12-2016

Cast2Face: Assigning character names onto faces in movie with actor-character correspondence

Guangyu GAO

Mengdi XU

Jialie SHEN
Singapore Management University, jlshen@smu.edu.sg

Huangdong MA

Shuicheng YAN

Published in IEEE Transactions on Circuits and Systems for Video Technology, Dec 2016, Volume: 26, Issue: 12, pp. 2299-2312.
http://doi.org/10.1109/TCSVT.2015.2504738

Cast2Face: Assigning Character Names Onto Faces in Movie With Actor-Character Correspondence

Guangyu Gao, Mengdi Xu, Jialie Shen, Huadong Ma, *Senior Member, IEEE*,
and Shuicheng Yan, *Senior Member, IEEE*

***Abstract*— Automatically identifying characters in movies has attracted researchers' interest and led to several significant and interesting applications. However, due to the vast variation in character appearance as well as the weakness and ambiguity of available annotation, it is still a challenging problem. In this paper, we investigate this problem with the supervision of actor-character name correspondence provided by the movie cast. Our proposed framework, namely, Cast2Face, is featured by: 1) we restrict the assigned names within the set of character names in the cast; 2) for each character, by using the corresponding actor and movie name as keywords, we retrieve from the Google image search and get a group of face images to form the gallery set; 3) the probe face tracks in the movie are then identified as one of the actors by a robust kernel multitask joint sparse representation and classification method; and 4) the conditional random field model with consideration of the constraints between face tracks is introduced to enhance the final labeling. Finally, the assigned actor name of a face track is then mapped to the character name based on the cast again. Besides face naming, we further apply the proposed method to spotlight the summarization of a particular actor in his/her movies. We conduct extensive experiments and empirical evaluations on several feature-length movies to demonstrate the satisfying performance of our method.**

***Index Terms*— Cast analysis, character identification, conditional random field (CRF), face recognition, multitask learning.**

I. Introduction

With rapid advances in digital technologies, there has been profound development in videos, especially the feature movies. In order to feasibly browse and index these movies, it is very crucial and urgent to provide efficient and effective techniques for movie content analysis and understanding. Automatic character identification is one of the most important techniques to deal with the problem, since character identification is to identify the faces of the characters in a movie and label them with their corresponding names. In a feature-length movie, the characters are often the most important contents to be indexed, and thus, the character identification becomes a critical step in film semantic analysis.

As has been noted in [1]–[3], although very intuitive to humans, automatic character identification is still tremendously challenging due to: 1) the lack and ambiguity of available annotations; 2) many other factors, such as pose, light, and expression, influence the way a face appears; and 3) when there are many uncontrolled data quality factors, such as low resolution, occlusion, nonrigid deformation, large motion, and complex background, which make the results of face detection and tracking unreliable for most image-based recognition, the situation is even worse in movies.

In order to deal with these challenges, in this paper, we present a novel cast analysis and an image retrieval-based approach for automatically naming the faces of the characters in a movie. We find that the cast of a film, which typically contains the names of actors, characters, and their (one-to-one) correspondence, is always available, and the Internet also provides a vast of information about actors. Accordingly, to deal with the first challenge, we propose to do a matching between the faces detected from the movie and the face images in the gallery set, which have been searched from the Web. The assigned actor name of a face is then mapped to the character name by the actor-character correspondence. For the second challenge, when each face is detected and tracked, a multitask joint sparse representation and classification (MTJSRC) is used to accurately recognize the face tracks based on the gallery face set. In addition, the kernel-view MTJSRC (KMTJSRC) has even achieved more robust performance. In order to deal with the third challenge, we introduce the conditional random field (CRF) model, which considers the constraints between those neighboring tracks, to enhance the final recognition and labeling results. This paper is different from the state-of-the-art name-to-face methods [1], [2], [4], where subtitle and/or scripts are required. Based on the results of character identification, a further application to generate spotlights summarization and digestion of a particular actor in many of his/her movies is also presented.

A. Related Work

The task of associating faces with names in a movie or a TV program is typically accomplished by combining multiple

Manuscript received July 29, 2014; revised October 23, 2014, January 5, 2015, and March 16, 2015; accepted April 26, 2015. Date of publication December 1, 2015; date of current version December 2, 2016. This work was supported in part by the National Natural Science Foundation of China under Grant 61401023, in part by the Funds for Creative Research Groups of China under Grant 61421061, in part by the Specialized Research Fund for the Doctoral Program of Higher Education under Grant 20120005130002, and in part by the Cosponsored Project of Beijing Committee of Education. This paper was recommended by Associate Editor J. Zhang. *(Guangyu Gao and Mengdi Xu equally contributed to this work.)*

G. Gao is with the School of Software, Beijing Institute of Technology, Beijing 100081, China (e-mail: guangyu.ryan@gmail.com).

M. Xu is with the Agency for Science, Technology and Research, Singapore 138632 (e-mail: mengdi.xu@gmail.com).

J. Shen is with the School of Information, Singapore Management University, Singapore 178902 (e-mail: jlshen@smu.edu.sg).

H. Ma is with the School of Computer, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: mhd@bupt.edu.cn).

S. Yan is with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117576 (e-mail: eleyans@nus.edu.sg).

sources of information, e.g., image, video, and text, under little or even no manual intervention. In the early stage, the most similar application is identifying faces in news videos [5]–[7], especially recognizing the announcers or specific characters (politicians or star actors). In news videos, the labeled names are always available in captions or transcripts, and the appearances of these people are also very clear and distinct. However, in movies or TV series, the names of characters are not always available, and the appearances of characters vary in different conditions, which make it hard to detect, track, and recognize these characters.

Over the past two decades, extensive research efforts have been actively concentrated on this task in movies or TV series. Since we need to assign the character names to faces or bodies in videos, the set of names is necessary in advance. According to the utilized contents or clues for these names, the previous work can be roughly classified into two groups.

Group 1 studies the labeling task for character recognition by utilizing manually labeled visual or audio data as the training data set. In this group, the supervised learning or semisupervised learning methods are used; namely, the researchers collect several samples as training data to generate the recognition model, and the labeling information is used as the final labeling text. Arandjelovic and Zisserman [1] used the face image as the query to retrieve particular characters. Affine warping and illumination correcting were utilized to alleviate the effects of pose and illumination variations. In [8], a multicue approach combining facial features and speaker voice models was proposed for major cast detection. In [9], we also proposed a semisupervised learning strategy to address celebrity identification with collected celebrity data. In addition, there are several methods using audio clues or both audio and vision clues, such as [10]–[12]. However, these approaches cannot automatically assign real names to the characters. Therefore, most of the researchers use the manually labeled training data. For example, Tapaswi *et al.* [13] presented a probabilistic method for identifying the characters in TV series or movies, and the face and speaker models were trained on Episodes 4–6 with manual labeling. Compared with a TV series which may consist of many seasons and episodes, there might be insufficient data used for training in a movie, which has only one episode with a duration of $\sim$2 h.

Group 2 handles the problem of assigning real names to the characters by using the textual sources, such as scripts and captions [2], [14], [15]. Guillaumin *et al.* [16] presented the methods for face recognition using a collection of images with captions, especially for news videos. Everingham *et al.* [2] proposed to employ readily available textual sources, the film script and subtitle, for text video alignment and, thus, obtained certain annotated face exemplars. The rest of the faces were then classified into these annotated exemplars. Their approach was also followed in [17] for human action annotation. However, in the approach [2], the subtitle text and time-stamps were extracted by optical character recognition, which required extra computation cost on spelling error correction and text verification. In addition, Zhang *et al.* [4] investigated the problem of identifying characters in feature-length films using video and film scripts with global face-name matching. Bojanowski *et al.* [18] learned a joint model of actors and actions in movies using weak supervision provided scripts. In fact, except for the extra errors and computation cost, for some movies, the scripts cannot be found easily or may be quite different from subtitles.

The advantage of the second group is that the methods can assign real and accurate names to those visual or audio samples. However, the script is not always available, and sometimes, new errors can be introduced due to unguaranteed scripts to face alignment. In order to include both advantages of the two groups, in this paper, we aim to develop an efficient and accurate name-to-face method with a flexible gallery set but no costly textual analysis. Our approach can be considered as the combination of both advantages of the two groups. We search the gallery data from the Google image search, and unlike methods in Group 1, our approach does not need data training, and instead, it uses the KMTJSRC directly to recognize face tracks. In order to get the real names, we use the cast list, which is always available on the Web, since it is published with the movie, instead of the scripts used in Group 2.

In addition, the above-mentioned previous methods always take each face image or face track for recognition individually. They are strictly limited to the characters' face appearance, including face size, angle, resolution, and so on. Furthermore, since the identification is performed for individual track, constraints that the same person cannot appear twice in one frame and faces in continuous shots are less likely to be the same character cannot be integrated. Thus, there have been some methods that consider these constraints with the probability generated models [13], [19], [20]. For example, Anguelov *et al.* [19] proposed a method to recognize faces using the Markov random field (MRF) model on photo albums, which performed recognition primarily based on the face, and incorporated clothing features from a region below the face.

Meanwhile, Lu *et al.* [20] proposed to identify players in broadcast sports videos using CRF model. In their work, human region in each frame is treated as a CRF node, and the constraints that human region belongs to the same tracklet and the same player should not appear more than once in one frame are considered. However, their approach may be robust for player identification in broadcast identification, but it cannot be flexibly applied to movie/TV series that have more complicated backgrounds, personal appearance, and motion. Furthermore, since each human region is denoted by a CRF node, the whole CRF model will be very sophisticated and it is difficult to do inference in movie videos with this model. Therefore, the random field model based on track level will be more efficient and robust, and Tapaswi *et al.* [13] modeled each TV episode as an MRF, integrating information of face, clothing, speaker, and contextual constraints in a probability manner of tracklet level. Nevertheless, clothing and speaking features are not so reliable in feature movies, and the whole model's accuracy utterly relies on the performance of face recognition (i.e., clothing needs be labeled by face recognition). Besides, Bäuml *et al.* [15] integrated weakly supervised faces from subtitle, unlabeled faces, and constraints together into a common learning framework.

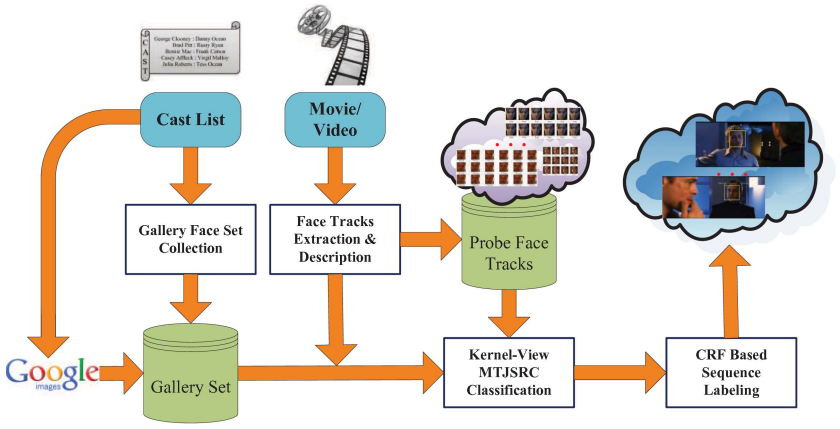

Fig. 1. Cast2Face framework diagram with four components: 1) gallery face set collection; 2) face tracks extraction and description; 3) KMTJSRC classification; and 4) CRF-based sequence labeling.

Therefore, besides considering the cast information and using the robust KMTJSRC recognition algorithm, another contribution of this paper is adopting the CRFs to model the constraints among face tracks. In general, the higher a character's name ranks in the cast list, the more frequently the character appears in the movie. Therefore, a prior probability is assigned to each character in advance. Then, the final face labeling is obtained with the CRF model based on both the prior probabilities and the robust recognition in the individual face track with KMTJSRC.

B. Outline of Our Approach

The Cast2Face method we propose is a novel framework for labeling the faces of the characters in a movie with cast. Our method comprises four components, as shown in Fig. 1.

1) Gallery face set collection with cast analysis and Web image search. Most of the previous methods use supervised or semisupervised training data, which are manually labeled or prepared to train the learning model. Unlike them, by using the textual source of the cast, our approach collects the gallery face data from the Google image search accurately and automatically. The collected gallery set not only contains sufficient face features, but also can be obtained efficiently.
2) Probe face tracks extraction and description using the state-of-the-art face detection and tracking algorithms to generate the face tracks. This step helps to obtain sufficient probe faces efficiently. After that, a robust face feature description method, which uses the scale-invariant feature transform (SIFT) descriptors, is introduced to more robustly represent each face track.
3) Face tracks identification using a robust KMTJSRC. We address the computation of joint SR of visual signals across multiple kernel-based representations, using the form of kernel matrices to represent each probe face track with the gallery set. Then, the recognition is finished by choosing the character name with the biggest ℓ_1 distance in the weight parameters.
4) CRF model-based tracks sequence labeling considering constraints among face tracks. Unlike the real-time face recognition, faces in movies are always with various angles, resolutions, and expressions; thus, face recognition directly performed on these faces is always with unsatisfactory accuracy. However, there are many constraints in these face tracks considered as a time sequence. Therefore, we consider the CRF model, which considers context information, since an ordinary classifier predicts a label for a single sample without regard to neighboring samples. By applying the CRF model on face tracks sequence labeling and minimizing the energy function, we get more robust labeling performance in terms of the initial recognition of KMTJSRC.

Compared with previous studies on name-to-face studies, the main contributions of this paper include the following.

1) To the best of our knowledge, Cast2Face proposed in this paper as well as its conference version [21] is the first work combining the character identification with the cast analysis and Web image retrieval.
2) A robust multitask joint SR method and the KMTJSRC are developed to classify each face track without training on a possibly contaminated gallery set.
3) The prior probability is introduced based on the character name order in the cast list, and a CRF model is used to relabel the whole face track more efficiently and effectively with the consideration of the neighboring constraints.
4) We design a novel application of our method to automatically generate the spotlights summarization of a particular actor in many of his/her movies.

More visual details can be seen in Fig. 2, which shows the working mechanism of our proposed Cast2Face method.

II. Cast2Face: Assigning Character Names Onto Faces

A. Cast-Based Web Image Search and Gallery Generation

The gallery data set and the real name for the final labeling are very important for the character identification. We can employ readily available textual annotation for TV and movie

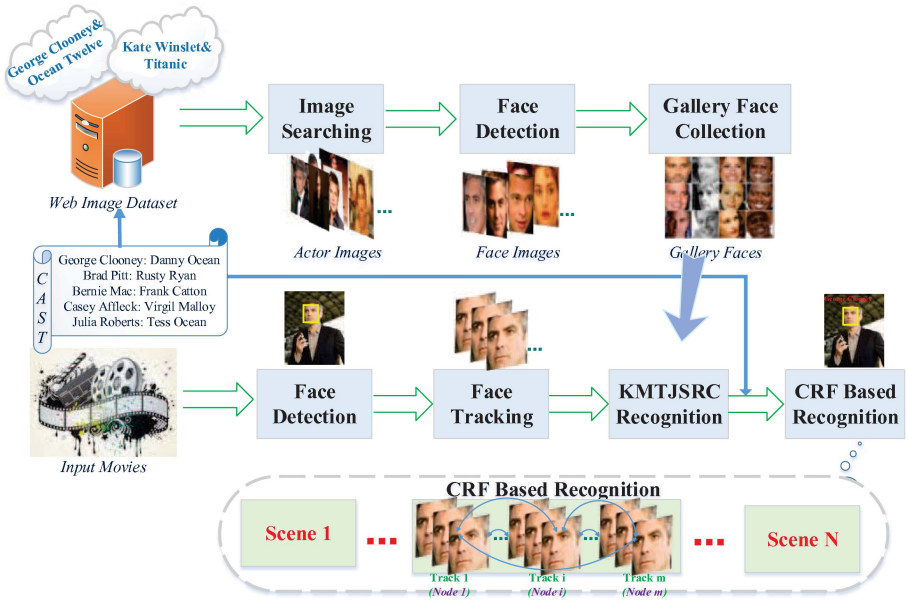

Fig. 2. Scheme illustration of the Cast2Face approach.

Fig. 3. Some exemplar faces of the star actor George Clooney in the gallery set generated by Web image search and face detection.

footage, in the forms of subtitles and transcripts, to automatically assign the correct name to each face image. However, it is feasible only when the subtitles or transcripts are available, which is sometimes not the case for movies, newly released movies particularly. Another form of textual information, the cast list, is always effective, since it directly keeps company with the original movie or TV series. Therefore, in order to associate names with characters detected in a movie, we use the movie cast list which is always available on the Web [such as the Internet movie database (IMDb)][1] or at the end of the movie. We restrict the names within the set of character names in the cast.

For each character, by using the corresponding actor's name as the keyword, we can retrieve from the Google image search and get a set of images. However, for many old but classic movies, the actors look older now than they did when the movies were produced, and the Google image search returns the actors's recent images, which may be very different from their appearances in these old movies. In order to deal with this problem, we combine the actor name with the movie name together as the keywords in the Google image search. Finally, we observe that the top hundreds of the returned images belong to the actor and the character in the movie with high precision. We then employ a frontal face cascade detector [22] included in OpenCV2.0[2] to detect and crop faces from the downloaded images. In this way, the gallery set is established and then used for labeling the faces of the characters extracted from the movie. Take the movie Ocean's Twelve (OT) as an example. Some gallery face images for the key actor, such as George Clooney, are shown in Fig. 3. Note that a few incorrect faces are inevitably introduced in the gallery set due to image retrieval and face detection errors. As we shall see later in experiments, our face identification method is quite robust for such noises contained in the gallery set.

B. Probe Face Tracks Extraction and Description

The frontal face cascade detector of [22] is very robust for frontal face detection, even with various face sizes. However, this method is sensitive to some nonfrontal faces, namely, profile faces. Yet, the automatic character identification, including face tracking, needs to start with a successful face detection, no matter frontal or profile face. Therefore, if the frontal face cascade detector cannot detect any faces in a frame, another robust face detector, OKAO face detector 3, is then used to detect profile faces with 30° toward left or right.

[1] http://www.imdb.com

[2] http://sourceforge.net/projects/opencvlibrary

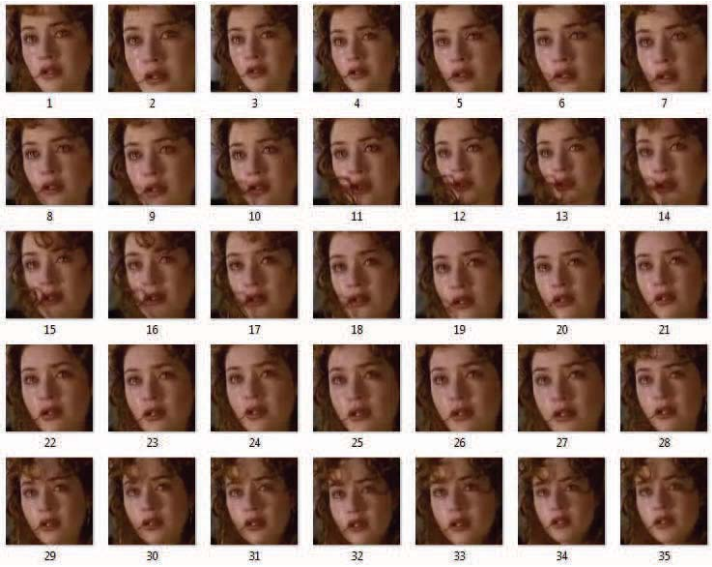

Fig. 4. Examples of the first 35 faces of a face track for Rose in TT, and the number below the image is the index of the face in the face track.

Fig. 5. Examples of detected face with facial feature points.

Actually, a typical movie may contain tens of thousands of detected faces. However, these faces merely arise from a few hundred tracks of a particular character. Therefore, it is feasible to discover the correspondences between faces and reduce the volume of the data that needs to be processed. Furthermore, stronger appearance models can be built for each character, since a face track provides multiple examples of the character's appearance. To obtain face tracks, a robust foreground correspondence tracker [23] is applied for each shot.

Here, shot changes are automatically detected using our previous accelerating shot boundary detection method [24]. Concretely, for each frame, the closer the pixels are to the center of the frame, the more important the pixels are. Thus, the focus region in each frame is defined. Furthermore, by using a skip interval of 40 frames, not only the detection process is sped up, but also more gradual transitions can be found. Besides, the camera and object caused motions are detected as the candidate shot boundaries, and using corner distribution analysis, all of them are excluded as false boundaries. Note that these camera or object caused motions are also used in key frames extraction.

Using the tracking algorithm in [23], with the assumption that the target face region can be represented by a set of superpixels without significantly destroying the boundaries between the target and the background, we model the prior knowledge regarding the target and the background appearance by

$$y_t(r) = \begin{cases} 1, & \text{if } \mathrm{sp}(t,r) \in \text{target} \\ -1, & \text{if } \mathrm{sp}(t,r) \in \text{background.} \end{cases} \quad (1)$$

Here, $\mathrm{sp}(t,r)$ denotes the rth superpixel in the tth frame, and $y_t(r)$ denotes its corresponding label. A robust superpixel-based discriminative appearance model is generated based on four factors: 1) cluster confidences; 2) cluster centers; 3) cluster radius; and 4) cluster members. This discriminative appearance model facilitates a tracker to discriminate the face region and the background with midlevel cues. After that, the target–background confidence map is used to formulate the tracking task, and the best candidate is obtained by the maximum *a posteriori* estimates. With the superpixels tracking, we collect faces belonging to tracks efficiently and accurately, and more details about the tracking algorithm can be seen in [23]. However, short tracks which are often introduced by false positive detections are discarded, and an example of the final face tracks is shown in Fig. 4.

To extract the face features and construct the representations, a part-based descriptor extracted around local facial features [2] is utilized. Here, we first use a generative model [1] to locate the nine facial key-points in the detected face region, including the left and right corners of each eye, the two nostrils and the tip of the nose, and the left and right corners of the mouth. Then, we extract the 128-D SIFT [25] descriptor from each key-point and directly concatenate them together to form our final face descriptor with dimensionality 1152. Fig. 5 shows some selected faces with facial feature points marked in our approach.

1) Kernel-View Multitask Joint Sparse Representation and Classification: Given a set of retrieved gallery face images and the extracted probe face tracks, we present in this section

a simple yet efficient algorithm for face track identification. Each unlabeled face track is simply represented as a set of feature vectors extracted from all images in the track. One simple method for identification, as conducted in [2], is to directly calculate the feature distance between a probe face track and the labeled exemplar faces, and then assign probe face track to the nearest neighborhood. Another feasible method is to classify each image in the track independently via, e.g., SR classification [26], and then assign the face track to the subject that achieves the highest frequency.

In this paper, by viewing the identification of each image in a probe face track as a task, the face track identification can be naturally casted to a multitask face recognition problem. This motivates us to apply the multitask joint SR model [27] to face track classification. The key advantage of multitask learning lies in that it can make use of complementary information contained in different subtasks. In addition, we also extend the multitask learning into kernel-view, which is more competitive than the state-of-the-art multiple kernel learning methods for face tracks recognition.

2) Multitask Joint Sparse Representation-Based Recognition: Suppose we have a set of exemplar faces with M subjects. Here, a subject means a person, which refers to a set of the same person's faces. Denote $X = [X_1, \ldots, X_M]$ as the feature matrix, and $X_m \in \mathbb{R}^{d \times p_m}$ is associated with the mth subject, where d is the dimensionality of features, and $p = \sum_{m=1}^{M} p_m$ means the total number of training samples. Here, we consider a supervised L-task linear representation problem as follows:

$$\boldsymbol{y}^l = \sum_{m=1}^{M} X_m \omega_m^l + \varepsilon^l, \quad l = 1, \ldots, L \tag{2}$$

where $\boldsymbol{y} = \boldsymbol{y}^l$ means a face track and $\boldsymbol{y}^l$ as a task is the lth face image in this track. Meanwhile, $\omega_m^l \in \mathbb{R}^{p_m}$ is a reconstruction coefficient vector associated with the mth subject, and ε^l is the residual term. Denote $\omega^l = [(\omega_1^l)^T, \ldots, (\omega_M^l)^T]^T$ the representation coefficients for the probe image $\boldsymbol{y}^l$, and $\omega_m = [\omega_m^1, \ldots, \omega_m^L]$ the representation coefficients from the mth subject across different tasks (faces). Furthermore, we denote $W = [\omega_m^l]$. Therefore, our proposed multitask joint SR model is formulated as the solution to the following problem of multitask least square regressions with $\ell_{1,2}$ mixed-norm regularization:

$$\min_W F(W) = \frac{1}{2} \sum_{l=1}^{L} \left\| \boldsymbol{y}^l - \sum_{m=1}^{M} X_m \omega_m^l \right\|_2^2 + \lambda \sum_{m=1}^{M} \| \omega_m \|_2 . \tag{3}$$

Here, we use the popular optimization method of accelerated proximal gradient (APG) [28], [29] to efficiently solve (3) with a fast convergence rate guaranteed. The APG is composed of a weight matrix sequence $\hat{W}^t = [\omega_m^{l,t}]_{t \geq 1}$, and an aggregation matrix sequence $\hat{V}^t = [v_m^{l,t}]_{t \geq 1}$. The $\hat{W}^{t+1}$ is updated according to the result in [30]

$$\hat{\omega}^{l,t+1} = \hat{v}^t - \eta \nabla^{l,t}, \quad l = 1, \ldots, L \tag{4}$$

$$\hat{\omega}_m^{t+1} = \left[1 - \frac{\lambda \eta}{\|\hat{\omega}_m^{t+1}\|_2} \right]_+, \quad m = 1, \ldots, M. \tag{5}$$

Here, $\nabla^{l,t} = -(X^l)^T y^l + (X^l)^T X^l \hat{v}^{l,t}$, η is the step size parameter, and $[\bullet]_+ = \max(\bullet, 0)$. In addition, the matrix

$$\hat{V}^{t+1} = \hat{W}^{t+1} + \frac{\alpha_{t+1}(1 - \alpha_t)}{\alpha_t} (\hat{W}^{t+1} - \hat{W}^t) \tag{6}$$

where α_t is directly set as $2/(t+2)$ [28] in our approach.

When the optimal $\hat{W} = [\hat{\omega}_m^l]$ is obtained, a probe image y^l can be approximated as $\hat{y}^l = X_m \hat{\omega}_m^l$. For classification, the decision is ruled in favor of the subject with the lowest total reconstruction error accumulated over all the L tasks

$$m^* = \arg\max_m \sum_{l=1}^{L} \left\| y^l - X_m \hat{\omega}_m^l \right\|_2^2. \tag{7}$$

We call model (3) along with classification rule (7) the MTJSRC in this paper.

3) Kernel-View Extensions Recognition: Heretofore, the face track identification is feasibly realized by the MTJSRC algorithm for SR and classification. In order to combine multiple feature kernels for face track recognition, we extend the MTJSRC algorithm to the kernel version as described in [31].

For a reproducing kernel Hilbert space, the kernel trick is to use a nonlinear function $\phi^l(x_i)^T \phi^l(x_j) = g^l(x_i, x_j)$ for some given kernel function g^k. Let $G^l = \phi^l(X^l)^T \phi^l(X^l)$ be the training kernel matrix associated with the lth modality of the feature and $h^l = \phi^l(X^l)\phi^l(y^l)$ be the test kernel vector associated with the lth modality. In our approach, the simple and available kernel matrix is constructed by directly using vector h^l and the column of each kernel matrix G^k as the extracted new features. In this new space, the original multitask least square regressions with $\ell_{1,2}$ mixed-norm regularization problem can be written as

$$\min_W F(W) = \frac{1}{2} \sum_{l=1}^{L} \left\| h^k - \sum_{m=1}^{M} G_m^l \omega_m^l \right\|_2^2 + \lambda \sum_{m=1}^{M} \|\omega_m\|_2. \tag{8}$$

Actually, in the experiment, the kernel matrices are computed as $\exp(-\chi^2(x, x')\mu)$, and μ is set to be the mean value of the pairwise χ^2 distance on the training set.

III. Face Track Sequences Labeling With CRF

After the MTJSRC algorithm, each face track has been labeled with one character name. However, the whole movie or the movie clip totally is seen as a story consisting of several shots or scenes, and there are plenty of correlations between these shots, including semantic correlation and scenario correlation. That is to say, there are also quantities of correlations in the audio and visual appearance of those semantic contents and scenarios. Thus, besides using the KMTJSRC-based face recognition on each track, we also consider these correlations among neighboring tracks using the probabilistic model of CRFs.

A. CRF Model for Sequence Labeling

As mentioned in [32], naive Bayes model, hidden Markov model (HMM), and maximum entropy models (MEMs) are among the most well-known probabilistic models. Naive Bayes

model is a very basic and simple model considering the Bayes decision, and HMM can be viewed as the sequence version of naive Bayes model: instead of single independent decisions, the HMM models a linear sequence of decisions. Similarly, CRFs can be seen as the sequence version of MEMs, and both of them are discriminative models. For HMM, there is a disadvantage of strong independence assumptions between the observation variables, and this reduces the accuracy of the model. CRF was proposed in [33], and there is no assumption on the dependence among observation variables that needs to be made for CRF.

Compared with naive Bayes model and HMM that are proposed to calculate the joint probability $P(X, Y)$ of the class and observation variable, and have the disadvantage of computational complexity, the CRF model which aims to calculate the conditional probability $P(Y|X)$, is more reasonable. The probability graph model of CRF is an undirected graph, and it also considers the dependence of adjacent nodes. Moreover, the factorization of this graph is performed in such a way that conditionally independent nodes do not appear within the same factor, which means that they belong to different cliques.

From the Hammersley Clifford theorem, the general model formulation $p(\mathbf{Y}\|\mathbf{O})$ of CRFs is derived

$$p(\mathbf{Y}\|\mathbf{O}) = \frac{1}{Z}\exp\left(-\sum_{c\in\mathcal{C}}\Psi_c(Y_c)\right). \tag{9}$$

Here, $\mathbf{O}$ is the observation value, Y is the labeling vector which refers to a sequence, and $\mathcal{C}$ is the maximum clique. Each factor $\Psi_c(Y_c)$ corresponds to a potential function over the variables $(Y_c = y_i, i \in c)$ constituting the clique c, and $\mathcal{C}$ is the set of all cliques [33]. The normalization constant Z is known as the partition function

$$Z = \sum_{\mathbf{Y}'}\prod_{c\in\mathcal{C}}\Psi(\mathbf{Y}'\|O). \tag{10}$$

Meanwhile, the Hammersley Clifford theorem introduces the relationship between Gibbs distribution and MRF. Accordingly, the Gibbs distribution is always used to get the solution for MRF. In general, the CRF in essence is an MRF with given observations, and thus, we can solve the CRF-based labeling problem by minimizing the energy functions. The corresponding Gibbs energy is defined as

$$E(\mathbf{Y}) = -\log p(\mathbf{Y}\|\mathbf{O}) - \log Z = \sum_{c\in\mathcal{C}}\Psi(\mathbf{Y}_c\|\mathbf{O}). \tag{11}$$

The maximum *posteriori* labeling $\mathbf{Y}^*$ of the random field is defined as

$$\mathbf{Y}^* = \arg\max_{\mathbf{Y}\in\mathcal{L}} p(\mathbf{Y}\|\mathbf{O}) = \arg\min_{\mathbf{Y}\in\mathcal{L}} E(\mathbf{Y}) \tag{12}$$

where $\mathcal{L}$ is the label set. To be concise, we drop the symbol $\mathbf{O}$, and just use $\Psi(\mathbf{Y}_c)$ to denote the potential functions of a CRF in the following sections.

B. Cast2Face Tracks Labeling Consistency

In our cast2face approach, when each track is seen as a node, and the correlations among neighboring tracks are considered as the property of conditional independence, the whole face tracks sequence can be modeled as a CRF model. That is to say, face tracks in a scene are viewed as a node sequence, and the CRF model is adopted to refine the labels for each node assigned with KMTJSRC. Therefore, the CRF model commonly used for cast2face track labeling is characterized by energy functions defined on unary and pairwise cliques as

$$E(\mathbf{y}) = \sum_{i\in\nu}\Psi_i(y_i) + \sum_{(i,j)\in\xi}\Psi_{ij}(y_i, y_j). \tag{13}$$

Here, ν corresponds to the set of all face tracks, while ξ is the set of all edges connecting the tracks $i, j \in \nu$. The edge set is commonly chosen to define neighborhood tracks, which means the face tracks belong to the same video scene. The labels constituting the label set $\mathcal{L}$ represent different characters. The random variable y_i denotes the labeling of tracks i in the tracks sequence. Every possible assignment of the random variable $\mathbf{y}$ (or the configuration of CRF) defines a tracks label scheme.

In order to solve the problem of minimizing a large class of energy functions, we adopt the method proposed in [34], which uses graph cuts to compute a local minimum even when very large moves are allowed. The unary potential Ψ_i of the CRF is defined as the negative log of the likelihood of a label assigned to the face track i. It can be computed from the nine points-based face features and the KMTJSRC classification for face recognition. In addition, there is also valuable prior knowledge in the cast, which can be used for more effective recognition. In general, in the cast list, the first or second character is always the starring actor/actress, and usually the higher the character name is on the cast list, the more frequent the character's occurrence is. Therefore, we analyze the raw relations between the character's frequency of occurrence in a video and its order in the cast list, and then assign an empirical prior probability to each character. Finally, the unary potential can be written as

$$\Psi_i(y_i) = -\log(p(y_i)p(o_i\|y_i)). \tag{14}$$

Here, $p(y_i)$ is the prior probability from the character name order in the cast list, and $p(o_i\|y_i)$ is the probability of the observed face features for the specific characters, namely, the recognized probability of the character label y_i using the KMTJSRC algorithm.

Although we have used the valuable prior knowledge in the cast list, employing single face track features alone is not very discriminative and robust for accurate labeling. However, this problem can be solved by using the sophisticated potential functions based on single character's face features and also the spatial and temporal relationships between different tracks in a scene. Thus, the pairwise terms Ψ_{ij} of the CRF take the form of a contrast sensitive Potts model

$$\Psi_{ij}(y_i, y_j) = \begin{cases} 0, & \text{if } \ y_i = y_j \\ \lambda(i,j), & \text{otherwise.} \end{cases} \tag{15}$$

Here, the function $\lambda(i, j)$ is an edge feature based on the difference in the feature distance of neighboring tracks. It is typically defined as: $\lambda(i, j) = \theta\exp(-\|f_i - f_j\|^2)$, where f_i is the feature vector extracted in Section II-B. In addition, although nodes in the same scene are considered neighbors,

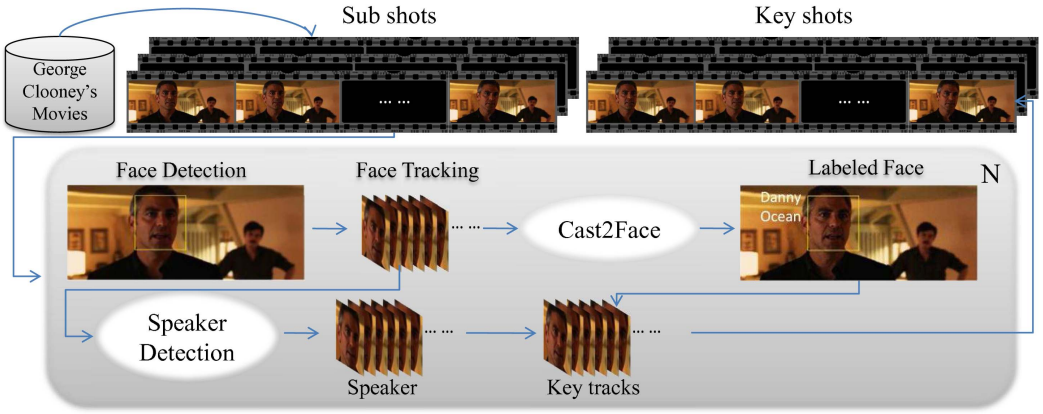

Fig. 6. Framework of actor-specific spotlights summarization.

TABLE I
SUMMARY OF TEST MOVIES

Movies	Duration(min)	Resolution	#Shot	Genres
Titanic	195	720 × 304	1550	Drama&Romance
Ocean's Twelve	120	1280 × 528	962	Crime&Thriller
Twilight	122	640 × 272	1205	Drama&Fantasy&Romance
The Bigbang 1	21	624 × 352	374	Comedy
The Bigbang 2	21	624 × 352	385	Comedy

nodes or face tracks with the same shot number should not be assigned with the same label. Therefore, an additional cost is added when two face tracks belong to the same shot and have the same labels, and this cost is directly set to a maximum number in the experiment.

IV. Application: Actor-Specific Spotlights Summarization

So far, we have assigned a character name to each face track, which means all the detected faces are successfully recognized. Based on the results of character identification, there are many applications, such as character-specific movie retrieval, personalized video summarization, intelligent playback and video semantic mining, and so on. In this section, we demonstrate an actor-specific spotlights summarization system using the Cast2Face, on which the users can input the actor names to search and digest the film. Fig. 6 shows the working scheme of the proposed actor-specific movie summarization method.

First, an accelerating shot boundary detection method is applied to divide the movie into several shots, and each shot is about 1∼2-min long. Second, the face detection and tracking processing are applied, and after the identification of all the detected face tracks in these shots, we rank the tracks associated with a particular actor according to the reconstruction error calculated in rule (7). The video shots containing the top ten tracks are then taken as the candidate spotlight videos. Third, we make further restriction that the actor should be speaking in the summarized video. The speaker is identified by using the visual information, i.e., for visual information, finding face detections with significant lip motion [2]. We then combine the obtained key shots together as a digested movie. Finally, by using the character name or actor name as the query entry, the corresponding actor's spotlights clips are presented to the user.

V. Experimental Results

In this section, we evaluate the effectiveness and the efficiency of the Cast2Face approach along with its application in spotlights summarization on several feature-length movies. All the movie casts are obtained from the IMDb. Ground truth names for face detection are produced manually. Note that the ground truth is only established for the face detections produced by the frontal face detector [22] or the small angle profile detector by OKAO. The results reported here are, therefore, related to the proportion of occurrences of a character detected by the state-of-the-art face detectors.

A. Data Sets

As a quantitative study of Cast2Face, we evaluate in this experiment the accuracy of our proposed KMTJSRC method as well as that of CRF sequence labeling for character identification. We report the corresponding results on three films OT (2004), Titanic (TT) (1997), and Twilight (TL) (2008) and two episodes of The Big Bang Theory, Season 3, Episodes 2 and 4 (The BigBang1 and BigBang2). These movies are characterized by several familiar scenarios and famous actors, as well as significant and visible face appearances. The resolution of these movies varies from 624 × 352 to high definition of 1280 × 720, and the frame rate is about 25 ∼ 30 frames/s. Details of these movies are shown in Table I. Besides, the data set used in [13], i.e., the

TABLE II

QUANTITATIVE RESULTS ON EVALUATION OF CAST2FACE METHOD

Movie Name	Actor Name	Gallery Set Size	Probe Face Tracks	Total Faces	Accuracy(%)			
					NN	SR	EASR	MTJSRC
Ocean's Twelve	George Clooney	90	27	1,477	63.4	63.1	66.7	**73.3**
	Julia Roberts	94	28	1,677	30.0	41.9	53.6	**82.1**
	Matt Damon	96	32	1,310	72.6	73.8	74.9	**87.5**
	Brad Pitt	90	13	522	43.8	72.6	76.8	**92.3**
Titanic	Leonardo DiCaprio	43	19	429	37.0	69.6	**73.7**	68.4
	Kate Winslet	57	41	1,015	62.4	72.2	71.8	**82.9**
	Billy Zane	47	18	386	57.7	80.0	77.7	**88.8**
Twilight	Kristen Stewart	96	122	3,321	82.4	84.5	83.6	**89.3**
	Robert Pattinson	92	98	2,612	70.8	82.8	82.6	**89.8**
	Taylor Lautner	47	6	86	16.6	**53.3**	50.0	50.0

TABLE III

PERFORMANCE COMPARISON WITH DIFFERENT GALLERY COLLECTIONS

Schemes / Movies	MA+ANS	NMA+ANS	MA+AMNS	NMA+AMNS
Titanic	82.6%	77.1%	83.4%	83.4%
Ocean Twelve	80.2%	80.2%	81.7%	81.7%
The BigBang 2	91.3%	91.7%	92.4%	92.4%

first season of The Big Bang Theory, is also involved to evaluate the performance of our approach (using KMTJSRC&CRF).

In our experiment, the test movies are first quickly and accurately segmented into shots and the face tracks are extracted. In addition, the cast lists of these movies are downloaded and stored as one-to-one correspondence of the character name and the actor name from IMDb. Then, the actor name combined with the movie name is used to search images from the Google image data set, and gallery sets are collected from the Google image search. The sizes of the constructed gallery sets for some selected actors are listed in Table II. In addition, the prior probability is assigned to these characters in accordance with their orders in the cast list. Finally, the KMTJSRC algorithm and CRF model-based sequence labeling are applied to recognize the face tracks. After that, we compare the performance of our approach with several other algorithms, and we also do comparisons among the MTJSRC algorithm, KMTJSRC algorithm, and the CRF-based enhanced sequence labeling for character identification.

B. Performance of Automatically Collected Gallery Set

One of the greatest contributions of this paper is that our approach does not need any training images. We collect the gallery set from the Internet image search using the actor's name from the cast list. In addition, as shown previously, a few incorrect faces are inevitably introduced in the gallery set due to image retrieval and face detection errors, but our approach is quite robust for such noises by using the MTJSRC algorithms. Besides, in an image search engine, it is easy to obtain those makeup and aged faces. Therefore, we find that some of these makeup or aged faces are always contained in the gallery set which can be accurately recognized with the SR. In order to evaluate the performance of the automatically collected gallery set from the Google image search, we use three clips, respectively, from the movie of TT (23 min), OT (25 min), and The BigBang2 (18 min).

After that, the actor name combined with the movie name is input into the Google image search to get the gallery data set. Then, the face detection is applied on both the test movie clips and the gallery set. In addition, we use the tracking algorithm to get the probe face tracks. In order to assess the fault tolerance of our approach, we manually adjust the gallery set by removing those images without apparent actor faces and those with more than one face. Besides, we also consider assessing the effectiveness of using both the actor name and the movie name.

In order to evaluate the efficiency of our automatically collected gallery set, we perform the KMTJSRC-based recognition on galleries collected by different schemes, as shown in Table III. We evaluate the performance of four schemes, including accuracy measure on the association of Actor Name Searching or Actor&Movie Name Searching, and with or without manual adjustment (MA). Here, MA means choosing images with noticeable and big-sized faces as well as stage photos for the processed movie. We can see that our KMTJSRC algorithm-based approach can tolerate noises introduced by searching result outliers, since schemes with or without MA get nearly the same performance. Meanwhile, it can be seen that the manually adjusted gallery set performs 1%–3% better on the old movie TT. That is to say, we can add some human intelligence to adjust the search results, especially for old movies.

Nevertheless, the results are observably different for schemes using different query keywords. That is to say, combining the actor name and the movie name together can archive more robust performance. Some movies were filmed several years ago, in which the actors appearances might have changed as a result of aging. However, with the movie name constraint, the Google image search mainly returns stage photos from the specific movie or life photos in the same age of the actor.

C. Performance of Character Identification via MTJSRC and KMTJSRC

Two baseline methods are employed for comparison: 1) the nearest neighbor classifier used in [2], which directly

calculates the feature distance between a probe face track and the labeled exemplar faces, and then assigns the probe face track to the nearest neighborhood and 2) the SR classifier [26] that classifies each image in the track independently and then assigns the face track to the subject that most frequently occurs in this track. In addition, for SR algorithm in [26], we give some details about how to use it in our track level face recognition.

Suppose the matrix $X = \{X_m\}$ for the entire gallery set is the concatenation of the $p = \sum_{i=1}^{m} p_m$ training samples of all M subject classes. Denote $X_m = [v_{m,1}, v_{m,2}, \ldots, v_{m,p_m}] \in \mathbb{R}^{d \times p_m}$ as the mth subject samples. For a new (test) face track $\boldsymbol{y}$ with K face images, we first classify the kth face into the class $c_k \in \{1, \ldots, M\}$, and also define $C = [c_1, \ldots, c_K]$ as the class vector for the test face track. Then, we assign $c = \arg\max_m \|C - m\|_0$, which means the most frequently occurred subject class, as the final subject class for the test track. Meanwhile, the class label c_k of the kth face in the track is obtained as follows.

$\boldsymbol{y_k}$ is the kth face in the face track, and represented as

$$\boldsymbol{y_k} = X\boldsymbol{\alpha} + \boldsymbol{e} \tag{16}$$

where $\boldsymbol{\alpha} \in \mathbb{R}^p$ is the coefficient vector. Then, to get the informative vector, $\boldsymbol{\alpha} = [\alpha_1^T, \ldots, \alpha_M^T]^T$ is equivalent to the solution of the following ℓ_1-minimization problem:

$$\hat{\boldsymbol{\alpha}}_{\mathbf{1}} = \arg\min \|\alpha\|_1 \quad \text{s.t. } \boldsymbol{y_k} = X\boldsymbol{\alpha} + \boldsymbol{e}. \tag{17}$$

That is to solve the following problem:

$$\min_{\boldsymbol{\alpha}} F(\boldsymbol{\alpha}) = \frac{1}{2}\|\boldsymbol{y_k} - X\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1. \tag{18}$$

This problem can be solved in polynomial time by standard linear programming methods [35]. After that, we classify $\boldsymbol{y_k}$ to the subject class that minimizes the residual between $\boldsymbol{y_k}$ and $\hat{\boldsymbol{y_k}}_{\boldsymbol{m}}$

$$c_k = \arg\min_m \|\boldsymbol{y_k} - X_m\boldsymbol{\alpha_m}\|_2. \tag{19}$$

As aforementioned, there always exist a few incorrect faces in the gallery set, and thus, training-based methods, e.g., support vector machine (SVM) and subspace analysis, are not applicable in our setting. In contrast, our multitask linear representation-based method is quite robust for the condemnation, since the joint representation ability of noise images is lower compared with those good samples.

The evaluation results are listed in Table II, from which we can see that the MTJSRC significantly outperforms both baselines for 8 out of the 10 testing actors. For computational cost, the Cast2Face method is training free and the most expensive calculation lies in the testing phase, where a multitask regression problem [see (3)] is optimized. In our experiment, the APG algorithm converges at roughly 10∼20 rounds of iterations. The average running time is 0.31 s per probe face track. The parameter λ in (3) is set to 0.1 throughout our experiment.

In addition, we also construct a baseline method, which is named error accumulated SR (EASR), by assigning a face track to the class with the lowest total reconstruction error accumulated over all the faces in a track, and this is based on SR algorithm in (18) as follows:

$$\min_{\boldsymbol{\alpha}} F(\boldsymbol{\alpha}) = \frac{1}{2}\sum_{k=1}^{K} \|\boldsymbol{y_k} - X\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1. \tag{20}$$

Finally, as shown in Table II, we find that when considering the total reconstruction error accumulated over all the faces, the performance of EASR is better than the basic SR. However, the MTJSRC algorithm also improves results since the construction of the coefficient is more flexible especially with $\ell_{1,2}$.

So far, with the MTJSRC algorithm [21], we have obtained a robust recognition performance. Moreover, the KMTJSRC is more robust for character identification and achieves a more efficient performance. So as to evaluate the performance of the KMTJSRC algorithm, we illustrate the comparison results on those test movies in Fig. 8.

D. Enhanced Performance With CRF Model

We have obtained many face tracks corresponding to different characters, and also collected the recognition results for each face track with the KMTJSRC algorithm. In order to assign names to these tracks more accurately, we consider constraints between neighboring tracks, which are involved in the CRF model. To assess the efficiency of the CRF model-based enhanced labeling, we choose four clips from four movies (TT, OT, TL, and The Big Bang1), respectively, each of them with the length of about 20∼30 min, in total ∼2 h of video. Generally speaking, faces in the same scene will be more similar than the faces belonging to different scenes. Therefore, face tracks in the same scene are set as neighbors for each other, and the state-of-the-art scene segmentations methods [36]–[38] are also used to obtain the scene boundaries based on the shot segmentation results.

After face detection, tracking and KMTJSRC algorithm-based recognition, we assign labels to each face track with a different labeling cost. The unary potential or labeling cost is obtained using (14). Here, $p(x_i)$ is the prior probability which measures the character name order in the cast list and is empirically set as one of the values among $0.8, 0.8, 0.6, 0.6, 0.5$ according to its order in the list. However, the user always pays more attention to the staring characters in the front of the cast, and thus, we consider performing recognition on no more than five staring actors in a movie. Meanwhile, the observation probability $p(o_i|x_i)$ of the ith track labeled with x_i is calculated using the weights that measure the confidence of different tasks in a final decision in (7).

For classification, the decision is ruled in favor of the class with the lowest total reconstruction error accumulated over all the K tasks in the KMTJSRC algorithm. That is to say, only the class with the minimum error value is used to label the face tracks. However, if the lowest and the second lowest total reconstruction error are nearly the same, it may be incorrect to assign the face track with the class label of the minimum value. In addition, we need to calculate the data cost using the probability of each face track assigned with each label in the CRF model. In order to deal with this problem, we transfer these total reconstruction errors into probability

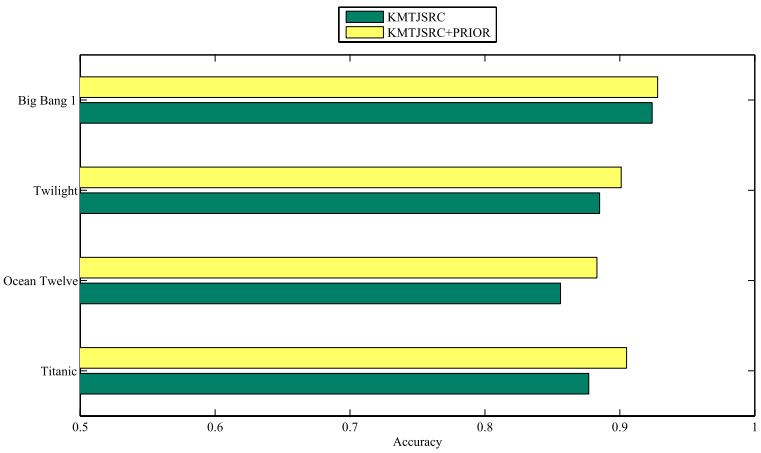

Fig. 7. Performance improved by prior probability.

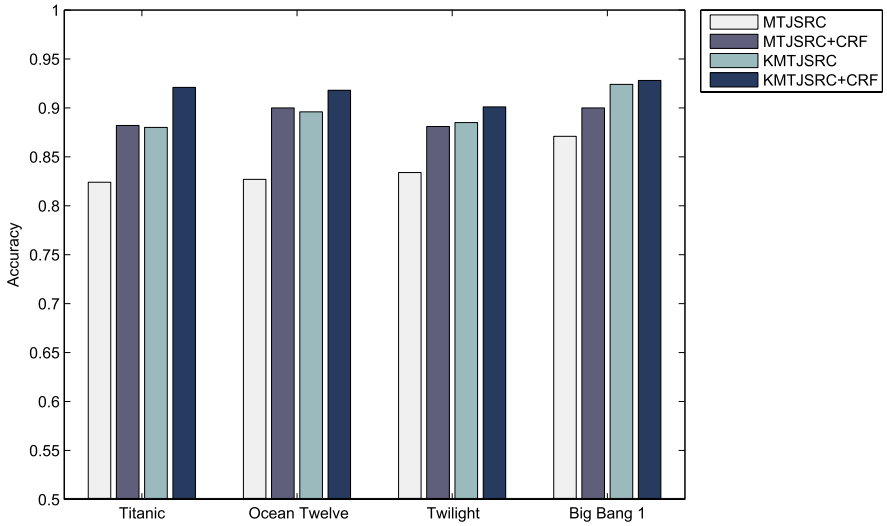

Fig. 8. Recognition accuracy (%) of different algorithm schemes on the probe set.

representation using the softmax active function.[3] The softmax activation function is a neural transfer function. In the field of reinforcement learning, a softmax function can be used to convert the values into action probabilities. The function used in our approach is

$$P(l_i) = \frac{\exp(-q(l_i)/\tau)}{\sum_{i=1}^{n} \exp(-q(l_i)/\tau)} \tag{21}$$

where the action value $q(l_i)$ corresponds to the expected reward of the following action l_i, namely, the total reconstruction error, and τ is called a temperature parameter (in allusion to chemical kinetics).

We first assess the efficiency of using the prior knowledge in the cast list. As shown in Fig. 7, the performance is improved with the prior probability, especially in the movie of TT. Actually, these improvements mainly come from the recognition of the third or fourth character. That is to say, the KMTJSRC recognition may label the first or second character with names, which is back in the cast. However, with the prior knowledge consideration, the top-listed character always has more probability to appear in the movie. Thus, we pull down the probability of false recognition of classifying the front characters' face with the back characters' name, especially when the faces are of small size or unclear appearance for feature extraction. After that, we also evaluate the performance with KMTJSRC and CRF model together, and the results of the comparisons between the two and the KMTJSRC are shown in Fig. 8. We also show some examples of the correction process for false alarm identifications using different recognition schemes in Fig. 9.

Finally, in order to validate the satisfactory performance of our approach using KMTJSTC&CRF, we also compare it to the performance obtained by using the MRF which integrates face recognition, clothing appearance, speaker recognition, and contextual constraints [13] as well as their follow-up work [15] in their data set of the first season of The Big Bang Theory. For reasonable comparison, we only extract the five main character names in the cast to test recognition, i.e., Sheldon Cooper, Leonard Hofstadter, Penny, Howard Wolowitz, and Raj Koothrappali. In [13], some identity tracks without faces

[3]http://en.wikipedia.org/wiki/Softmaxactivationfunction#citenote-4

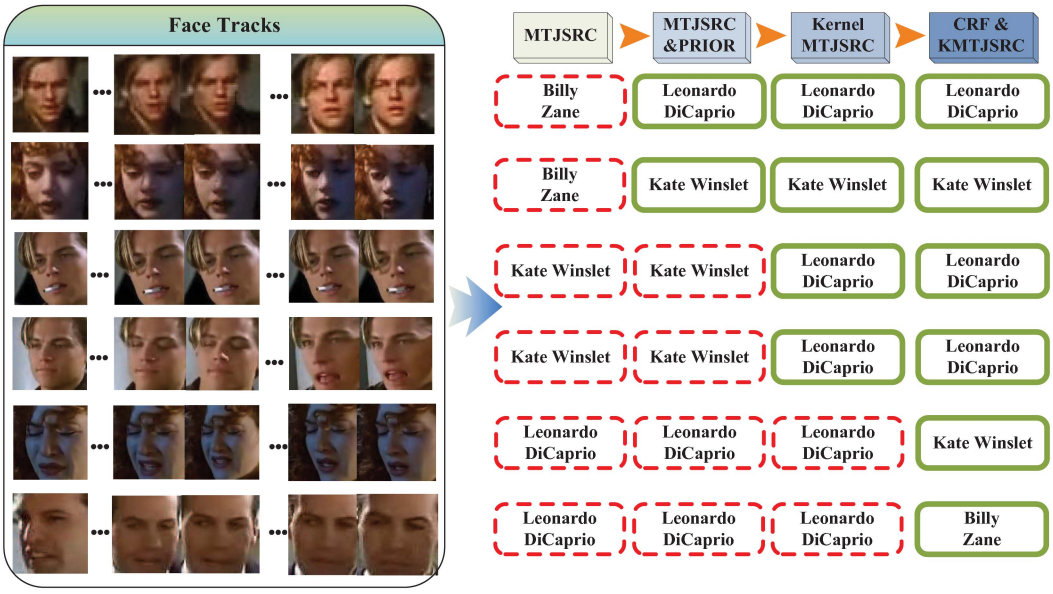

Fig. 9. Examples of false alarm correction via different algorithm schemes on the probe set. MTJSRC, MTJSRC&PRIOR, Kernel MTJSRC, and CRF&KMTJSRC are different schemes, and the rear scheme is extended on the front one. Dotted boxes: wrong recognition results. Solid boxes: accurate recognition results. We find that the rear one among the schemes always has more satisfactory results for character identification, i.e., many false labeling can be verified with the CRF&KMTJSRC scheme.

TABLE IV
COMPARISON OF OUR APPROACH, OUR APPROACH WITH DIRECT SPEAKER LABELING [13], [15]

Episode	E1	E2	E3	E4	E5	E6	Avg
FullModel in [13]	89.3%	84.9%	80.8%	86.7%	73.3%	84.1%	83.2%
Model in [15]+MRF	95.2%	94.2%	77.8%	79.4%	79.9%	75.9%	83.7%
Ours&Speaker Labelling	89.1%	92.7%	91.2%	88.3%	89.1%	86.5%	89.5%
KMTJSRC	91.0%	89.7%	86.5%	89.2%	83.6%	92.1%	88.7%
KMTJSRC&CRF	92.8%	93.0%	89.7%	94.4%	88.8%	93.4%	92.0%

are also recognized with clothes and speeches. In order to do a reasonable and fair comparison, we only compare the result of the face recognition accuracy with our approach.

Besides, in the baseline work, since they all assume that the subtitles and transcripts are both available, a novel compared scheme is also constructed, named Ours&Speaker Labeling. In this scheme, we directly set the recognized probability of a speaker's (i.e., y_i) face tracks $p(o_i||y_i)$ as 1. That is to say, face tracks with speaking are labeled with the speaker's name in transcripts. Finally, the comparisons of these four schemes are shown in Table IV.

In Table IV, compared with the method of [13], we can see that our approach achieves obviously more accurate results, and the reasons may be as follows: 1) compared with the method of [13], which used the DCT features of each face, we extract the more robust SIFT features in nine facial key-points; 2) meanwhile, by looking each face in a track as a task, the MTJSRC method outperforms most of the existing methods, such as the SVM classifier or the full model in [13]; and 3) in sitcoms, the assumption of constraints between face tracks involved in our CRF-based approach (for example, face tracks of the same identity do have stronger similarity in a scene) is more convincing and strong in these sitcom videos. However, our approach strictly depends on the ability of face detection and tracking algorithms, and thus, the identity recognition recall will be not as strong compared with methods such as [13].

In the experimental results of the method [15], the performance is improved averagely more than 4% with MRF model compared with that using the multinomial logistic regression for weakly labeled data as well as the unlabeled data and constraints. That is to say, random field model, which considers the global consistency, can improve the recognition performance with sequence labeling. Actually, while MRF models the joint probabilities $p(X, Y)$, another random field model, i.e., CRF is essentially a structured extension of logistic regression, it models the conditional probabilities $P(Y|X)$. With the same idea that the MRF improves the recognition performance in sequence labeling, our approach using the CRF model achieves a more robust performance in the average accuracy, as shown in Table IV.

In addition, in [15], the recognition accuracy deeply depends on the script alignment and speaking detection accuracy which is ~87% for all those 22% tracks in their data set. In general, face tracks with high confidence of speaking detection always have great probability to be recognized accurately with the KMTJSRC algorithm, since the faces will have good appearance. Thus, as shown in Table IV, the performance of our approach combined with directly replacing the KMTJSRC result by the speaker's label in the script is better on

TABLE V
RESULTS ON SPOTLIGHTS SUMMARIZATION FOR GEORGE CLOONEY

Movie Name	Total Tracks	Positive	True Positive
Ocean's Eleven	232	23	19
Ocean's Twelve	228	17	15
Up In The Air	274	40	38

Episodes 2 and 3 (with high speaker precision), but the average accuracy lacks competitiveness to our approach. The reason may be that the speaker labeling is more efficient on these two episodes, but has lower recognition accuracy compared with KMTJSRC on the whole data set, and this directly pulls down the final recognition performance. Moreover, except for the challenge that speaker detection is a difficult problem itself with noisy labels described in [39], the script is not always available and sometimes new errors can be introduced due to the unguaranteed script to face alignment.

E. Actor-Specific Spotlights Summarization

In this experiment, we apply Cast2Face to spotlights summarization and evaluate the performance. We build a gallery set which contains face images of 21 actors from three films Oceans Eleven (2001), OT (2004), and Up In the Air (2009). Taking actor George Clooney as an example, we aim to extract these key shots for him from these films. After the MTJSRC, we obtain a set of tracks identified as George Clooney, among which the tracks, which include the speaking George Clooney, are taken as the key tracks. Table V shows the tracks detection and identification results. The subshots, including key tracks, are called key shots. By assembling these key shots, we can get the final spotlights summarization for George Clooney. The results of this experiment are available at YouTube: http://www.youtube.com/user/cast2face.

VI. CONCLUSION

Cast2Face is a novel cast and image retrieval-based movie character identification method proposed in this paper. We demonstrate that using the KMTJSRC algorithm and CRF model, high precision can be achieved by combining multiple sources of information, including the cast, Web image, and movie. Compared with the subtitle and script-based methods, one appealing aspect of our method is that it is textual analysis free. We have also explored an application of our method for actor-specific spotlights summarization. Empirical evaluations on feature-length movies show the satisfying performance of the Cast2Face method.

REFERENCES

[1] O. Arandjelovic and A. Zisserman, "Automatic face recognition for film character retrieval in feature-length films," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, pp. 860–867.

[2] M. Everingham, J. Sivic, and A. Zisserman, "'Hello! My name is... Buffy'—Automatic naming of characters in TV video," in *Proc. 17th Brit. Mach. Vis. Conf.*, 2006, pp. 889–908.

[3] M. Everingham and A. Zisserman, "Identifying individuals in video by combining 'generative' and discriminative head models," in *Proc. 10th IEEE Int. Conf. Comput. Vis.*, Oct. 2005, pp. 1103–1110.

[4] Y.-F. Zhang, C. Xu, H. Lu, and Y.-M. Huang, "Character identification in feature-length films using global face-name matching," *IEEE Trans. Multimedia*, vol. 11, no. 7, pp. 1276–1288, Nov. 2009.

[5] S. Satoh, Y. Nakamura, and T. Kanade, "Name-It: Naming and detecting faces in news videos," *IEEE Multimedia*, vol. 6, no. 1, pp. 22–35, Jan./Mar. 1999.

[6] M.-Y. Chen and A. Hauptmann, "Searching for a specific person in broadcast news video," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May 2004, pp. III-1036–III-1039.

[7] T. L. Berg *et al.*, "Names and faces in the news," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun./Jul. 2004, pp. II-848–II-854.

[8] Z. Liu and Y. Wang, "Major cast detection in video using both speaker and face information," *IEEE Trans. Multimedia*, vol. 9, no. 1, pp. 89–101, Jan. 2007.

[9] C. Xiong, G. Gao, Z. Zha, S. Yan, H. Ma, and T.-K. Kim, "Adaptive learning for celebrity identification with video context," *IEEE Trans. Multimedia*, vol. 16, no. 5, pp. 1473–1485, Aug. 2014.

[10] A. Kanak, E. Erzin, Y. Yemez, and A. M. Tekalp, "Joint audio-video processing for biometric speaker identification," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2003, pp. III-561–III-564.

[11] Y. Li, S. S. Narayanan, and C.-C. J. Kuo, "Adaptive speaker identification with audiovisual cues for movie content analysis," *Pattern Recognit. Lett.*, vol. 25, no. 7, pp. 777–791, 2004.

[12] S. Kwon and S. Narayanan, "Unsupervised speaker indexing using generic models," *IEEE Trans. Speech Audio Process.*, vol. 13, no. 5, pp. 1004–1013, Sep. 2005.

[13] M. Tapaswi, M. Bäuml, and R. Stiefelhagen, "'Knock! Knock! Who is it?' Probabilistic person identification in TV-series," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 2658–2665.

[14] M. Everingham, J. Sivic, and A. Zisserman, "Taking the bite out of automated naming of characters in TV video," *Image Vis. Comput.*, vol. 27, no. 5, pp. 545–559, 2009.

[15] M. Bäuml, M. Tapaswi, and R. Stiefelhagen, "Semi-supervised learning with constraints for person identification in multimedia data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3602–3609.

[16] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid, "Face recognition from caption-based supervision," *Int. J. Comput. Vis.*, vol. 96, no. 1, pp. 64–82, 2012.

[17] I. Laptev, M. Marszałek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[18] P. Bojanowski, F. Bach, I. Laptev, J. Ponce, C. Schmid, and J. Sivic, "Finding actors and actions in movies," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2280–2287.

[19] D. Anguelov, K.-C. Lee, S. B. Gökturk, and B. Sumengen, "Contextual identity recognition in personal photo albums," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2007, pp. 1–7.

[20] W.-L. Lu, J.-A. Ting, K. P. Murphy, and J. J. Little, "Identifying players in broadcast sports videos using conditional random fields," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2011, pp. 3249–3256.

[21] M. Xu, X. Yuan, J. Shen, and S. Yan, "Cast2Face: Character identification in movie with actor-character correspondence," in *Proc. 18th ACM Int. Conf. Multimedia*, 2010, pp. 831–834.

[22] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2001, pp. I-511–I-518.

[23] S. Wang, H. Lu, F. Yang, and M.-H. Yang, "Superpixel tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 1323–1330.

[24] G. Gao and H. Ma, "Accelerating shot boundary detection by reducing spatial and temporal redundant information," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2011, pp. 1–6.

[25] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[26] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–226, Feb. 2009.

[27] G. Obozinski, B. Taskar, and M. I. Jordan, "Joint covariate selection and joint subspace selection for multiple classification problems," *Statist. Comput.*, vol. 20, no. 2, pp. 231–252, 2010.

[28] P. Tseng, "On accelerated proximal gradient methods for convex-concave optimization," *SIAM J. Optim.*, to be published.

[29] X. Chen, W. Pan, J. T. Kwok, and J. G. Carbonell, "Accelerated gradient method for multi-task sparse learning problem," in *Proc. 9th IEEE Int. Conf. Data Mining*, Dec. 2009, pp. 746–751.

[30] M. Schmidt, E. van den Berg, M. P. Friedlander, and K. Murphy, "Optimizing costly functions with simple constraints: A limited-memory projected quasi-Newton algorithm," in *Proc. 12th Conf. Artif. Intell. Statist.*, 2009, pp. 456–463.

[31] X.-T. Yuan and S. Yan, "Visual classification with multi-task joint sparse representation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2010, pp. 3493–3500.

[32] R. Klinger and K. Tomanek, "Classical probabilistic models and conditional random fields," Dept. Comput. Sci., Dortmund Univ. Technol., Dortmund, Germany, Tech. Rep. TR07-2-013, Dec. 2007.

[33] J. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proc. 18th Int. Conf. Mach. Learn.*, 2001, pp. 282–289.

[34] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 11, pp. 1222–1239, Nov. 2001.

[35] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 1, pp. 129–159, 2001.

[36] Z. Rasheed and M. Shah, "Scene detection in Hollywood movies and TV shows," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2003, pp. II-343–II-348.

[37] M. Kyperountas, C. Kotropoulos, and I. Pitas, "Enhanced eigen-audioframes for audiovisual scene change detection," *IEEE Trans. Multimedia*, vol. 9, no. 4, pp. 785–797, Jun. 2007.

[38] V. T. Chasanis, A. C. Likas, and N. P. Galatsanos, "Scene detection in videos using shot clustering and sequence alignment," *IEEE Trans. Multimedia*, vol. 11, no. 1, pp. 89–100, Jan. 2009.

[39] M. Kostinger, P. Wohlhart, P. M. Roth, and H. Bischof, "Learning to recognize faces from videos and weakly related information cues," in *Proc. 8th IEEE Int. Conf. Adv. Video Signal-Based Surveill.*, Aug./Sep. 2011, pp. 23–28.