Who is the Hero?
Semi-supervised Person Re-identification in Videos

Umar Iqbal[1], Igor D. D. Curcio[2] and Moncef Gabbouj[1]
[1]*Department of Signal Processing, Tampere University of Technology, Tampere, Finland*
[2]*Nokia Research Center, Tampere, Finland*

Keywords: Semi-supervised Person Re-identification, Important Person Detection, Face Tracks, Clustering.

Abstract: Given a crowd-sourced set of videos of a crowded public event, this paper addresses the problem of detecting and re-identifying all appearances of every individual in the scene. The persons are ranked according to the frequency of their appearance and the rank of a person is considered as the measure of his/her importance. Grouping appearances of every person from such videos is a very challenging task. This is due to unavailability of prior information or training data, large changes in illumination, huge variations in camera viewpoints, severe occlusions and videos from different photographers. These problems are made tractable by exploiting a variety of visual and contextual cues i.e., appearance, sensor data and co-occurrence of people. A unified framework is proposed for efficient person matching across videos followed by their ranking. Experimental results on two challenging video data sets demonstrate the effectiveness of the proposed algorithm.

1 INTRODUCTION

At public events people usually record videos as a user generated content, and one of the general problems of consumers is that they do not know how to edit the content and find a video sequence that contains the most important person they have been shooting. The objective of this paper is to analyze the crowd-sourced videos of a single event to detect important/mainstream persons appearing in the event. The importance of an individual is subjective and it can vary considerably from one person to another. However, in some cases it can also be generalized based on some fair assumptions, such as the fact that in public events people usually capture notable individuals. For instance, singers/performers in concerts, bride/groom during a wedding ceremony, etc. Hence, these persons happen to appear relatively often in the data. It can be considered that the person who is captured by most of the people has high importance among the majority. For example, videos captured at an indoor public concert can be seen in Figure 1. By looking thoroughly on the example videos, we can find occurrences of the same persons in multiple videos, despite the fact that these videos are captured by different people.

An automatic detection of such individuals from multiple videos has a variety of applications that can easily be realized in multimedia content retrieval, automatic video remixing, etc. In this work, we refer to these individuals as “Important Persons”, and all others are called “Casual Persons”. The problem posed here is similar to person re-identification where appearances of each person across multiple videos are detected and identified. Subsequently, a method to rank them according to the amount of occurrence is needed to finally detect the important persons.

Figure 1: Examples of videos captured at the same event but by different photographers.

From Figure 1, a large amount of appearance variation can be seen. To tackle with these challenges, we aim to utilize multiple visual cues. In addition to traditional face and clothing color information, we also exploit the semantic information about the person’s appearance by detecting several high-level facial attributes (e.g., ethnicity, hair colors, etc.). Contex-

Iqbal U., D. D. Curcio I. and Gabbouj M..
Who is the Hero? - Semi-supervised Person Re-identification in Videos.
DOI: 10.5220/0004738801620173
In *Proceedings of the 9th International Conference on Computer Vision Theory and Applications* (VISAPP-2014), pages 162-173
ISBN: 978-989-758-004-8

tual data obtained from electronic compass is utilized to analyze the camera movements and ease the processing of larger videos. Finally, information about co-occurrence of individuals is utilized to develop a semi-supervised Hierarchical Agglomerative Clustering (HAC) algorithm for grouping the individuals.

The contribution of our work is twofold. First, a unified framework is proposed: this does not require any prior information about the individuals. The framework is completely automatic and does not require any human intervention. Second, we demonstrate how usage of multiple visual modalities and contextual cues can be exploited to enhance the performance of persons matching. Experimental results on two video data sets demonstrate the effectiveness of the framework and ensure that the proposed system provides competitive results as compared to the state-of-the-art algorithms.

1.1 Related Work

Recently, an increased amount of research has been carried out in the direction of person re-identification and clustering of individuals in videos or images. In this regard, the closest prior art is that of (Barr et al., 2011), where the most appearing persons are detected from the videos recorded in various capturing conditions. First, the face tracks are pre-processed to eliminate outliers followed by the grouping of the same individuals using HAC. In result, persons appearing in videos more than a certain threshold are considered as the most appearing persons. We build upon the similar idea and further extend it to incorporate multiple visual and contextual cues.

The problem of important person detection shares many common properties with identity specific video indexing. Recently, (Hao and Kamata, 2012) have proposed an automatic algorithm for individual retrieval from videos. Information about co-occurrence of individuals is utilized to gather training data, followed by the learning of a distance metric to perform person matching. (Bäuml et al., 2013) utilize the transcription and subtitles of TV videos to obtain weakly labeled data and use it along with other contextual and visual cues for person re-identification. Similary, (Cinbis et al., 2011) also learn a self-supervised similarity metric from face tracks of the characters appearing in TV-videos. Other works focusing on similar problems include constraint propagation based unsupervised person re-identification (Tao and Tan, 2008) and a divide and conquer based strategy (Gou et al., 2012). However, most of these works are targeted to TV videos captured by professional cameramen. Unlike crowd-sourced videos, TV videos are more structured, and often contain more close-up scenes. Moreover, a very little variability in video quality can be found from one episode to another. Hence, person detection and matching is easier in these scenarios. We, on the other hand, focus on videos recorded by amateurs with various hand-held cameras, which implies more challenges as compared to the aforementioned works. In addition, in case of crowd-sourced videos, no prior information or subtitles can be obtained. Therefore, no training data is available.

Many photo/video album organization methods rely on face and clothes information to find similarities in people, as proposed in (Zhang et al., 2003) and (Sivic et al., 2006). In addition to visual cues, (Suh and Bederson, 2004) also utilize the time stamp information to group images that belong to the same event, and perform person matching based on clothing color information. Recently, a very interesting work for the automatic face association in photo albums has been proposed by (Lo Presti and La Cascia, 2012) where an online learning method is employed to group individuals using face information. However, all these works are intended for the collection of images and are not directly applicable to videos.

Furthermore, all previously mentioned works rely only on face or clothing color information as visual cues. We, in addition, utilize also the high-level facial attributes (e.g., gender, age, eye-wears, etc.) as they provide very strong clues about the appearance of a person and are proven to be robust against face pose variation (Kumar et al., 2011). This can help especially for videos that are captured in a same event.

1.2 Proposed Framework

The schematic diagram of the proposed framework is depicted in Figure 2. The framework is composed of four processing units. We start with the temporal video segmentation to divide larger videos, for which the sensor data is available, into smaller subsequences (Sec. 2.1). The second step represents each person with three appearance models; facial feature, clothing color and high-level attributes. First, we obtain face tracks by detecting and tracking faces in consecutive frames and subsequently form clothes tracks by considering a bounding box below every face element of the face tracks (Sec. 2.2). Afterwards, we extract useful features for all three appearance models to be used in the identification process (Sec. 3.1). The third step utilizes the extracted features to perform person grouping (Sec.4) and assigns unique identities to the grouped persons. Finally, in the fourth step, we rank every individual according to a criterion and

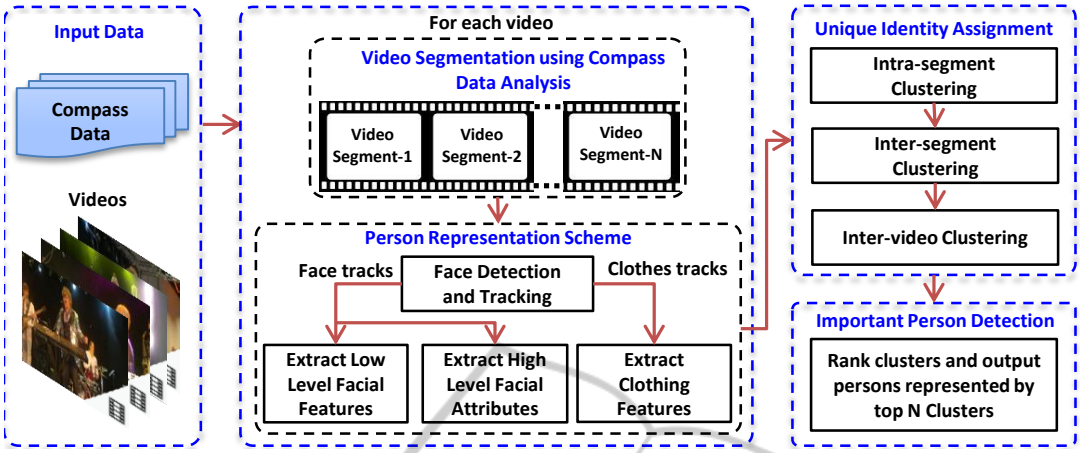

Figure 2: Schematic diagram of the proposed framework.

output the persons represented by top N clusters as the most important persons (Sec. 4.3).

2 VIDEO ANALYSIS

We start with the temporal video segmentation to divide the larger videos into smaller sub-sequences that are then passed to the face tracking algorithm to obtain face and clothes tracks. The rest of this section explains these steps in detail.

2.1 Temporal Video Segmentation

In addition to tractability of larger videos, this step ensures that the clustering within a sub-sequence is accurate and wrong clusters are not propagated into the next clustering stages. For this purpose, an obvious choice is to analyze the camera motion and divide the videos where shot changes are detected. Unlike traditional techniques that are based on content analysis of videos and therefore are computationally expensive, we adopt the sensor based camera panning detection and classification algorithm proposed by (Cricri et al., 2012). Their technique utilizes the compass orientations, provided by the built-in electronic compass available in modern camera devices, to analyze the horizontal motion of hands and detect the camera panning. The algorithm is computationally very efficient and does not require content analysis. Therefore, it is robust to object movements in videos.

Given the compass orientations (temporally aligned with the content of the videos) with respect to magnetic North, first, a low-pass filter is applied on the raw compass data to eliminate peaks due to shaky camera movements. Afterward, a first-order derivative is computed to analyze camera movements. Peaks that are greater than a certain threshold are considered as the points where camera panning is detected. Subsequently, camera panning is classified as slow or fast based on its speed; gradual change in compass orientation represents slow panning, whereas rapid change corresponds to fast camera panning. Slow panning represents that the photographer is following an object or trying to cover a panoramic scene, whereas fast panning corresponds to the photographer's intention to change the whole scene. We exploit this observation and divide the video from all the points where fast panning is detected.

2.2 Face Detection and Tracking

We employ a detector-assisted particle filters based multi-view face tracking approach, similar to the work of (Bauml et al., 2010), to exploit the temporal information in videos. We utilize the readily available implementation of Local Binary Pattern (LBP) based face detector of OpenCV (Bradski, 2000) and integrate it into the tracking algorithm as explained next.

We train several face detectors at different pose angles to detect and track faces under pose variations. The detectors are trained at the following angles;

$$\theta = \{0, \pm 15, \pm 30, \pm 45, \pm 60\} \quad (1)$$

Five detectors $(0, \pm 30, \pm 60)$ are run in parallel over the entire frame after every k (k = 10) frames. A detection is considered legitimate if at least three detections are spatially close to each other and far enough from already known faces. Afterward, we initiate an independent particle filter, consisting of 1000 particles, for every detected face to track them in the remaining video. The state of each particle consists of the location (x, y) of the face, size and the yaw angle

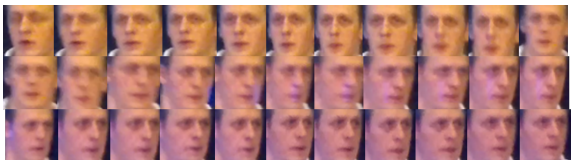

Figure 3: Sample face track generated by the detector assisted face tracker. We can see the variations in appearance due to changes in facial pose, noise due to motion, and illumination changes.

of the face as follows

$$\boldsymbol{x} = [x, y, s, \alpha] \tag{2}$$

The particles are propagated based on a 2^{nd} order auto-regressive process. We compute the weights w_i of the particles by running a face detector at the region defined by the state $\boldsymbol{x}$ of every particle. For this purpose, we chose the detector that has the lowest angular distance between its pose angle, at which it is trained, and the particle's yaw-angle α. This ensures that the particles whose pose angles best describes the current face pose are assigned the higher weights. As the classifiers used for face detectors are insensitive to small localization errors, each detector gives multiple overlapping detections around the face regions. This normally happens because of running detectors at different scales and multiple locations within a region. Such overlapping detections usually appear with fewer consistencies for less confident face regions or background patches. We use this assumption and consider the number of such overlapping detections as the weights w of the particles;

$$D_\rho = \arg\max_\rho(\alpha - \rho), \rho \in \theta \tag{3}$$

$$w = |D_\rho(\boldsymbol{x})| \tag{4}$$

A tracker is terminated when no detection is found in the regions defined by its particles. Furthermore, care has to be taken to avoid identity switching of overlapping tracks that is when particles from different trackers come in the vicinity very close to each other; they may be scored on the same track or be replaced with each other. For this purpose, we adopt a very simple technique such that if the distance between the centers $\overline{X}_1$ and $\overline{X}_2$ of two trackers becomes less than a certain threshold τ, we terminate one of these tracks;

$$\|\overline{X}_1 - \overline{X}_2\| < \tau \tag{5}$$

The tracks are re-initialized once they become far enough from each other. This gives us several disjoint face tracks for the same individual. An example of face track can be seen in Figure 3.

2.3 Upper-body Detection

For every face track, we form clothes tracks by detecting the upper-body region utilizing the spatial information of the face regions. This is done by taking a bounding box below every face element. Although body detection in this case is completely dependent on the face detector, and we are not able to detect persons whose faces are not visible, as in (Sivic et al., 2006), we use clothing information to aid the identification process. An independent body detection and tracking algorithm can be used to enhance the recall of person detection and the proposed framework can directly benefit from it. In the rest of this paper, a combination of face and clothes track will be referred as person track.

3 APPEARANCE MODELS FOR PERSON IDENTIFICATION

Once the person tracks are obtained, we represent each of their elements with three appearance models as shown in Figure 4.

3.1 Facial Features

It is a well-known fact that images of different identities in the same pose are more similar compared to images of the same identity in different poses. Despite a face track comprises more than one image of a person's face, it is still unsure whether all possible face poses and expressions are available. Moreover, faces in a face track are temporally related and contain less appearance variations as compared to face tracks of the same person but extracted from a different video. Hence, to cope with these challenges, we represent every face track with low-level textural features and high-level facial attributes. Low-level features help to extract the underlying texture of the person's face, whereas high-level attributes provide the semantic information about his/her appearance.

3.1.1 Facial Landmarks Detection and Face Alignment

Before feature extraction, we align all face regions such that their eyes and mouth appear at fixed spatial locations. For this purpose, eyes and mouth are detected using the deformable part models based facial landmark detector by (Uřičář et al., 2012) and its publicly available implementation is used. Alignment is done by warping the face image using a 2D Affine transformation matrix. In addition, under the

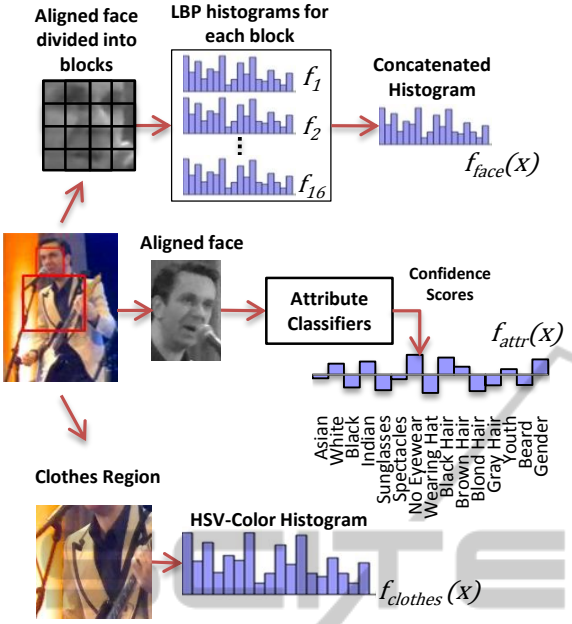

Figure 4: Example of Person Representation by different appearance models.

assumption of facial symmetry, every face is rotated to the negative yaw angle, utilizing the pose information provided by the face tracker. This little tweak helps in the identification stage as the features are always extracted from the good side of the face.

3.1.2 Low-level Facial Features

We first resize every aligned face sample to a fixed size of 80×80 pixels and convert it to grayscale. Subsequently, we apply LBP operator over the entire image to extract the underlying texture of the face (Ahonen et al., 2004). LBP is proven to be robust against illumination variation and is computationally very effective. At a given pixel location the LBP operator is defined by the binary comparisons between the pixel intensity with the intensities of its P neighboring points lying on the circumference of a circle defined by the radius R. For P=8 and R=1, a 59 bin histogram is often used to represent the frequencies of binary patterns in a given face image (58 for uniform patterns and 1 for the remaining (Ahonen et al., 2004)).

To include the spatial information, we divide the image into 4×4 blocks and extract a 59-bin histogram from each block. A final feature vector, f_{face}, of length 944 (4×4×59) is formed by concatenating all block-wise histograms as depicted in Figure 4.

3.1.3 High-level Attributes

We have selected 15 most evident attributes (Table 1)

Table 1: Selected attributes and 5-fold cross validation accuracies of their classifiers.

Attribute	Acc. (%)	Attribute	Acc. (%)
Asian	81.79	Wearing Hat	97.96
White	86.01	Black Hair	94.45
Black	94.22	Brown Hair	94.04
Indian	88.34	Blond Hair	94.42
Sunglasses	98.05	Gray Hair	96.80
Spectacles	94.84	Gender	95.84
No-Eyewear	94.72	Youth	92.82
Beard	94.10		

that cannot vary considerably in an event. However, more attributes can also be considered, such as in (Kumar et al., 2011), where 73 attributes are used for person identification. Our aim is to label every face element with the confidence values about the presence of the selected attributes. For this purpose, we train binary classifiers for each attribute and directly use the confidence values provided by the classifiers as the features. To automatically train the attribute classifiers we adopt the framework proposed by (Kumar et al., 2011) that is briefly explained next.

The classifiers are trained in a supervised manner. The training samples (500 to 2000 for each attribute) are gathered from two publicly available datasets; PubFig (Kumar et al., 2011) and FaceTracer (Kumar et al., 2008). Images are manually labeled by multiple persons and have been cross-checked once. First, we align all training samples following the same technique as described in Sec. 3.1.1. However, in this case we select a slightly larger face region to keep the hair and chin visible. Different pixel value types and face regions can be crucial to efficiently learn a classifier for a particular attribute. For example, for the attribute "Eyeglasses", regions around the eyes are more crucial than the regions like cheeks, hairs, etc. Therefore, we divide every face region into 13 functional parts i.e., eyes, nose, forehead, etc. Subsequently, each face part is converted to various pixel types including different color spaces (RGB, HSV, and Image Intensity), edge magnitudes and orientations. Furthermore, the extracted pixel values can be further normalized for better generalization and robustness to illumination variations and can also be aggregated in different forms, i.e., raw pixel values, histograms and mean variances. In total, we obtain 585 different combinations of face region, pixel value type, normalization and aggregation type. The aim is to select the best combinations for classification of a particular attribute.

For this, we train Support Vector Machine (SVM) classifiers with RBF kernel, using LibSVM (Chang

and Lin, 2011), for each combination and select the best six combinations using a forward feature selection algorithm based on their combined cross-validation accuracy. The parameters C and γ for SVMs are selected using the grid search. The cross-validation accuracies of the classifiers are given in Table 1. For further detail on feature selection and attribute classifiers, the readers can refer to (Kumar et al., 2011). A pictorial explanation of the features extracted using the attribute classifiers can be seen in Figure 4.

In addition, this is to note that adding high-level attributes for face representation increase the computational complexity of the feature extraction module (slightly more than twice as compared to the combination of low-level facial and clothing color features). However, the aim of using them in this paper is to demonstrate how they can contribute to person identification, and achieving low computational complexity is not the main objective here.

3.2 Clothing Color Features

Clothing color information can provide very important clues about the appearance of a person, particularly within a single video, or videos captured at the same event. We extract 3-dimensional HSV color histogram from every element of clothes tracks that are then used to represent the clothing color information of the persons. To deal with partial occlusion and avoid background regions, we estimate the color distribution $f_{clothes}$ of the clothes region utilizing the weighted kernel profile proposed in (Comaniciu et al., 2003). The weighting kernel assigns smaller weights to the pixels farther from the center of the clothes region, as these pixels are more prone to occlusions and background variations. The color distribution $f_{clothes}$ is defined as follows: Let z_c be the center of the clothes region and z_i be the location (x, y and z coordinates) of a pixel in this region. The weighting kernel k, to assign smaller weights to the pixels farther from the center, is defined as

$$k(r) = \begin{cases} 1-r^2, & \text{if } r < 1 \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

where r is the normalized distance between the pixel location z_i and the region center z_c. Then the m-bin weighted color histogram $f^u_{clothes}, u = 1,...,m$ of the clothes regions centered at z_c would be

$$f^u_{clothes} = C\sum_{i=1}^{n} k(\| \frac{z_c - z_i}{h \times w} \|^2)\delta[b(z_i) - u] \tag{7}$$

In eq. 7, $b : R^3 \rightarrow \{1,...,m\}$ is a function that gives the index, for the pixel at location z_i, of the bin in the histogram. The term n gives the total number of pixels in the clothes region, δ is the Kronecker's delta, h and w are the height and width of the region respectively. C is the normalization factor to ensure that distribution is legitimate, i.e., $\sum_{u=1}^{m} f^u_{clothes} = 1$ and is defined as

$$C = \frac{1}{\sum_{i=1}^{n} k(\| \frac{z_c - z_i}{h \times w} \|^2)} \tag{8}$$

The weighted color histograms from every clothes element will be represented as $f_{clothes}$ in the rest of this paper.

4 PERSON TRACK CLUSTERING

We employ a Semi-supervised HAC algorithm to group the person tracks of the same individuals. HAC takes a distance matrix and forms a dendrogram (tree), where a pair of clusters (person tracks) is merged at each level. We begin from the leaf node (each leaf node represents a singleton person track) and start merging the closest pairs at each level. Finally, the last level (root node) represents a single cluster that contains all the person tracks. The levels of the dendrogram represent the distances between clusters such that the clusters that are near the leaf nodes are more similar. We perform HAC at three different steps:

- First, all person tracks within the sub-sequences are grouped.
- Afterwards, within-video clustering is performed. In case the video is divided into sub-sequences, the clusters obtained from all sub-segments are grouped. If the video is not segmented, clustering is done for the person tracks from the complete video.
- In the third stage, clusters originated from all videos are clustered to globally group the same individuals.

The distance d between two clusters, C_i and C_j, is calculated as follows:

As a cluster can contain more than one person tracks P_k, we decompose all person tracks, within a cluster, and aggregate their elements into a single set $S = \{p_1, p_2, ..., p_n\}$, $n = \sum_k |P_k|$, $\forall P_k \in C_i$ where each element of the set consists of a face, attribute and clothing color feature vector. Once the sets S_i and S_j are formed for each cluster, we find N closest pairs of elements from both sets and form a new set Q of these pairs as $Q = \{\{a_1, b_1\}, \{a_2, b_2\}, ...\{a_N, b_N\}\}, a \in S_i, b \in S_j$.

The mean of the distances between these pairs is taken as the distance between two clusters as follows:

$$d(C_i,C_j) = \frac{1}{N}\sum_{n=1}^{N} f(a_n,b_n) \quad (9)$$

where $N = min(|S_i|,|S_j|)$ to ensure that the distance is not biased toward the set with higher length. We define the distance $f(a,b)$ as the weighted sum of distances according to each appearance model as follows;

$$f(a,b) = d_{face} * w_{face} + d_{attr} * w_{attr} + d_{clothes} * w_{clothes} \quad (10)$$

where d_{face} and d_{attr} are the distances between facial and high-level attributes respectively and are taken as the Euclidean distance between their feature vectors. $d_{clothes}$ is the Bhattacharya distance between clothes patterns of both elements a_n and b_n. The weights w_{face}, w_{attr} and $w_{clothes}$ are the weights assigned to each model. The discussion on weight selection is given in Sec. 5.3. To ensure that no modality takes higher weight than the one assigned to it, we normalize all feature vectors such that their distances range between 0 and 1.

Once the symmetric distance matrix, $D(i,j) = d(C_i,C_j)$, from the distances between all clusters is computed, HAC can be performed with any linkage criterion to develop a dendrogram. However, by utilizing the uniqueness constraints, there are possibilities to further enhance the quality of clustering as described in the next Section.

4.1 Uniqueness Constraints

If two persons appear in overlapping frames, it is sure that they represent different identities. This little information is extracted directly from the frame numbers of the person tracks, and the distance matrix for the HAC is slightly changed to enforce these constraints. The distance matrix can be seen as a fully connected graph where each node C_i is connected to others C_j with a weight $d(C_i,C_j)$. Our aim is to update the graph such that the distance between all the persons that appear in overlapping frames is increased to infinity. For this, we create a new distance matrix D' as

$$D'(i,j) = \begin{cases} \max(D)+1, & \text{if } C_i \cap C_j \neq \phi \\ D(i,j), & \text{otherwise} \end{cases} \quad (11)$$

In D' the distance between all overlapping person-tracks is increased. However, updating it this way results in the loss of its metricity. Moreover, the triangular inequality is also violated, and therefore chances are there that two clusters, with uniqueness constraint as true, may be clustered due to any other connecting node. To ensure that none of the clusters with uniqueness constraints be merged, we perform HAC with complete linkage (Klein et al., 2002). Complete linkage always considers the maximum distance between the elements of two clusters, and therefore, enforces the uniqueness constraints. Complete linkage also gives more compact cluster that decreases the chances of grouping person tracks of different individuals. The resulting dendrogram is more optimized and does not cluster co-occurring persons.

We use uniqueness constraints in all three stages of HAC. First they are used for all the person tracks which appear in the overlapping frames in a sub-segment. In the second stage, constraints are used for all the clusters originating from the same sub-segment. This ensures that clusters from first clustering level are not merged in the later stages. Subsequently, constraints are also used at the last clustering stage, such that the clusters originating from the same videos are not merged again.

4.2 Cutoff Selection

A cutoff level has to be selected to achieve the optimal clustering. Our goal is to select the optimal level where the clusters are homogeneous with respect to identity, and are less redundant. For this reason, we select a cutoff level that minimizes the ratio between intra-cluster distances and inter-cluster distances. This ensures that the person tracks within a cluster are closer to each other and are far from the person tracks in other clusters. To compute the intra-cluster distance for cluster C_i we form the set $S_i = \{p_1, p_2, ..., p_n\}$, as done before. The intra-cluster distance is then defined as

$$dist_{intra}(C_i) = \frac{1}{\sum_{i=1}^{n} i}\sum_{i=1}^{n}\sum_{j=i+1}^{n} f(p_i,p_j) \quad (12)$$

Inter-cluster distance, $dist_{inter}$, is calculated by taking the mean of pairwise distances between all clusters at that level. The distance between a pair of clusters C_i and C_j is computed in the same way as done before in eq. 9. Finally, we select the cutoff level that minimizes the following cost function:

$$J = \beta\frac{\sum_{i=1}^{n} dist_{intra}(C_i)}{dist_{inter}} + (1-\beta) * c \quad (13)$$

where c is the total number of clusters formed at a given level, and the constant β defines the trade-off between clustering accuracy and redundancy. The larger value of β will result in very compact and accurate clusters but with larger redundancy. On the other

hand, smaller β will contain less redundant clusters but with less homogeneity. As the uniqueness constraints are used at all stages of clustering, the value of β should be selected carefully such that no errors are propagated to the next stages of clustering. If two person tracks belonging to the same person, within a sub-segment, are not clustered during the first stage, then they will not be merged in later stages due to the uniqueness constraint. The first and second stage of clustering leverages more benefits from usage of multiple modalities as the clothing color and attribute features would be homogeneous. Therefore, the clustering can be performed with more confidence in these stages. Hence, for the first and second stage, we select a higher value of β, and a slightly lower for the third stage.

4.3 Global Ranking of Individuals and Important Person Detection

After the completion of all clustering stages, each cluster represents a unique individual. To detect important persons, we rank all individuals based on the count of sub-sequences from which the person tracks in a cluster originate. Finally, persons represented by the first N clusters are taken as important persons and all others are classified as casual persons. The experimental results of the complete framework are discussed in detail in the next section.

5 EXPERIMENTAL RESULTS

We evaluate the performance of the proposed framework, based on four performance metrics, over two video datasets − a single-event and a multi-event dataset.

5.1 Performance Metrics

For the evaluation of clustering performance, we use the same quality metrics as used in (Barr et al., 2011). This includes Self-Organization Rate (SOR) and Cluster Conciseness (CON).
SOR gives the information about the homogeneity of clusters, such that the amount of data samples that are grouped into correct identity. SOR is described as follows;

$$SOR = (1 - \frac{\sum N_{AB} + Ne}{N}) \tag{14}$$

where N_{AB} indicates the number of data samples representing person-A that are grouped into a cluster dominated by the samples of person-B. N_e denotes the number of person tracks that are assigned to a cluster in which no person dominates with more than half, and N represents the total number of person tracks available for the clustering. Higher value of *SOR* represents more homogeneous clusters and higher accuracy of clustering.

CON provides the information about redundant clusters such that if more than one cluster represents the same person, then all clusters except one are redundant. If person tracks representing person-A show majority in C_A clusters, then $C_A - 1$ of those clusters are redundant. The total number of redundant clusters C_r is given by

$$C_r = \sum_A C_A - 1 \tag{15}$$

and the CON is defined as follows;

$$CON = (1 - \frac{C_r}{C}) \tag{16}$$

where C is the total number of clusters obtained after the final stage of clustering. Similar to the *SOR*, higher values of *CON* represent good clustering efficiency.

Since our goal is to detect important persons, it is possible that an important person is not detected as important or vice versa. Detection rate of important persons is captured by False Positive Rate (FPR) and False Negative Rate (FNR). FPR represents the ratio of casual persons mis-classified as important. FNR denotes the ratio of important persons that are missed by the system (not classified as important). Both FPR and FNR range from 0 to 1 and the lower value of both measures shows better detection accuracy.

5.2 Datasets Detail

In order to assess the performance of the proposed architecture, a dataset containing videos from five unique crowded events is collected. We refer to this dataset as single-event dataset in the rest of this paper. To compare the proposed framework with the state-of-the-art, we also evaluate it on a publicly available multi-event dataset, named as ND-QO-Flip, proposed in (Barr et al., 2011). Both datasets exhibit unique properties and complexities as described next.

5.2.1 Single-event Dataset

Videos are recorded by different photographers from different distances, view angles and using various mobile phone cameras (i.e., Nokia Lumia 800, Nokia Lumia 900, Nokia Pureview 808 and Nokia N8). The length of the videos varies from 1 minute up to 3 minutes. The number of videos in a unique event varies from three to seven. Similarly, the number of persons appearing in these videos also differs with the event.

Figure 5: Example video frames from single-event dataset, each row representing a unique event.

Figure 6: Example video frames from ND-QO-Flip dataset.

Among the five events, one is recorded in the real indoor public concert, and all others are simulated by us. Videos in the public concert are completely unconstrained and are captured from different view-

points around the stage. In this event, the five band members appear in almost all videos, whereas the audience varies from video to video. Hence, our aim is to detect band members as important persons. In simulated events, three are recorded outdoor in different weather conditions (i.e., sunny, rainy, etc.) and one is recorded indoor. Few (one to three) of the persons in these videos are considered as important, and videos are recorded such that important persons appear in more videos than others. The appearance of the persons varies across the videos as all photographers were situated at different locations and view angles. Furthermore, all crowd members were allowed to make any kind of facial expressions and vary their facial and body pose. Photographers were allowed to move, zoom & pan the camera but under the constraint that important persons appear in more videos. The resolution of the videos is either Full-HD (1920×1080) or HD (1080×720). Example video frames from single-event dataset can be seen in Figure 5.

5.2.2 Multi-event Dataset

The multi-event dataset evaluates the proposed framework in cases when the videos are not recorded in the same event. In such cases, the situation normally becomes even worse as the clothing color, and facial appearance can vary across the videos. The dataset consists of fourteen 25-59 second videos recorded during a period of seven months under various weather and capturing conditions. Unlike our dataset, creators of the ND-QO-Flip dataset have restricted the facial pose to near frontal, which reduces the complexity of the dataset. However, crowd members were allowed to vary the facial expressions. The dataset contains 90 subjects overall, five of them appeared in more than one video and all others appeared in a single video. Hence, for this dataset, the goal is to detect five persons who appear in multiple videos. Example video frames from single-event dataset can be seen in Figure 6.

5.3 Weights for Each Appearance Model

The weights, w_{face}, w_{attr} and $w_{clothes}$, for each modality in eq. 10 should be assigned differently for each dataset due to their different nature and complexities. For the single-event dataset, we assign fixed weights for all stages of clustering. It is due to the assumption of same clothing of individuals across different videos. The weights are selected such that the biometrics such as low-level facial features are the most discriminating features among all modalities. Therefore, the higher value is taken for w_{face}. On the other hand, high-level attributes and clothing colors can be similar for two different individuals. Hence, we assign a relatively low weight to high-level attributes and minimum weight to clothing features.

The clothing information in the multi-event dataset is not the same across different videos. Therefore, for the last stage of clustering, we eliminate the clothing color information. For comparison of different combinations, we also use the combination of face and clothing color information. In this case, we assign a higher value to w_{face} and a relatively lower one to $w_{clothes}$.

5.4 Results

To verify that the usage of multiple modalities and uniqueness constraints really aid person identification, we start with a baseline method that utilizes only the facial features for person representation. Afterwards, new features are added to the baseline one by one until we reached the proposed framework that utilizes all three modalities and uniqueness constraints.

5.4.1 Results on Single-event Dataset

For the single event dataset, results are evaluated on each event separately. Moreover, we also calculate the averages of all quality measures in order to have a holistic picture of the performance as given in Table 2. We can see how the final results depict the effectiveness of all intuitions discussed earlier. The increase in SOR and CON with the introduction of new modalities into the baseline is clearly evident. The success of using clothing color can also be seen by the increase in SOR and CON and also the decrease in the FNR for important person detection. However, the use of clothing color slightly increases the FPR. This is likely due to the merging of persons with similar clothing color. Despite the usage of high-level attributes with facial features, the overall results do not show a significant increase in performance. However, the capability of high-level attributes can

Table 2: Experimental results on the single-event dataset, averaged over all events.

Method	SOR	CON	FPR	FNR
Only Face Features	0.67	0.56	0.42	0.32
Face+Clothes	0.70	0.60	0.48	0.29
Face+Attributes	0.67	0.58	0.41	0.32
Face+Clothes+Attributes	0.71	0.60	0.39	0.28
Face+Clothes+Attributes+ Uniqueness constraint	0.74	0.65	0.29	0.18

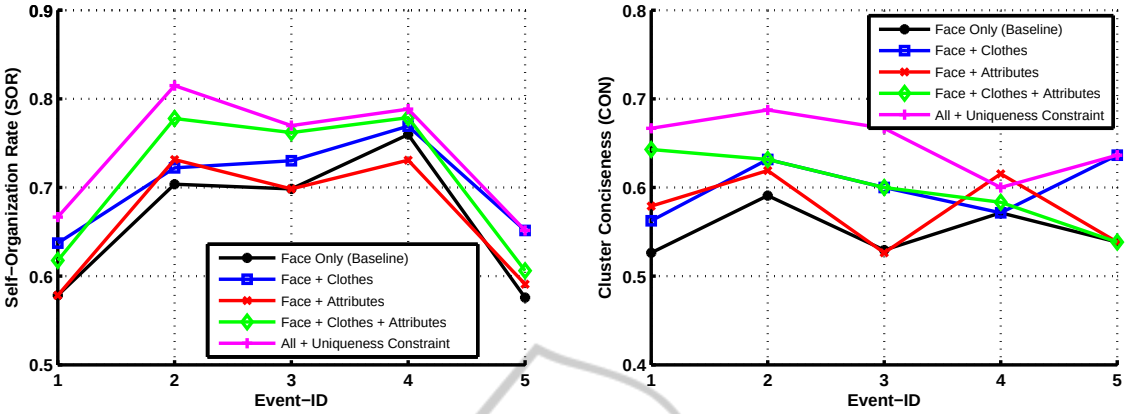

Figure 7: Experimental results obtained using different methods over all events of the single-event dataset. (Left) Comparison of the SOR using different methods. (Right) Comparison of the CON using different methods.

be seen on the results of individual events as seen in Figure 7 where comparisons between SOR and CON obtained using all methods on individual events are given. The usage of high-level attributes increases the performance of clustering for most of the events. Similarly, the combination of all three modalities also enhances the performance as compared to the other methods. The usage of the uniqueness constraint also shows promising results and appears to be very effective in almost all events. Finally the proposed algorithm that utilizes all three modalities and the uniqueness constraints increases the SOR from 0.67 (baseline) to 0.74. Also the FPR and FNR are decreased to 0.29 and 0.18 as compared to 0.42 and 0.32, achieved using the baseline method. All these results clearly demonstrate the potency of using multiple modalities and uniqueness constraints in unconstrained environments where there is no restriction on human pose, lightening conditions, movements, etc.

5.4.2 Results on Multi-event Dataset

Table 3 summarizes the results obtained over the multi-event dataset. We can see that usage of multiple modalities again works well and improves the performance. A clear increment of all quality measures can be seen from the baseline to the proposed algorithm. We are able to detect all important persons of the multi-event dataset with SOR, CON and FPR equal to 0.95, 0.70 and 0.04 respectively. These results demonstrate the capacity of the proposed algorithm in videos captured in different illumination, weather conditions, occlusions and severe appearance variations. In this case, no clothing color information is used for inter-videos clustering.

Table 4 gives a comparison between the results obtained by our framework and the one stated by (Barr et al., 2011). We can see that our approach achieved

Table 3: Experimental results obtained over the multi-event dataset.

Method	SOR	CON	FPR	FNR
Only Face Features	0.89	0.69	0.05	0.40
Face+Clothes	0.92	0.73	0.06	0.60
Face+Attributes	0.91	0.71	0.05	0.40
Face+Clothes+Attributes	0.94	0.73	0.04	0.20
Face+Clothes+Attributes+ Uniqueness constraint	0.95	0.70	0.04	0.00

Table 4: Comparison of the proposed algorithm with state-of-the-art for the multi-event dataset.

Method	SOR	CON	FPR	FNR
Proposed	0.95	0.70	0.04	0.00
(Barr et al., 2011)	0.96	0.66	0.06	0.00

almost equal SOR and higher value of CON. Moreover, FPR is also lower than achieved by their method. This shows that our method gives reduced amount of redundant clusters and also provides low number of false positives. However, it is important to note that this comparison is not completely well-founded due to the differences in face-tracks caused by different face detection and tracking algorithms.

Despite the results obtained for multi-event dataset are relatively better than the one obtained on single-event dataset, it should be remembered that the facial pose in this case is restricted to near frontal. This shows the increase in complexity due to the variations in face pose and camera view angles.

6 CONCLUSIONS

In this paper the problem of semi-supervised person re-identification, with application to important person detection was addressed. A standalone frame-

work was proposed that utilizes several visual modalities and contextual constraints to group the occurrences of every individual across different videos. Experimental results on two challenging datasets illustrate the effectiveness of usage of multiple modalities. The use of clothing colors and high-level attributes demonstrates encouraging results and provides sufficient increase in the performance. Similarly the combination of all modalities (face, high-level attributes, clothing color) showed promising results. Enhancements have been achieved by enforcing the uniqueness constraints into the clustering algorithm. The final approach that utilizes all modalities and uniqueness constraints exhibits a clear increase in performance for both datasets. Experimental results validate the performance of the proposed framework on various challenging situations, emphasize on the importance of face pose variations in real life scenarios and encourage us to strive for better person representation techniques.

ACKNOWLEDGEMENTS

Authors are thankful to the band Eternal Erection and other crowd members for allowing us to use their videos in this research.

REFERENCES

Ahonen, T., Hadid, A., and Pietikainen, M. (2004). Face Recognition with Local Binary Patterns. In *European Conference on Computer Vision*.

Barr, J. R., Bowyer, K. W., and Flynn, P. J. (2011). Detecting questionable observers using face track clustering. In *IEEE Workshop on Applications of Computer Vision*.

Bauml, M., Bernardin, K., Fischer, M., Ekenel, H., and Stiefelhagen, R. (2010). Multi-pose face recognition for person retrieval in camera networks. In *IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 441–447.

Bäuml, M., Tapaswi, M., and Stiefelhagen, R. (2013). Semi-supervised Learning with Constraints for Person Identification in Multimedia Data. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Bradski, G. (2000). The OpenCV Library. *Dr. Dobb's Journal of Software Tools*.

Chang, C.-C. and Lin, C.-J. (2011). LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2.

Cinbis, R. G., Verbeek, J., and Schmid, C. (2011). Unsupervised Metric Learning for Face Identification in TV Video. In *International Conference on Computer Vision*, Barcelona, Spain.

Comaniciu, D., Ramesh, V., and Meer, P. (2003). Kernel-based object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(5):564–577.

Cricri, F., Curcio, I. D. D., Mate, S., Dabov, K., and Gabbouj, M. (2012). Sensor-based analysis of user generated video for multi-camera video remixing. In *IEEE 18th International Conference on Multimedia Modeling*, pages 255–265.

Gou, G., Huang, D., and Wang, Y. (2012). A novel video face clustering algorithm based on divide and conquer strategy. In *Proceedings of the 12th Pacific Rim international conference on Trends in Artificial Intelligence*, pages 53–63.

Hao, P. and Kamata, S. (2012). Unsupervised people organization and its application on individual retrieval from videos. In *21st International Conference on Pattern Recognition*, pages 2001–2004.

Klein, D., Kamvar, S. D., and Manning, C. D. (2002). From instance-level constraints to space-level constraints: Making the most of prior knowledge in data clustering. In *International Conference on Machine Learning*, pages 307–314, San Francisco, CA, USA.

Kumar, N., Belhumeur, P. N., and Nayar, S. K. (2008). FaceTracer: A Search Engine for Large Collections of Images with Faces. In *European Conference on Computer Vision*, pages 340–353.

Kumar, N., Berg, A., Belhumeur, P., and Nayar, S. (2011). Describable visual attributes for face verification and image search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(10):1962–1977.

Lo Presti, L. and La Cascia, M. (2012). An on-line learning method for face association in personal photo collection. *Image and Vision Computing*, 30(4-5):306–316.

Sivic, J., Zitnick, C. L., and Szeliski, R. (2006). Finding people in repeated shots of the same scene. In *British Machine Vision Conference*.

Suh, B. and Bederson, B. B. (2004). Semi-automatic image annotation using event and torso identification. Technical report, Computer Science Department, University of Maryland, College Park, MD.

Tao, J. and Tan, Y.-P. (2008). Efficient clustering of face sequences with application to character-based movie browsing. In *IEEE International Conference on Image Processing*, pages 1708–1711.

Uřičář, M., Franc, V., and Hlaváč, V. (2012). Detector of facial landmarks learned by the structured output SVM. In *Proceedings of the 7th International Conference on Computer Vision Theory and Applications*.

Zhang, L., Chen, L., Li, M., and Zhang, H. (2003). Automated annotation of human faces in family albums. In *Proceedings of the eleventh ACM international conference on Multimedia*, pages 355–358.