Deep Learning Architectures for Face Recognition in Video Surveillance

Saman Bashbaghi, Eric Granger, Robert Sabourin and Mostafa Parchami

Abstract Face recognition (FR) systems for video surveillance (VS) applications attempt to accurately detect the presence of target individuals over a distributed network of cameras. In video-based FR systems, facial models of target individuals are designed a priori during enrollment using a limited number of reference still images or video data. These facial models are not typically representative of faces being observed during operations due to large variations in illumination, pose, scale, occlusion, blur, and to camera inter-operability. Specifically, in still-to-video FR application, a single high-quality reference still image captured with still camera under controlled conditions is employed to generate a facial model to be matched later against lower-quality faces captured with video cameras under uncontrolled conditions. Current video-based FR systems can perform well on controlled scenarios, while their performance is not satisfactory in uncontrolled scenarios mainly because of the differences between the source (enrollment) and the target (operational) domains. Most of the efforts in this area have been toward the design of robust video-based FR systems in unconstrained surveillance environments. This chapter presents an overview of recent advances in still-to-video FR scenario through deep convolutional neural networks (CNNs). In particular, deep learning architectures proposed in the literature based on triplet-loss function (e.g., cross-correlation matching CNN, trunk-branch ensemble CNN and HaarNet) and supervised autoencoders (e.g., canonical face representation CNN) are reviewed and compared in terms of accuracy and computational complexity.

Saman Bashbaghi, Eric Granger, Robert Sabourin
Laboratoire. d'imagerie de vision et d'intelligence artificielle,
École de technologie supérieure, Université du Québec, Montreal, Canada
e-mail: bashbaghi@livia.etsmtl.ca, {eric.granger, robert.sabourin}@etsmtl.ca

Mostafa Parchami
Computer Science and Engineering Department, University of Texas at Arlington, TX, USA
e-mail: mostafa.parchami@mavs.uta.edu

1 Introduction

Face recognition (FR) systems in video surveillance (VS) has received a significant attention during the past few years. Due to the fact that the number of surveillance cameras installed in public places is increasing, it is important to build robust video-based FR systems [38]. In VS, capture conditions typically range from semi-controlled with one person in the scene (e.g. passport inspection lanes and portals at airports), to uncontrolled free-flow in cluttered scenes (e.g. airport baggage claim areas, and subway stations). Two common types of applications in VS are: (1) still-to-video FR (e.g., watch-list screening), and (2) video-to-video FR (e.g., face re-identification or search and retrieval) [23, 11, 4]. In the former application, reference face images or stills of target individuals of interest are used to design facial models, while in the latter, facial models are designed using faces captured in reference videos. This chapter is mainly focused on still-to-video FR with a single sample per person (SSPP) under semi- and unconstrained VS environments.

The number of target references is one or very few in still-to-video FR applications, and the characteristics of the still camera(s) used for design significantly differ from the video cameras used during operations [3]. Thus, there are significant differences between the appearances of still ROI(s) and ROIs captured with surveillance cameras, according to various changes in ambient lighting, pose, blur, and occlusion [21, 1]. During enrollment of target individuals, facial regions of interests (ROIs) isolated in reference still images are used to design facial models, while during operations, the ROIs of faces captured in videos are matched against these facial models. In VS, a person in a scene may be tracked along several frames, and matching scores may be accumulated over a facial trajectory (a group of ROIs that correspond to the same high-quality track of an individual) for robust spatio-temporal FR [7].

In general, methods proposed in the literature for still-to-video FR can be broadly categorized into two main streams: (1) conventional, and (2) deep learning methods. The conventional methods rely on hand-crafted feature extraction techniques and a pre-trained classifier along with fusion, while deep learning methods automatically learn features and classifiers cojointly using massive amounts of data. In spite of improvements achieved using the conventional methods, yet they are less robust to real-world still-to-video FR scenario. On the other hand, there exists no feature extraction technique that can overcome all the challenges encountered in VS individually [4, 15, 34].

Conventional methods proposed for still-to-video FR are typically modeled as individual-specific face detectors using one- or 2-class classifiers in order to enable the system to add or remove other individuals and easily adapt over time [23, 2]. Modular systems designed using individual-specific ensembles have been successfully applied in VS [23, 11]. Thus, ensemble-based methods have been shown as a reliable solution to deal with imbalanced data, where multiple face representations can be encoded into ensembles of classifiers to improve the robustness of still-to-video FR [4]. Although it is challenging to design robust facial models using a single training sample, several approaches have addressed this problem, such as multiple face representations, synthetic generation of virtual faces, and using auxiliary data from other people to enlarge the training set [2, 18, 16, 36]. These techniques seek to enhance the robustness of face models to intra-class variations. In multiple representations, different patches and face descriptors are employed [2, 4], while 2D morphing or 3D reconstructions are used to synthesize artificial face images [16, 22]. A generic auxiliary dataset containing faces of other persons can be exploited to perform domain adaptation [20], and sparse representation classification through dictionary learning [36]. However, techniques based on synthetic face generation and auxiliary data are more complex and computationally costly for real-time applications, because of the prior knowledge required to locate the facial components reliably, and the large differences between the quality of still and video ROIs, respectively.

Recently, several deep learning based solutions have been proposed to learn effective face representations directly from training data through convolutional neural networks (CNNs) and nonlinear feature mappings [30, 31, 6, 13, 28]. In such methods, different loss functions can be considered in the training process to enhance the inter-personal variations, and simultaneously reduce the intra-personal variations. They can learn non-linear and discriminative feature representations to cover the existing gaps compared to the human visual system [34], while they are computationally costly and typically require a large number of labeled data to train. To address the SSPP problem in FR, a triplet-based loss function have been introduced in [27, 28, 8, 25, 24] to discriminate between a pair of matching ROIs and a pair of non-matching ROIs. Ensemble of CNNs, such as trunk-branch ensemble CNN (TBE-CNN) [8] and HaarNet [25] have been shown to extracts features from the global appearance of faces (holistic representation), as well as, to embed asymmetrical features (local facial feature-based representations) to handle partial occlusion. Moreover, supervised autoencoders have been proposed to enforce faces with variations to be mapped to the canonical face (a well-illuminated frontal face with neutral expression) of the person in the SSPP scenario to generate robust feature representations [9, 26].

2 Bachground of Video-Based FR Through Deep Learning

Deep CNNs have recently demonstrated a great achievement in many computer vision tasks, such as object detection, object recognition, etc. Such deep CNN models have shown to appropriately characterize different variations within a large amount of data and to learn a discriminative non-linear feature representation. Furthermore, they can be easily generalized to other vision tasks by adopting and fine-tuning pretrained models through transfer learning [28, 6]. Thus, They provide a successful tool for different applications of FR by learning effective feature representations directly from the face images [6, 13, 28]. For example, DeepID, DeepID2, and DeepID2+ have been proposed in [29, 31, 32], respectively, to learn a set of discriminative high-level feature representations.

For instance, an ensemble of CNN models was trained in [31] using the holistic face image along with several overlapping/non-overlapping face patches to handle the pose and partial occlusion variations. Fusion of these models is typically carried out by feature concatenation to construct over-complete and compact representations. Followed by [31], feature dimension of the last hidden layer was increased in [29, 32], as well as, exploiting supervision to the convolutional layers in order to learn hierarchical and non-linear feature representations. These representations aim to enhance the inter-personal variations due to extraction of features from different identities separately, and simultaneously reduce the intra-personal variations. In contrast to DeepID series, an accurate face alignment was incorporated in Microsoft DeepFace [34] to derive a robust face representation through a nine-layer deep CNN. In [30], the high-level face similarity features were extracted jointly from a pair of faces instead of a single face through multiple deep CNNs for face verification applications. Since these approaches are not considered variations like blurriness and scale changes (distance of the person from surveillance cameras), they are not fully adapted for video-based FR applications.

Similarly, for the SSPP problems, a triplet-based loss function has been lately exploited in [28, 27, 8, 25, 24] to learn robust face embeddings, where this type of loss seeks to discriminate between the positive pair of matching facial ROIs from the negative non-matching facial ROI. A robust facial representation learned through triplet-loss optimization has been proposed in [24] using a compact and fast cross-correlation matching CNN (CCM-CNN). However, CNN models like the trunk-branch ensemble CNN (TBE-CNN) [8] and HaarNet [25] can further improve robustness to variations in facial appearance by the cost of increasing computational complexity. In such models, the trunk network extracts features from the global appearance of faces (holistic representation), while the branch networks embed asymmetrical and complex facial traits. For instance, HaarNet employs three branch networks based on Haar-like features, while facial landmarks are considered in TBE-CNN. However, these specialized CNNs represent complex solutions that are not perfectly suitable for real-time FR applications [5].

Moreover, autoencoder neural networks can be typically employed to extract deterministic non-linear feature mappings robust to face images contaminated by different noises, such as illumination, expression and poses [9, 26]. An autoencoder network contains encoder and decoder modules, where the former module embed the input data to the hidden nodes, while the latter returns the hidden nodes to the original input data space with minimizing the reconstruction error(s) [9]. Several autoencoder networks inspired from [35] have been proposed to remove the aforementioned variances in face images [9, 17, 19]. These networks deal with faces containing different types of variations (e.g., illumination, pose, etc.) as noisy images. For instance, a facial component-based CNN has been learned in [40] to transform faces with changes in pose and illumination to frontal view faces, where pose-invariant features of the last hidden layer are employed as face representations. Similarly, several deep architecture have been proposed using multi-task learning in order to rotate faces with arbitrary poses and illuminations to target-pose faces [37, 39]. In addition, a general deep architecture was introduced in [10] to encode a desired attribute and combine it with the input image to generate target images as similar as the input image with a visual attribute (a different illumination, facial appearance or new pose) without changing other aspects of a face.

3 Deep Learning Architectures for FR in VS

In this section, the most recent deep learning architectures proposed for video-based FR considering the SSPP problem are addressed. These architectures can be categorized into two groups: (1) Deep CNN models trained using triplet-loss function, and (2) deep autoencoders.

3.1 Deep CNNs Using Triplet-Loss

Recently, deep learning algorithms specialized for FR mostly utilize triplet-loss in order to train the deep architecture and thereby learning a discriminant face representation [28, 8]. However, careful triplet sampling is a crucial step to achieve a faster convergence [28]. In addition, employing triplet-loss is challenging since the global distributions of the training samples are neglected in optimization process.

Triplet-loss approach was first proposed in [28] to train CNNs for robust face verification. To that end, the representation of triplets (three faces containing an anchor and a positive image of the same subject and a negative image of other subjects) are L_2-normalized as the input of triplet-loss function. It therefore ensures that the input representations of face images lie on a unit hypersphere prior to apply triplet-loss function [8]. Deep CNN models proposed for video-based FR that employed triplet-loss for training are reviewed in the following subsections.

3.1.1 Cross-Correlation Matching CNN

An efficient CNN architecture has been proposed in [24] for accurate still-to-video FR from a single reference facial ROI per target individual. Based on a pairwise cross-correlation matching (CCM) and a robust facial representation learned through triplet-loss optimization, CCM-CNN architecture is a fast and compact network (requires few network branches, layers and parameters). It exploits a matrix Hadamard product followed by a fully connected layer that simulates the adaptive weighted cross-correlation technique [12]. A triplet-based optimization approach has been exploited to learn discriminant facial representations based on triplets containing the positive, negative video ROIs and the corresponding still ROI. In particular, the similarity between the representations of positive video ROIs and the reference still ROI is enhanced, while the similarity between negative video ROIs and the both reference still and positive video ROIs is increased. To further improve robustness of facial models, the CCM-CNN fine-tuning process incorporates diverse knowledge by generating synthetic faces based on still and video ROIs of non-target individuals.

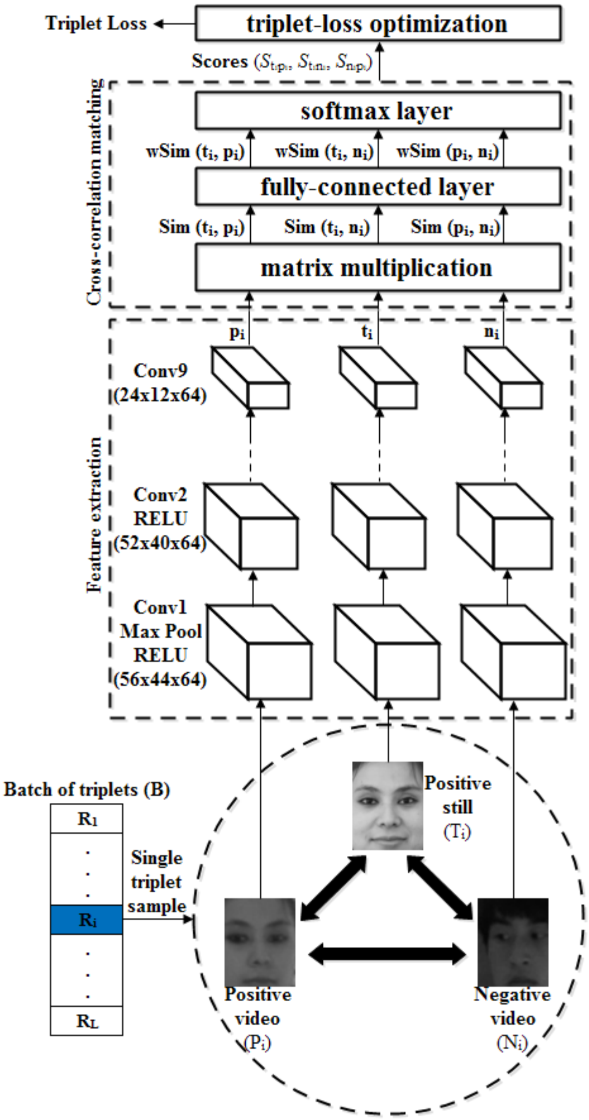

Fig. 1: Training pipeline of the CCM-CNN [24].

As shown in Figure 1, the CCM-CNN learns a robust facial representation by iterating over a batch of training triplets $B = \{R_1, \ldots, R_L\} = \{(T_1, P_1, N_1), \ldots, (T_L, P_L, N_L)\}$, where L is the batch size, and each triplet R_i contains a still ROI T_i along with a corresponding positive ROI P_i and a negative ROI N_i from operational videos. This architecture was inspired by Siamese networks containing identical subnetworks with the same configurations, parameters and weights. Therefore, fewer parameters are required for training that can avoid overfitting. The CCM-CNN consists of three main components – feature extraction, cross-correlation matching and triplet-loss optimization. The feature extraction pipeline extracts discriminative feature maps from ROIs that are similar for two images of the same person under different capture conditions (e.g., illumination and pose). The cross-correlation matching component inputs feature maps extracted from the ROIs and calculates the likelihood of the faces belonging to the same person. Finally, triplet-loss optimization computes a loss function to maximize similarity of the still ROIs and their respective positive samples in the batch, while minimizing similarity between still ROIs and their negative ROIs, as well as, positive and negative ROIs.

Despite differences in the domains between reference target still ROIs and target/non-target video ROIs, the CCM-CNN can effectively extract discriminant features. As shown in Figure 1, feature extraction is carried out by 3 identical subnetworks for still, positive and negative faces. These subnetworks process three input faces and the weights are shared across them. Each subnetwork consists of 9 convolutional layers each followed by a spatial batch normalization, drop-out, and RELU layers. Contrary to former convolutional layers, the last convolutional layer is not followed by a RELU in order to maintain the representativeness of the final feature map and to avoid losing informative data for the matching stage. Moreover, a single max-pooling layer is added after the first convolution layer to increase the robustness to small translation of faces in the ROI.

In the CCM-CNN, all three feature extraction pipelines share the same set of parameters. This ensures that the features extracted from target still ($\mathbf{t_i}$), positive ($\mathbf{p_i}$) and negative ($\mathbf{n_i}$) are consistent and comparable. Each convolutional layer has 64 filters of size 5x5 without padding. Thus, given the input size of 120x96, the output of each branch is of size N_f = 24x12x64 features.

After extracting features from the still and video ROIs, a pixel-based matching method is employed to effectively compare these feature maps and measure the matching similarity. The process of comparison in the CCM-CNN has three stages: matrix Hadamard product, fully connected neural network, and finally a softmax. Instead of concatenating feature vectors of different branches as input to the fully connected layer, the feature maps representing the ROIs are multiplied with each other to encode pixel-wise correlation between each pair of ROI in the triplet. This approach eliminates the complexity of matching by replacing the concatenation with a simple element-wise matrix multiplication and directly encodes similarity as opposed to let the network learn how to match input concatenated feature vectors.

The matrix Hadamard product is exploited to simulate cross-correlation, where Hadamard product of the two matrices provides a single feature map that represents the similarity of the two ROIs. For example, the similarity $Sim(\mathbf{t}_i, \mathbf{p}_i)$ and cross-correlation $\mathbf{w}Sim(\mathbf{t}_i, \mathbf{p}_i)$ of still $\mathbf{t}_i$ and positive $\mathbf{p}_i$ feature maps is computed as follows, respectively, using matrix Hadamard product:

$$Sim(\mathbf{t}_i, \mathbf{p}_i) = (\mathbf{t}_i \odot \mathbf{p}_j) \tag{1}$$

$$wSim(\mathbf{t}_i, \mathbf{p}_i) = \omega_m \cdot RELU(\omega_n \cdot Sim(\mathbf{t}_i, \mathbf{p}_i) + \mathbf{b}_n) + \mathbf{b}_m \tag{2}$$

where ω_m, ω_n, $\mathbf{b}_m$ and $\mathbf{b}_n$ are the weights and biases of the two fully-connected layers applied to the vectorized output of the matrix multiplication. Furthermore, a softmax layer is applied to obtain a probability-like similarity score for each of the two classes (match and non-match).

A multi-stage approach is considered to efficiently train the CCM-CNN based on reference stills ROI per individual and operational videos. To that end, pre-training is performed using a large generic FR dataset, and a domain specific dataset for still-to-video FR is used for fine-tuning. To that end, a set of matching and non-matching images is selected from the Labeled Faces in the Wild (LFW) [14]. Images from this set are augmented to roughly 1.3M training triplets. In order to consistently update the set of training triplets, the on-line triplet sampling method [28] is used for 50 epochs.

In contrast with FaceNet [28], a pair-wise triplet-loss optimization function was proposed to effectively train the network. In order to adapt the network for pairwise triplet-based optimization, it is modified by incorporating additional feature extraction branches. Each batch contains several triplets, and for each triplet, the network seeks to learn the correct classification. During the training, each branch of the feature extraction pipeline is assigned to a component of the triplet – the main branch is responsible for processing the reference still ROI, while the positive (negative) branch extracts features from the positive (negative) video ROI of the triplet. Moreover, the cross correlation matching pipeline is modified to benefit from the triplets by introducing an Euclidean loss layer followed by softmax which computes the similarity for each pair of ROIs in the triplet. The loss layer is exploited to compute the overall loss of the network as follows:

$$\text{Triplet Loss} = \frac{1}{L} \sum_{R_i \in B} \sqrt{(1 - S_{t_i p_i})^2 + S^2_{t_i n_i} + S^2_{n_i p_i}} \tag{3}$$

where S_{tp}, S_{tn}, and S_{np} are the similarity scores from cross-correlation matching between (1) the reference (positive) still ROI and positive video ROI, (2) still ROI and negative video ROI, and (3) negative and positive video ROIs of the triplet, respectively, computed using the aforementioned approach. During operations (once the network training is completed) the additional feature extraction branch (negative branch, N) is removed from the network, and only the still and the positive branches (P) are taken into account. Thus, the main branch (T) extracts features from a reference still ROIs, while the positive branch extracts features from the probe video ROI to determine whether they belong to the same person.

During fine-tuning, CCM-CNN acquires knowledge on the similarities and dissimilarities between the target individuals of interest enrolled to the system. In order to improve the robustness of facial models intra-class variation, the network is fine-tuned with synthetic facial ROIs generated from the high-quality still ROIs that account for the operation domain. For each still image, a set of augmented images are generated using different transformations, such as shearing, mirroring, rotating and translating the original still image. In contrast with the pre-training, the focus of the fine-tuning stage is to learn dissimilarities between the subjects of interest.

3.1.2 Trunk-Branch Ensemble CNN

An improved triplet-loss function has been introduced in [8] to promote the robustness of face representations. To that end, a trunk-branch ensemble CNN (TBE-CNN) model has been proposed to extract complementary features from holistic face images, as well as, face patches around facial landmarks through trunk and branch networks, respectively. To emulate real-world video data, artificially blur training data are synthesized from still images by applying artificial out-of-focus and motion blur to learn blur-insensitive face representations. The architecture of TBE-CNN is shown in Figure 2.

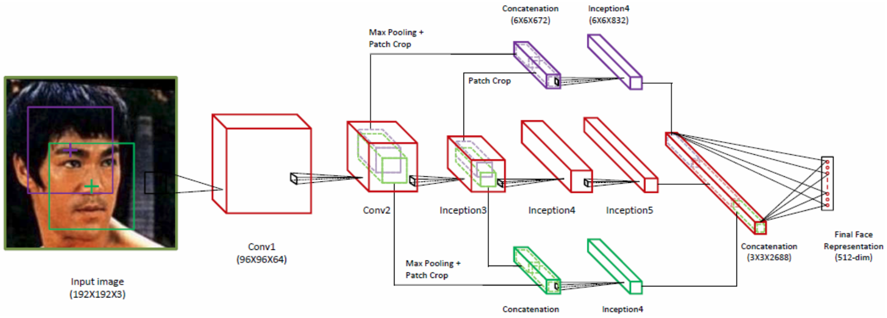

Fig. 2: The architecture of TBE-CNN [8].

As shown in Figure 2, TBE-CNN contains one trunk network along with several branch networks, where the trunk and branch networks share some layers in order to embed global and local information. This sharing strategy may lead to reduce the computational cost and also efficient convergence. The output feature maps of these networks are concatenated to feed into the fully-connected layer to generate final face representations.

During training as illustrated in Figure 3, TBE-CNN is given still images and simulated video frames, where the network aims to classify each still image and its corresponding artificially blurred face image correctly into the same class. The training process is performed using a stage-wise strategy, where the trunk network and each of the branch networks are trained separately with fixed parameters.

To improve the discriminative power of face representations, mean distance regularized triplet-loss (MDR-TL) function is considered to fine-tune the entire network.

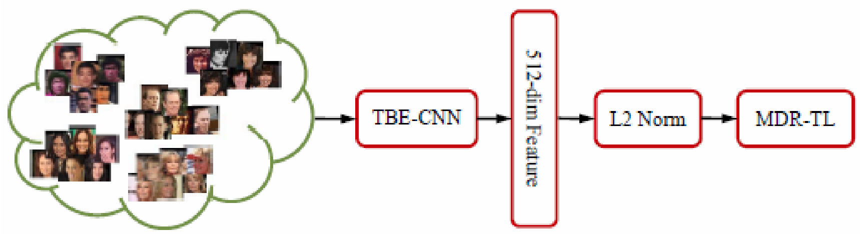

Fig. 3: Training pipeline of the CCM-CNN [8].

Compared to the original triplet-loss function proposed in [28], MDR-TL regularizes the triplet-loss to provide uniform distributions for both inter- and intra-class distances. Figure 4 represents the principle of MDR-TL.

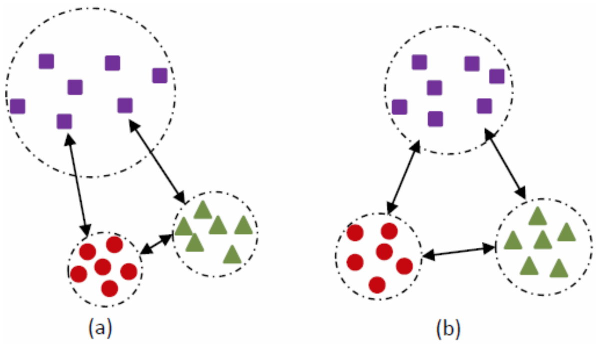

Fig. 4: The mean distance regularized triplet-loss. (a) Training triplet with non-uniform inter- and intra-class distance distributions, and (b) triplets with uniform inter- and intra-class distance distributions using MDR-TL regularization [8].

As demonstrated in Figure 4(a), it is difficult to appropriately discriminate between matching and non-matching pairs of face images because the training samples have non-uniform inter- and intra-class distance distributions. To tackle this problem, the triplet loss is regularized using MDR-TL loss function by constraining the distances between mean representations of different subjects (Figure 4(b)).

3.1.3 HaarNet

An ensemble of deep CNNs called HaarNet has been proposed in [25] to efficiently learn robust and discriminative face representations for video-based FR applications. Similar to TBE-CNN [8], HaarNet consists of a trunk network with three diverging branch networks that are specifically designed to embed facial features, pose, and other distinctive features. The trunk network effectively learns a holistic representation of the face, whereas the branches learn more local and asymmetrical features related to pose or special facial features by means of Haar-like features. Furthermore, to increase the discriminative capabilities of the HaarNet, a second-order statistic regularized triplet-loss function has been introduced to take advantage of the inter-class and intra-class variations existing in training data to learn more distinctive representations for subjects with similar faces. Finally, a fine-tuning stage has been performed to embed the correlation of facial ROIs stored during enrollment and improve recognition accuracy.

The overall architecture of the HaarNet is presented in Figure 5. It is composed of a global trunk network along with three branch networks that can effectively learn a representation that is robust to changing capture conditions.

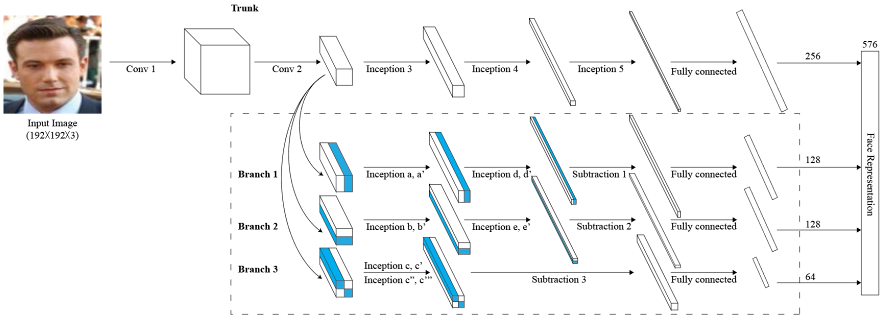

Fig. 5: HaarNet architecture for the trunk and three branches [25]. (Max pooling layers after each inception and convolution layer are not shown for clarity).

As shown in Fig. 5, the trunk is employed to learn the global appearance face representation, whereas three branches diverged from the trunk are designed to learn asymmetrical and more locally distinctive representations. For the trunk network, the configuration of GoogLeNet [33] is employed with 18 layers.

In contrast with [8], instead of training each branch network on different face landmarks, Haarnet utilizes three branch networks in order to compute one of the Haar-like features, respectively as illustrated in Fig. 6. Haar features have been exploited to extract distinctive features from faces based on the symmetrical nature of facial components, and on contrast of intensity between adjacent components. In general, these features are calculated by subtracting sum of all pixels in the black areas from the sum of all pixels in the white areas. To avoid information loss, the Haar-like features are calculated by matrix summation, where black matrices are negated. Thus, instead of generating only one value, each Haar-like feature returns a matrix.

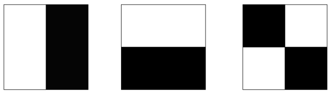

Fig. 6: Haar-like features used in branch networks [25].

In the Haarnet architecture (see Fig. 5), the trunk network and its three branches share the first two convolutional layers. Then, the first and second branches split the output of Conv2 into two sub-branches, and also apply two inception layers to each sub-branch. Subsequently, the two sub-branches are merged by a subtraction layer to obtain a Haar-like representation for each corresponding branch. Meanwhile, the third branch divides the output of Conv2 into four sub-branches and one inception layer is applied to each of the sub-branches. Eventually, a subtraction layer is exploited to combine those for sub-branches and feed to the fully connected layer. The final representation of the face is obtained by concatenating the output of the trunk and all three Haar-like features.

Fig. 7 illustrates the training process of the HaarNet using a triplet-loss concept, where a batch of triplets composed of <anchor, positive, negative> is input to the architecture is translated to a face representation.

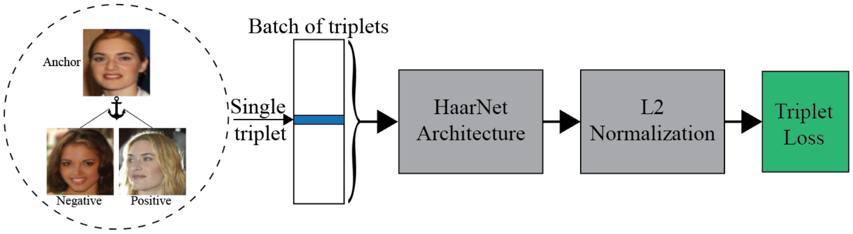

Fig. 7: Processing of triplets to compute the loss function. The network inputs a batch of triplets to the HaarNet architecture followed by an $L2$ Normalization [25].

As shown in Fig. 7, output of the HaarNet is then $L2$ normalized prior to feed into the triplet-loss function in order to represent faces on a unit hyper-sphere. Let's denote the $L2$ normalized representation of a facial ROI x as $f(x) \in R^d$ where d is the dimension of the face representation.

A multi-stage training approach is hereby considered to effectively optimize the parameters of the HaarNet. The first three stages are designed for initializing the parameters with a promising approximation prior to employ the triplet-loss function. Moreover, these three stages are beneficial to detect a set of hard triplets from the dataset in order to initiate the triplet-loss training. In the first stage, the trunk network is trained using a softmax loss, because the softmax function converges much faster than triplet-loss function. During the second stage, each branch is trained separately by fixing the shared parameters and by only optimizing the rest of the parameters.

Similar to the first stage, a softmax loss function is used to train each of the branches. Then, the complete network is constructed by assembling the trunk and the three branch networks. The third stage of the training is indeed a fine-tuning stage for the complete network in order to optimize these four components simultaneously. In order to consider the inter- and intra-class variations, the network is trained for several epochs using the hard triplets detected during the previous stages.

As suggested in [8], adding mean distance regularization term to the triplet-loss function can promote distinctiveness of the face representations. Inspired from [8], the main idea of the second-order statistics regularization term is illustrated in Figure 8 illustrates.

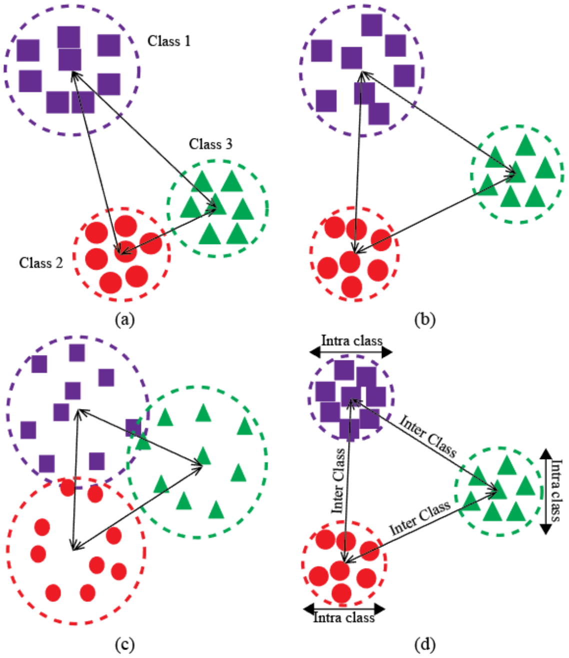

Fig. 8: Illustration of the regularized triple loss principles based on the mean and standard deviation of three classes, assuming 2D representations of the ROIs [25].

In Fig 8 (a), triplet-loss function may suffer from nonuniform inter-class distances that leads to failure of using simple distance measures, such as Euclidean and cosine distances. In this regard (see Fig. 8 (b)), a mean distance regularization term can be added to increase the separation of class representations. On the other hand, representations of some facial ROIs may be confused with representation of the adjacent facial ROIs in the feature space due to high intra-class variations. Fig. 8 (c) shows such a configuration, where the mean representation of the classes are distant from each other but the standard deviations of classes are very high, leading to overlap among class representations. To address this issue, a new term in the loss function is introduced to examine the intra-class distribution of the training samples.

The triplet constraint can be expressed as a function of the representation of anchor, positive and negative samples as follows [28]:

$$\left\|f\left(x_i^a\right)-f\left(x_i^p\right)\right\|_2^2+a<\left\|f\left(x_i^a\right)-f\left(x_i^n\right)\right\|_2^2 \tag{4}$$

where $f\left(x_i^a\right)$, $f\left(x_i^p\right)$, and $f\left(x_i^n\right)$ are the face representations of the anchor, positive, and negative, respectively. All the triplets sampled from the training set should satisfy the constraint. Thus, during training, HaarNet minimizes of the loss function:

$$L_{HaarNet} = \delta_1 L_{triplet} + \delta_2 L_{mean} + \delta_3 L_{std} \tag{5}$$

where δ_i denotes the weight for each term in the loss function. Furthermore, $L_{triplet}$ can be defined based on (4) as follows:

$$L_{triplet}=\frac{1}{2N}\sum_{i=1}^{N}\left[\left\|f\left(x_i^a\right)-f\left(x_i^p\right)\right\|_2^2-\left\|f\left(x_i^a\right)-f\left(x_i^n\right)\right\|_2^2+\alpha\right]_+ \tag{6}$$

Similar to [8], assuming that the mean distance constraint is $\beta < \|\hat{\mu}_c-\hat{\mu}_c^n\|_2^2$, L_{mean} is defined as:

$$L_{mean} = \frac{1}{2P}\sum_{c=1}^{C}\max\left(0,\ \beta - \|\hat{\mu}_c-\hat{\mu}_c^n\|_2^2\right) \tag{7}$$

In addition, the standard deviation constraint is defined to be $\sigma_c > \gamma$, where σ_c is the standard deviation of the class c. Therefore, L_{std} can be computed as follows:

$$L_{std} = \frac{1}{M}\sum_{c=1}^{C}\max\left(0,\ \gamma - \sigma_c\right) \tag{8}$$

where N, P, and M are the number of samples that violate the triplet, mean distance, and standard deviation constraints, respectively. Likewise, C is the number of subjects in the current batch and α, β,and γ are margins for triplet, mean distance, and standard deviation constraints, respectively. The loss function (5) can be optimized using the regular stochastic gradient descent with momentum similar to [8]. The gradient of loss w.r.t. the facial ROI representation of ith image for subject c (denoted as $f\left(x_{ci}\right)$) is derived as follows:

$$\frac{\partial L_{std}}{\partial f\left(x_{ci}\right)} = -\frac{1}{M}\sum_{c=1}^{C}\omega_c\,\frac{\partial \sigma_c}{\partial f\left(x_{ci}\right)} \tag{9}$$

where ω_c equals to 1 if the standard deviation constraint is violated, and equals to 0 otherwise. Moreover, the derivative of L_{std} can be computed by applying the chain rule as follows:

$$\frac{\partial \sigma_c}{\partial f\left(x_{ci}\right)} = \frac{\partial \sqrt{\frac{1}{N_c} \sum_{j=1}^{N_c} \left\| f\left(x_{cj}\right) - \mu_c \right\|_2^2}}{\partial f\left(x_{ci}\right)} = \frac{\left[\sum_{j=1}^{N_c} \frac{1}{N_c} \left\| \mu_c - f\left(x_{cj}\right) \right\|_2\right] - \left\| \mu_c - f\left(x_{ci}\right) \right\|_2}{2 \sqrt{\frac{1}{N_c} \sum_{j=1}^{N_c} \left\| f\left(x_{cj}\right) - \mu_c \right\|_2^2}} \tag{10}$$

As shown in Fig. 8 (d), the discriminating power of the face representations can be improved by setting margins such that $\gamma < \beta$. This ensures a high inter-class and a low intra-class variations to increase the overall classification accuracy.

3.2 Deep CNNs Using Autoencoder

An efficient Canonical Face Representation CNN (CFR-CNN) has been proposed in [26] for accurate still-to-video FR from a SSPP, where still and video ROIs are captured under various conditions. The CFR-CNN is based on a supervised autoencoder that can represent the divergence between the source (still ROI) and target (video ROI) domains encountered in still-to-video FR scenario. The autoencoder network is trained using a weighted pixel-wise loss function that is specialized for SSPP problems, and allows to reconstruct canonical ROIs (frontal and less blurred faces) for matching that correspond to the conditions of reference still ROIs. In addition, it can generate discriminative face embeddings that are similar for the same individuals, and robust to variations typically observed in unconstrained real-world video scenes. A fully-connected classification network is also trained to perform face matching using the face embeddings extracted from the deep autoencoder, and accurately determine whether the pairs of still and video ROIs correspond to the same individual.

Autoencoder CNNs are typically utilized to normalize variations in face capture conditions from probe video ROIs to those in still reference ROIs. The architecture of the autoencoder is shown in Figure 9, where the input image is a probe video ROI captured using a surveillance camera, while the output is a reconstructed image. This network consists of (1) three convolutional layers each followed by a max-pooling layer to extract robust convolutional maps, and then (2) a two-layer fully-connected network that generates a 256-dimensional face embedding. The decoder reverses these operations by applying a fully-connected layer to generate the original vector and three deconvolutional layers, each one followed by un-pooling layers designed for generating the final reconstruction of the face.

A development set (assumed to be collected from unknown individuals captured from the operational domain) is employed for training of the deep autoencoder network. A batch of video ROIs are fed into the network, where still ROIs of the corresponding persons are used for facial reconstructions. Using higher-quality still images that are captured during enrollment under controlled conditions as target faces, the autoencoder network simultaneously learns invariant face embeddings to

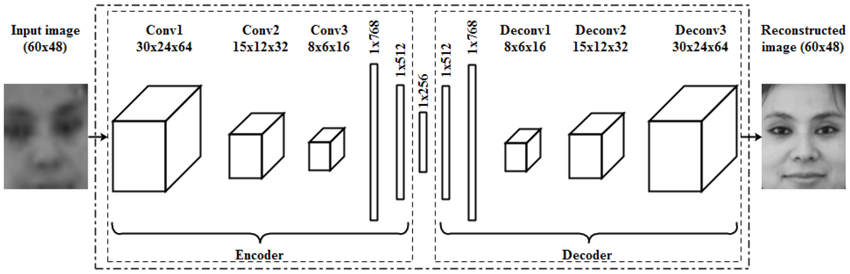

Fig. 9: Block diagram of the autoencoder network in the CFR-CNN [26].

normalize the input video ROIs. The parameters of this autoencoder network are optimized by employing a weighted Mean Squared Error (MSE) criterion, where a T-shaped region (illustrated in Figure 10) is considered to assign a higher significance to discriminative facial components like eyes, nose and mouth. This loss function of is formulate as:

$$
\begin{gathered}
L_{CFR-CNN} = \sum_{i \in rows} \sum_{j \in cols} \tau_{i,j} \left\| X^2 - \hat{X}^2 \right\| \\
\tau_{i,j} = \begin{cases} \alpha & \text{if (i,j) belongs to T} \\ \beta & \text{if (i,j) otherwise} \end{cases}
\end{gathered}
\tag{11}
$$

where $rows \times cols$ is the size of ROIs, X is the target still ROI and $\hat{X}$ is the reconstructed ROI. The weight α is considered for the T region, while the weight β is considered for pixels outside the T region.

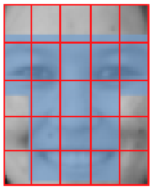

Fig. 10: T-shaped weight mask used for the loss function of CFR-CNN [26].

A fully-connected network is then integrated with the deep convolutional autoencoder, and the output of the intermediate layer is then considered as a face representation that is invariant to the different nuisance factors commonly encountered in unconstrained surveillance environments. Finally, face matching is performed using a fully-connected classification network as shown in Figure 11. This network is implemented to match the face representations of still and video ROIs.

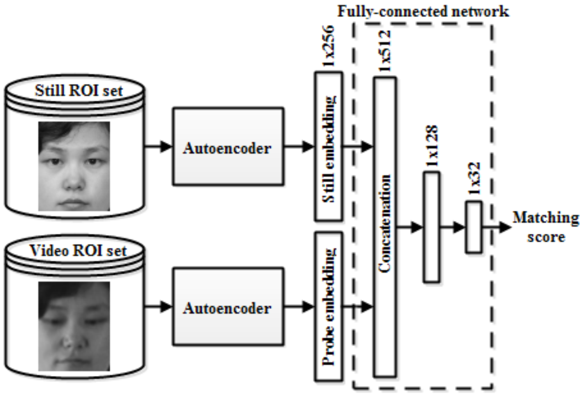

Fig. 11: Block diagram of the classification network in the CFR-CNN [26].

The fully-connected classification network is trained using a regular pairwise-matching scheme, where the face embeddings of the reference still and probe video ROIs are fed into the classification network. The network can thereby learn to classify each pair of still and video ROIs as either matching or non-matching.

4 Performance Evaluation

The performance of the aforementioned video-based FR systems is evaluated using Cox Face DB [15]. This dataset is specifically collected for video surveillance applications, where it is composed of high-quality still faces captured with still cameras under controlled conditions and low-quality video faces captured with different off-the-shelf camcorders under uncontrolled conditions. Videos are recorded per subject when they are walking through a designed-S curve containing changes in pose, illumination, scale and blur. An example of still and videos of one subject is shown in Figure 12.

The systems are evaluated according to experimental protocol suggested in [15], where each probe video ROI is compared against the reference still ROIs, and rank-1 recognition is reported as the FR accuracy. Meanwhile, since video-based FR systems are often required to perform real-time processing in surveillance applications, the computational complexity of such systems should be also taken into consideration. In this regards, the complexity can be determined in terms of the number of operations (to match a video probe ROI to a reference still ROI), the number of network parameters and layers [5].

In order to confirm the viability of the CNN-based video FR systems for real-time surveillance applications, Table 1 presents the accuracy and compares their computational complexity.

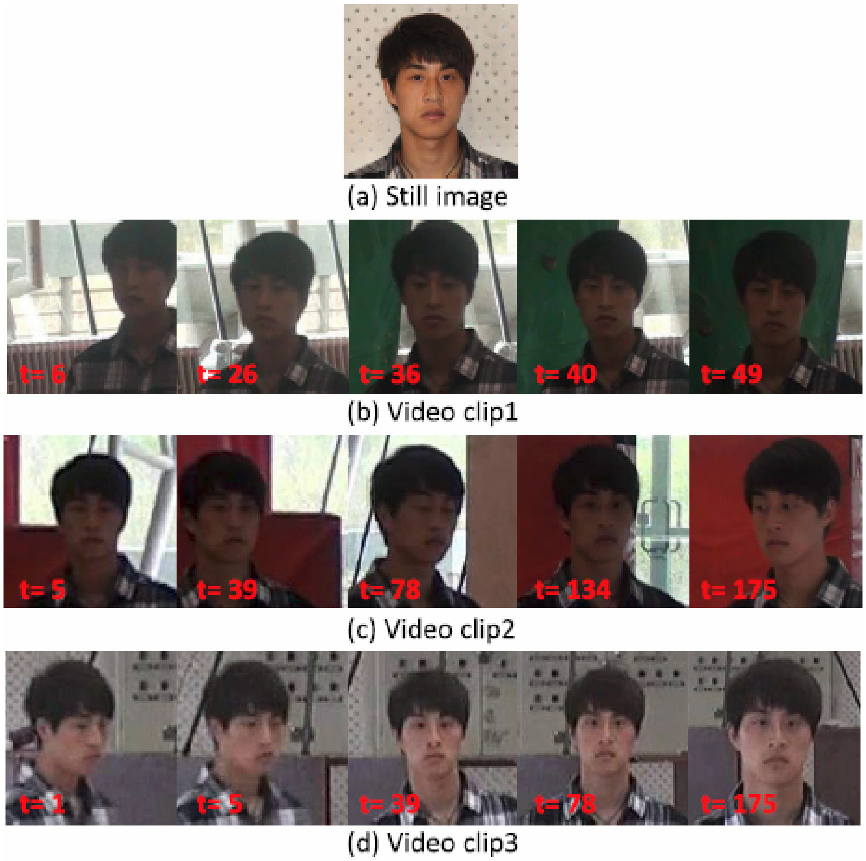

Fig. 12: Example of still images (a) and video frames (b-d) captured by the still camera and three camcorders in the COX Face DB, where the t value indicates the frame in each video sequence [15].

It can be seen in Table 1 that the TBE-CNN and HaarNet provide the highest level of accuracy, while they are very complex. Although the CCM-CNN and CFR-CNN cannot outperform these deep architectures, but they can achieve satisfactory results with significantly lower computational complexity. Moreover, the number of network parameters and layers are key factors in designing deep CNN that can greatly affect the convergence and training time. Considering these criteria, the proposed CCM-CNN and CFR-CNN have the lowest design complexity, and subsequently the shortest convergence time.

Table 1: Rank-1 recognition and computational complexity of video-based FR systems over videos of Cox Face DB.

FR system	**Rank-1 recognition**	**Computational complexity**		
		# operations	**# parameters**	**# layers**
CCM-CNN [24]	89.53±0.9	33.3M	2.4M	30
TBE-CNN [8]	90.61±0.6	12.8B	46.4M	144
HaarNet [25]	91.40±1.0	3.5B	13.1M	56
CFR-CNN [26]	87.29±0.9	3.75M	1.2M	7

5 Conclusion and Future Directions

In this chapter, the most recent deep learning architectures proposed for robust face recognition in video surveillance were thoroughly investigated. To overcome the existing challenges in real-world surveillance unconstrained environments, the single training reference sample and domain adaptation problems have been taken into account during the system design. On the other hands, computational complexity is also a key issue to provide an efficient solution for real-time video-based FR systems. In particular, this chapter reviewed deep learning architectures proposed based on triplet-loss function and autoencoder CNNs.

Triplet-based loss optimization method allows to learn complex and non-linear facial representations that provide robustness across inter- and intra-class variations. CCM-CNN proposes a cost-effective solution that is specialized for still-to-video FR from a single reference still by simulating weighted CCM. TBE-CNN and HaarNet can extract robust representations of the holistic face image and facial components through an ensemble of CNNs containing one trunk and several branch networks. In addition, to compensate the limited robustness of facial model in the case of single reference still, they were fine-tuned using synthetically-generated faces from still ROIs of non-target individuals. In contrast, CFR-CNN employed a supervised autoencoder CNN to generate canonical face representations from low-quality video ROIs. It can therefore reconstruct frontal faces that correspond to capture conditions of reference still ROIs and generate discriminant face representations. Experimental results obtained with the COX Face DB indicated that TBE-CNN and HaarNet can achieve higher level of accuracy with heavy computational complexity, while CCM-CNN and CFR-CNN can provide convincing performance with significantly lower computational costs.

Since the use of deep learning is increasingly growing, one of the future direction is to integrate conventional methods with deep learning methods in order to incorporate statistical and geometrical properties of faces into the deep features. In addition, future research can focus on utilizing temporal information, where facial ROIs can be tracked over frames to accumulate the predictions over time. Thus, the combination of face detection, tracking, and classification in a unified deep learning-based network will lead to a robust spatio-temporal recognition suitable for real-world video surveillance applications. Thus, 3D CNNs and recurrent neural networks such as long short-term memory can be exploited to consider convolutions through the time, due to capturing temporal information among successive video frames.

Acknowledgements This work was supported by the Fonds de Recherche du Québec - Nature et Technologies and MITACS.

References

1. Barr, J.R., Bowyer, K.W., Flynn, P.J., Biswas, S.: Face recognition from video: A review. International Journal of Pattern Recognition and Artificial Intelligence **26**(05) (2012)
2. Bashbaghi, S., Granger, E., Sabourin, R., Bilodeau, G.A.: Watch-list screening using ensembles based on multiple face representations. In: ICPR, pp. 4489–4494 (2014)
3. Bashbaghi, S., Granger, E., Sabourin, R., Bilodeau, G.A.: Dynamic ensembles of exemplar-svms for still-to-video face recognition. Pattern Recognition **69**, 61 – 81 (2017)
4. Bashbaghi, S., Granger, E., Sabourin, R., Bilodeau, G.A.: Robust watch-list screening using dynamic ensembles of svms based on multiple face representations. Machine Vision and Applications **28**(1), 219–241 (2017)
5. Canziani, A., Paszke, A., Culurciello, E.: An analysis of deep neural network models for practical applications. arXiv preprint arXiv:1605.07678 (2016)
6. Chellappa, R., Chen, J., Ranjan, R., Sankaranarayanan, S., Kumar, A., Patel, V.M., Castillo, C.D.: Towards the design of an end-to-end automated system for image and video-based recognition. CoRR **abs/1601.07883** (2016)
7. Dewan, M.A.A., Granger, E., Marcialis, G.L., Sabourin, R., Roli, F.: Adaptive appearance model tracking for still-to-video face recognition. Pattern Recognition **49**, 129 – 151 (2016)
8. Ding, C., Tao, D.: Trunk-branch ensemble convolutional neural networks for video-based face recognition. IEEE Trans on PAMI **PP**(99), 1–14 (2017). DOI 10.1109/TPAMI.2017.2700390
9. Gao, S., Zhang, Y., Jia, K., Lu, J., Zhang, Y.: Single sample face recognition via learning deep supervised autoencoders. IEEE Transactions on Information Forensics and Security **10**(10), 2108–2118 (2015)
10. Ghodrati, A., Jia, X., Pedersoli, M., Tuytelaars, T.: Towards automatic image editing: Learning to see another you. In: BMVC (2016)
11. Gomerra, M., Granger, E., Radtke, P.V., Sabourin, R., Gorodnichy, D.O.: Partially-supervised learning from facial trajectories for face recognition in video surveillance. Information Fusion **24**(0), 31–53 (2015)
12. Heo, Y.S., Lee, K.M., Lee, S.U.: Robust stereo matching using adaptive normalized cross-correlation. IEEE Trans on PAMI **33**(4), 807–822 (2011)
13. Huang, G.B., Lee, H., Learned-Miller, E.: Learning hierarchical representations for face verification with convolutional deep belief networks. In: CVPR (2012)
14. Huang, G.B., Ramesh, M., Berg, T., Learned-Miller, E.: Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Tech. Rep. 07-49 (2007)
15. Huang, Z., Shan, S., Wang, R., Zhang, H., Lao, S., Kuerban, A., Chen, X.: A benchmark and comparative study of video-based face recognition on cox face database. IP, IEEE Trans on **24**(12), 5967–5981 (2015)
16. Kamgar-Parsi, B., Lawson, W., Kamgar-Parsi, B.: Toward development of a face recognition system for watchlist surveillance. PAMI, IEEE Trans on **33**(10), 1925–1937 (2011)
17. Kan, M., Shan, S., Chang, H., Chen, X.: Stacked progressive auto-encoders (spae) for face recognition across poses. In: CVPR (2014)
18. Kan, M., Shan, S., Su, Y., Xu, D., Chen, X.: Adaptive discriminant learning for face recognition. Pattern Recognition **46**(9), 2497–2509 (2013)

19. Le, Q.V.: Building high-level features using large scale unsupervised learning. In: ICASSP (2013)
20. Ma, A., Li, J., Yuen, P., Li, P.: Cross-domain person re-identification using domain adaptation ranking svms. IP, IEEE Trans on **24**(5), 1599–1613 (2015)
21. Matta, F., Dugelay, J.L.: Person recognition using facial video information: A state of the art. Journal of Visual Languages and Computing **20**(3), 180 – 187 (2009)
22. Mokhayeri, F., Granger, E., Bilodeau, G.A.: Synthetic face generation under various operational conditions in video surveillance. In: ICIP (2015)
23. Pagano, C., Granger, E., Sabourin, R., Marcialis, G., Roli, F.: Adaptive ensembles for face recognition in changing video surveillance environments. Information Sciences **286**, 75–101 (2014)
24. Parchami, M., Bashbaghi, S., Granger, E.: Cnns with cross-correlation matching for face recognition in video surveillance using a single training sample per person. In: AVSS (2017)
25. Parchami, M., Bashbaghi, S., Granger, E.: Video-based face recognition using ensemble of haar-like deep convolutional neural networks. In: IJCNN (2017)
26. Parchami, M., Bashbaghi, S., Granger, E., Sayed, S.: Using deep autoencoders to learn robust domain-invariant representations for still-to-video face recognition. In: AVSS (2017)
27. Parkhi, O.M., Vedaldi, A., Zisserman, A.: Deep face recognition. In: BMVC (2015)
28. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: CVPR (2015)
29. Sun, Y., Chen, Y., Wang, X., Tang, X.: Deep learning face representation by joint identification-verification. In: NIPS (2014)
30. Sun, Y., Wang, X., Tang, X.: Hybrid deep learning for face verification. In: ICCV (2013)
31. Sun, Y., Wang, X., Tang, X.: Deep learning face representation from predicting 10,000 classes. In: CVPR (2014)
32. Sun, Y., Wang, X., Tang, X.: Deeply learned face representations are sparse, selective, and robust. In: CVPR (2015)
33. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: CVPR (2015)
34. Taigman, Y., Yang, M., Ranzato, M., Wolf, L.: Deepface: Closing the gap to human-level performance in face verification. In: CVPR (2014)
35. Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P.A.: Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. JMLR **11**, 3371–3408 (2010)
36. Yang, M., Van Gool, L., Zhang, L.: Sparse variation dictionary learning for face recognition with a single training sample per person. In: ICCV (2013)
37. Yim, J., Jung, H., Yoo, B., Choi, C., Park, D., Kim, J.: Rotating your face using multi-task deep neural network. In: CVPR (2015)
38. Zheng, J., Patel, V.M., Chellappa, R.: Recent developments in video-based face recognition. In: Handbook of Biometrics for Forensic Science, pp. 149–175. Springer (2017)
39. Zhu, Z., Luo, P., Wang, X., Tang, X.: Multi-view perceptron: a deep model for learning face identity and view representations. In: NIPS (2014)
40. Zhu, Z., Luo, P., Wang, X., Tang, X.: Recover canonical-view faces in the wild with deep neural networks. arXiv preprint arXiv:1404.3543 (2014)