Unsupervised Celebrity Face Naming in Web Videos

Lei Pang and Chong-Wah Ngo

***Abstract*—This paper investigates the problem of celebrity face naming in unconstrained videos with user-provided metadata. Instead of relying on accurate face labels for supervised learning, a rich set of relationships automatically derived from video content and knowledge from image domain and social cues is leveraged for unsupervised face labeling. The relationships refer to the appearances of faces under different spatio-temporal contexts and their visual similarities. The knowledge includes Web images weakly tagged with celebrity names and the celebrity social networks. The relationships and knowledge are elegantly encoded using conditional random field (CRF) for label inference. Two versions of face annotation are considered: within-video and between-video face labeling. The former addresses the problem of incomplete and noisy labels in metadata, where null assignment of names is allowed—a problem seldom been considered in the literature. The latter further rectifies the errors in metadata, specifically to correct false labels and annotate faces with missing names in the metadata of a video, by considering a group of socially connected videos for joint label inference. Experimental results on a large archive of Web videos show the robustness of the proposed approach in dealing with the problems of missing and false labels, leading to higher accuracy in face labeling than several existing approaches but with minor degradation in speed efficiency.**

***Index Terms*—Celebrity face naming, social network, unconstrained web videos, unsupervised.**

I. Introduction

LABELING celebrities in Web videos is a challenging problem due to large variations in face appearance. The problem becomes increasingly important due to the massive growth of videos in Internet. According to YouTube trends map,[1] about 80% of popular videos are people-related and among the people-related videos, about 75% are about celebrities. To date, most search engines index these videos with user-provided text descriptions (e.g., title, tag), which are often noisy and incomplete. The descriptions are given globally, and hence the correspondences between celebrity names and faces are not explicit. It is not unusual that a mentioned celebrity does not appear in the video, and vice versa, a celebrity actually appearing in a video is not mentioned. For these reasons, searching people-related videos may yield unsatisfactory retrieval performance, either because of low recall or low precision. Ideally, finding the direct correspondences between names and faces could help rectify the potential errors in text descriptions and thus serve as a preprocessing step for video indexing. Furthermore, user search experience could be improved if the name-face correspondence is visualized, for example, by showing the name of a celebrity when a cursor moves over a face [1].

Fig. 1. Example of Web video illustrating the challenge of associating the names (italic) in metadata with the detected faces (with bounding boxes) in the video. Among the 14 faces of four celebrities (Hillary Clinton, Barack Obama, Wolf Blitzer and John Edwards), only four faces (dotted bounding boxes) of two celebrities (Hillary Clinton and Barack Obama) are mentioned in the metadata. In addition, only two (Hillary Clinton and Barack Obama) out of three celebrities who are mentioned appear in the video.

The problem of celebrity naming can be traced back to name-face association [2], where the goal is to align the observed faces with a given set of names. In the literature, this problem has been attempted in the domains of news videos [1], [2], movies [3] and TV series [4], capitalizing on the rich set of time-coded information including speech transcripts and subtitles. Nevertheless, these approaches often assume the ideal situation where the text cue is "rich" such that the given name set is free-of-noise and can perfectly match the observed faces. As a consequence, directly extending these approaches to Web video domain is not straightforward. Utilization of rich context information for face naming is also studied in the domain of personal album collection [5]–[8], by using timestamps, geotags, personal contact lists and social networks. Nevertheless, these approaches cannot be directly applied for domain unrestricted videos, because of the absence of context cues and prior knowledge such as family relationships for problem formulation.

Fig. 1 illustrates the problem with a real example of Web video. Out of the fourteen faces (of four celebrities) detected in the video, only four of them have names mentioned in the metadata. Furthermore, among the three celebrities who are mentioned, only two of them appear in the video. In other words,

Manuscript received June 18, 2014; revised October 24, 2014 and March 20, 2015; accepted March 22, 2015. Date of publication April 02, 2015; date of current version May 13, 2015. This work was supported by the Research Grants Council of the Hong Kong Special Administrative Region, China under CityU Grant 11210514, the City University of Hong Kong under Project 7008178, and the National Natural Science Foundation of China under Project 61272290. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Xiao-Ping Zhang.

The authors are with the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong (e-mail: leipang3-c@my.cityu.edu.hk; cwngo@cs.cityu.edu.hk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2015.2419452

[1][Online]. Available: http://www.youtube.com/trendsmap

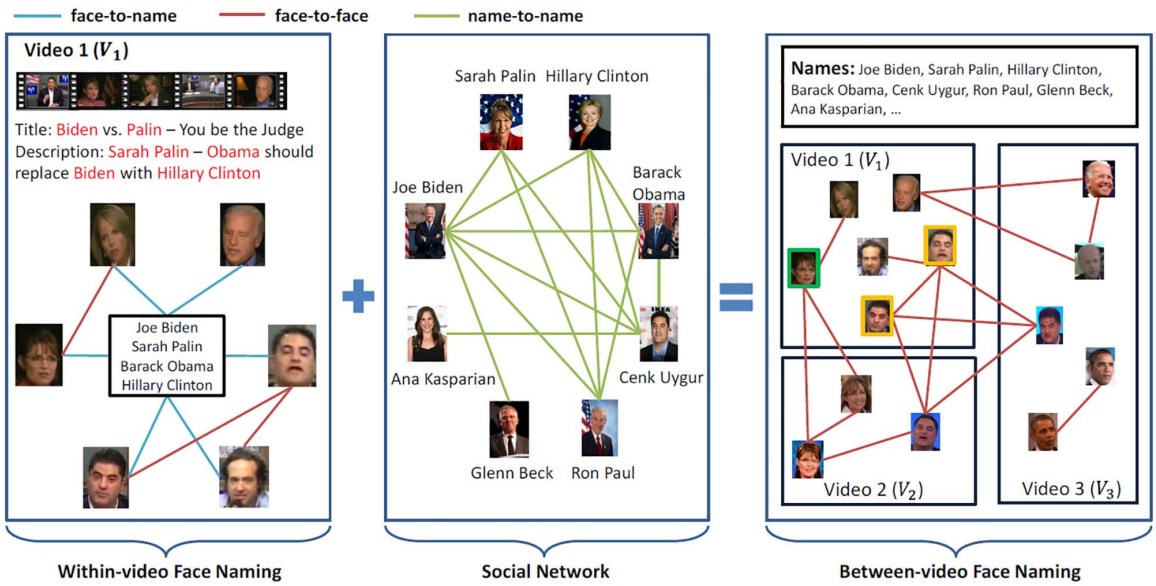

Fig. 2. Within-video naming constructs a graph modeling the face-to-name and face-to-face relationships among the faces and names found in a video (V_1). By social network, between-video labeling expands the graph by connecting to the graphs of two other videos (V_2 and V_3) that share social relations. The expanded graph is additionally modeled with name-to-name relationship inferred from the social network.

there are missing faces and names in the video and text respectively. Imposing constraints such as a face must be assigned to a name [8] and every given name must match to at least one face [3] apparently will result in erroneous labels. Such examples of videos are not rare. A statistic on a dataset used in this paper indicates that there are as high as 52% and 54% of Web videos suffering from the missing faces and names problems respectively. Additionally, a common characteristic of Web videos, as shown in Fig. 1, is that faces appear wildly different as a result of motion blur, lighting and resolution changes. In brief, the challenge of name-face association can be attributed to incomplete text labels, noisy text and visual cues.

In this paper, leveraging on rich relationships rather than rich texts [1]–[4] in Web video domain, an algorithm based on conditional random field (CRF) [9], [10] is proposed to address the problem of face naming. We consider three major kinds of relationships as follows.

- *Face-to-name resemblance* models how likely a face should be assigned to a name based on external knowledge from image domain.
- *Face-to-face constraint* considers factors such as background context, spatial overlap, temporal disconnectivity and visual similarity for relating faces from different frames and videos.
- *Name-to-name relationship*, or social relation, considers the joint appearance of celebrities by leveraging social network constructed based on the co-occurrence statistics among celebrities.

The first two relationships are exploited for labeling faces in a video, which we term as "within-video" face labeling. The task is to assign the names mentioned in metadata to the faces detected in a video, with the problem of missing faces and names in mind such that "null assignment" of names is allowed. The social relationship extends naming within a single video to "between-video" naming, by performing labeling of faces on a group of videos whose celebrities fall in the same social network. Compared to "within-video" naming, the relationships established among videos allow the rectification of names incorrectly tagged and the filling in of missing names not found in metadata.

Fig. 2 depicts two major tasks in this paper. Given a Web video V_1, "within-video" labeling constructs a graph with the names and faces in the video as vertices. Based upon the face-to-name and face-to-face relationships, edges are established among the vertices for inference of face labels by CRF. The inference can be affected by situations such as there are faces whose names are not mentioned in the metadata (e.g., Cenk Uygur), and similarly names mentioned in the metadata but faces do not appear in the video (e.g., Barack Obama). "Between-video" face labeling, by associating V_1 to a social network, crawls relevant videos (i.e., V_2 and V_3) and forms a larger graph composing of names and faces from multiple videos. Using social cues, additional edges modeling name-to-name relationships are also established. As shown in the example of Fig. 2, the expanded graph has the advantages that the missing name "Cenk Uygur" (marked in yellow rectangle) in V_1 can be propagated from V_2 and V_3 and the corresponding faces are assigned with the name replacing the "null" label, while the face wrongly labeled as "Hillary Clinton" (marked in green rectangle) can be rectified with name-to-name relationship as well as the similar faces found in V_2.

The main contribution of this paper is on the extension of name-face association to domain unrestricted Web videos for celebrity face naming. Particularly, this paper exploits three major relationships in addressing the problems of missing names and faces commonly happened in weakly-tagged videos, which are issues yet to be fully explored. CRF has been

employed in the literature for various labeling tasks, but in different contexts such as image annotation [11] and association of faces and time-coded overlaid text [12], which are different from this paper. We consider CRF in this paper mainly for its power in integrating diverse sets of relationships and off-the-shelf algorithms for label inference [13]–[15]. CRF, nevertheless, is known to be suffered from slow inference speed and high memory consumption, and hence is prohibited in some applications where scalability is a concern. In this paper, we suggest a practical way to bypass this problem by leveraging social relation to constrain the complexity of inference.

The remaining sections are organized as following. Section II presents the related works. Section III elaborates the problem formulation and describes our solution based on CRF for within-video celebrity naming, with the consideration of the missing names and faces problem. Section IV extends the solution to between-video celebrity naming, by leveraging social cues to rectify the potential errors in user-provided text descriptions. Section V presents experimental results, and finally Section 6 concludes this paper.

II. Related Work

The existing research efforts for face naming are mostly dedicated to the domain of Web images [16]–[18] and constrained videos [4], [19] such as TV series, news videos and movies. These works can be broadly categorized into three groups: model-based, search-based and constrained clustering-based face naming.

Model-based approaches seek to learn classifiers for face recognition. Due to the requirement of labeled samples as training examples for each face model, these approaches generally do not scale with the increase number of names. There have been numerous efforts strived for learning effective classifiers from smaller size of training samples. For example, by using Fisher discriminant analysis the approach in [20] wisely incorporates the labeled and unlabeled samples into kernel learning for face annotation. In [21], partial label information derived from the domain of broadcast videos are exploited for face naming using multiple instance learning. In this case, labeling a face is equivalent to judging whether a face is anonymous or not, which significantly cut short the labeling time. In [22], weakly-labeled images directly crawled from Web are leveraged for learning of face models. To minimize labeling efforts, a bootstrap learning strategy, which is named as consistency learning in [22], is employed to automatically filter out false samples from weakly labeled Web images for model training. A slight deviation of model-based learning is the so-called face verification, which determines whether a face pair belongs to the same person identity. DeepFace [23] is the most recent work achieving great success by using deep learning techniques. However, the requirement of large training samples for adequately covering visual appearance variations, for example, 4.4 million face labels for around 4,000 persons in DeepFace, is resourcefully expensively.

In contrast to model-based learning, search-based approaches mine the names from the retrieved examples deemed to be similar to the query faces. Therefore, the need for training examples is not applicable here since no classifier will be explicitly trained. Generally speaking, the main challenge for this line of approaches is to conquer the problem of noisy labels, for example, by unsupervised label refinement [24], when no supervisory information is available. The problem of name mining is then straightforwardly tackled by majority voting among the top–n retrieved images [24]. In [25], the local coordinate coding (LCC) is applied to enhance the weak labels while minimizing the impact of noisy labels during the voting of top–n images. The most recent effort by [26] posts this problem as the measuring of the weights for votes casted by images, through learning distance functions from multimodal features and the optimal fusion of these functions. Specifically, distance functions and fusion weights are offline learnt in a query independent manner using training examples. During retrieval, a vote from a top-n retrieved image to a candidate name is weighted based on its multimodal similarities to a query, determined by the learnt distance metrics and their optimal fusion weights.

The most related works to this paper are clustering-based approaches. The underlying assumption is that faces belonging to a person can be densely clustered and hence be exploited for face naming. These approaches generally perform well when there are only a few name candidates to be considered for a face. Existing approaches include constraint Gaussian mixture models (CGMM) [17], [18], graph-based clustering (GC) [17] and face-name association by commute distance (FACD) [27]. Using Expectation-Maximization (EM) algorithm, CGMM [17], [18] learns a Gaussian mixture model for each name. The learning iterates between assigning faces to the best possible model (E-step) and updating of model parameters (M-step). A null category for dealing with missing names problem is also learnt by treating all the faces as a mixture model. GC [17] and FACD [27] adopt a different strategy by using graph representation to model the density of faces. Started from the candidate names given in metadata, GC first retrieves images tagged with these names. A graph is then online constructed with faces in these images as vertices and their similarities as edges. The problem of name assignment is formulated as finding the densest sub-graphs, each corresponding to a name, from the graph. With the constraint that each face in a picture can be assigned to at most one name and vice versa, the problem of name assignment is shown to be equivalent to min-cost-max-flow problem, which can be solved using simplex algorithm. A merit of GC is that null category assignment can be naturally considered in the problem and no extra parameter is required. FACD strategically speeds up the graph construction by offline indexing the name-face pairs into an inverted index. Different from GC, FACD assigns names by explicitly enumerating the steps, named as commute distance, required to traverse from a face to a name through the random walk algorithm. Compared to GC, an extra threshold needs to learn in order to gate the activation of null category assignment, when the commute distances between a face and the candidate names are considered far. Different from the proposed work in this paper, these approaches [16]–[18], [27] are designed for Web images, and do not exploit inter-image correlation for a more global way of name-face association.

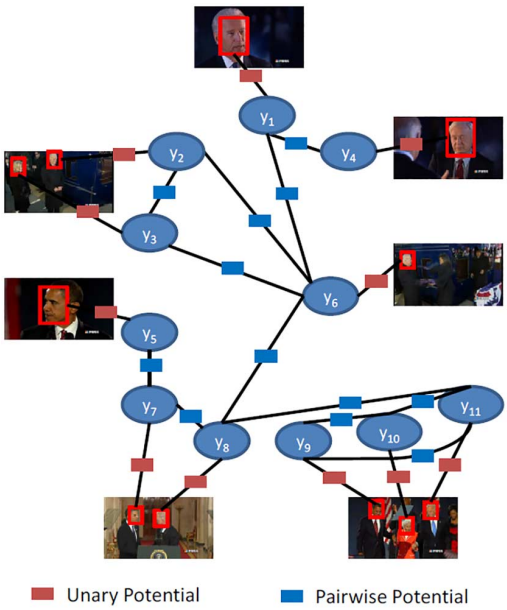

Fig. 3. Example of graph depicting the modeling of relationships for face naming as an optimization problem. The objective function is to maximize the probability of assigning the right names or labels (denoted as $\mathbf{y}_i$) to faces based upon the unary and pairwise potentials defined by various relationships.

III. Relationship Modeling

This section begins by formulating the problem of within-video face labeling as an optimization problem under conditional random field (CRF). Multiple relationships are then defined to characterize the sets of faces and names in the CRF.

A. Problem Definition and Notation

Given a video, the inputs to the problem of name-face association are the observed (or detected) faces from the video and the celebrity names found in metadata. Denote the celebrity names as a set $\mathcal{N} = \{c_1, c_2, \ldots, c_M\}$ and the detected faces as a sequence $S = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)$, where M and N represent the number of names and faces respectively. The problem here is to assign at most one name $c_i \in \mathcal{N}$ to a face $\mathbf{x}_i \in S$, such that every face in a video is given either a name or no name (i.e., null assignment). The output of the problem is a label sequence, denoted as $Y = (y_1, y_2, \ldots, y_N)$, where each element y_i is an indexed variable indicating $\mathbf{x}_i$ face in the sequence S is assigned with a name $c_i \in \mathcal{N}$ or "null".

Under CRF, the face and label sequences are modeled as a graph for name inference. The graph is undirected, denoted as $G = (V, E)$, where $V = \{S, Y\}$ is the set of vertices and E is the set of edges connecting vertices. The edges are established based upon different relationships defined between faces and between faces and names. Fig. 3 shows an example of the graph with 11 faces in the observed sequence. There is an index variable, y_i, encoding the label for each of the eleven faces.

Basically, the problem now is to enumerate each possible label assignment, and then eventually select one among the assignments as the best solution that maximizes the probability of assignment. With a little abuse of notations, let's denote each of such assignment as a vector $\mathbf{y} = [y_1, y_2, \ldots, y_n]$ for a vector of observed faces $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n]$. Here, we would like to estimate the conditional probability $p(\mathbf{y}|\mathbf{x})$. Following the local Markov property in CRF, we assume that two indexed variables $y_i, y_j \in Y$ are independent of each other if there is no edge (or relationship) between them. With reference to Fig. 3 as example, the variable y_1 is dependent on the variable y_4, but not the variable y_2. Following the convention of CRF in naming notation [10], we name the set of dependent labels and observations as "factor". For example, $\{y_1, \mathbf{x}_1\}$ is a factor, and $\{y_1, y_4, \mathbf{x}_1, \mathbf{x}_4\}$ is also a factor. To this end, the conditional probability can be factorized into the following forms

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{c \in C} \Phi(\mathbf{y}_c, \mathbf{x}_c) \tag{1}$$

where C is the set of factors in G, Φ is a potential function, and $Z(\mathbf{x}) = \sum_y \prod_{c \in C} \Phi(\mathbf{y}_c, \mathbf{x}_c)$ is a partition function served for normalizing the probability score. We consider two kinds of potentials in characterizing Φ, namely the unary potential $\mu(y_i, \mathbf{x}_i)$ and pairwise potential $\psi(y_i, y_j, \mathbf{x}_i, \mathbf{x}_j)$, which model the face-to-name and face-to-face relationships respectively as shown in Fig. 3. The conditional probability can thus be rewritten as

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{c \in C_\mu} \mu(y_i, \mathbf{x}_i) \prod_{c \in C_\psi} \psi(y_i, y_j, \mathbf{x}_i, \mathbf{x}_j) \tag{2}$$

where $C = \{C_\mu, C_\psi\}$. Furthermore, the pairwise potential is a linear combination of three functions, each features one kind of relationships, as follows:

$$\psi(y_i, y_j, \mathbf{x}_i, \mathbf{x}_j) = \theta_{sr} f_{sr} + \theta_{tr} f_{tr} + \theta_{vr} f_{vr}. \tag{3}$$

Each of these feature functions models spatial (f_{sr}), temporal (f_{tr}) or visual (f_{vr}) relationship, and is weighted by $\theta = \{\theta_{sr}, \theta_{tr}, \theta_{vr}\}$ respectively. In the remaining subsections, we will further detail the unary potential (Section III-B) and the feature functions under pairwise potential (Section III-C).

In brief, the problem of face naming can be elegantly stated as to maximize the probability in (1). The inference of names can be rigorously solved with off-the-shelf algorithms such as Markov Chain Monte Carlo (MCMC) [13] or Loopy Belief Propagation (LBP) [14], [15]. As shown in Fig. 3, the challenges of face naming originate from the large variations in visual appearance and face resolution. Relying merely on face similarity for naming is likely to fail in this kind of examples.

B. Unary Potential

The unary potential energy measures the likelihood of a face $\mathbf{x}_i$ being labeled with a name or "null". To do so, we model each name n as a multivariate Gaussian distribution of faces as

$$p(\mathbf{x}_i^{face}|\lambda_n) = \mathcal{N}(\mathbf{x}_i^{face}; \mu_n, \Sigma_n) \tag{4}$$

where $\lambda = \{\mu, \Sigma\}$ is the set of Gaussian parameters, and $\mathbf{x}_i^{face}$ represents the facial feature extracted from face $\mathbf{x}_i$. The faces used for modeling (4) are extracted from Web images crawled from search engines (See Section V-A2).

We model the assignment of a face to "null" category as a problem of information uncertainty. Specifically, considering the probability distribution of labeling a face with the names in

$\mathcal{N}$, the uncertainty in labeling can be characterized by the normalized entropy as

$$E_{x_i} = \frac{-\sum_{n\in\mathcal{N}} p(\mathbf{x}_i^{face}|\lambda_n)\log_2(p(\mathbf{x}_i^{face}|\lambda_n))}{\log_2|\mathcal{N}|}. \quad (5)$$

The uncertainty reaches the highest (i.e., higher entropy value) when the probabilities are uniformly distributed. Reversely, when the probability of assigning to a name is noticeably high than other names, the uncertainty becomes lower. To this end, the unary potential characterizing the edge between a face $\mathbf{x}_i$ and a label y_i is defined as

$$\mu(y_i,\mathbf{x}_i) = \begin{cases} p(\mathbf{x}_i^{face}|\lambda_{y_i}), & \text{if } y_i \in \mathcal{N} \\ E_{x_i}, & \text{if } y_i = \text{null} \end{cases} \quad (6)$$

where the probability of labeling a face as belonging to "null" category is proportional to the uncertainty of assigning the face to the given names. Note that (6) contributes to the conditional probability $p(\mathbf{y}|\mathbf{x})$. CRF will numerate all possible name assignments and eventually select the one with the highest probability as the name assignment for $\mathbf{x}_i$.

C. Pairwise Potential

The pairwise potential energy characterizes the possible relationships between two faces, as described by (3). This sub-section defines the feature functions of each relationship in characterizing the pairwise potential energy.

Spatial Relationship: Given two frames of different shots, the spatial locations of faces, as well as their overlapping area, give clue to the identity of face. Generally speaking, by cinematography practice, the position and size of a face shall not change dramatically across shots for maintaining temporal coherence. Nevertheless, this clue is weak considering that any two faces at the center of frames will overlap regardless of their identities. A more robust way of modeling spatial relationship is to also consider the background of the frames where faces reside. Specifically, when two frames sharing similar background, the spatial relationship can be leveraged to establish edges in G for linking the labels assigned to faces.

Denote $\mathbf{x}_i^{back}$ as a color histogram [28] for the background of a frame where the face x_i resides, the feature function for spatial relation is defined as

$$f_{sr}(y_i,y_j,\mathbf{x}_i,\mathbf{x}_j) = 1_{\{cos(\mathbf{x}_i^{back},\mathbf{x}_j^{back})\geq\eta\}}. \quad (7)$$

The similarity between background frames is measured by cosine similarity, i.e., $\cos(.)$. The parameter η is an empirical threshold, which will be discussed in Section V-B. Equation (7) specifies the condition for an edge to be established between two faces. Note that the notation $1_{\{...\}}$ is an indicator function which means that an edge between the faces $\mathbf{x}_i$ and $\mathbf{x}_j$ is established when the condition $cos(\mathbf{x}_i^{back},\mathbf{x}_j^{back}) \geq \eta$ is met.

To weight the significance of the knowledge, the parameter θ_{sr} (3) is considered to be characterized by the positions and sizes of faces. Denote area($\mathbf{x}_i$) and overlap($\mathbf{x}_i$, $\mathbf{x}_j$) as the size of a face $\mathbf{x}_i$ and the area of overlap with face $\mathbf{x}_j$ respectively, the proportion of face overlap is

$$prop(\mathbf{x}_i,\mathbf{x}_j) = \frac{overlap(\mathbf{x}_i,\mathbf{x}_j)}{area(\mathbf{x}_i)\cup area(\mathbf{x}_j)}. \quad (8)$$

The value for model parameter θ_{sr} is set on-the-fly depending on the assigned labels as follows:

$$\theta_{sr} = \begin{cases} prop(\mathbf{x}_i,\mathbf{x}_j), & \text{if } y_i = y_j \neq \text{null} \\ 1-prop(\mathbf{x}_i,\mathbf{x}_j), & \text{if } y_i \neq y_j \\ \max(prop(\mathbf{x}_i,\mathbf{x}_j), 1-prop(\mathbf{x}_i,\mathbf{x}_j)), & \text{if } y_i = y_j = \text{null} \end{cases}. \quad (9)$$

The face overlap is utilized to boost (penalize) when the same (different) names are assigned during label inference. For the case of two faces assigned to null category, the knowledge of whether they belonging to the same identity is unknown. To model the uncertainty, max operator is used such that the assignments will not be punished, regardless of their actual identities.

Temporal Relationship: The appearance of faces at different timestamps along the temporal axis gives clue of whether the names assigned to faces should be exclusive of each other. Specifically, faces, which coincide in any of a frame along the timeline of a video, should belong to different identities and hence can be assigned different labels throughout the video. Denote $\mathbf{x}_i^{time}$ as the timestamp where a face $\mathbf{x}_i$ appears, the feature function for temporal relationship is defined as

$$f_{tr}(y_i,y_j,\mathbf{x}_i,\mathbf{x}_j) = 1_{\{\mathbf{x}_i^{time}=\mathbf{x}_j^{time}\}}. \quad (10)$$

where an edge between x_i and x_j is established if the condition $\mathbf{x}_i^{time} = \mathbf{x}_j^{time}$ is fulfilled. Nevertheless, note that this relationship is further controlled by the model parameter θ_{tr}, which can still assign a weight of zero to this relationship if both faces have the same labels, i.e., $y_i = y_j$, as following

$$\theta_{tr} = \begin{cases} 1, & \text{if } y_i \neq y_j \text{ or } y_i = y_j = \text{null} \\ 0, & \text{if } y_i = y_j \neq \text{null} \end{cases}$$

A special case is when two faces are both assigned to "null" category. Since this case does not imply that the two faces should belong to the same person, the value for θ_{tr} is set equal to 1.

Visual Relationship: Like spatial relationship, face similarity only provides weak clue to the name identity in the Web video domain. Generally speaking, the dissimilarity between two faces can always be attributed to situations such as changes in viewpoint and lighting conditions. The inference of labels based on face dissimilarity can thus be uncertain. On the other hand, two highly similar faces nevertheless is a necessary clue to evidence the name identity. Based on these two facets of perception on face similarity, the feature function for visual relationship is modeled as

$$f_{vr}(y_i,y_j,\mathbf{x}_i,\mathbf{x}_j) = 1_{\{cos(\mathbf{x}_i,\mathbf{x}_j)\geq\delta\}} \quad (12)$$

where δ is an empirical threshold for filtering low similar faces from taking part in the label propagation in CRF. Cosine similarity, $\cos(\mathbf{x}_i,\mathbf{x}_j)$, is used for measuring the similarity between two faces $\mathbf{x}_i$ and $\mathbf{x}_j$. The parameter θ_{vr} characterized by facial similarity is set as

$$\theta_{vr} = \begin{cases} cos(\mathbf{x}_i,\mathbf{x}_j), & \text{if } y_i = y_j \neq null \\ 1-cos(\mathbf{x}_i,\mathbf{x}_j), & \text{if } y_i \neq y_j \\ \max(cos(\mathbf{x}_i,\mathbf{x}_j), 1-cos(\mathbf{x}_i,\mathbf{x}_j)), & \text{if } y_i = y_j = null \end{cases}. \quad (13)$$

Similar to (9), max operator is used when both labels are assigned to null category.

D. Complexity Analysis

Algorithm 1 summarizes the major steps of within-video face naming. The run time complexity is governed by two major parts: graph construction (particularly step-2) and CRF label propagation (step-3). As shown in Algorithm 1, the first two steps involve mainly the establishment of unary and pairwise potentials for the graph $G(V, E)$, given the sets of faces S and names $\mathcal{N}$ in a video. For unary term, the complexity is $\mathcal{O}(|\mathcal{N}| \times |S|)$, where $|\mathcal{N}|$ and $|S|$ are the number of names and faces respectively. Since pairwise potential considers any two face pairs, the complexity is $\mathcal{O}(|S|^2)$. For the third step, we employ Loopy Belief Propagation (LBP) [14], [15] for label propagation. For each face under investigation, the speed of LBP is dominated by message passing, or more specifically, the possible pairwise name assignments $|\mathcal{N}|^2$ for every edge of the face. In the worst case when the graph is fully connected, the complexity is $\mathcal{O}(k \times |S| \times |E| \times |\mathcal{N}|^2)$, assuming the inference converges in k iterations.

The required memory cost is proportional to the size of a graph. Using adjacency matrix as the graph representation, the space complexity is $\mathcal{O}(|S|^2)$. Furthermore, each vertex and edge are represented as a vector of size $|\mathcal{N}|$ and a matrix of size $|\mathcal{N}|^2$ respectively, resulting in extra $\mathcal{O}(|S| \times |\mathcal{N}|)$ and $\mathcal{O}(|E| \times |\mathcal{N}|^2)$ for storing vertices and edges. In practice, as the spatial and visual relationships connect only faces with high background and visual similarities respectively, the constructed graph is generally sparse. As a consequence, the complexity in terms of speed and space for a video is not considered high.

Algorithm 1 Within-video face labeling.

Input: The sets of faces S and names $\mathcal{N}$ in a video V
Output: Face labels Y that maximizes $p(\mathbf{y}|\mathbf{x})$ in (2)

1) Constructing a graph G by modeling the unary potential for each face $\mathbf{x}_i \in S$, where an edge between $\mathbf{x}_i$ and $y_i \in Y$ is weighted with (6).
2) Establishing edges for any pairs of $y_i \in Y$ and $y_j \in Y$ in G that satisfy the condition in (7), (10) and (12), with their edge weights set respectively based on spatial (9), temporal (11) and visual (13) relationships.
3) Performing loopy belief propagation [14], [15] on G for face labeling.

IV. Leveraging Social Cues

Performing face naming within a video has the limitation that only the names mentioned in the metadata will be considered. In the situation where missing names or faces exist, probabilistic inference of labels could become arbitrary due to lack of sufficient clues. In the extreme case, there maybe no edges established for some faces after the evaluation of pairwise relationships. Under this situation, the amount of messages passing between faces will be limited, which directly impacts the effect of label propagation. By extending the proposed approach in Section III to beyond a single video, faces originally lacking channels for effective message passing should have higher chance to be connected. To be explicit, the advantages of involving multiple videos in graph construction are twofold. By the candidate names from third party videos, faces originally labeled as null can be named as far as possible through CRF optimization. Second, ambiguous labels due to information uncertainty can be resolved with additional cues derived from other videos.

A. Social-Driven Graph Splitting

With a video collection as input, the graph G presented in Section III-A is expanded with a vertex set V including all the observed faces and names mentioned in the collection. The expansion will result in dramatic increase of edges under the modeling of unary and pairwise potentials. Specifically, unary potential considers all the available names as admissible labels for a face, while visual relationships establish links for similar faces across videos. Note that the spatial and temporal relationships, which are only valid within a video itself, cannot be leveraged for between-video connectivity. Due to the quadratic complexity of CRF, the significant increase in size of a graph is expected to affect the speed efficiency. As an example in the experiment, considering a collection of 2,000 videos, processing each video sequentially will take 27 minutes in total. Jointly processing the videos as a whole will slow down the speed by 200 times theoretically.

A vivid cue that can be exploited for reducing the size of graph is social networks among the celebrities under consideration. For example, the former president of China Jintao Hu is not likely to be linked to any face in a video about "Britain's Got Talent show". In other words, social network helps trimming potentially superfluous relations, giving light of splitting a large graph into subgraphs of each depicting a social network. With this intuition, we first exploit the co-occurrence of celebrities in mining social networks, followed by constructing one graph per social network for CRF optimization. Fig. 4 summarizes the name inference using social networks, where there are three major steps.

Denote $G_{name} = (V, E, W)$ as a graph depicting the relationship among celebrities, where V is the vertex set representing n names, and E is the edge set linking celebrities. The notation W denotes a weight matrix of size $n \times n$, whose element $W_{i,j}$ describes the relationship between two celebrities n_i and n_j defined as

$$W_{i,j} = \frac{2|s_i \cap s_j|}{|s_i| + |s_j|} \tag{14}$$

where s_i is the set of videos tagged with celebrity n_i, and $|s_i|$ denotes the set cardinality. Equation (14) basically calculates the co-occurrence statistics of celebrities as the proportion of videos where both names are tagged. As can be seen in the step-1 of Fig. 4, using the matrix W, a social graph G_{name} is constructed. We connect each name in G_{name} to five other names with the largest weights, such that the resulting graph is sparse and efficient to be processed. Subsequently, by Walktrap algorithm [29], the graph is further partitioned into sub-graphs corresponding to social networks. The algorithm is highly efficient and capable of estimating the number of communities automatically.

Having the social networks, we distribute each video to one or multiple networks based on the names mentioned in a video

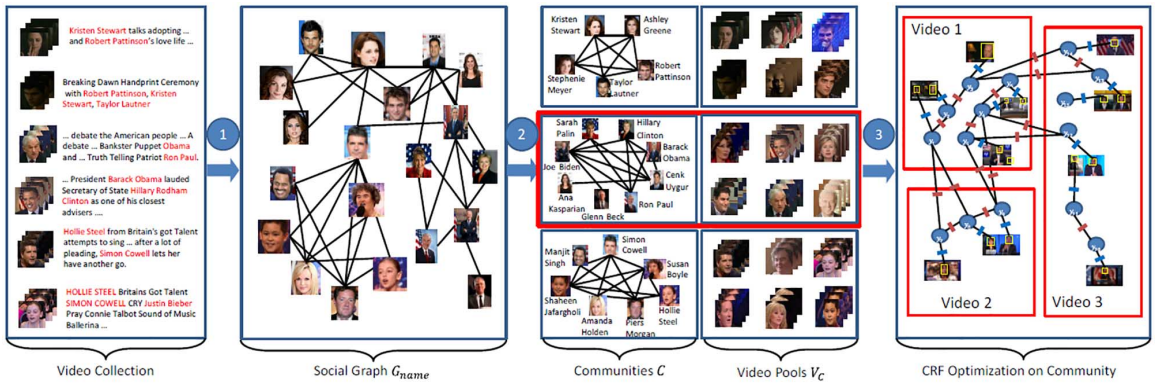

Fig. 4. Between-video naming: the process of constructing social networks and performing CRF on multiple smaller graphs.

(step-2 of Fig. 4). Denote Ω_C as the set of celebrities in a community C, and N_{v_i} as the set of names tagged in a video v_i. A video v_i is assigned to C if $\mathcal{N}_{v_i} \cap \Omega_C \neq \emptyset$. By doing so, each network C will be associated with a video pool V_C, and meanwhile the community size will also be expanded with new names from the pool, i.e., $\Omega_C^* = \Omega_C \cup_{v_i \in V_C} \mathcal{N}_{v_i}$. In other words, Ω_C^* is composed of celebrities not only from the network but also all names tagged in the videos assigned to C. To this end, as depicted in step-3 of Fig. 4, CRF only needs to separately consider the videos and celebrities Ω^* in the domain of a community for name inferencing, which overall cuts short the running time.

B. Name-to-Name Social Relation

Recall that the joint modeling of videos as a graph has the advantage that a name y_j not mentioned in a video v_i can still be exploited for labeling faces in v_i. Intuitively, the chance that there is a celebrity named y_j appearing in v_i is proportional to the co-occurrence of y_i with other names mentioned in v_i. For example, Bill Clinton has a higher chance than George Bush to appear in a video tagged with the name Barack Obama, giving the fact that Clinton has closer political relationship with Obama. With this intuition, the unary potential term in (6), which characterizes the edge between a face $\mathbf{x}_i$ and a label y_i, can be augmented with social information, or name-to-name relationship, as follows:

$$\mu(y_i, \mathbf{x}_i) = \begin{cases} ws_{y_i,\mathbf{x}_i}, & \text{if } y_i \in \mathcal{N} \\ E_{x_i}, & \text{if } y_i = \text{null} \end{cases} \tag{15}$$

where

$$ws_{y_i,\mathbf{x}_i} = \alpha \times p(\mathbf{x}_i^{face}|\lambda_{y_i}) + (1-\alpha) \times \max(W_{n,y_i}), n \in \mathcal{N} \tag{16}$$

and W is the matrix capturing the co-occurrence statistics of celebrities. The max operator basically picks a name $n \in \mathcal{N}_{v_i}$ who co-occurs most with y_i, and uses the corresponding score in W for adjusting the significance of unary potential. Basically, the equation takes into account whether the label y_i is actually socially connected to any names mentioned in a video v_i. The probability of being y_i is boosted, if the statistics supports such claim. Equation (16) combines both clues from visual and social in a linear fashion. The empirical parameter α, which is set equal to 0.5, is for trading off the importance of visual and social cues.

C. Algorithm Implementation

While running CRF separately on each community as shown in Fig. 4 has accelerated face naming, a practical concern is the memory cost. For a community with 13 celebrities, 279 videos (i.e., $|V_C| = 279$) and 34 candidate names (i.e., $|\Omega_C^*| = 34$), the memory consumption can be as high as 5G bytes. Furthermore, for videos with celebrities who could be assigned to more than one community, the result of labeling may not be consistent across communities. For practical consideration, we thus propose CRF to be conducted on video rather than on community basis.

Algorithm 2 summarizes the details. The algorithm starts by processing each video individually. Given a video v_i, step-1 constructs a graph G_i by establishing the face-to-name and face-to-face relationships based on the unary and pairwise potentials. Step-2 crawls videos sharing the same social networks as v_i. Subsequently, step-3 builds a new graph G^* to establish edges among the involved videos based on face similarity and name-to-name relationship. Among the crawled videos, G^* eventually only includes videos that establish edges with v_i, which is usually a sparse and small graph in size. For example, there are typically about 30 videos per graph in our experiments. Performing face inference on such graph, step-4 of the algorithm, will take about one second and occupy only 0.04G bytes of memory.

Algorithm 2 Between-video face labeling.

Input: A Web video v_i and the associated metadata
Output: Face labels

1) Construct a graph G_i connecting faces and names in v_i by (4), (7), (10) and (12).
2) Crawl the videos, and their graphs, sharing the same social networks as v_i from the dataset.
3) By (12) and (15), a new graph G^* is constructed by establishing edges to connect all the graphs in steps 1 and 2.
4) Perform loopy belief propagation [14], [15] on G^* for face labeling.

In Algorithm 2, it is important to note that steps 1 to 3 only need to be performed once for all the videos in a dataset. Precisely, with reference to the social networks, multiple large

graphs can be constructed for all the videos in a dataset. When labeling faces for a video v_i, only videos which connect to v_i will be involved during face inference. Algorithm 2 can also be directly applied for unseen or newly arrived videos. Basically, by crawling all the videos sharing the same social networks as an unseen video (step-2), a graph G^* can be built (step-3) on-the-fly by relationship modeling among these videos for name inference (step-4).

V. Experiments

The experiments include within-video (Section V-C) and between-video (Section V-D) face naming. The empirical studies investigate the effectiveness of various relations proposed in this paper, with comparison to state-of-the-art approaches. The runtime efficiency is also detailed in Section V-E.

A. Dataset and Evaluation Metrics

Dataset: A dataset named Cele-WebV [30] is constructed for experiments. The dataset is originated from the core dataset of MCG-WEBV [31], composing of 14,473 Web videos uploaded to 15 YouTube channels during December of year 2008 to November of year 2009. To preprocess the dataset, candidate person names are extracted from the video metadata, by stepwise matching of a word as well as a succession of words against Wikipedia. A person name is verified if the category tag for birth year is found in the matched Wikipedia pages. By filtering out names that appear in less than 10 videos, finally there are 141 celebrity names being retained for experiments. The dataset Cele-WebV is formed by pooling together 2,583 videos containing the 141 celebrities. A total of 41,047 frontal faces are extracted[2] from 409,900 keyframes of the dataset, including 20% of close-up faces with resolution larger than 150×150 pixels.

To facilitate the result analysis, we further split the dataset into three subsets: *Easy*, *Average* and *Hard*, representing the potential difficulty in face naming. The *Easy* (*Hard*) subset contains videos with no more than 2 (more than 4) celebrity names found in the metadata. The *Average* dataset includes the remaining videos with 3 or 4 names. Table I shows the detailed statistics of Cele-WebV dataset. To generate ground-truth, each face is labeled with a celebrity name found in the video metadata. By doing so, each celebrity has on average 136.5 faces. However, there are only 46% (19,240 out of 41,047) of faces being labeled with names. On average, each video has 8.56 faces without assigning a name. On the other hand, there are 52% of celebrity names do not associate with any faces in the videos. The large number of faces without a name, as well as names without corresponding faces, basically hint the challenge of this dataset.

We also create another subset named Cele-WebV* by fully labeling all the faces in a video regardless of whether the celebrity names are mentioned in the metadata of the video. There are 300 videos randomly selected from Cele-WebV being included in this subset. Different from Cele-WebV, we expand the number of celebrities from 141 to 200 names. Out of the 2,487 faces without names found in the metadata, 146 faces are labeled with a name among the 200 celebrities. Finally, we create another dataset named Cele-WebV$^+$ consisting of 800 videos not in the core dataset of MCG-WEBV. Note that there is no overlap of videos between Cele-WebV$^+$ and Cele-WebV, though both sets of videos are originated from MCG-WEBV. As Cele-WebV*, the 7,663 faces in the dataset are fully labeled with the names of 200 celebrities. Among 3,739 faces (48.8%) without celebrity names in the metadata, 223 of them are labeled. Compared with Cele-WebV, there is a higher percentage of names, on average around 1.27 out of 2.1 names in the metadata, without any faces found in the video.

TABLE I
Cele-WebV and its Subsets. The Second Column Shows the Number of Videos, Followed by the Average Number of Faces, Tagged Names and Celebrities Per Video in the Remaining Columns. The Numbers Inside Parenthesis Indicate the Percentage of Faces Without Names in the Metadata (3rd column) and the Percentage of Names Without Faces Appearing in the Videos (4th column)

	video	face	name	celebrity
Easy	2223	15.0 (54%)	1.2 (46%)	0.65
Average	257	20.0 (52%)	3.4 (59%)	1.35
Hard	103	25.8 (59%)	5.9 (67%)	1.96
Cele-WebV	2583	15.9 (54%)	1.6 (52%)	0.77

Supporting Image Dataset and Features: A face dataset consisting of Web images of the 200 celebrities is also constructed. The pictures are crawled from Google image search engine using the celebrity names as the keywords. The top-150 pictures presented by the search engine for each celebrity are crawled. No human intervention is involved throughout the procedure. To reduce noises, pictures with more than one faces are filtered out. Finally, there is a total of 19,851 images in the dataset. The dataset is used for modeling the unary potential as described in Section III-B. For each celebrity, a multivariate Gaussian model is learnt, by treating the Web images of the celebrity as positive examples. Note that the relatively small number of Web images, around 100 per celebrity, hinders the use of more complex model such Gaussian mixture model (GMM).

Two features, facial feature and color histogram, are extracted for measuring the face and background similarities respectively. The facial feature is represented by a 1,937 dimensional vector, describing the salient points extracted from 13 facial regions [4]. The dimension of the facial feature is reduced to 100 by principal component analysis (PCA). The background feature is represented by a color histogram of 300 dimensions in RGB space. Cosine similarity is employed as proximity measurement for background features [28].

Performance Evaluation: Similar to [16], [17], [19], [27], the performance is measured by accuracy and precision. Both measures count the number of faces correctly labeled, but differ where accuracy also includes the counting of faces without labels. Denote C as the set of faces correctly labeled (including "null assignment"), and T as the groundtruth labels of all the faces in a dataset. The set $T = \{L, U\}$, which is composed of a subset L with all the faces assigned with names and a subset U

[2]We employ the commercial software developed by ISVision for face detection. [Online]. Available: http://www.isvision.com/cn/index

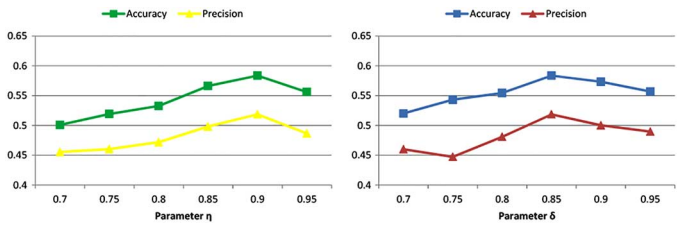

Fig. 5. Sensitivity of the parameters η (left) and δ (right) towards the accuracy and precision of face naming.

with faces without assigned a name. The definitions of accuracy and precision are

$$\text{Accuracy} = \frac{|C \cap T|}{|T|} \tag{17a}$$

$$\text{Precision} = \frac{|C \cap L|}{|L|}. \tag{17b}$$

Note that accuracy and precision are calculated across all the faces in a test collection, rather than averaged over videos. Recall is not used here because we do not consider the problem of "retrieving all faces given a name", rather we are dealing with the problem of whether a face is labeled with a correct name (precision), and otherwise labeled as "null" if the name is missing from metadata (accuracy).

There are three datasets used in the experiments. The experiments on within-video and between-video face naming are conducted on Cele-WebV and Cele-WebV* respectively. Cele-WebV is also used for empirical parameter tuning. The parameters of the proposed as well as compared approaches are tuned using grid search in a brute-force manner for the best possible performance. For more objective evaluation, the dataset Cele-WebV$^+$ is used as an independent dataset for evaluation, where no parameter tuning is allowed.

B. Parameter Sensitivity

There are two empirical parameters, η and δ, used in our method for controlling the graph sparsity. These parameters are for filtering out the faces with low facial and background similarities, as outlined in (7) and (12) respectively. Fig. 5 shows the sensitivity of these two parameters on Cele-WebV. Both parameters exhibit similar performance trends, where a lower value will result in inclusion of noises, and hence lower performances in both accuracy and precision. On the other hand, an extremely high value will filter most of the true positive relationships. The performances drop in this case since there are only few edges left to be leveraged for multiple relationships modeling. Basically, for both parameters the performances peak at certain values, which are relatively high to filter most noises while still capable of retaining a good number of true positives for estimation. With a reasonable setting, the performances vary within the ranges of 46% - 52% for precision, and 50% - 58% for accuracy. In the remaining sections, the parameters of η and δ are set to 0.9 and 0.85 respectively.

C. Within-Video Face Naming

We first investigate the effect of considering multiple relationships in our approach (CRF-L). Table II contrasts the performances when different relationships are incorporated. On top of the unary potential (UP) defined based upon external Web images, each relationship basically improves the accuracy (precision) by at least 7% (5%). Visual relationship is observed to be contributing less than other relationships mainly due to the large variation of face appearance in Web videos, attributed to the uncontrolled video capturing environment. By integrating all the three relationships, an improvement of 16% (18%) is attained in accuracy (precision). It is also observed that the performance is inversely proportional to the number of names mentioned in metadata and the number of faces in a video. It is worth noting that, when multiple relationships are leveraged, the relative improvement is indeed proportional to the number of faces and names. Generally speaking, the large number of faces and names increases the uncertainty of naming, but results in a graph with more relationships to be exploited on the other hand. For example, the appearance of multiple faces in a frame hints the exclusive relationships when assigning the names. As shown in Table II, the temporal relationship contributes more to the performance in the *Hard* subset than the *Average* and *Easy* subsets. It is worthwhile to note that the accuracy of null assignment (because of missing names) is also improved when more names are given in the metadata. For *Hard* subset, the accuracy is 52.3%, against the 47.9% and 48.5% achieved by the *Average* and *Easy* subsets respectively. This is simply because the entropy \(5) used for measuring the uncertainty of name-face assignment becomes more reliable, when more names are available for providing a more complete picture of statistics. Similar observation is also noted for the missing faces problem. Lower error rate (31.6%) is attained in *Hard* subset than *Average* (37.2%) and *Easy* (35.8%), due to the presence of more relationships to be exploited for inference.

Next, we compare our proposed method CRF-L with five other approaches: random assignment (RA), threshold-based null assignment (TA), constrained Gaussian Mixture Model (CGMM) [17], [18], graph-based clustering (GC) [17] and commute distance (FACD) [27]. RA is a baseline, which randomly associates a face either to a name in the metadata or to null category. TA is based on our method but considering only the unary potential, and an empirical threshold optimally tuned for null assignment. Note that CGMM, GC and FACD are originally designed for name-face association in the domain of Web images. In the experiments, similar to CRF-L, these approaches operate on the keyframe-level and directly use the names tagged in metadata as the candidate names for labeling. Both CGMM and GC involve iterative optimization. In the experiment, the iteration stops as soon as no more than 3% of face labels are updated. The learning process normally converges within 10 iterations. For FACD, there are two key parameters: top-p similar faces of a name to be used for graph construction, and the threshold ϵ for null category assignment. We tune the parameters in the range of values suggested by [27], and choose $p = 50$, and ϵ equals to a value where 40% of faces in the dataset are assumed belonging to null category based on commute distance. For all the three approaches,

TABLE II
WITHIN-VIDEO CELEBRITY NAMING: EFFECT OF COMBINING MULTIPLE RELATIONSHIPS. THE IMPROVEMENTS OF CRF-L AGAINST UP IN ACCURACY (PRECISION) IN "EASY," "AVERAGE," AND "HARD" ARE 12.9% (22.6%), 31.7% (24.2%), AND 38.6% (46.5%), RESPECTIVELY

	Accuracy				*Precision*			
	Easy	*Average*	*Hard*	Cele-WebV	*Easy*	*Average*	*Hard*	Cele-WebV
Unary Potential (UP)	0.534	0.404	0.308	0.503	0.477	0.327	0.215	0.438
UP+Temporal (TR)	0.573	0.437	0.339	0.536	0.509	0.352	0.246	0.461
UP+Visual (VR)	0.555	0.422	0.330	0.530	0.500	0.354	0.226	0.459
UP+Spatial (SR)	0.576	0.425	0.355	0.542	0.519	0.369	0.254	0.475
CRF-L	**0.603**	**0.532**	**0.427**	**0.584**	**0.585**	**0.406**	**0.315**	**0.519**

TABLE III
WITHIN-VIDEO CELEBRITY NAMING: PERFORMANCE COMPARISON ACROSS DIFFERENT SUBSETS OF CELE-WEBV

	Accuracy				*Precision*			
	Easy	*Average*	*Hard*	Cele-WebV	*Easy*	*Average*	*Hard*	Cele-WebV
RA	0.471	0.232	0.148	0.413	0.399	0.145	0.079	0.320
TA	0.536	0.360	0.256	0.468	0.449	0.222	0.105	0.405
CGMM [17], [18]	0.535	0.403	0.317	0.499	0.511	0.341	0.214	0.440
GC [17]	0.545	0.419	0.326	0.511	0.523	0.342	0.236	0.452
FACD [27]	0.590	0.457	0.372	0.523	0.539	0.366	0.241	0.461
CRF-L	**0.603**	**0.532**	**0.427**	**0.584**	**0.585**	**0.406**	**0.315**	**0.519**

note that temporal relationship is also considered and utilized as the "cannot link" constraint for restricting the assignment of names.

Table III lists the comparison of six different approaches. The general trend is that all the approaches outperform the baselines RA and TA, and the performance gaps become wider with the increase difficulty level of the dataset. CRF-L exhibits the overall best performance across all the subsets. Using the unary potential and temporal relation alone (UP + TR) as shown in Table II, the accuracy (0.536) is already higher than all other compared approaches. A key observation is that facial similarity alone is not always reliable in Web video domain, where the appearance can be wildly different even within the same video. As a consequence, CGMM, GC and FACD, which depends heavily on facial similarity for model learning and graph construction, suffer from imprecise modeling. Although temporal relation is also considered by these approaches, the fact that the relation is utilized as a hard constraint, rather than soft constraint as in CRF-L, also limits its power in label estimation. In short, CRF-L enjoys the advantages that multiple relations, in addition to facial similarity, can softly interact to reach consensus, resulting in more robust face labeling than other approaches. By integrating multiple relationships softly, CRF-L can also effectively minimize the error propagation in message passing.

D. Between-Video Face Naming

This section verifies the performance of our approach (CRF-G), which models multiple relationships not only within videos but also among videos. The experiment is conducted on Cele-WebV*. For CRF-G, the 200 celebrities are split into 12 communities using Walktrap algorithm [29]. The community size ranges from as small as 15 videos with 4 celebrities to as large as 81 videos with 28 celebrities. Depending on which communities the tagged celebrities belong to, each video in Cele-WebV is assigned to one or multiple communities. As presented in Section IV-C, CRF-G builds an extended graph for each video based on the connections of a video with other videos in the communities. We compare CRF-G to consistency learning (CL) [22], which label faces regardless of whether names are tagged. CL builds a face model per name using images crawled from Web. In the implementation, we use the face dataset crawled from Google (see Section V-A2) as training examples for learning 200 k-nearest-neighbor (k-NN) classifiers as the face models. Here, the value of k is empirically tuned to 5, which exhibits the best performance when tested across different values of k. As in [22], the bootstrapping strategy is employed for selecting the best possible samples for learning k-NN. By doing so, there are 2,831 Web images in the face dataset being filtered out by CL. The motivation of CL can be viewed as similar to the unary potential energy in CRF-G, except with a more sophisticated way of sample selection and classifier learning.

TABLE IV
BETWEEN-VIDEO FACE NAMING. THE IMPROVEMENT OF CRF-G AGAINST CRF-L IS SHOWN IN THE PARENTHESIS

	Accuracy	*Precision*
UP	0.480	0.395
CL	0.483	0.401
CRF-L	0.561	0.503
CRF-G	**0.586** (+4.5%)	**0.521** (+3.6%)

Table IV shows the performance comparison. Recall that in Cele-WebV*, out of 2,487 faces not being tagged, there are 146 faces belonging to one of the 200 celebrities. By CRF-G, there are 79 out of these 146 faces being correctly named. Furthermore, there are also 687 incorrectly labeled faces by CRF-L being rectified, where among them 55 of the labels are due to the missing faces problem. On the other hand, CRF-G does generate false alarms, where 345 faces, correctly assigned with "null", are incorrectly labeled, and 51 faces, correctly named by CRF-L, are falsely labeled. The false labels are mostly attributed to facial features, which adjust the uncertainty of labeling (5) for "null" assignment. Overall, CRF-G outperforms CRF-L by 4.5% and 3.6% respectively in terms of accuracy and precision. Comparing to CL and UP, it is apparent that leveraging multiple relations has advantage over classifier learning. Similar to the observation as in within-face naming, visual information alone suffers from imprecise estimation due to wildly different appearances of faces across videos.

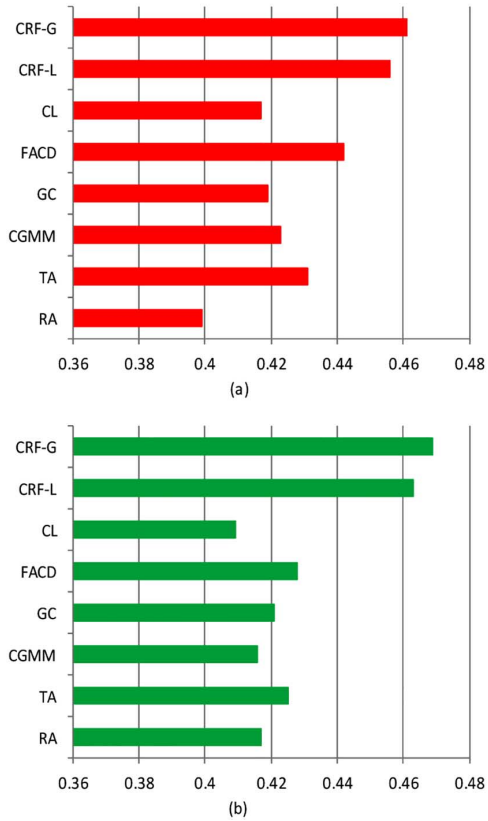

Fig. 6. Performance comparison of eight different approaches on Cele-WebV$^+$ dataset. (a) Accuracy. (b) Precision.

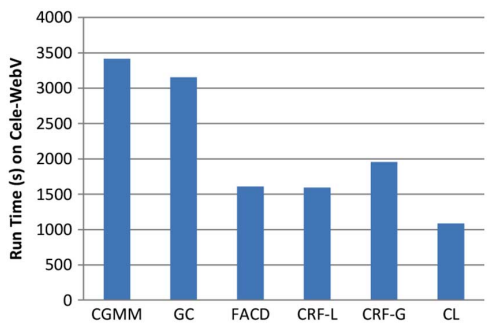

Fig. 7. Time cost (seconds) of various methods on Cele-WebV.

E. Performance Comparison

This section shows the experimental results on Cele-WebV$^+$, where no parameter tuning is allowed. Performance comparison is conducted for a total of eight different approaches listed in Fig. 6. As shown in the results, CRF-G shows the overall best performances in accuracy and precision followed by CRF-L. The performance trend is similar to that of observed on Cele-WebV and Cele-Web* datasets. To verify that the performance of different methods presented in Fig. 6 is not by chance, we also conduct significance test using randomization test [32]. The target number of iterations used in the randomization is 100,000. At the significance level of 0.05, CRF-G significantly outperforms all other approaches including CRF-L. Meanwhile, the performance of CRF-L is also significantly better than all other six compared approaches.

Fig. 7 details the online processing time of six different approaches on Cele-WebV using a PC with 8-core 2.67GHZ cpu and 20 GB memory. Note that the decomposition of celebrities into communities (in CRF-G), model learning (in CL), and construction of inverted index (in FACD) are all considered offline and the time for these operations are not shown. For CRF, online processing includes the time spent for graph construction and face naming. In CRF-L, the average size of a graph is 16 vertices (faces), 19 edges (pairwise potential) and 3.1 labels (candidate names and null label). Considering that a total of 27 minutes is required for processing a video collection of 880 hours, CRF-L is fairly efficient. When introducing external knowledge from other videos and considering celebrity relationships by social networks, the average graph size is grown to 84 vertices, 113 edges and 12.3 labels. Additional 40% of time is required for CRF-G compared to CRF-L. CGMM and GC, in contrast to CRF-G and CRF-L which process each video individually, consider all the videos and labels in one run, resulting in slower speed. Among all the approaches, CL is the most efficient in terms of online processing. The efficiency, however, is traded off by the expensive offline processing in sampling training examples for classifier learning.

F. Discussion

This subsection further discusses the factors that could impact the performance and practicality of CRF-G.

Impact of social relation. We examine the impact of social cue in face naming, by comparing CRF-G to the case when social relation is not considered. Precisely, CRF is run on a graph constructed using all the videos in Cele-WebV*, and without social relation to split the graph and adjust the unary potential (15). The experimental result shows that social relation speeds up CRF-G by 16 times, from 3,809 seconds to 241 seconds. In terms of performance effectiveness, social relation contributes 20% and 29% of improvements for the accuracy and precision of CRF-G respectively. The results basically indicate that running CRF-G on multiple smaller sub-graphs will not degrade the effectiveness, and meanwhile, significantly speed up the efficiency.

Scalability. To study the scalability of CRF-G, we further expand Cele-WebV* from 200 to 1,000 celebrities. This results in a relatively large social network, which is split into 87 communities after applying Walktrap algorithm on the network. The accuracy and precision attained for this expanded dataset are 0.556 and 0.508 respectively. The results correspond to the drops of 5.1% in accuracy and 2.5% in precision when comparing to running CRF-G on 200 celebrities. As CRF-G operates on multiple smaller communities rather the whole social network, the increase in the number of celebrities basically means more number of communities to be processed, but does not necessarily imply that the scale (in terms of the number of faces and names) of each community will also expand proportionally. As a result, the overall performance is not impacted adversely.

Variations in face appearances. Next, we investigate the robustness of CRF-G to face variations. Among the 4,564 faces in Cele-WebV*, there are 3,441 faces, or 75% of faces, being manually picked up and regarded as suffering from changes in pose, illumination and resolution. Fig. 8 shows some samples of clean and noisy faces. Labeling this subset of faces is generally challenging, for example, using unary potential alone can only attain the accuracy of 0.404 and precision of 0.260.

Fig. 8. Samples of "noisy" (bottom) versus "clean" (top) faces, due to different effects: (left to right) pose, illumination, resolution, make-up, occlusion, aging, and drawing.

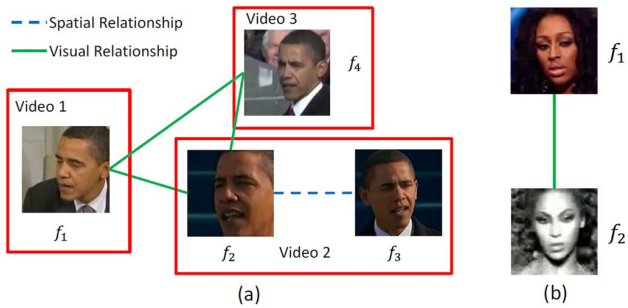

Fig. 9. True and false positives by CRF-G. (a) The face f_3 is strongly connected to the label "Barack Obama," and indirectly influences the labeling of three other faces suffering from changes in face appearances. (b) Two visually similar faces (f_1: Alexandra Burke, f_2: Beyonce Knowles) result in false labeling of f_2.

CRF-G is able to achieve the accuracy of 0.514 and precision of 0.404, which basically show the benefit of exploiting different relationships for this problem. When clean and noisy faces are linked in a graph, CRF-G possesses the capability in propagating the strong beliefs from clean to relatively noisy faces, which is the key reason leading to performance improvement. Fig. 9(a) shows an example illustrating how noisy faces can be correctly labeled. On the other hand, when two faces are incorrectly linked, as an example shown in Fig. 9(b), false labeling is also likely to happen. Comparing to the other subset of clean faces (accuracy $=$ 0.806, precision $=$ 0.784), the performance drops due to face variations are 36% in accuracy and 48% in precision. Basically, the success in labeling depends mostly on whether the right and correct relationships are established among the clean and noisy faces for message passing by CRF-G.

In modeling unary potential, we employ the multivariate Gaussian with single distribution for modeling unary potential. Considering that Gaussian mixture model (GMM) has better capability in capturing face variations, we also conduct additional experiment investigating the advantage of GMM. Using the expanded Cele-WebV* dataset with 1,000 celebrities for the experiment, a total of 96,314 Web images for these celebrities were crawled. We assume that the images of a celebrity contain ten Gaussian components. Using GMM, the accuracy (precision) is boosted to 0.569 (0.511), corresponding to 2.9% (1.9%) of improvement over the approach that does not employ GMM. Nevertheless, our analysis shows that GMM helps very little in rectifying errors due to severe face variations. We speculate that, due to the different data distributions in image and video domains, the effectiveness of GMM is limited tough helpful in capturing variations peculiar to the video domain, such as the effects of resolution and occlusion shown in Fig. 8.

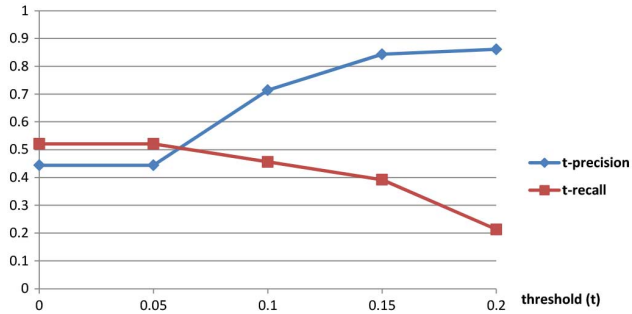

Fig. 10. Tradeoff between t-precision and t-recall when thresholding the results of CRF-G on Cele-WebV*.

Practicality. The CRF-G formulation is to maximize the conditional probability $p(\mathbf{y}|\mathbf{x})$ as in (1). In the application where it is better not to suggest the name of a celebrity than assigning an incorrect name, we can set a threshold to gate whether a name should be assigned to a face given the value of $p(\mathbf{y}|\mathbf{x})$. With this intuition, we set a threshold t such that a face $\mathbf{x}_i$ is assigned to a name indexed by $\mathbf{y}_i$ only if $p(\mathbf{y}_i|\mathbf{x}_i) \geq t$. Fig. 10 shows the tradeoff between precision and recall by thresholding on $p(\mathbf{y}|\mathbf{x})$. Note that t-precision is defined as the number of correctly labeled faces over the number of labeled faces given the threshold set as t. Similarly, t-recall measures the proportion of faces being correctly labeled out of all the faces with labels in the dataset. "Null asignment" is not considered here because the assignment means not to assign a name to a face. From Fig. 10, it can be seen that setting the threshold to the value of 0.15 can achieve a precision of 0.843, while still with a recall of 0.4. Beyond this threshold point, precision continues improving slightly but recall tends to drop significantly.

VI. Conclusion and Future Work

We have presented the modeling of multiple relationships using CRF for celebrity naming in the Web video domain. In view of the incomplete and noisy metadata, CRF softly encodes these relationships while allowing null assignments by considering the uncertainty in labeling. Experimental results basically show that these nice properties lead to performance superiority over several existing approaches. The consideration of between-video relationships also results in further performance boost, mostly attributed to the capability of rectifying the errors due to missing names and persons. The price of improvement, nevertheless, also comes along with increase in processing time and the number of false positives. Fortunately, the proposals of leveraging social relation and joint labeling by sequential video processing still make CRF scalable in terms of speed and memory efficiency.

While the overall performance of the proposed approach is encouraging, the effectiveness is still limited by facial feature similarity, which is used in the unary energy term and pairwise visual relationship. With the recent advancement in facial feature representations such as DeepFace [23] and face track [33], we plan to investigate the effectiveness of incorporating these representations into the proposed CRF framework in the near future.

REFERENCES

[1] J. Yang and A. G. Hauptmann, "Naming every individual in news video monologues," in *Proc. ACM Int. Conf. Multimedia*, 2004, pp. 580–587.

[2] 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.

[3] Y. F. Zhang, C. S. Xu, H. Q. 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.

[4] M. R. Everingham, J. Sivic, and A. Zisserman, ""Hello! My name is . . . Buffy"–automatic naming of characters in TV video," in *Proc. Brit. Mach. Vis. Conf.*, 2006, pp. 92.1–92.10.

[5] Z. Stone, T. Zickler, and T. Darrell, "Toward large-scale face recognition using social network context," *Proc. IEEE*, vol. 98, no. 8, pp. 1408–1415, Aug. 2010.

[6] L. Y. Zhang, D. V. Kalashnikov, and S. Mehrotra, "A unified framework for context assisted face clustering," in *Proc. Int. Conf. Multimedia Retrieval*, 2013, pp. 9–16.

[7] Y. Y. Chen, W. H. Hsu, and H. Y. M. Liao, "Discovering informative social subgraphs and predicting pairwise relationships from group photos," in *Proc. ACM Int. Conf. Multimedia*, 2012, pp. 669–678.

[8] J. Choi, W. De Neve, K. N. Plataniotis, and Y. M. Ro, "Collaborative face recognition for improved face annotation in personal photo collections shared on online social networks," *IEEE Trans. Multimedia*, vol. 13, no. 1, pp. 14–28, Feb. 2011.

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

[10] C. Sutton and A. McCallum, "An introduction to conditional random fields," *Found. Trends Mach. Learn.*, vol. 4, no. 4, pp. 267–373, 2012.

[11] W. Li and M. S. Sun, "Semi-supervised learning for image annotation based on conditional random fields," in *Proc. Conf. Image Video Retrieval*, 2006, vol. 4071, pp. 463–472.

[12] G. Paul, K. Elie, M. Sylvain, O. Marc, and D. Paul, "A conditional random field approach for face identification in broadcast news using overlaid text," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2014, pp. 318–322.

[13] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods (Springer Texts in Statistics)*. New York, NY, USA: Springer-Verlag, 2005.

[14] M. J. Wainwright and M. I. Jordan, "Graphical models, exponential families, and variational inference," *Found. Trends Mach. Learn.*, pp. 1–305, 2008.

[15] J. S. Yedidia, W. Freeman, and Y. Weiss, "Constructing free-energy approximations and generalized belief propagation algorithms," *IEEE Trans. Inf. Theory*, vol. 51, no. 7, pp. 2282–2312, Jul. 2005.

[16] V. F. Mert Özcan, L. Jie, and B. Caputo, "A large-scale database of images and captions for automatic face naming," in *Proc. Brit. Mach. Vis. Conf.*, 2011, pp. 29.1–29.11.

[17] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid, "Automatic face naming with caption-based supervision," in *Proc. IEEE Comput. Vis. Pattern Recog.*, Jun. 2008, pp. 1–8.

[18] T. Berg, A. Berg, J. Edwards, and D. Forsyth, "Who's in the picture?," in *Proc. Neural Inf. Process. Syst.*, 2005, pp. 137–144.

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

[20] J. K. Zhu, S. C. H. Hoi, and M. R. Lyu, "Face annotation using transductive kernel fisher discriminant," *IEEE Trans. Multimedia*, vol. 10, no. 1, pp. 86–96, Jan. 2008.

[21] J. Yang, R. Yan, and A. G. Hauptmann, "Multiple instance learning for labeling faces in broadcasting news video," in *Proc. ACM Int. Conf. Multimedia*, 2005, pp. 31–40.

[22] M. Zhao, J. Yagnik, H. Adam, and D. Bau, "Large scale learning and recognition of faces in web videos," in *Proc. Int. Conf. Automat. Face Gesture Recog.*, 2008, pp. 1–7.

[23] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the gap to human-level performance in face verification," in *Proc. IEEE Comput. Vis. Pattern Recog.*, Jun. 2014, pp. 1701–1708.

[24] D. Y. Wang, S. Hoi, Y. He, and J. K. Zhu, "Mining weakly labeled web facial images for search-based face annotation," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 1, pp. 166–179, Jan. 2014.

[25] D. Y. Wang, S. C. Hoi, Y. He, J. K. Zhu, T. Mei, and J. B. Luo, "Retrieval-based face annotation by weak label regularized local coordinate coding," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 3, pp. 550–563, Mar. 2014.

[26] D. Y. Wang, S. C. Hoi, P. C. Wu, J. K. Zhu, Y. He, and C. Y. Miao, "Learning to Name Faces: A Multimodal Learning Scheme for Search-Based Face Annotation," in *Proc. ACM Conf. Res. Develop. Inf. Retrieval*, 2013, pp. 443–452.

[27] J. Bu *et al.*, "Unsupervised face-name association via commute distance," in *Proc. ACM Int. Conf. Multimedia*, 2012, pp. 219–228.

[28] S. W. Smoliar and H. Zhang, "Content-based video indexing and retrieval," *IEEE Multimedia*, vol. 1, no. 2, pp. 62–72, Jun. 1994.

[29] P. Pons and M. Latapy, "Computing communities in large networks using random walks," *J. Graph Algorithms Appl.*, pp. 191–218, 2006.

[30] Z. Chen, C.-W. Ngo, W. Zhang, J. Cao, and Y.-G. Jiang, "Name-face association in web videos: A large-scale dataset, baselines, and open issues," *J. Comput. Sci. Technol.*, vol. 29, no. 5, pp. 785–798, 2014.

[31] J. Cao, Y. D. Zhang, Y. C. Song, Z. N. Chen, and J. T. Li, "MCG-WEBV: A Benchmark Dataset for Web Video Analysis," Inst. Comput. Technol., Chinese Academy Sci., Beijing, China, Tech. Rep. ICT-MCG-09-001, 2009.

[32] J. P. Romano, "On the behavior of randomization tests without a group invariance assumption," *J. Amer. Statist. Assoc.*, vol. 85, no. 411, pp. 686–692, 1990.

[33] B. C. Chen *et al.*, "Scalable face track retrieval in video archives using bag-of-faces sparse representation," *IEEE Trans. Circuits Syst. Video Technol.*, submitted for publication.

Lei Pang received the B.Eng degree from College of Software, Nankai University, Tianjin, China, in 2010. He is currently working toward the Ph.D. degree in computer science at the City University of Hong Kong, Kowloon, Hong Kong.

He is now with the VIREO Group, City University of Hong Kong. His research interest include multimedia content analysis, covering Web video face naming, multimedia question answering, and emotion prediction on Web videos.

Chong-Wah Ngo received the M.Sc. and B.Sc. degrees in computer engineering from Nanyang Technological University of Singapore, Singapore, and the Ph.D. degree in computer science from the Hong Kong University of Science & Technology, Clear Water Bay, Hong Kong.

He was previously a Postdoctoral Scholar with the Beckman Institute, University of Illinois in Urbana-Champaign, Champaign, IL, USA. He was also a Visiting Researcher at Microsoft Research Asia, Beijing, China. He is currently a Professor with the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong. His recent research interests include large-scale multimedia information retrieval, video computing, multimedia mining, and visualization.

Dr. Ngo was an Associate Editor of the IEEE TRANSACTIONS ON MULTIMEDIA (2011–2014). He was Conference Co-Chair of the ACM International Conference on Multimedia Retrieval 2015 and the Pacific Rim Conference on Multimedia 2014. He also served as Program Co-Chair of ACM Multimedia Modeling 2012 and ICMR 2012. He was the Chairman of the Hong Kong Chapter of ACM from 2008 to 2009.