Joint Face Representation Adaptation and Clustering in Videos

Zhanpeng Zhang[1], Ping Luo[2,1], Chen Change Loy[1,2], and Xiaoou Tang[1,2]

[1] Dept. of Information Engineering, The Chinese University of Hong Kong
[2] Shenzhen Key Lab of Comp. Vis. & Pat. Rec., Shenzhen Institutes of Advanced Technology, CAS, China

Abstract. Clustering faces in movies or videos is extremely challenging since characters' appearance can vary drastically under different scenes. In addition, the various cinematic styles make it difficult to learn a universal face representation for all videos. Unlike previous methods that assume fixed handcrafted features for face clustering, in this work, we formulate a joint face representation adaptation and clustering approach in a deep learning framework. The proposed method allows face representation to gradually adapt from an external source domain to a target video domain. The adaptation of deep representation is achieved without any strong supervision but through iteratively discovered weak pairwise identity constraints derived from potentially noisy face clustering result. Experiments on three benchmark video datasets demonstrate that our approach generates character clusters with high purity compared to existing video face clustering methods, which are either based on deep face representation (without adaptation) or carefully engineered features.

Keywords: Convolutional Network, Transfer Learning, Face Clustering, Face Recognition

1 Introduction

Face clustering in videos aims at grouping detected faces into different subsets according to different characters. It is a popular research topic [1–5] due to its wide spectrum of applications, *e.g.* video summarization, content-based retrieval, story segmentation, and character interaction analysis. It can be even exploited as a tool for collecting large-scale dataset for face recognition [4].

Clustering faces in videos is challenging. As shown in Fig. 1, the appearance of a character can vary drastically under different scenes as the story progresses. The viewing angles and lighting also vary widely due to the rich cinematic techniques, such as different shots (*e.g.* deep focus, follow shot), variety of lighting techniques, and aesthetics. In many cases, the face is blur due to fast motion or occluded due to interactions between characters. The blurring and occlusion are more severe for fantasy and action movies, *i.e. Harry Potter* series.

Conventional techniques that assume fixed handcrafted features [2, 4] may fail in the cases as shown in Fig. 1. Specifically, handcrafted features are susceptible

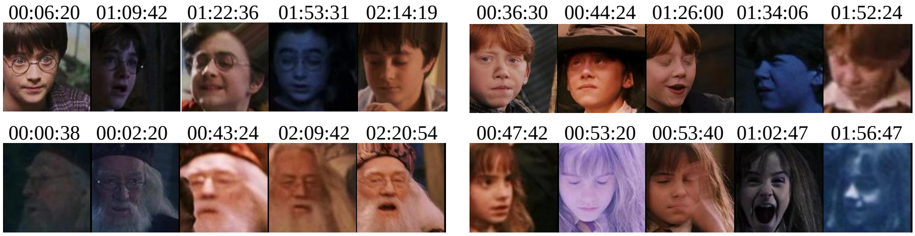


Fig. 1. Faces at different time of the movie *Harry Potter*. Face clustering in videos is challenging due to the various appearance changes as the story progresses.

to large appearance, illumination, and viewpoint variations, and therefore cannot cope with drastic appearance changes. Deep learning approaches have achieved substantial advances for face representation learning [6–8]. These methods could arguably provide a more robust representation to our problem. However, two issues hinder a direct application of deep learning approaches. Firstly, contemporary deep models [6–8] for face recognition are trained with web images or photos from personal albums. These models overfit to the training data distributions thus will not be directly generalizable to clustering faces in different videos with different cinematic styles. Secondly, faces detected in videos usually do not come with identity labels[3]. Hence, we cannot adopt the popular transfer learning approach [11] to adapt these models for our desired videos.

In the absence of precise face annotations, we need to provide deep models with the new capability of learning from weak and noisy supervisions to achieve model adaptation for unseen videos. To this end, we formulate a novel deep learning framework that jointly performs representation adaptation and face clustering in a target video. On one hand, deep representation adaptation provides robust features that permit for better face clustering under unconstrained variations. On the other hand, the clustering results, in return, provide weak pairwise constraints (whether two faces should/should not be assigned to the same cluster) for learning more robust deep representation.

We note that pairwise constraints derived from face tracks (*i.e.* detection or tracking result of face image subsequences) have been used in previous studies to improve video face clustering [3–5]. In particular, faces appearing in the same frame unlikely belong to the same person while any two faces in the same face track should belong to the same person. Our approach differs to these studies in that we not only exploit such static constraints. Our method also takes advantage of weak dynamic constraints obtained from joint clustering. How to carefully utilize such noisy constraints is challenging and we show that our approach is capable of forming a positive and alternating feedback loop between representation adaptation and clustering.

[3] Unless we perform joint matching of visual appearance with video's script [9, 10]. However, an accurate visual-script matching is still far from addressed. This option is beyond the scope of this study.

Contributions: 1) We formulate the video face clustering in a novel deep learning framework. An alternating feedback loop between representation adaptation and clustering is proposed to adapt the deep model from a source domain to a target video domain. To our knowledge, this is the first attempt to introduce deep learning for video face clustering. 2) Different from existing methods that construct static pairwise constraints from the face trajectories, we iteratively discover inter and intra person constraints that allow us to better adapt the face representation in the target video. Experiments on three benchmark video datasets show that the proposed method significantly outperforms existing state-of-the-art methods [2, 4, 12]. In addition, we apply the adapted representation for face verification in the target video. The results demonstrate the superiority of our method compared to deep face representation without adaptation [7]. Code will be released to provide details of our algorithm[4].

2 Related Work

Traditional face clustering methods [13–16] are usually purely data-driven and unsupervised. In particular, these algorithms mainly focus on clustering the photo albums. How to find a good distance metric between faces or effective subspace for face representation is the key point for these algorithms. For example, Zhu *et al.* [14] propose a rank-order distance that measures similarity between two faces using their neighboring information. Fitzgibbon and Zisserman [17] develop a joint manifold distance (JMD) that measures the distance between two subspaces. Each subspace is invariant to a desired group of transformations. In addition, there are also techniques that utilize the user interaction [18], extra information on the web [19] and prior knowledge of family photo albums [20] to improve the performance. Another line of work on clustering employs linear classification cost as a clustering criterion, such as DIFFRAC discriminative clustering framework [21].

Recently, clustering face in videos has attracted more attention. Existing algorithms aim at exploiting the inherent pairwise constraints obtained from the face tracks for better clustering performance. Cinbis *et al.* [3] learn a cast-specific metric, adapted to the people appearing in a particular video, such that pairs of faces within a track are close and faces appearing together in a video frame are far form each other. More recently, Xiao *et al.* [2] introduce subspace clustering to solve this problem, and design a weighted block-sparse regularizer to incorporate the pairwise constraints. These algorithms usually employ handcrafted feature thus the representation effectiveness is limited. For example, the algorithm in [2] extracts SIFT descriptor from the detected facial landmarks. It cannot deal with profile faces. In addition, in these works, the constraints extracted from the face tracks are sparse and not updated in the clustering process. It may fail to provide enough information to guide the clustering. To mitigate this problem, Wu *et al.* [4] augment the constraints by a local smoothness assumption before clustering.

[4] http://mmlab.ie.cuhk.edu.hk/projects/DeepFaceClustering/index.html

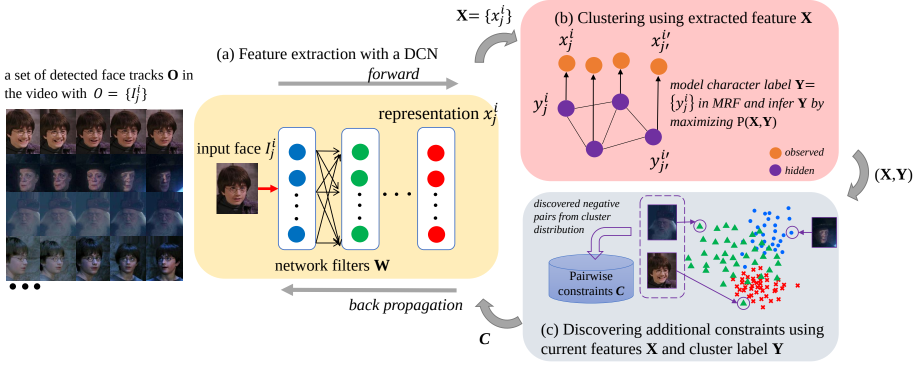

a set of detected face tracks **O** in the video with $O = \{I_j^i\}$
(a) Feature extraction with a DCN
forward
representation x_j^i
input face I_j^i
network filters **W**
back propagation
X= $\{x_j^i\}$
(b) Clustering using extracted feature **X**
x_j^i
$x_{j'}^{i'}$
y_j^i
$y_{j'}^{i'}$
model character label **Y**= $\{y_j^i\}$ in MRF and infer **Y** by maximizing P(**X**,**Y**)
observed
hidden
(**X**,**Y**)
discovered negative pairs from cluster distribution
Pairwise constraints ***C***
(c) Discovering additional constraints using current features **X** and cluster label **Y**
C

Fig. 2. Illustration of the proposed framework. We propose an alternating feedback loop between representation adaptation (a) and clustering (b). In each loop, the DCN extracts face representation (a) by which we perform face clustering in an MRF (b). After that, we discover new negative pairs and add them to the pairwise constraints for DCN adaptation (c). The deep face representation is gradually adapted from an external source domain (e.g., a large-scale face photo dataset) to a target video domain.

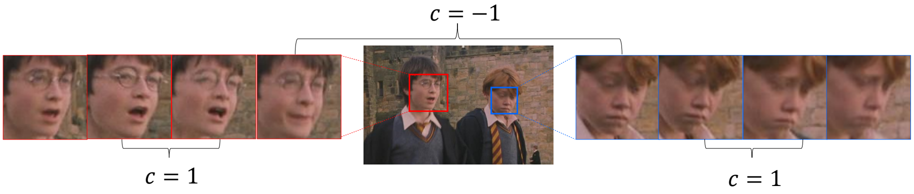


Fig. 3. The pairwise identity constraints set initialized by the face tracks.

Different from these studies, we gain richer guidance by iteratively generating constraints based on the clustering process.

In addition to the inherent pairwise constraint, recent works on video face clustering also incorporate contextual information [1]. For example, clothing [22], speech [23], gender [12], video editing style [24], and cluster proportion prior [25] are employed as additional cues to link faces of the same person. While additional context may introduce uncertainty and its availability will limit the application scenario, in this work, we focus on adapting better face representation via dynamic clustering constraints, which are robust and readily obtainable.

3 Our Approach

This section presents our formulation of joint face representation adaptation and clustering as a probabilistic framework, and provides an alternating optimization method to solve it.

Following previous video face clustering methods [2, 4, 5], given a set of face tracks $\mathbf{O} = \{I_j^i\}$ in a target video, where I_j^i is the j-th face image of the i-th track, our goal is to obtain representation of face images as well as to partition all the face images according to different characters of the target video. We define a set of filters $\mathbf{W}$, which transform the raw pixels of each face image I_j^i into its high-level hidden representation $\mathbf{x}_j^i$ in a Deep Convolutional Network (DCN), as shown in Fig. 2 (a). The filters $\mathbf{W}$ are initialized by training on external large-scale face dataset (see Sec. 3.3). To guide the clustering process, we also define a set of pairwise identity constraints $\mathbf{C} = \{c(I_j^i, I_{j'}^{i'})\}$ for any pair of face images:

$$c(I_j^i, I_{j'}^{i'}) = \begin{cases} 1 & I_j^i \ and \ I_{j'}^{i'} \ belong \ to \ the \ same \ identity, \\ -1 & I_j^i \ and \ I_{j'}^{i'} \ belong \ to \ different \ identities, \\ 0 & not \ defined. \end{cases} \tag{1}$$

Note that different from previous studies [3–5], the identity constraints $\mathbf{C}$ will be updated iteratively instead of kept static. As shown in Fig. 3, at the very beginning, we initialize the identity constraints (denoted as $\mathbf{C}_0$) by assuming all the face images in the same track have the same identity, *i.e.* $c(I_j^i, I_{j'}^{i'}) = 1$, $i = i'$. In addition, for faces in partially or fully overlapped face tracks (*e.g.* faces appearing in the same frame of the video), their identities should be exclusive. Thus, we define $c(I_j^i, I_{j'}^{i'}) = -1$. The constraints between the remaining face pairs are undefined, *i.e.* $c(I_j^i, I_{j'}^{i'}) = 0$.

Then we define a set of cluster labels $\mathbf{Y} = \{y_j^i\}$, where $y_j^i = \ell$ and $\ell \in \{1, 2, ..., K\}$, indicating the corresponding face image I_j^i belongs to which one of the K characters, as shown in Fig. 2 (b). To this end, the clusters and face representation can be obtained by maximizing a posteriori probability (MAP)

$$\mathbf{X}^*, \mathbf{Y}^*, \mathbf{W}^* = \arg\max_{\mathbf{X},\mathbf{Y},\mathbf{W}} p(\mathbf{X}, \mathbf{Y}, \mathbf{W}|\mathbf{O}, \mathbf{C}), \tag{2}$$

where $\mathbf{O} = \{I_j^i\}$ and $\mathbf{X} = \{\mathbf{x}_j^i\}$. $\mathbf{C}$ is the dynamic identity constraint. By factorization, Eqn.(2) is proportional to $p(\mathbf{C}|\mathbf{O}, \mathbf{W})P(\mathbf{C}|\mathbf{X}, \mathbf{Y}, \mathbf{O}, \mathbf{W})p(\mathbf{X}, \mathbf{Y}|\mathbf{O}, W)$ $P(\mathbf{W}|\mathbf{O})$. Note that the image set $\mathbf{O}$ is given and fixed, then we can remove it in the last term. Here we also make the following assumptions: 1) the update of constraints $\mathbf{C}$ is independent to $\mathbf{W}$, *i.e.* $P(\mathbf{C}|\mathbf{X}, \mathbf{Y}, \mathbf{O}, \mathbf{W}) = P(\mathbf{C}|\mathbf{X}, \mathbf{Y})$; 2) $\mathbf{O}$ is independent to the inference process of $\mathbf{Y}$ because $\mathbf{Y}$ is inferred from $\mathbf{X}$, *i.e.* $p(\mathbf{X}, \mathbf{Y}|\mathbf{O}, W) = p(\mathbf{X}, \mathbf{Y}|\mathbf{W})$; 3) inference of the cluster label $\mathbf{Y}$ is independent to $\mathbf{W}$, *i.e.* $p(\mathbf{X}, \mathbf{Y}|\mathbf{W}) = p(\mathbf{X}, \mathbf{Y})$. Then we have

$$p(\mathbf{X}, \mathbf{Y}, \mathbf{W}|\mathbf{O}, \mathbf{C}) \propto p(\mathbf{C}|\mathbf{O}, \mathbf{W})p(\mathbf{C}|\mathbf{X}, \mathbf{Y})p(\mathbf{X}, \mathbf{Y})p(\mathbf{W}), \tag{3}$$

where the first term $p(\mathbf{C}|\mathbf{O}, \mathbf{W})$ solves filters $\mathbf{W}$ of the DCN by using the pairwise identity constraints as supervision. This can be implemented by imposing a contrastive loss in the DCN training process (see Sec. 3.3 for details). As a result, the hidden representation $\mathbf{X}$ can be obtained using the learned filters $\mathbf{W}$. The second term $p(\mathbf{C}|\mathbf{X}, \mathbf{Y})$ updates these constraints leveraging $\mathbf{X}$ and the estimated character labels $\mathbf{Y}$, as discussed in Sec.3.2. The forth term $p(\mathbf{W})$ regularizes the network filters.

In Eqn.(3), the third term $p(\mathbf{X}, \mathbf{Y})$ infers the character label $\mathbf{Y}$ given the hidden representation $\mathbf{X}$. Motivated by the fact that if two face images are close in the space of the hidden representation, the character labels are likely to be the same, we establish the relation between face pairs by Markov Random Field (MRF), where each node represents a character label y_j^i and each edge represents the relation between the character labels. For each node y_j^i, we associate it with the observed variable $\mathbf{x}_j^i$. Then we have

$$p(\mathbf{X}, \mathbf{Y}) = p(\mathbf{X}|\mathbf{Y})p(\mathbf{Y}) \propto \prod_{i,j} \Phi(\mathbf{x}_j^i|y_j^i) \prod_{i,j} \prod_{i',j' \in \mathcal{N}_j^i} \Psi(y_j^i, y_{j'}^{i'}), \tag{4}$$

where $\Phi(\cdot)$ and $\Psi(\cdot)$ are the unary and pairwise term, respectively. $\mathcal{N}_j^i$ signifies a set of face images, which are the neighbors of y_j^i and defined by the representation similarity.

The parameters of Eqn.(3) are optimized by alternating between the following three steps as illustrated in Fig. 2, (1) fix the filter $\mathbf{W}$ of DCN, obtain the current face representation $\mathbf{X}$, and infer character labels $\mathbf{Y}$ by optimizing MRF as defined in Eqn.(4), (2) update the identity constraints $\mathbf{C}$ given $\mathbf{X}$ and the inferred character labels $\mathbf{Y}$, and (3) update the hidden face representation using $\mathbf{W}$ by minimizing the contrastive loss of the identity constraints, corresponding to maximizing $p(\mathbf{C}|\mathbf{O}, \mathbf{W})p(\mathbf{W})$. This optimization process is conducted for $T = 3$ iterations in our implementation. We will describe these three steps in Sec. 3.1, 3.2, and 3.3 respectively.

3.1 Inferring Character Labels

Given the current face representation $\mathbf{X}$, we infer the character labels $\mathbf{Y}$ by maximizing the joint probability $p(\mathbf{X}, \mathbf{Y})$. We employ the Gaussian distribution to model the unary term $\Phi(\cdot)$ in Eqn.(4)

$$\Phi(\mathbf{x}_j^i|y_j^i = \ell) \sim \mathcal{N}(\mathbf{x}_j^i|\mu_\ell, \Sigma_\ell), \tag{5}$$

where μ_ℓ and Σ_ℓ denote the mean vector and covariance matrix of the ℓ-th character, which are obtained and updated in the inference process. For the pairwise term $\Psi(\cdot)$ in Eqn.(4), it is defined as

$$\Psi(y_j^i, y_{j'}^{i'}) = \exp\left\{\alpha v(\mathbf{x}_j^i, \mathbf{x}_{j'}^{i'}) \cdot \left(\mathbf{1}(y_j^i, y_{j'}^{i'}) - \mathbf{1}(v(\mathbf{x}_j^i, \mathbf{x}_{j'}^{i'}) > 0)\right)\right\}, \tag{6}$$

where $\mathbf{1}(\cdot)$ is an indicator function and α is a trade-off coefficient updated in the inference process. Furthermore, $v(\cdot, \cdot)$ is a pre-computed function that encodes the relation between any pair of face images $\mathbf{x}_j^i$ and $\mathbf{x}_{j'}^{i'}$. Similar to [4], positive relation (*i.e.* $v(\cdot, \cdot) > 0$) means that the face images are likely from the same character. Otherwise, they belong to different characters. Specifically, the computation of v is a combination of two cues: (1) the similarity between appearances of a pair of face images and (2) the pairwise spatial and temporal constraints of the face images. For instance, face images within a face track belong to the same character, while face images appearing in the same frame belong to different characters. Intuitively, Eqn. (6) encourages face images with positive relation to be the same character. For example, if $v(\mathbf{x}_j^i, \mathbf{x}_{j'}^{i'}) > 0$ and $y_j^i = y_{j'}^{i'}$, we have $\Psi(y_j^i, y_{j'}^{i'}) = 1$. However, if $v(\mathbf{x}_j^i, \mathbf{x}_{j'}^{i'}) > 0$ but $y_j^i \neq y_{j'}^{i'}$, we have $\Psi(y_j^i, y_{j'}^{i'}) < 1$, indicating the character label assignment is violating the pairwise constraints.

To solve Eqn. (4), we employ the simulated field algorithm [26], which is a classic technique for MRF optimization. To present the main steps of our work clearly, we provide the details of this algorithm and the computation of $v(\cdot, \cdot)$ in the *supplementary material.*

3.2 Dynamic Pairwise Identity Constraints

Different from previous methods [12, 2, 4], where the identity constraints between a pair of face images are fixed after initialized at the very beginning, the identity constraints $\mathbf{C}$ in our approach is updated iteratively in the adaptation process to obtain additional supervision to adapt the face representation. In particular, after inferring the character labels $\mathbf{Y}$ in Sec.3.1, we compute the confidence value that measures the possibility of a face pair from different characters, *i.e.* negative pair. After that, we append pairs with high confidence to the current set of pairwise constraints $\mathbf{C}$. The negative pair generation process is motivated by the facts that: diverse clusters contain large noise, while clusters with high purity are compact; and faces from the same character are likely to be close in the representation space. Specifically, for the face pairs in each cluster, we define the confidence Q by

$$Q(i_\ell, i'_\ell) = \frac{1}{1 + \gamma e^{-trace(\Sigma_\ell) D_{i_\ell, i'_\ell}}} \tag{7}$$

where i_ℓ and i'_ℓ denote the faces in cluster ℓ. $trace(\Sigma_\ell)$ is the trace of the covariance matrix, which describes the variations within the cluster. D_{i_ℓ, i'_ℓ} is the L2-distance between the faces in the learned face representation space $\mathbf{X}$. γ is a scale factor for normalization. In this case, face pairs in diverse clusters with large distances will have high confidence. In our implementation, face pairs with confidence value $Q(i_\ell, i'_\ell) > 0.5$ are selected as additional negative pairs.

3.3 Face Representation Adaptation

Pre-training DCN. The network filter $\mathbf{W}$ is initialized by pre-training DCN to classify massive identities as discussed in DeepID2+ [7]. We adopt its network architecture due to its exceptional performance in face recognition.

Specifically, DCN takes face image of size 55×47 as input. It has four successive convolution layers followed by one fully connected layer. Each convolution layer contains learnable filters and is followed by a 2×2 max-pooling layer and Rectified Linear Units (ReLUs) [27] as the activation function. The number of feature map generated by each convolution layer is 128, and the dimension of the face representation generated by the final fully connected layer is 512. Similar to [7], our DCN is pre-trained on CelebFace [28], with around $290,000$ faces images from $12,000$ identities. The training process is conducted by back-prorogation using both the identification and verification loss functions.

Fine-tuning Face Representation by C. After updating the identity constraints $\mathbf{C}$ in Sec.3.2, we update the hidden face representation by back-propagating the constraint information to the DCN. In particular, given a constraint in $\mathbf{C}$, we minimize a contrastive loss function [7], $E_c(\mathbf{x}_j^i, \mathbf{x}_{j'}^{i'})$, which is defined as

$$E_c = \begin{cases} \frac{1}{2} \parallel \mathbf{x}_j^i - \mathbf{x}_{j'}^{i'} \parallel_2^2, & c(I_j^i, I_{j'}^{i'}) = 1, \\ \frac{1}{2} \max(0, \tau - \parallel \mathbf{x}_j^i - \mathbf{x}_{j'}^{i'} \parallel_2^2), & c(I_j^i, I_{j'}^{i'}) = -1, \end{cases} \tag{8}$$

where τ is the margin between different identities. Eqn. (8) encourages face images of the same character to be close and that of the different characters to be far away from each other.

To facilitate representation adaptation, beside E_c, we fine-tune DCN by back-propagating the errors of the MRF defined in Sec.3.1. We take the negative logarithm of Eqn.(4), drop the constant terms, and obtain $\frac{1}{2}\sum_{i,j}\sum_{\ell=1}^{K} \mathbf{1}(y_j^i = \ell)\big(\ln|\Sigma_\ell| + (\mathbf{x}_j^i - \mu_\ell)^{\mathsf{T}} \Sigma_\ell^{-1} (\mathbf{x}_j^i - \mu_\ell)\big)$. Note that in the step of representation adaptation, we update network filters $\mathbf{W}$ while keeping the remaining parameters fixed, such as $\mathbf{Y}$, Σ, and μ. Therefore, minimizing the above function is equivalent to optimize $\mathbf{W}$, such that the distance between each face image and its corresponding cluster center is minimized. We define this loss function as below

$$E_{MRF} = \frac{1}{2} \sum_{\ell=1}^{K} \mathbf{1}(y_j^i = \ell) \parallel x_j^i - \mu_\ell \parallel_2^2 . \tag{9}$$

By minimizing Eqn.(9), the representation naturally reduces the intra-personal variations.

Combining Eqn. (8) and (9), the training process is conducted by back-propagation using stochastic gradient descent (SGD) [29]. Algorithm 1 shows the entire pipeline of the proposed joint face representation adaptation and clustering.

4 Experiments

4.1 Datasets

Experiments are conducted on three publicly available face clustering datasets: Accio [30], BF0502 [31] and Notting-Hill [32]. The Accio dataset is collected from

Algorithm 1 Joint face representation adaptation and clustering with dynamic constraints

```
Input:
    Face tracks $\mathbf{O} = \{\mathbf{I}_j^i\}$, character number $K$ of the target video.
Output:
    Character labels $\mathbf{Y}$ and filters $\mathbf{W}$ of the DCN.
Generate a set of initial pairwise constraints, denoted as $\mathbf{C}_0$, using the face tracks
    $\mathbf{O}$ as introduced at the beginning of Sec.3.
Pre-train the filters $\mathbf{W}$ of DCN with an external face dataset as discussed in Sec.3.3.
Fine-tune filters $\mathbf{W}$ of DCN with $\mathbf{C}_0$ as discussed in Sec.3.3.
for $t = 1$ to $T$ do
    Generate the face representation $\mathbf{x}_j^i$ for each face image $I_j^i$.
    Infer the corresponding character label $y_j^i$ with fixed $\mathbf{W}$ by maximizing Eqn. (4)
        (Sec. 3.1).
    Discover additional negative face pairs $\mathbf{C}_t$ and append them to the pairwise
        identity constraints $\mathbf{C}$ (Sec. 3.2).
    Fine-tune the face representation by minimizing Eqn. (8) and (9) using back-
        propagation on the DCN (Sec. 3.3).
end for
```

the eight "*Harry Potter*" movies and we use the first instalment of this series in our experiment (denoted as Accio-1 in the following text). Accio-1 contains multiple challenges, such as a large number of dark scenes and many tracks with non-frontal faces. In addition, the number of the faces of each character is unbalanced (e.g., there are 51,620 faces of the character "*Harry Potter*", while 4,843 faces for "*Albus Dumbledore*"). In particular, there are 36 characters, 3,243 tracks, and 166,885 faces in the test movie. The face tracks are obtained by tracking-by-detection using a particle filter [30]. BF0502 [31] is collected from the TV series "Buffy the Vampire Slayer". Following the protocol of other face video clustering studies [2, 4, 12], we evaluate on 6 main casts including 17,337 faces in 229 face tracks. The dataset Notting-Hill is gathered from the movie "*Notting Hill*". It includes faces of 5 main casts, with 4,660 faces in 76 tracks.

4.2 Evaluation Criteria and Baselines

The clustering performance is measured in two different ways. In the first one, we evaluate how the algorithm balances the precision and recall. In particular, we employ the *B-cubed precision and recall* [1, 33] to compute one series of score pairs for the tested methods given different numbers of clusters. Specifically, the B-cubed precision is the fraction of face pairs assigned to a cluster with matching identity labels. The B-cubed recall is the average fraction of face pairs belonging to the groundtruth identity assigned to the same cluster [15]. To combine the precision and recall, we use the F_1-score (the harmonic mean of these two metrics).

For the second evaluation metric, we use *accuracy* computed from a confusion matrix, which is derived by the best match between the cluster labels and

Table 1. B-cubed precision (P), recall (R), and F_1-score (F) with different iterations (T) of the proposed method on the Accio-1 [30] dataset, with cluster number $K = 36$.

T=1			T=2			T=3			T=4		
P	R	F	P	R	F	P	R	F	P	R	F
0.63	0.30	0.41	0.68	0.32	0.44	0.69	0.35	0.46	0.67	0.33	0.44

Table 2. B-cubed precision (P), recall (R), and F_1-score (F) of different methods on the Accio-1 [30] (Harry Potter) dataset.

Methods	#cluster=40			#cluster=50			#cluster=60			#cluster=120			#cluster=240		
	P	R	F	P	R	F	P	R	F	P	R	F	P	R	F
K-means	.246	.114	.156	.262	.105	.150	.289	.089	.136	.321	.059	.100	.379	.044	.079
K-means-DeepID2$^+$	.543	.201	.293	.574	.181	.275	.581	.155	.244	.594	.099	.169	.612	.074	.132
DIFFRAC [21]	.307	.109	.160	.326	.080	.129	.338	.089	.141	.336	.057	.098	.347	.032	.059
DIFFRAC-DeepID2$^+$	.557	.213	.301	.586	.181	.277	.607	.160	.253	.622	.120	.201	.620	.068	.122
WBSLRR [2]	.296	.153	.202	.322	.117	.172	.346	.092	.145	.354	.087	.140	.384	.033	.061
WBSLRR-DeepID2$^+$	.502	.206	.292	.533	.184	.274	.551	.161	.249	.599	.114	.192	.637	.054	.100
HMRF [4]	.272	.128	.174	.295	.101	.151	.303	.093	.142	.342	.067	.112	.403	.041	.074
HMRF-Fisher	.583	.234	.334	.591	.184	.281	.604	.176	.273	.667	.127	.213	.712	.086	.154
HMRF-DeepID2$^+$	.599	.230	.332	.616	.211	.314	.621	.174	.272	.644	.128	.214	.669	.075	.135
DeepID2$^+\cdot\mathbf{C}_0$	.655	.253	.365	.676	.238	.352	.684	.192	.300	.713	.155	.255	.785	.132	.226
DeepID2$^+\cdot\mathbf{C}_0\cdot$Intra	.657	.312	.423	.685	.286	.404	.698	.229	.345	.735	.201	.316	.781	.158	.263
Full model	**.711**	**.352**	**.471**	**.739**	**.312**	**.439**	**.768**	**.242**	**.368**	**.779**	**.203**	**.322**	**.841**	**.172**	**.286**

groundtruth identities. The best match is obtained by using the Hungarian method [34]. This evaluation metric is widely employed in current video face clustering methods [2, 4, 12, 25].

We compare the proposed method with the following classic and state-of-the-art approaches: (1) K-means [35]; (2) Unsupervised Logistic Discriminant Metric Learning (ULDML) [3]; (3) Penalized Probabilistic Clustering (PPC) [36]; (4) DIFFRAC [21] discriminative clustering; (5)HMRF-based clustering [4]; (6) Weighted Block-Sparse Low Rank Representation (WBSLRR) method [2]; (7) Multi-cue Augmented Face Clustering (McAFC) [12]. The latter three recent approaches are specifically designed for face clustering in videos.

4.3 Experiments on Accio-1 (Harry Potter) [30]

Effects of the iterations in the adaptation process. The evaluation is first conducted on the Accio-1 dataset [30]. Firstly, to demonstrate the effectiveness of the alternating adaptation process, we report the performance in different iterations in Table 1. Given that there are 36 characters in this movie, we set the cluster number $K = 36$ here. It is observed that the performance increases and it converges when $T = 3$. This demonstrates the benefits of the alternating adaptation process.

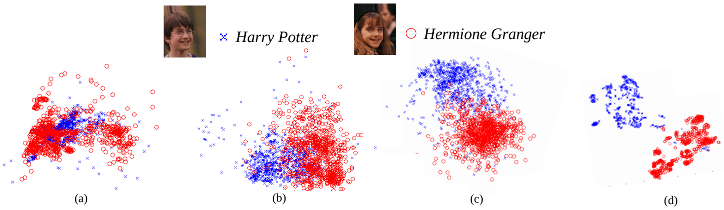


Fig. 4. Visualization of different characters' face representation in a chapter of the movie "*Harry Potter*". (a)-(c): projecting different representations to a 2D space by PCA: (a) raw pixel value, (b) DeepID2$^+$, and (c) our adapted representation. (d): projecting the adapted representation to a 2D space by t-SNE [38].

Performance of different variants and competitors. To verify other components of the proposed method, we further test different variants of our method, as well as other existing models:

- DeepID2$^+$$\cdot\mathbf{C}_0$: We perform clustering with fixed DCN filters $\mathbf{W}$ and pairwise constraints set $\mathbf{C}_0$. That means we do not perform representation adaptation after training the network on the face photo dataset and initial pairwise constraints. This variant corresponds to the typical transfer learning strategy [11] adopted in most deep learning studies. Since our network structure and pre-training data are identical to that of [7], We use the notation DeepID2$^+$.
- DeepID2$^+$$\cdot\mathbf{C}_0\cdot$Intra: We finetune DCN filters $\mathbf{W}$ only with the intra person constraints (Eqn. (9)) but not the inter person constraints (Eqn. (8)).
- "HMRF$^+$" and "HMRF-DeepID2$^+$": Since HMRF [4] only uses the raw pixel value or handcrafted features, for fair comparison, we also use the DCN representation initially trained on the face photo dataset for this method. Similar notation scheme is used for K-means [35], DIFFRAC [21] and WBSLRR [2] algorithms, and Fisher Vector [37] representation.

We report the B-cubed precision and recall, as well as the F_1-score of different methods in Table 2. It is observed that:

- As the cluster number increases, the precision increases while the recall decreases. This is intuitive since larger number of clusters decreases the cluster size and improves the cluster purity.
- This dataset is very challenging. For example, the K-means [35] only achieved 0.379 in precision even the cluster number is nearly six times of the identities.
- The DCN representation improves the performance substantially (e.g., the DIFFRAC [21], HMRF [4], and WBSLRR [2] method have 0.2∼0.3 improvements in terms of precision when employing the DCN representation).
- The proposed method (*i.e.* full model) performs the best, and the comparison on different variants of the proposed method demonstrates the superiority of the alternating adaptation process (e.g., the performance of full model is better than that of "DeepID2$^+$$\cdot\mathbf{C}_0$").

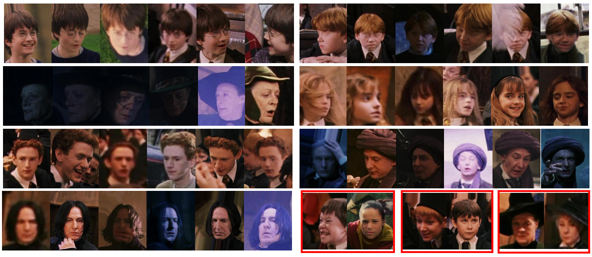


Fig. 5. Example results in different clusters generated by the proposed method on Accio-1 [30]. Face pairs in red rectangles are incorrectly assigned to a same cluster.

- Interestingly, by comparing "DeepID2$^+$$\cdot\mathbf{C}_0$" and "DeepID2$^+$$\cdot\mathbf{C}_0$$\cdot$Intra", we can observe obvious improvement on recall, but the precision can hardly increase. This is because using only the intra person constraints can decrease the distances between the faces of the same character, but can not provide discriminative information directly to correct the wrong pairs in the cluster. Thus, both intra- and inter-person constraints are important for discriminative face clustering.

Representation visualization. Fig. 4 visualizes different representations by projecting them to a 2D space. Firstly, in Fig. 4 (a)-(c), we project the representations by PCA. We can observe that for the original pixel values, the representations are severely overlapped. By pre-training DCN with face dataset and adapting the representation, we can gradually obtain more discriminative representation. After that, we use the t-SNE [38] dimensionality reduction and Fig. 4 (d) shows that the characters can be almost linearly separated. This demonstrates the effectiveness of the adapted face representation.

Example results. Fig. 5 shows some clustering examples, where each bank except the right bottom one denotes a cluster. It is observed that each cluster covers a character's faces in different head pose, lighting conditions, and expressions. This demonstrates the effectiveness of the adapted face representation. We also show some failed cases indicated by the red rectangles, where each pair with different characters is incorrectly partitioned in the same cluster. These faces fail mainly because of the unbalanced face number of the identity (*e.g.* , some characters just appear in a few shots) and some extreme lighting conditions.

4.4 Experiments on BF0502 [31] and Notting Hill [32]

We report the accuracy of our method and other competitors in Fig. 6 and 7. Following previous research [2, 4, 12], each algorithm is repeated for 30 times, and the mean accuracy and standard deviation are reported. The results of the competitors are gathered from the literatures [2, 4, 12]. Fig. 6 shows that our method achieves substantial improvement compared to the best competitor (from 62.76% to 92.13%), demonstrating the superiority of our method.

Methods	Accuracy(%)
K-means	39.31 ± 4.51
ULDML [3]	41.62 ± 0.00
PPC [36]	78.88±5.15
HMRF [4]	50.30 ± 2.73
WBSLRR [2]	62.76 ±1.10
Our method	**92.13 ±0.90**

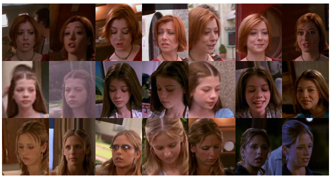


Fig. 6. Left: Clustering accuracies of the state-of-the-art methods and our method on the BF0502 [31] dataset. Right: Example clustering results. Each row denotes a cluster.

Methods	Accuracy(%)
K-means	69.16 ± 3.22
ULDML [3]	73.18 ± 8.66
PPC [36]	78.88±5.15
HMRF [4]	84.39 ± 1.47
CMVFC [39]	93.42 ± 0.00
McAFC [12]	96.05 ±0.39
WBSLRR [2]	96.29 ±0.00
Our method	**99.04 ±0.20**

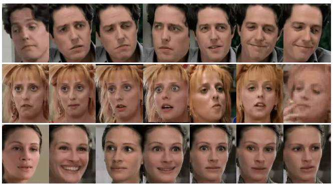


Fig. 7. Left: Clustering accuracies of the state-of-the-art methods and ours on the Notting Hill [32] dataset. Right: Example clustering results. Each row denotes a cluster.

4.5 Computational Cost

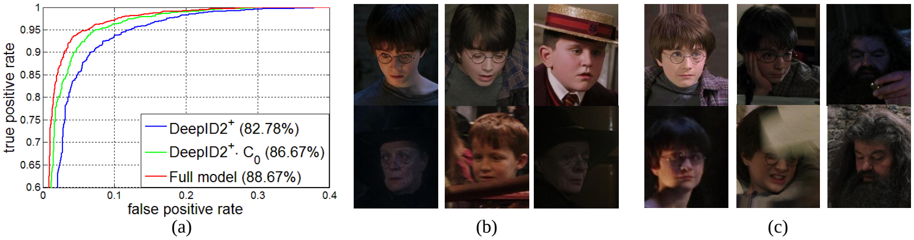
Training a high-capacity DCN from scratch is time consuming due to the large amount of training data. However, given the DCN pre-trained on a large face dataset, for a new target video, we only need to perform representation adaptation. Table. 3 shows the running time of our algorithm on the videos. In particular, the DCN adaptation in Table. 3 is the time that we use to train the DCN with a Nvidia Titan GPU and the total time additionally includes the computation cost of other steps (*i.e.* inferring the character label $\mathbf{Y}$ in Sec. 3.1 and updating the constraints in Sec. 3.2). It is observed that the time cost is feasible in many applications, where face clustering can be performed off-line.

4.6 Application to Face Verification

To further demonstrate the effectiveness of the adapted face representation, we perform face verification on the Accio-1 dataset [30]. To evaluate the representation directly, for each face pair, we calculate the L2 distance of the representation to measure the pairwise similarity, instead of training a joint Bayesian model as in [7]. If the distance is larger than a threshold, the face pair is regarded as negative (*i.e.* different identities). The threshold is determined by 1,000 validation face pairs (500 positive and 500 negative samples) randomly chosen from Accio-1 [30] dataset. Evaluation is performed on another 1,000 randomly chosen face pairs (500 positive and 500 negative samples) from this dataset. The validation and test faces are exclusive in terms of scenes and identities. Similar to Sec. 4.3, we compare the performance among different representations, including (1) DeepID2$^+$, (2) DeepID2$^+ \cdot \mathbf{C}_0$, and (3) full model.

Table 3. Running time for the Accio-1 [30] (Harry Potter), BF0502 [31], and Notting Hill [32] dataset (in minutes).

Accio-1		BF0502		Notting Hill	
DCN adaptation	total	DCN adaptation	total	DCN adaptation	total
13.8	30.3	2.2	5.3	0.5	1.4



(a) (b) (c)

Fig. 8. (a): ROC of face verification on Accio-1 [30] dataset. The number in the legend indicates the verification accuracy. (b) and (c): negative and positive pairs failed to be matched by DeepID2+ [7] but successfully matched after adaptation by our approach.

Fig. 8 shows the Receiver Operating Characteristic Comparison (ROC). It is evident that representation adapted by the proposed method outperforms the original deep representation and can handle different cinematic styles better.

5 Conclusion

In this work, we have presented a novel deep learning framework for joint face representation adaptation and clustering in videos. In the absence of precise face annotations on the target video, we propose a feedback loop in which the deep representation provides robust features for face clustering, and the clustering results provide weak pairwise constraints for learning more suitable deep representation with respect to the target video. Experiments on three benchmark video datasets demonstrate the superiority of the proposed method when compared to the state-of-the-art video clustering methods that either use handcrafted features or deep face representation (without adaptation). The effectiveness of the adapted face representation is further demonstrated by a face verification experiment.

Acknowledgements

This work is partially supported by SenseTime Group Limited, the Hong Kong Innovation and Technology Support Programme, and the General Research Fund sponsored by the Research Grants Council of the Kong Kong SAR (CUHK 416312).

References

1. Zhang, L., Kalashnikov, D.V., Mehrotra, S.: A unified framework for context assisted face clustering. In: ACM Conference on International Conference on Multimedia Retrieval. (2013)
2. Xiao, S., Tan, M., Xu, D.: Weighted block-sparse low rank representation for face clustering in videos. In: ECCV. (2014)
3. Cinbis, R., Verbeek, J., Schmid, C.: Unsupervised metric learning for face identification in tv video. In: ICCV. (2011)
4. Wu, B., Zhang, Y., Hu, B.G., Ji, Q.: Constrained clustering and its application to face clustering in videos. In: CVPR. (2013)
5. Wu, B., Lyu, S., Hu, B., Ji, Q.: Simultaneous clustering and tracklet linking for multi-face tracking in videos. In: ICCV. (2013)
6. Taigman, Y., Yang, M., Ranzato, M., Wolf, L.: DeepFace: Closing the gap to human-level performance in face verification. In: CVPR. (2014)
7. Sun, Y., Wang, X., Tang, X.: Deeply learned face representations are sparse, selective, and robust. In: CVPR. (2015)
8. Schroff, F., Kalenichenko, D., Philbin, J.: FaceNet: A unified embedding for face recognition and clustering. In: CVPR. (2015)
9. Ding, L., Yilmaz, A.: Learning relations among movie characters: A social network perspective. In: ECCV. (2010)
10. Tapaswi, M., Bauml, M., Stiefelhagen, R.: Improved weak labels using contextual cues for person identification in videos. In: FG. (2015)
11. Yosinski, J., Clune, J., Bengio, Y., Lipson, H.: How transferable are features in deep neural networks? In: NIPS. (2014)
12. Zhou, C., Zhang, C., Fu, H., Wang, R., Cao, X.: Multi-cue augmented face clustering. In: ACM Multimedia Conference. (2015)
13. Li, Z., Tang, X.: Bayesian face recognition using support vector machine and face clustering. In: CVPR. (2004)
14. Zhu, C., Wen, F., Sun, J.: A rank-order distance based clustering algorithm for face tagging. In: CVPR. (2011)
15. Otto, C., Klare, B., Jain, A.: An efficient approach for clustering face images. In: International Conference on Biometrics. (2015)
16. Cao, X., Zhang, C., Fu, H., Liu, S., Zhang, H.: Diversity-induced multi-view subspace clustering. In: CVPR. (2015)
17. Fitzgibbon, A., Zisserman, A.: Joint manifold distance: a new approach to appearance based clustering. In: CVPR. (2003)
18. Tian, Y., Liu, W., Xiao, R., Wen, F., Tang, X.: A face annotation framework with partial clustering and interactive labeling. In: CVPR. (2007)
19. Berg, T., Berg, A., Edwards, J., Maire, M., White, R., Teh, Y.W., Learned-Miller, E., Forsyth, D.: Names and faces in the news. In: CVPR. (2004)
20. Xia, S., Pan, H., Qin, A.: Face clustering in photo album. In: ICPR. (2014)
21. Bach, F.R., Harchaoui, Z.: Diffrac: a discriminative and flexible framework for clustering. In: NIPS. (2008) 49–56
22. El-Khoury, E., Senac, C., Joly, P.: Face-and-clothing based people clustering in video content. In: ACM International Conference on Multimedia Information Retrieval. (2010)
23. Paul, G., Elie, K., Sylvain, M., Jean-Marc, O., Paul, D.: A conditional random field approach for audio-visual people diarization. In: ICASSP. (2014)

24. Tapaswi, M., Parkhi, O.M., Rahtu, E., Sommerlade, E., Stiefelhagen, R., Zisserman, A.: Total cluster: A person agnostic clustering method for broadcast videos. In: Proceedings of Indian Conference on Computer Vision Graphics and Image Processing. (2014)
25. Tang, Z., Zhang, Y., Li, Z., Lu, H.: Face clustering in videos with proportion prior. In: IJCAI. (2015)
26. Celeux, G., Forbes, F., Peyrard, N.: EM procedures using mean field-like approximations for markov model-based image segmentation. Pattern recognition **36**(1) (2003) 131–144
27. Nair, V., Hinton, G.E.: Rectified linear units improve restricted boltzmann machines. In: ICML. (2010)
28. Sun, Y., Wang, X., Tang, X.: Deep learning face representation from predicting 10,000 classes. In: CVPR. (2014)
29. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NIPS. (2012)
30. Ghaleb, E., Tapaswi, M., Al-Halah, Z., Ekenel, H.K., Stiefelhagen, R.: Accio: A data set for face track retrieval in movies across age. In: ACM International Conference on Multimedia Retrieval. (2015)
31. Everingham, M., Sivic, J., Zisserman, A.: Hello! My name is... buffy –automatic naming of characters in TV video. In: BMVC. (2006)
32. Zhang, Y., Xu, C., Lu, H., Huang, Y.: Character identification in feature-length films using global face-name matching. IEEE Transactions on Multimedia **11**(7) (2009) 1276–1288
33. Amigó, E., Gonzalo, J., Artiles, J., Verdejo, F.: A comparison of extrinsic clustering evaluation metrics based on formal constraints. Information retrieval **12**(4) (2009) 461–486
34. Kuhn, H.W.: The hungarian method for the assignment problem. Naval research logistics quarterly **2**(1-2) (1955) 83–97
35. Bishop, C.M.: Pattern recognition and machine learning. Springer (2006)
36. Lu, Z., Leen, T.K.: Penalized probabilistic clustering. Neural Computation **19**(6) (2007) 1528–1567
37. Parkhi, O., Simonyan, K., Vedaldi, A., Zisserman, A.: A compact and discriminative face track descriptor. In: CVPR. (2014) 1693–1700
38. Van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of Machine Learning Research **9**(2579-2605) (2008)
39. Cao, X., Zhang, C., Zhou, C., Fu, H., Foroosh, H.: Constrained multi-view video face clustering. IEEE Transactions on Image Processing **24**(11) (2015) 4381–4393