based on image set. In *CVPR*, 2008.

[20] X. Song, X. Shao, H. Zhao, J. Cui, R. Shibasaki, and H. Zha. An online approach: Learning-semantic-scene-by-tracking and tracking-by-learning-semantic-scene. In *CVPR*, 2010.

[21] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *CVPR*, 2001.

[22] N. Vretos, V. Solachidis, and I. Pitas. A mutual information based face clustering algorithm for movie content analysis. *Image and Vision Computing*, 2011.

[23] B. Y. Wu, Y. F. Zhang, B. G. Hu, and Q. Ji. Constrained clustering and its application to face clustering in videos. In *CVPR*, 2013.

[24] B. Yang and R. Nevatia. Multi-target tracking by online learning of non-linear motion patterns and robust appearance models. In *CVPR*, 2012.