Video-based Face Recognition using Deep Learning

for Single Sample Per Person (SSPP) surveillance applications

Mostafa Parchami

Department of Computer Science and Engineering
University of Texas at Arlington

This dissertation is submitted for the degree of
Doctor of Philosophy

College of Engineering May 2017

I would like to dedicate this thesis to my loving parents ...

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains fewer than 65,000 words including appendices, bibliography, footnotes, tables and equations and has fewer than 150 figures.

Mostafa Parchami
May 2017

Acknowledgements

A very special gratitude goes out to all down at Ford Motor Company for helping and providing the funding for the work.

With a special mention to my mentor and supervisors Dr. Gian-Luca Mariottini, Dr. Ramez Elmasri who made this possible and Saman Bashbaghi, Saif Sayed, Aaron Staranowicz, Gustavo Puerto, and Mohammadhani Fouladgar. It was fantastic to have the opportunity to work majority of my research at Astra Lab. What a cracking place to work!

And finally, last but by no means least, also to everyone in the department of computer science at UTA, it was great with all of you during last four years.

Thanks for all your encouragement!

Abstract

Face Recognition (FR) is the task of identifying a person based on images of the face of the identity. Systems for video-based face recognition in video surveillance seek to recognize individuals of interest in real-time over a distributed network of surveillance cameras. These systems are exposed to challenging unconstrained environments, where the appearance of faces captured in videos varies according to pose, expression, illumination, occlusion, blur, scale, etc. In addition, facial models for matching must be designed using a single reference facial image per target individual captured from a high-quality still camera under controlled conditions. Deep learning has shown great improvement in both low-level and high-level computer vision tasks. More specifically, deep learning outperforms traditional machine learning algorithms in FR applications. Unfortunately, such methods are not designed to overcome the challenges in video-based FR such as difference in source and target domain, single sample per person (SSPP) issue, low quality images, etc. Therefore, more sophisticated algorithms should be designed to overcome these challenges. We propose to design different deep learning architectures and compare their capabilities under such circumstances. Deep learning can not only learn how to discriminate between faces, it can also learn how to extract more distinctive features for FR applications. Thus, in each chapter we pursue a different type of deep convolutional neural networks to extract meaningful face representations that are similar for faces of the same person and different for faces of different persons. Chapter 2 provides a novel method for implementing cross-correlation in deep learning architectures and benefits from transfer learning to overcome SSPP aspect of the problem. Later, chapter 3 improves the results by employing a triplet-loss training method. Chapter 4, uses a much complex architecture for face embedding to achieve better accuracy. Chapter 5, employs a convolutional autoencoder to frontalize faces and finally, chapter 6, shows another application of cross-correlation in deep learning. Extensive experiments confirm that all of the proposed methods outperform traditional computer vision systems.

Table of contents

List of figures **xv**

List of tables **xix**

1 Introduction **1**
1.1 Motivation 1
1.2 Research Thrusts 3

2 Paper I: Robust Video-Based Single Sample Face Recognition Using a Specialized CNN **5**
2.1 Introduction 7
2.2 Background of techniques 9
2.3 Proposed method 10
2.3.1 Feature Extraction Pipeline 11
2.3.2 Patch Matching 12
2.3.3 Training 12
2.4 Experiments 14
2.4.1 Video datasets 14
2.4.2 Experimental Setup 15
2.4.3 Experimental Results 17
2.5 Conclusion 19

3 Second Paper: Video-Based Face Recognition Using Ensemble of Haar-Like Deep Convolutional Neural Networks **21**
3.1 Introduction 24
3.2 HaarNet Architecture 25
3.2.1 Face embedding: 26
3.2.2 Second-order statistics regularized loss function: 28
3.2.3 Training phase: 31

3.2.4 Recognition process: . . . 32
3.3 Experiments . . . 32
3.3.1 Datasets: . . . 32
3.3.2 Protocol: . . . 33
3.3.3 Performance metrics: . . . 36
3.3.4 Results . . . 36
3.4 Conclusion . . . 39

4 Convolutional NNs for Face Recognition in Video Surveillance Using a Single Training Sample Per Person 41
4.1 Introduction . . . 43
4.2 Proposed system . . . 45
4.2.1 Feature extraction . . . 45
4.2.2 Cross-correlation matching . . . 47
4.3 Pairwise triplet-loss training . . . 47
4.3.1 Pre-training . . . 47
4.3.2 Fine-tuning . . . 48
4.4 Experiments . . . 49
4.4.1 Video datasets . . . 49
4.4.2 Experimental setup . . . 49
4.4.3 Experimental results . . . 50
4.5 Conclusion . . . 53

5 Video-Based Single Sample Face Recognition Using Face Frontalization via Autoencoders Deep Neural Networks 55
5.1 Introduction . . . 57
5.2 Proposed network . . . 59
5.2.1 Training frontalization network . . . 60
5.2.2 Training Classification network . . . 62
5.3 Experiments . . . 63
5.3.1 Video datasets . . . 63
5.3.2 Experimental setup . . . 63
5.3.3 Experimental results . . . 64
5.4 Conclusion . . . 66

6 Deep Feature Tracker: A Novel Application for Deep Convolutional Neural Networks 67
6.1 Introduction 69
6.2 Proposed Method 72
6.2.1 Feature Detection Network 72
6.2.2 Feature Tracking Network 73
6.2.3 Training The Architecture 74
6.3 Experimental Results 77
6.3.1 Evaluation on KITTI Flow 2015 78
6.3.2 Evaluation on MIS dataset 79
6.3.3 Evaluation on UBC Patches dataset 80
6.4 Conclusion 82

References 83

List of figures

1.1 Video-based face recognition scenarios. (a): Video-to-still scenario, (b): Still-to-video scenario. (c): Video-to-video scenario. 1

2.1 The block diagram of the deep learning-based face recognition framework. . 10
2.2 The block diagram of feature extraction convolutional neural network. . . . 11
2.3 Sample of augmented face images generated for the fine-tuning stage. The first row represents face thumbnails generated by one level of sub-sampling where the second row represents images generate by two levels of subsampling followed by upsampling. *a*: the original still image, *b*: flip, *c*: rotation, *d*: shearing, *e*: translation. 14
2.4 Examples of Cox Face DB (top row) and Chokepoint dataset (bottom row) video face thumbnails, where they contain variations in camera viewpoints, pose, expression, blurriness, and occlusion. 15

3.1 Haar-like features used in branch networks. 26
3.2 HaarNet architecture for the trunk and three branches. (Max pooling layers after each inception and convolution layer are not shown for clarity). 27
3.3 Processing of triplets to compute the loss function. The network inputs a batch of triplets to the HaarNet architecture followed by an $L2$ Normalization. 29
3.4 Illustration of the regularized triple loss principle based on the mean and standard deviation of 3 classes, assuming a 2D representation of the facial ROIs. 30
3.5 Examples of LFW (top row), Cox Face DB (middle row) and Chokepoint (bottom row) datasets, where they contain different variations in camera viewpoints, pose, expression, blurriness, and occlusion. The left most column represents high-quality frontal still faces (for Cox Face DB and Chokepoint datasets). 33

3.6 Sample of augmented facial images generated from a Chokepoint still for the fine-tuning stage. The first row represents facial ROIs generated by one level of sub-sampling, while the second row represents images generate by two levels of subsampling followed by upsampling. 34
3.7 ROC curves of HaarNet and baseline FR methods for videos of each camera in the Cox Face DB. 38

4.1 The block diagram of the proposed video-based FR system illustrating pair-wise triplet-loss training. 46
4.2 ROC curves of the proposed method and baseline FR methods for videos of each camera in the Cox Face DB. 50

5.1 The block diagram of the proposed frontalisation autoencoder network. . . 60
5.2 T-shaped weight mask used for the proposed CFR-CNN loss function. . . . 61
5.3 The block diagram of the proposed classification network. 61
5.4 Sample outputs of the frontalization network. The top row are the probe ROIs used as input and the bottom row are their corresponding reconstructed canonical faces. 62
5.5 ROC curves of the proposed method and baseline FR methods for videos of each camera in the Cox Face DB. 64

6.1 Application of pixel based target tracking in biopsy. Left: The image where an optical biopsy site is selected. Right: The image where the site is tracked from previous frames using tracked keypoints. [81] 70
6.2 Application of pixel based feature tracking in AR where the tracked pixels are used as anchor points to overlay a pre-operative 3-D model. Left: The tracked points visualized on the current frame. Right: The overlaid CT-scan model on top of the image. [56] 71
6.3 Overall diagram for the Deep-PT. The method takes in input the live video from the single camera and detects and tracks features over time. 72
6.4 Main diagram for the feature detection pipeline. 73
6.5 Main diagram for the feature tracking pipeline. 74
6.6 Sample of training points generated for KITTTI Flow 2012 dataset. Each row presents a single pair of consecutive images with features marked with green dots. 75
6.7 UBC Patches dataset [23] contains several viewpoints of each 3D point and is challenging due to different levels of rotation, translation and scale. . . . 76

6.8 Qualitative comparison of Deep-PT Vs. forward-backward KLT-tracker where the lines show correspondences. Top row: visualization of the tracking performed by the Deep-PT over a cropped region of an image from KITTI Flow dataset. Bottom row: visualization of the tracking performed on the same image by the KLT-tracker 79
6.9 Qualitative comparison of Deep-PT Vs. forward-backward KLT-tracker where the lines show correspondences. Top row: visualization of the tracking performed by the Deep-PT over a pair of consecutive frames from the MIS dataset. Bottom row: visualization of the tracking performed on the same images by the KLT-tracker 81

List of tables

2.1 Rank-1 recognition accuracy of the proposed still-to-video scenario against state-of-the-art methods on COX Face DB 17
2.2 Rank-1 recognition accuracy of the proposed video-to-still scenario against state-of-the-art methods on COX Face DB 18
2.3 Area Under PR curves on ChokePoint Dataset 18
2.4 Running time (in seconds) comparison of different methods in the S2V/V2S face recognition 19

3.1 Specifications of the trunk network. 27
3.2 Specifications of the 3 branch networks. 28
3.3 Parameters of the regularized triplet-loss function used during the training process. 36
3.4 Rank-1 accuracy for still-to-video FR over the COX Face DB. 36
3.5 Rank-1 recognition for video-to-still FR over the COX Face DB. 37
3.6 The comparison of complexity (number of parameters that need to be estimated) by TBE-CNN and HaarNet architectures. 37
3.7 Average AUPR for videos of the Chokepoint. 38

4.1 Rank-1 accuracy of the proposed network against state-of-the-art FR systems on COX Face DB. 51
4.2 Average AUPR for videos of the Chokepoint along with the comparison of complexity (number of parameters and operations). 52

5.1 Rank-1 accuracy of the proposed network against state-of-the-art FR systems on COX Face DB. 64
5.2 Average AUPR performance for Chokepoint videos along with the comparison of complexity (number of operations, network parameters and layers). . 65

6.1 Tracker's training parameters. Note that in addition to the learning rate decay, the learning rate is decreased by factor of 0.2 every 30 epochs after the epoch number 120. 76
6.2 Score Network's training parameters. Note that in addition to the learning rate decay, the learning rate is decreased by factor of 0.1 every 30 epochs after the epoch number 120. 77
6.3 X-pixel tracking accuracy of Deep-PT and forward-backward KLT tracker in percentage. 78
6.4 Pixel back-projection error and inlier percentage for the MIS dataset. 80
6.5 UBC matching results. Numbers are Error at %95 recall in percentage. . . . 81

Chapter 1

Introduction

1.1 Motivation

Recognizing faces of the same subjects in unconstrained environments makes face recognition (FR) challenging, due to variations in face appearances of the same identity. Such variations are entitled to changes in ambient lighting, different poses, facial expressions, occlusions, blurriness, etc. [33], [64]. Systems designed for video-based FR aim to identify a target individual over a network of video cameras, where video face thumbnails are matched against still face images such as mug-shots, passport or driver license photos [5], [24].

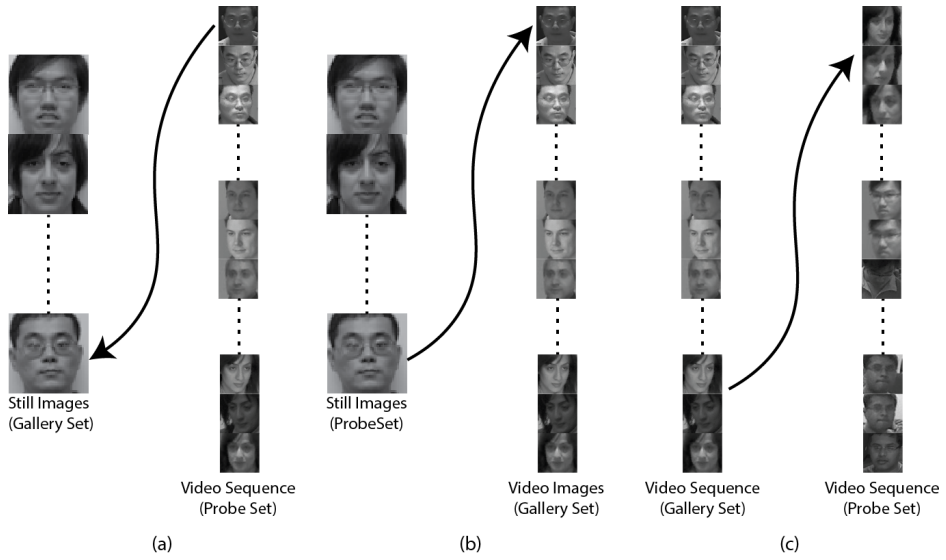


Fig. 1.1 Video-based face recognition scenarios. (a): Video-to-still scenario, (b): Still-to-video scenario. (c): Video-to-video scenario.

Three different distinct scenarios including still-to-video, video-to-still, and video-to-video scenarios defined in video-based FR [31] are illustrated in figure 1. The still-to-video scenario compares a high-quality still face image captured by a still camera under controlled conditions against a database of low-quality video sequences captured by video cameras under uncontrolled conditions [5]. On the contrary, the video-to-still scenario matches a video sequence against still face images stored in the gallery, while the video-to-video scenario queries a given video sequence against a set of target video sequences recorded from surveillance videos [80].

State-of-the-art video-based FR systems yet decline to perform accurately on video face databases with real-world setting under limited data circumstances [5], [31], [6], [38]. For example, unified subspace FR methods such as PCA, LDA, Bayesian face, and methods based on metric learning models are failed to simultaneously reduce the complex intra-class variations and enlarging the inter-class variations, due to their linear nature or shallow structures [64], [67].

Recently, different techniques have shown to be successful to handle the problem of limited reference samples in order to generate a representative facial model, specifically in still-to-video and video-to-still FR [24], [31], [6], [16], [32]. These systems are typically proposed to compensate the lack of adequate face samples through employing multiple face representations, face synthesizing, and augmenting the target samples in order to enlarge the training set [25].

In addition, sparse representation-based classification methods lately provide a promising performance by taking the advantages of a generic auxiliary training set and dictionary learning [80], [16], [84]. In spite of the improvements achieved through the aforementioned methods, there still exists a significant gap compared to the human visual system [71]. Nevertheless, deep learning-based methods provide a robust and powerful tool to handle the intra-class and inter-class variations, and tackle the existing challenges in video-based FR [64], [19]. Such methods thus can learn an effective face representation directly from face images through their deep architecture and hierarchical nonlinear mapping [67], [68], [11], [28], [59].

To appropriately learn a face embedding that can reduce the intra-class variations, as well as, increase the inter-class variations, a triplet-based loss has been utilized in FaceNet [59] throughout a unit hyper-sphere space in order to disjoint the negative face thumbnail of other identities from the positive pair of two faces belonging to the same identity. Similarly, a Trunk-Branch Ensemble CNN (TBE-CNN) model has been proposed in [17] along with an improved triplet loss function to learn blur-insensitive face representations using a composition of both still face images and artificially blurred faces. This model is an end-to-end network that

shares the early- and mid-layer convolutional layers between the trunk (to extract holistic face features) and branch (to extract local face features) networks to efficiently extract discriminative face representations. The main drawback of the TBE-CNN is that it requires to detect the facial landmarks properly, where it can increase the complexity to perform in real-time applications, as well as, it may fail due to occlusion.

The goal of this thesis is to advance robustness of video-based face recognition systems using deep learning methods to empower surveillance networks for watch-list screening. We mainly focus on feature extraction from face image thumbnails to embed more representative information in face representations.

1.2 Research Thrusts

Thrust 1: Incorporate triplet-loss in face representation learning

As mentioned before, learning discriminative representation of face thumbnails is the key to success in face recognition. The goal is to utilized the triplet-loss function during the training process of the network to enhance the face representations and increase the discriminative power of the face representations. In order to achieve high performance in face recognition, we proposed to pursue the following steps:

- Design a more sophisticated network architecture to embed more informative knowledge in the face representations.
- Enhance the triplet-loss in order to achieve higher discriminative power.
- Provide a training scheme to effectively train the network using triplet-loss function.
- Train the network over a big dataset containing preferably millions of images to achieve higher level of accuracy and robustness.

Thrust 2: Design a network to frontalize face images

The most important challenge in video-based FR is changes in viewpoint, pose, and facial expression. The most intuitive method is to generate a neutral and frontal image based on the low-quality video image. In order to build such system we suggest to follow these instructions:

- Design an autoencoder or generative adversarial network to learn how to transform images.
- Design a novel training approach for SSPP problem.

Thrust 3: Utilizing the proposed network in low-level computer vision

Our preliminary studies confirm that the proposed patch-based matching scheme can be successfully applied to many low-level computer vision problems such as stereo reconstruction, optical flow and feature tracking. The goal of this research is to study the capability of such network for more basic computer vision tasks and promote learning in these applications. This proposed research roughly consists of two stages:

- Design and implementation of a similar network to address regression problems as opposed to classification required for face recognition.
- Study the performance of learning-based methods in one of the low-level computer vision applications.

Chapter 2

Paper I: Robust Video-Based Single Sample Face Recognition Using a Specialized CNN

Abstract

Video-based face recognition (FR) systems attempt to recognize individuals of interest precisely over a distributed network of surveillance cameras throughout unconstrained environments. These systems are subject to challenging operational conditions, where the appearance of faces changes severely due to variations in pose, scale, expression, illumination, occlusion, blur, and etc. In addition, still images are taken using a high-quality still camera under controlled condition, whereas lower quality video cameras are typically used to capture faces with a different view point and under uncontrolled conditions. However, considering the assumption of single training sample per person turns FR into a more complicated problem to design robust facial models. In order to perform video-based FR accurately in real-time applications, this paper presents a deep learning architecture to learn discriminative and consistent face representations. In particular, an specialized deep convolutional neural network (DCNN) is exploited to effectively extract features from the face thumbnails and compare the still face with the video face thumbnails. To that end, still faces of the target individuals are matched against probe faces through a patch-based template matching approach. In order to tackle the single training sample issue, the proposed framework makes use of transfer learning to fine-tune the network considering our assumptions. The proposed system is extensively evaluated on the challenging COX Face DB and Chokepoint datasets, where the experimental results reveal that our method efficiently outperforms the state-of-the-art video-based FR systems with much less design complexity.

2.1 Introduction

The aim of video-based FR systems is to recognize the target persons based on the facial biometric traits over a network of surveillance cameras in unconstrained environments [6, 34, 80]. To that end, faces captured from video cameras are concurrently matched against facial models generated a priori for each target person [5, 24]. To generate a facial model, different types of FR applications may be considered including still-to-video, video-to-still, and video-to-video FR [31]. Specifically, high-quality still images captured from an still

camera under controlled condition are exploited in still-to-video and video-to-still FR, while only low-quality video sequences captured from video cameras under uncontrolled conditions are employed in video-to-video FR applications [17]. In real-world still-to-video and video-to-still FR scenarios as considered in this paper, the number of existing reference stills for each target person is very limited. Therefore, constructing a discriminant and representative facial model is complicated [19]. In addition, design a robust FR system is a challenging task, due to nuisance factors occasionally observed in such environments, including variations in ambient lighting, pose, scale, expression, blurriness, and occlusion [50].

When only a single reference still is available to design a facial model for each target person, the problem is called single sample face recognition (SSFR) [19]. In this regard, FR systems are typically declined to perform accurately due to lacking of different profile views [5, 17]. To address the SSFR problems, several approaches have been proposed to date w.r.t. the FR literature containing techniques used for augmenting the target samples, extracting multiple representations, and using auxiliary data to enlarge the training set [6, 16, 24, 31, 32]. To that end, face synthesizing through morphing or 3D reconstruction can be employed to produce additional target samples and enhance the intra-class variations [25]. Different holistic and local appearance-based feature extraction techniques can be also used to generate multiple face representations [5, 6]. Moreover, extended sparse representation-based classification methods were also proposed to deal with the SSFR problems, where they utilize a generic auxiliary training set and dictionary learning to discriminate between the still and video faces [16, 80].

Although the aforementioned methods achieved convincing improvements to overcome the SSFR challenges, yet the current FR systems suffer from the significant performance gap in compare with the human visual system [71]. Recently, DCNNs yield to a higher level of performance compared to other approaches [19]. In this paper, we proposed a patch-based face matching framework by employing a DCNN to extract features from both still and video face thumbnails. The extracted features are shown to be a robust facial model and thus, the proposed framework achieves a higher accuracy. Moreover, the matching pipeline exploits a matrix dot product followed by a fully connected layer that resembles the cross-correlation in hand-crafted feature extraction techniques.

To the best of authors' knowledge, the proposed framework is the first deep learning method designed for patch-based face matching under single training sample condition. We propose to incorporate transfer learning by generating simulated video images using the existing still images to further fine-tune the DCNN. Thus, the network can learn the intra-class variations and subsequently, improve the recognition accuracy by more than 15%. The proposed method is extensively evaluated on two challenging datasets and the experimental

results suggest an immense improvement after applying the fine-tuning stage. By employing the aforementioned approaches, the proposed method achieves a high performance in both face identification and verification over a network of surveillance cameras.

The rest of the paper is organized as follows. A brief review of the background of techniques is outlined in Section 2. The proposed method is described in Section 3, where the deep architecture and design strategies are explained. Section 4 provides the extensive experimental results containing the datasets used for experiments, as well as, the experimental protocols. Finally, the Section 5 concludes the results obtained in this paper and discusses the future directions of the research.

2.2 Background of techniques

Learning effective feature representations directly from the face images through deep networks has been recently provided a successful tool for Video-based FR [11, 28, 59]. For instance, a facial component-based CNN has been learned in [88] to transform frontal and well-illuminated faces of target individuals in different poses and illuminations, where features of the last hidden layer are employed as face representations. Furthermore, DeepID, DeepID2, and DeepID2+ have been proposed in [65, 68, 69], respectively, to learn a set of discriminative high-level feature representations. Thus, ensemble of CNN models was trained in [68] using the holistic face image and several overlapping/non-overlapping face patches to handle the pose and occlusion variations. Fusion of these models is typically carried out by concatenation to construct over-complete and compact representations. Followed by [68], dimension of the last hidden layer representations was increased in [65, 69], as well as, exploiting supervision to the convolutional layers in order to learn hierarchical non-linear feature representations. These representations thus enhance the inter-personal variations due to extraction of features from different identities separately, and simultaneously reduce the intra-personal variations through feature extraction from the same identity together.

In contrast to DeepID series, a 3D accurate face alignment was incorporated in DeepFace to derive a robust face representation through a nine-layer deep CNN [71]. In [67], 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. Similarly, a triplet-based loss has been lately exploited in [59] and [54] to learn a face embedding, where the loss aims to split the positive pair of two matching face thumbnails from the negative non-matching face thumbnail. Moreover, in the case of SSFR, a deep supervised auto-encoder neural network has been recently proposed to learn a robust face representations [19]. To that end, non-frontal faces with different perturbation factors are mapped with the canonical face

(frontal face with normal illumination and neutral expression) of the same person in order to extract insensitive features.

2.3 Proposed method

In the case of SSFR, we can reduce the problem to patch matching between the still image and the face thumbnail extracted from the video. However, achieving a high accuracy under this scheme requires careful selection of features and matching method. On the other hand, DCNNs are proven to achieve higher accuracy and robustness than the traditional computer vision patch-based matching algorithms [3, 40, 86]. This superiority is even shown in low-level feature extraction and patch-matching applications such as Optical Flow, Stereo Matching, and etc. [24, 46, 76]. In order to perform SSFR using a patch-based DCNN algorithm, the problem should be redefined as a patch matching problem and then a suitable network should be designed to take advantage of this re-definition.

Given two images in the patch-based matching single image FR problem, one high-quality frontal still image and one face thumbnail extracted from a low-quality video, the problem is to assign a likelihood for these two images being the same person. Based on this definition, the face recognition problem is reduced to matching two image patches. Thus, the face recognition framework iterates over still images and compares them to the given image to detect which person it belongs to.

The block diagram for the proposed framework is shown in Figure 2.1. As shown in this diagram, the method utilizes a database of still images (gallery set) which are high-quality frontal still images taken from the subjects of interest. The framework consists of two major components including feature extraction and patch matching.

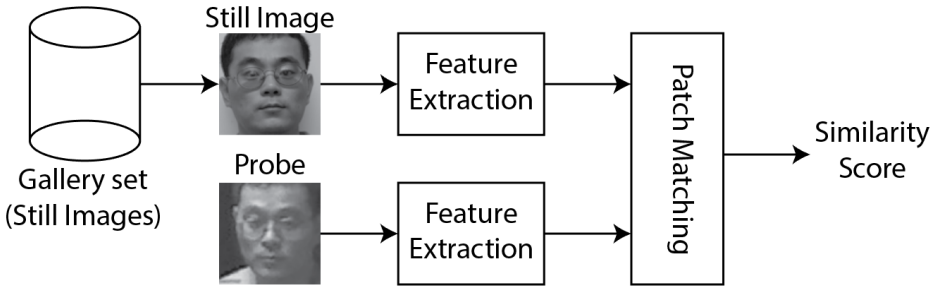


Fig. 2.1 The block diagram of the deep learning-based face recognition framework.

Feature extraction pipeline is responsible for extracting distinctive features from each face thumbnail such that these features are similar for two different images from the same person. The patch matching component takes features extracted from the images and computes the

likelihood of the faces belong to the same person. Each of these components is described in the following sections.

2.3.1 Feature Extraction Pipeline

The feature extraction pipeline is inspired by the DCNN proposed in [46]. In this paper, the goal is to extract features from two patches and localize the left patch in the right patch. Despite the difference in the domain, this pipeline is able to effectively extract complex features from a local patch. Therefore, in this paper we adopted a similar pipeline with minor modifications to extract good features for face matching. The block diagram of the feature extraction is presented in Figure 2.2.

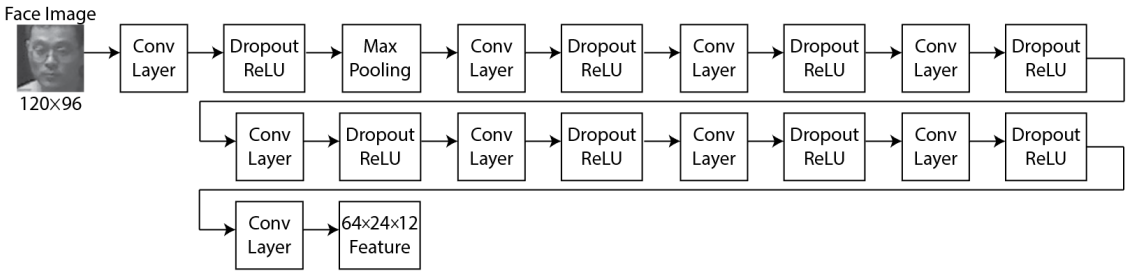


Fig. 2.2 The block diagram of feature extraction convolutional neural network.

Feature extraction is carried out by 9 convolutional layers each followed by a spatial batch normalization, drop-out, and RELU layers except the last convolutional layer which is not followed by a RELU in order to maintain the final feature representing the face thumbnail intact and avoid losing informative data for the classification. Moreover, similar to [71], a single max pooling layer is added after the first convolution layer. The purpose of the max pooling layer is increasing the robustness to small translation of faces in the patch. Most of the state-of-the-art methods for SSFR such as [31] heavily rely on accurately detecting and cropping the face thumbnail and do not consider any possible displacement which highly affects the matching score. However, the pooling layer would avoid such discrepancy between still and video face thumbnails.

It Is worth mentioning that the two feature extraction pipelines shown in Figure 2.1 share the same parameters. This makes sure that the features extracted from the two images are consistent and comparable. Each convolutional layer in Figure 2.2 has 64 filters of size 5x5 without applying any padding. Thus, given the input size of 120x96, the output will be of size 64x24x2.

2.3.2 Patch Matching

After extracting features from the still and the video thumbnails, a matching method should be employed to effectively compare these features and measure the similarity. The comparison in our proposed framework has three stages: matrix dot product, fully connected neural network, and finally a softmax. There are several approaches to join the two branches of the deep network. One option would be to concatenate the two feature vectors and form a single big vector and pass it as an input to the fully connected network. This approach has been employed in [24] and [83]. However, in our case, the resulted feature vector is much bigger and merging the two big vectors makes training the network more challenging. On the other hand, authors in [59], took a different approach and instead of having two separate pipelines, they utilized a triplet-based loss function. These triplets consist of two matching face thumbnails and one non-matching thumbnail. Therefore, the network minimizes the distance between an image to its positive samples.

However, in our scenario, we only have one single training sample for the subject of interest and the triplet loss function would not be appropriate. Thus, we followed the approach taken by [46] which uses matrix multiplication to simulate cross correlation in neural network framework. Dot product of the two matrices gives us a single three dimensional feature matrix that represents the two images. Then, this matrix is vectorized to obtain a one-dimensional feature vector of size 18432. This feature vector is then fed in to a two-layer fully connected neural network that classifies the input vector as either a match or a non-match. Furthermore, a softmax layer is applied to obtain a log score for each of the two classes (match and non-match).

2.3.3 Training

Being restricted to single training sample per subject, forced us to adopt a slightly different approach in training the network. The training consists of two phases where during the first phase the feature extraction pipeline is trained to obtain discriminative features from the face thumbnails. In this phase, the fully connected layer is also trained to be able to classify face thumbnails as matching or non-matching. However, at this point the network has no idea about the subjects that are supposed to be classified. Therefore, the second phase of training, fine-tuning, is responsible to train the pre-trained network using the still images such that it will be able to determine the right face among the faces in the gallery. The following sub-sections focus on the pre-training and fine-tuning stages and describe the essence of each stage.

Pre-training

During the pre-training, the network will be trained as a general face thumbnail matching system. Thus, it has no prior knowledge about the subjects of interest and the focus of this stage is to train mainly the feature extraction pipeline. For this purpose, we use a pool of matching and non-matching images from the COX Face DB. The trick to obtain a high accuracy is to generate an unbalanced training dataset. The dataset should contain positive matches as well as negative matches. Generating the positive samples is straightforward where a still image from a subject is paired with the video face thumbnails of the same subject. However, generating negative samples is more challenging and requires a well defined protocol. In our experiments, we generated the negative samples by pairing a still image with a randomly video thumbnail of other subjects. In order to make an unbalanced dataset, for each positive pair, we sampled two negative pairs to generate a dataset with wide variety of positive and negative samples. and our experiments has shown the effectiveness of this sampling scheme.

Fine-tuning

Fine-tuning stage is where the network actually acquires knowledge about the similarities and dissimilarities between the subjects of interest. So far, the network is pre-trained on face thumbnails that are not expected to be seen during the query. In order to improve the facial model and include the gallery information to enhance the intra-class variations, we propose to fine-tune the network with augmented images generated based on the still images. Thus, for each still image, a set of augmented images are generated by the following transformations: shearing, mirroring, rotating, and translating the original still image. Then, two levels of sub-sampling are applied to each of these images to obtain two images per transformation operation. While shearing, mirroring, rotating and translating is increasing the diversity in the viewpoint and facial appearance, sub-sampling encodes different distances from the camera as well as the quality of face thumbnail. After sub-sampling all images are up-sampled to the same size than the still image to replicates the low-quality video face thumbnails. Figure 2.3 presents some of these augmented images generated for the fine-tuning on the testing dataset.

For fine-tuning, similar to the pre-training an unbalanced number of matching and non-matching pairs are fed to the network. However, in contrast with the pre-training, the focus here is to learn dissimilarities between the still images and thus the parameters of the feature extraction pipeline are fixed. By fixing these parameters, we make sure that the feature extraction will not be biased by the still images. Fine-tuning in this case does not require

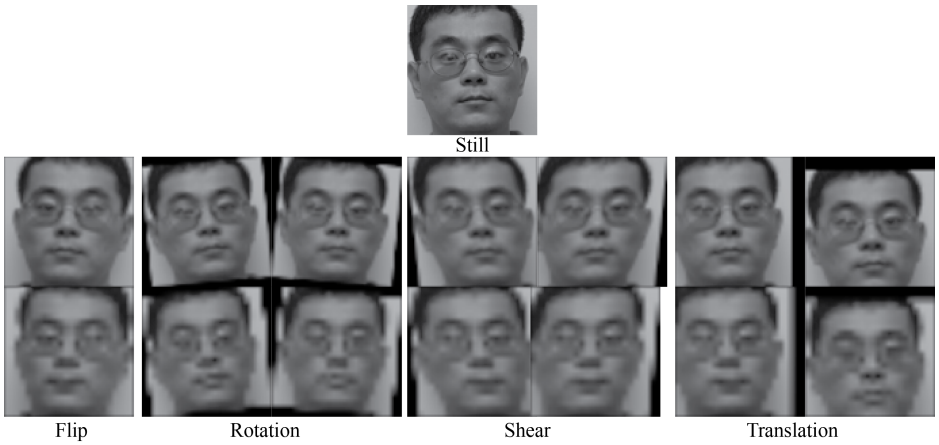


Fig. 2.3 Sample of augmented face images generated for the fine-tuning stage. The first row represents face thumbnails generated by one level of sub-sampling where the second row represents images generate by two levels of subsampling followed by upsampling. *a*: the original still image, *b*: flip, *c*: rotation, *d*: shearing, *e*: translation.

being extensive and only several epochs on the augmented dataset would boost the accuracy a lot.

2.4 Experiments

In this section, extensive experiments are performed to thoroughly assess performance of the proposed system against the state-of-the-art video-based FR systems. To evaluate different aspects of the proposed framework, several experiments are designed to perform still-to-video and video-to-still FR scenarios.

2.4.1 Video datasets

The experiments are conducted on two challenging datasets specifically designed for video-based FR: COX Face DB [31] and ChokePoint dataset [79]. Random examples face thumbnails of these datasets are shown in Figure 2.4, where they resemble the real-world surveillance environments.

The COX Face DB is constructed with participation of 1000 subjects. The dataset consists of one high quality still image and three uncontrolled video clips captured by low-resolution cameras for each subject. These videos are recorded while subjects are walking roughly along an S-shaped path to emulate different viewpoints and facial appearances similar to real-world scenarios. The video clips are captured by three different off-the-shelf cam-coders. Moreover, these videos are taken from the subjects walking in a large gymnasium with high

Fig. 2.4 Examples of Cox Face DB (top row) and Chokepoint dataset (bottom row) video face thumbnails, where they contain variations in camera viewpoints, pose, expression, blurriness, and occlusion.

ceiling, thus, the environment and camera setup approximates the outdoor lighting conditions. The ChokePoint dataset [79] is a collection of videos mainly obtained for experiments in person re-identification or face verification. The dataset contains still images of 25 subjects in portal 1 and 29 subjects in portal 2. In total, the dataset contains 64204 face thumbnails accurately extracted from the images from 48 video sequences captured using three cameras, in two portals and with subjects entering and leaving the scene.

2.4.2 Experimental Setup

In order to fairly compare the results of the proposed framework with sate-of-the-art systems, we followed standard experimental setups suggested by [5], [31], and [50]. For COX Face DB, we took the same training subjects to train the feature extraction pipeline, where 300 subjects are considered for training and 700 subjects for testing over a course of 10 iterations with random selection of training and testing subjects for each iterations. During training, all the still and video face thumbnails of the 300 subjects are adopted. On the other hand, the high-resolution still images from the rest 700 subjects are used during testing as the gallery set and the probe set contains the face thumbnails of the video clips from the corresponding 700 subjects. Thus, each probe is compared against all the gallery images and rank-1 recognition is reported as the accuracy of still-to-video FR system. Moreover, for video-to-still FR, the gallery set and probe sets are swapped and each still image is matched against video face thumbnails of all the 700 subjects. Furthermore, for fine-tuning, we use the still images of the 700 test subjects augmented to be similar to video face thumbnails. This allows the

network to gain knowledge about the possible appearance of people within the surveillance environments. However, we provide the results of our framework with or without fine-tuning stage.

In the experiment over ChokePoint dataset, 5 subjects of interest are randomly selected and thus the gallery set contains only the still thumbnails of these subjects. On the other hand, the probe set contains all video thumbnails of these subjects along with videos of 10 unknown subjects appeared in the capturing scene. Moreover, we utilize the pre-trained network that was already trained on COX Face DB to operate on the ChokePoint dataset. Except that the fine-tuning is performed using the still images of the ChokePoint dataset. Noted that in this experiment, the classifier does not have any knowledge about the background subjects and thus, this experiment is more realistic and challenging than the protocol used for COX Face DB. In another experiment, all the video thumbnails in the ChokePoint dataset are used as probe set and the gallery contains the 27 high quality controlled images. This experiment is similar to the above-mentioned experiment on the COX Face DB, however, the number of images in the gallery set is much lower and the number of video thumbnails for each of these subjects is extremely higher.

Meanwhile, in order to have a consistent neural network, we scale all the faces to 120x96 pixels. The convolutional neural network is implemented and trained using Torch 7.0 deep learning framework [13]. The training is performed for 30 epochs using the training data generated from the COX Face DB. Also, for the fine-tuning purpose on the COX Face DB, the network is trained for an additional 5 epochs on the simulated data generated from the still images. In order to fine-tune the network for ChokePoint dataset, the network is trained for 3 epochs on the simulated data generated from the still images from the same dataset. Rank-1 recognition accuracy of the proposed framework is compared against PSCL [31], learning Euclidean to Riemannian metric (LERM) [32], and TBE-CNN [17] on the COX Face DB and also ensemble-based method (EBM) [5], and [50] on the ChokePoint dataset.

Rank-1 recognition is computed based on the highest response in the gallery set for the given probe face. Although rank-1 recognition measure is an appropriate indication for face identification, However, the area under the ROC curve is a more desirable metrics for face verification [31]. The ROC space is defined as False Negative Rate (FNR) along x-axis and True Positive Rate (TPR) along the y-axis. TPR is the ratio of correctly classified face thumbnails as a target subject in the gallery over number of all probes with a corresponding thumbnail in the gallery. On the other hand, FNR is the ratio of incorrectly labeled probes as one of the target subjects over the number of non-target probes. Area Under the ROC Curve (AUC) is a well-known measure of detection performance and can be interpreted as the probability of classification over the range of TPR and FPR [5].

2.4.3 Experimental Results

The comparison of rank-1 recognition accuracy of the proposed still-to-video framework against state-of-the-art methods over COX Face DB is presented in Table 2.1.

Table 2.1 Rank-1 recognition accuracy of the proposed still-to-video scenario against state-of-the-art methods on COX Face DB

Cameras / Methods	Camera1	Camera2	Camera3
PSCL [31]	36.39 ± 1.6	30.87 ± 1.8	50.96 ± 1.4
LERM [32]	49.07 ± 1.5	44.16 ± 0.9	63.83 ± 1.6
TBE-CNN [17]	88.24 ± 0.4	87.86 ± 0.8	95.74 ± 0.7
Ours	71.47 ± 1.7	70.93 ± 1.1	76.67 ± 1.8
Ours+FT	97.81 ± 0.5	96.04 ± 0.9	98.79 ± 0.6

As shown in Table 2.1, the proposed method significantly outperforms other techniques that exploit hand-crafted features. However, the proposed patch-based matching DCNN fails to outperform the TBE-CNN [17] that uses a similar architecture than the one in FaceNet [59]. Authors in [17] trained the network on roughly 2.6 million training samples obtained from CASIA-WebFace database. Moreover, TBE-CNN employs an ensemble of convolutional neural networks to achieve a higher recognition accuracy. Despite the elegant and complex design of TBE-CNN, the proposed fine-tuning approach outperforms the TBE-CNN by a big margin with a simpler design and more sophisticated training methodology. Table 2.1 shows a remarkable improvement in rank-1 recognition after the proposed fine-tuning. The presented result supports our claim that most of the existing still-to-video FR systems lack in using the knowledge embedded in the still images. The proposed fine-tuning stage efficiently takes advantage of the still images in the gallery set to enhance the intra-class variations, as well as, inter-class variations between the subjects of interest. Moreover, the proposed face augmentation proved to be effective in reducing false negatives by learning the appearances of the face of the subjects in the gallery set.

Table 2.2 provides the rank-1 recognition for video-to-still scenario in comparison with the same methods. Due to existence of multiple video face thumbnails available in the gallery, a higher accuracy than still-to-video is expected.

As shown in Table 2.2, the proposed framework with fine-tuning exceeds the state-of-the-art methods in video-to-still FR scenario.

For comparison on the ChokePoint dataset, we adopted the trained network on COX Face DB and activated it without any modifications on the ChokePoint dataset. Thus, the network is fine-tuned using simulated video thumbnails augmented from the still images of the 5

Table 2.2 Rank-1 recognition accuracy of the proposed video-to-still scenario against state-of-the-art methods on COX Face DB

Cameras / Methods	**Camera1**	**Camera2**	**Camera3**
PSCL [31]	38.60 ± 1.4	33.20 ± 1.8	53.26 ± 0.8
LERM [32]	45.71 ± 2.0	42.80 ± 1.8	58.37 ± 3.3
TBE-CNN [17]	93.57 ± 0.6	93.69 ± 0.5	98.96 ± 0.2
Ours	83.23 ± 2.0	81.51 ± 2.8	85.42 ± 1.6
Ours+FT	95.43 ± 0.7	94.21 ± 1.0	95.90 ± 0.4

subjects and then the same operation is performed. The results and comparison with [5] are presented in Table 2.3, where the area under precision-recall (AUPR) curve is considered. AUPR is used to measure the performance under the imbalanced data circumstances, where the space is defined by Precison and TPR as Recall. PR is the ratio of true positives over the sum of true positives and false positives.

Table 2.3 Area Under PR curves on ChokePoint Dataset

Methods / Subjects	**EBM** [5]	**Ours**	**Ours+FT**
Individual #1	99.7 ± 0.09	82.62	97.52 ± 0.04
Individual #2	95.2 ± 1.8	85.42	98.32 ± 0.59
Individual #3	98.9 ± 1.6	80.25	98.95 ± 0.06
Individual #4	98.3 ± 1.4	84.95	99.74 ± 0.47
Individual #5	99.5 ± 0.06	86.85	99.33 ± 0.06

The final experiment is conducted similar to the protocol adopted by [50], where the training is performed on a separate dataset (in our case we have trained the network over COX Face DB) and evaluated on all of the face images in the ChokePoint dataset. Therefore, all video thumbnails are considered as probes and all still images are put in the gallery. In order to have a fair comparison, we have included the results of the proposed framework before and after fine-tuning stage. The rank-1 recognition rate documented in [50] for still-to-video FR is 62.7%, whereas we could reach up to 73.25% before fine-tuning. Moreover, by employing the same fine-tuning approach, we could achieve 97.46% rank-1 accuracy on all the images in the dataset.

Video-based FR requires real-time face verification and identification. The proposed framework is designed to be simple, yet accurate while maintaining the real-time aspect of the design. In order to confirm the feasibility of utilizing the proposed framework in video surveillance scenarios, the time complexity is compared with other state-of-the-art

approaches in Table 2.4. The training time is reported as the total training time over the COX Face DB dataset. However, the test times are reported as the running time for matching a single probe against all the 700 subjects in the gallery.

Table 2.4 Running time (in seconds) comparison of different methods in the S2V/V2S face recognition

Methods / Phase	**PSCL**[31]	**LERM**[32]	**Ours**
Train	865.36	1001.59	3652.36
Test	1.35	1.21	0.11
Fine-Tuning	*N.A.*	*N.A.*	186.57

Table 2.4 compares the run-time of the methods in train, test and fine-tuning (specific to our method) on an Intel(R) Core(TM) i7-37700M (3.40GHz) PC along with a GEFORCE GTX 1070 8GB, where the proposed framework achieves a significantly lower time complexity.

2.5 Conclusion

This paper presents a deep learning framework for video-based FR by adopting a patch-based matching DCNN architecture specialized for SSFR problems. In this framework, convolutional layers are employed to effectively extract discriminative features from the still and video face thumbnails. These rich features extracted from the face thumbnails provides robustness for face matching under variations in viewpoint of the camera and the facial appearance of the subjects. Feature matching of the two faces is performed by emulating cross correlation in deep neural network by applying a matrix dot product followed by a fully connected layer. The results suggest that the proposed matching scheme is very effective for the SSFR scenarios. Moreover, to overcome the single training sample challenge, transfer learning approach is applied in order to embed knowledge about the watch-list. The experimental results suggest that the proposed method is capable of learning a complex model for face matching that is effective for both still-to-video and video-to-still SSFR.

Three different sets of experiments designed to investigate the performance of the proposed framework under different real-world scenarios. The presented results over COX Face DB indicate the performance of the system in case of big watch-list and ensures that the system can identify the correct subject among a huge list of subjects of interest. On the other hand, the ChokePoint dataset has more video face thumbnails introducing more facial appearance variance in the dataset. Also, results indicate that the proposed fine-tuning

stage effectively increases the recognition accuracy of the network and by far outperforms the state-of-the-art methods. Moreover, fine-tuning is a natural way in deep learning to get around the lack of large amount of training data.

In order to achieve a higher level of performance, future research will be focused on utilizing spatio-temporal information. The idea is to track the subject over a set of frames and classify the face thumbnail for each frame and accumulate votes over time. The combination of face detection, tracking, and identification in a unified deep learning-based network not only would improve the accuracy, but also, can make a complete system that can be deployed for robust video-based FR.

Chapter 3

Second Paper: Video-Based Face Recognition Using Ensemble of Haar-Like Deep Convolutional Neural Networks

Abstract

Growing number of surveillance and biometric applications seek to recognize the face of individuals appearing in the viewpoint of video cameras. Systems for video-based FR can be subjected to challenging operational environments, where the appearance of faces captured with video cameras varies significantly due to changes in pose, illumination, scale, blur, expression, occlusion, etc. In particular, with still-to-video FR, a limited number of high-quality facial images are typically captured for enrollment of an individual to the system, whereas an abundance facial trajectories can be captured using video cameras during operations, under different viewpoints and uncontrolled conditions. This paper presents a deep learning architecture that can learn a robust facial representation for each target individual during enrollment, and then accurately compare the facial regions of interest (ROIs) extracted from a still reference image (of the target individual) with ROIs extracted from live or archived videos. An ensemble of deep convolutional neural networks (DCNNs) named HaarNet is proposed, where a trunk network first extracts features from the global appearance of the facial ROIs (holistic representation). Then, three branch networks effectively embed asymmetrical and complex facial features (local representations) based on Haar-like features. In order to increase the discriminativness of face representations, a novel regularized triplet-loss function is proposed that reduces the intra-class variations, while increasing the inter-class variations. Given the single reference still per target individual, the robustness of the proposed DCNN is further improved by fine-tuning the HaarNet with synthetically-generated facial still ROIs that emulate capture conditions found in operational environments. The proposed system is evaluated on stills and videos from the challenging COX Face and Chokepoint datasets according to accuracy and complexity. Experimental results indicate that the proposed method can significantly improve performance with respect to state-of-the-art systems for video-based FR.

3.1 Introduction

Systems for video-based FR attempt to accurately recognize individuals appearing in the field of view of a video camera. Three distinct scenarios can be considered in video-based FR – still-to-video, video-to-still, and video-to-video FR [31]. For example, the still-to-video FR scenario is relevant in watch-list screening applications, where individuals of interest are enrolled to a video surveillance system using reference facial images captured a priori under controlled conditions using a still camera (i.e., mug-shots, passport or driver license photos). Then, facial ROIs extracted from video captured over a distributed network of surveillance cameras are matched against those still ROIs stored during enrollment [5, 24]. In contrast, the video-to-video FR scenario is relevant, for example, in person re-identification applications, where individuals of interest are enrolled to a video surveillance system using reference facial trajectories captured a priori in videos [15, 80], and then matched against facial ROIs extracted from video trajectories captured over a network of cameras.

Recognizing the face of an individual in unconstrained real-world videos remains a challenging task, due in large part variations of facial appearances caused by changes in ambient lighting, poses, expressions, occlusions, scale, blur, etc. [34, 65]. The performance of state-of-the-art systems for video-based FR also declines in real-world environments when a limited number of ROIs is available during enrollment to design a robust facial model [31, 5, 6, 38]. In literature, unified subspace FR methods such as PCA, LDA, Bayesian Face, and metric learning methods cannot simultaneously reduce the complex intra-class variations and enlarge the inter-class discrimination due to their linear nature or shallow structures [65, 67]. Recently, some techniques have been successful for generating representative facial models given a limited number fo reference ROIs, specifically in still-to-video and video-to-still FR [31, 24, 6, 16, 32]. These systems are typically proposed to compensate the lack of representative reference facial ROIs face using multiple face representations, synthetic face generation, and augmenting the target samples in order to enlarge the training set [25, 8]. In addition, recent sparse representation-based classification methods have provided a promising performance by learning additional auxiliary (variational) dictionaries for robust modeling of intra-class variability in video environments [80, 16, 85, 51].

Despite the improvements achieved through the above-mentioned methods, there still exists a significant gap compared to the human visual system [71]. In this paper, deep learning methods are considered to provide robust modeling of intra-class and inter-class variations, and accurate video-based FR [65, 19]. Deep learning methods have been shown to learn effective face representations directly from face images through their deep architecture and hierarchical nonlinear mapping [67, 68, 11, 28, 59]. In particular, to learn a face embedding that can suitably reduce the intra-class variations, as well as, increase the inter-class variations,

a triplet-based loss has been utilized with FaceNet [59] in a compact Euclidean space in order to dissociate the negative facial ROIs of other identities from the positive pair of two faces corresponding to the same identity. Similarly, a Trunk-Branch Ensemble CNN (TBE-CNN) model has been proposed in [17] along with an improved triplet loss function to learn blur-insensitive face representations composed of both still face images and artificially blurred faces. This model is an end-to-end network that shares the early- and mid-layer convolutional layers between the trunk (to extract holistic features) and branch (to extract local features) networks to efficiently extract discriminative face representations. The main drawback of the TBE-CNN is that it requires the reliable detection of facial landmarks (that may fail due to occlusion), and thereby increase the complexity to perform in real-time applications.

In this paper, a novel end-to-end ensemble of DCNNs called HaarNet is proposed to efficiently learn robust and discriminative face representations for video-based FR applications. 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 is proposed for an end-to-end training process. The proposed triplet-loss function takes 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 is proposed to embed the correlation of facial ROIs stored during enrollment and improve recognition accuracy.

3.2 HaarNet Architecture

The overall architecture of the proposed HaarNet is presented in Fig. 3.2. Inspired by [17], this ensemble of deep convolutional neural networks (DCNNs) 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. As shown in Fig. 3.2, 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.

3.2.1 Face embedding:

Similar to [59] and [17], the face embedding is performed using a Haar-like deep neural network. In contrast with [17], instead of fusing the trunk and branch representations to obtain a final face representation using only one fully connected layer, we propose to concatenate the output of trunk and branches to obtain a final representation of the facial ROI. In particular, we propose to utilize three branch networks, where each branch computes one of the Haar-like features illustrated in Fig. 3.1. As outlined in [74] Haar features have been utilized for face detection 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.

In the architecture (see Fig. 3.2), 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.

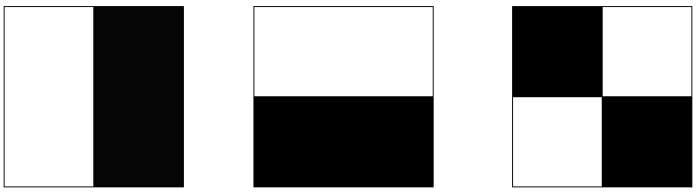


Fig. 3.1 Haar-like features used in branch networks.

As illustrated in Fig. 3.2, the first two convolutional layers (Conv1 and Conv2) extract low-level features representing local information [17]. These two layers share weights between all branches and the trunk. However, since the mid- and high-level features have different properties in each branch and the trunk, the corresponding layers don't share parameters.

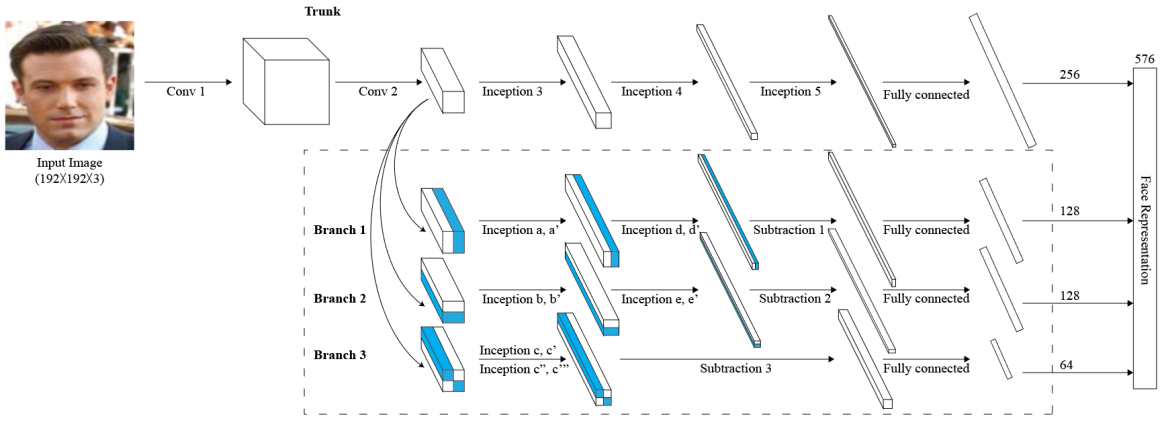


Fig. 3.2 HaarNet architecture for the trunk and three branches. (Max pooling layers after each inception and convolution layer are not shown for clarity).

The layers and specifications of the trunk network are presented in Table 3.1. For the trunk network, the configuration of GoogLeNet [70] is employed with 18 layers. In order to have a consistent input, all the face images are scaled to 192x192 pixels for all datasets.

Table 3.1 Specifications of the trunk network.

Features	Layer type	Kernel size/stride	Output size	Depth
Low-level features	Conv1	7x7/2	96x96x64	1
	Max pooling	2x2/2	48x48x64	0
	Conv2	3x3/1	48x48x192	2
	Max pooling	2x2/2	24x24x192	0
Mid-level features	Inception (3a)	-	24x24x256	2
	Inception (3b)	-	24x24x480	2
	Max pooling	2x2/2	12x12x480	0
High-level features	Inception (4a)	-	12x12x512	2
	Inception (4b)	-	12x12x512	2
	Inception (4c)	-	12x12x512	2
	Inception (4d)	-	12x12x528	2
	Inception (4e)	-	12x12x832	2
	Max pooling	2x2/2	6x6x832	0
	Inception (5a)	-	6x6x832	2
	Inception (5b)	-	6x6x1024	2
	Max pooling	2x2/2	3x3x1024	1
	Dropout	-	3x3x1024	1
	Fully connected	-	256	1

Table 3.2 presents the specification of the layers of the three branches of HaarNet, where each branch computes one of the Haar-like features. The specifications of those branches without some layers are marked by a hyphen.

Table 3.2 Specifications of the 3 branch networks.

Features	Layer type	Kernel size/stride	Branch1	Branch2	Branch3
Low-level features	Conv1	7x7/2	96x96x64	96x96x64	96x96x64
	Max pooling	2x2/2	48x48x64	48x48x64	48x48x64
	Conv2	3x3/1	48x48x192	48x48x192	48x48x192
	Max pooling	2x2/2	24x24x192	24x24x192	24x24x192
Mid-level features	Inception (a)	-	12x12x480	-	-
	Inception (a')	-	12x12x480	-	-
	Inception (b)	-	-	12x12x480	-
	Inception (b')	-	-	12x12x480	-
	Inception (c)	-	-	-	12x12x480
	Inception (c')	-	-	-	12x12x480
	Inception (c")	-	-	-	12x12x480
	Inception (c"')	-	-	-	12x12x480
	Max pooling	2x2/2	6x6x480	6x6x480	3x3x480
High-level features	Inception (d)	-	6x6x832	-	-
	Inception (d')	-	6x6x832	-	-
	Inception (e)	-	-	6x6x832	-
	Inception (e')	-	-	6x6x832	-
	Max pooling	2x2/2	3x3x832	3x3x832	-
	Dropout	-	3x3x832	3x3x832	3x3x480
	Subtraction 1	-	3x3x832	-	-
	Subtraction 2	-	-	3x3x832	-
	Subtraction 3	-	-	-	3x3x480
	Fully connected	-	128	128	64

3.2.2 Second-order statistics regularized loss function:

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 [59, 17, 75]. However, careful triplet sampling is a crucial step in order to achieve a faster convergence [59]. In addition, employing triplet-loss is challenging since the global distributions of the training samples are neglected in optimization process.

Ding and Tao [17] have shown that by adding a mean distance regularization term to the triplet-loss function, the distinctiveness of the face representation may improve. Fig. 3.4 illustrates the main idea of the proposed second-order statistics regularization term. In Fig 3.4 (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. 3.4 (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. 3.4 (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, this paper introduces a new term in the loss function to examine the intra-class distribution of the training samples.

Fig. 3.3 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.

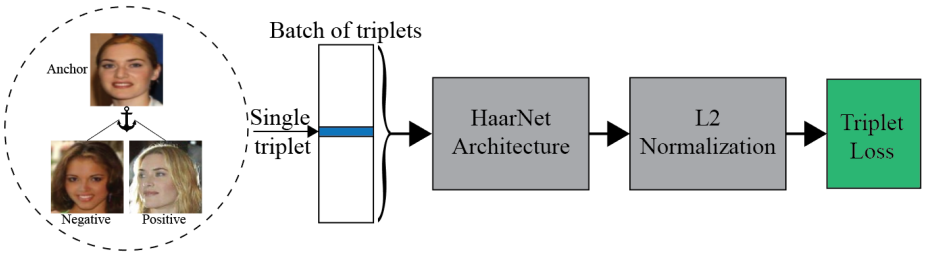


Fig. 3.3 Processing of triplets to compute the loss function. The network inputs a batch of triplets to the HaarNet architecture followed by an *L2* Normalization.

As shown in Fig. 3.3, 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.

The triplet constraint can be expressed as a function of the representation of anchor, positive and negative samples as follows [59]:

$$\left\| f\left(x_i^a\right) - f\left(x_i^p\right) \right\|_2^2 + a < \left\| f\left(x_i^a\right) - f\left(x_i^n\right) \right\|_2^2 \tag{3.1}$$

where $f(x_i^a)$, $f\left(x_i^p\right)$, and $f(x_i^n)$ 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:

$$L_{HaarNet} = \delta_1 L_{triplet} + \delta_2 L_{mean} + \delta_3 L_{std} \tag{3.2}$$

where δ_i denotes the weight for each term in the loss function. Furthermore, $L_{triplet}$ can be defined based on (3.1) as follows:

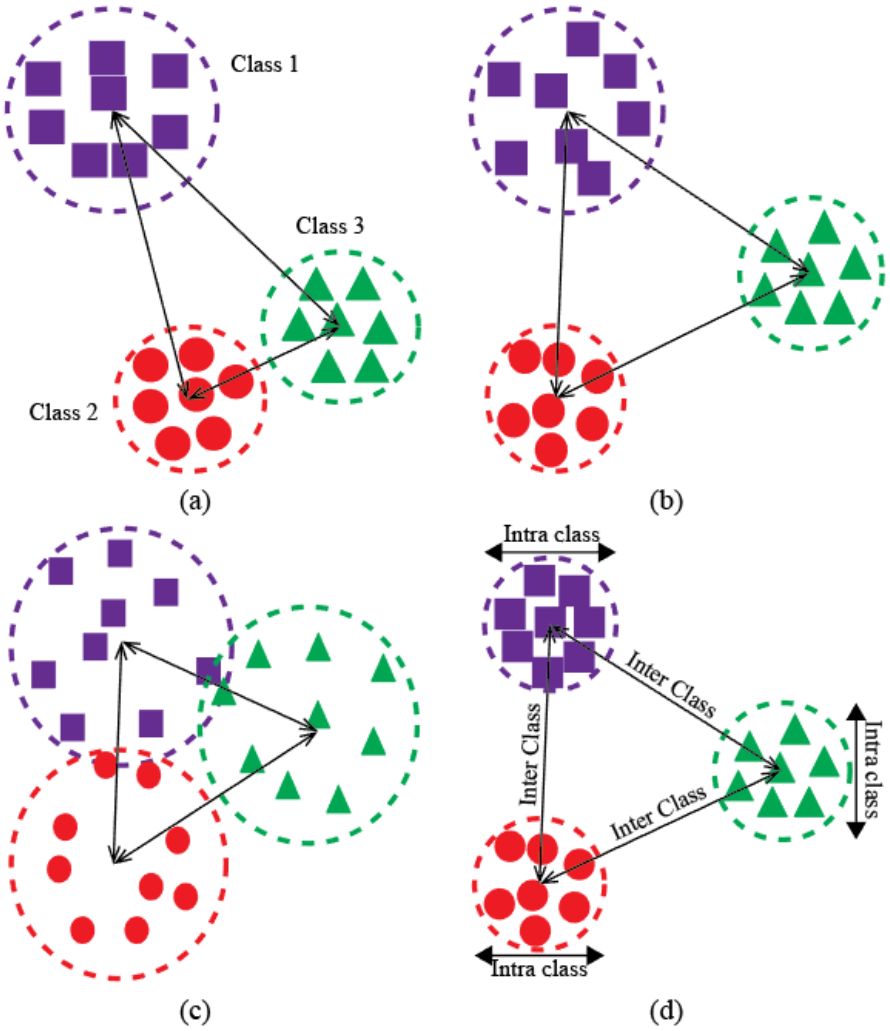


Fig. 3.4 Illustration of the regularized triple loss principle based on the mean and standard deviation of 3 classes, assuming a 2D representation of the facial ROIs.

$$L_{triplet} = \frac{1}{2N}$$

$$\sum_{i=1}^{N} \left[\left\| f\left(x_i^a\right) - f\left(x_i^p\right) \right\|_2^2 - \left\| f\left(x_i^a\right) - f\left(x_i^n\right) \right\|_2^2 + \alpha \right]_+ \tag{3.3}$$

Similar to [17], assuming that the mean distance constraint is $\beta < \|\hat{\mu}_c - \hat{\mu}_c^n\|_2^2$, we define L_{mean} as:

$$L_{mean} = \frac{1}{2P} \sum_{c=1}^{C} \max\left(0,\ \beta - \|\hat{\mu}_c - \hat{\mu}_c^n\|_2^2\right) \tag{3.4}$$

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

$$L_{std} = \frac{1}{M} \sum_{c=1}^{C} \max\left(0,\ \gamma - \sigma_c\right) \tag{3.5}$$

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 (3.2) can be optimized using the regular stochastic gradient descent with momentum similar to [17]. The gradient of loss w.r.t. the facial ROI representation of ith image for subject c (denoted as $f(x_{ci})$) is derived as follows:

$$\frac{\partial L_{std}}{\partial f(x_{ci})} = -\frac{1}{M} \sum_{c=1}^{C} \omega_c \frac{\partial \sigma_c}{\partial f(x_{ci})} \tag{3.6}$$

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:

$$\frac{\partial \sigma_c}{\partial f(x_{ci})} = \frac{\partial \sqrt{\frac{1}{N_c}\sum_{j=1}^{N_c} \left\| f\left(x_{cj}\right) - \mu_c \right\|_2^2}}{\partial f(x_{ci})} = \frac{\left[\sum_{j=1}^{N_c} \frac{1}{N_c} \left\| \mu_c - f\left(x_{cj}\right) \right\|_2\right] - \left\| \mu_c - f(x_{ci}) \right\|_2}{2\sqrt{\frac{1}{N_c}\sum_{j=1}^{N_c} \left\| f\left(x_{cj}\right) - \mu_c \right\|_2^2}} \tag{3.7}$$

As shown in Fig. 3.4 (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.

3.2.3 Training phase:

Training a network with multiple branches followed by a triplet-loss is tricky and requires careful attention to the details. A multi-stage training approach is hereby proposed to effectively optimize the parameters of the proposed 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.

3.2.4 Recognition process:

The HaarNet generates a 576 dimensional face representations consisting of a 256 dimensional feature extracted from the whole image concatenated with 320 dimensional Haar-like features. This heterogeneous face representation is incompatible with regular distance metrics such as Euclidean or Cosine distances. In order to employ the HaarNet method in a FR setup, we propose to train a fully connected layer followed by a "softmax" which takes two face representations as input and outputs a similarity score between zero and one. This layer is trained on LFW dataset for several epochs after the feature extraction pipeline is completely trained and later is fine-tuned on COX Face DB training set images.

3.3 Experiments

In this section, several experimental results are shown to evaluate and comparing the performance of the proposed HaarNet against the state-of-the-art video FR systems.

3.3.1 Datasets:

Experiments are conducted using challenging datasets designed specifically to video-based FR, LFW, COX Face DB and ChokePoint datasets. Example faces of three datasets used in this paper are presented in Fig. 3.5 that shows variations in the video ROIs for a specific subject similar to surveillance environments. Noted that LFW dataset has been only employed to train the HaarNet and adjust the network parameters with a large number of faces.

The COX Face DB [31] simulates real-world video surveillance data containing still and video images of 1000 subjects. For each subject, the dataset consists of one high-quality still image and three uncontrolled video clips recorded by low-resolution off-the-shelf cam-coders. These three videos are captured while subjects are walking roughly along an S-shaped path to

Fig. 3.5 Examples of LFW (top row), Cox Face DB (middle row) and Chokepoint (bottom row) datasets, where they contain different variations in camera viewpoints, pose, expression, blurriness, and occlusion. The left most column represents high-quality frontal still faces (for Cox Face DB and Chokepoint datasets).

emulate different poses and facial appearances similar to the real world scenarios. Moreover, these videos are taken from the subjects walking in a large gymnasium with high ceiling, thus, the environment and camera setup approximates the outdoor lighting conditions. In order to evaluate our proposed method on the COX Face DB, we adopted the still-to-video and video-to-still protocols introduced in [31].

The ChokePoint dataset [79] contains a collection of still images and videos for experiments in video-based FR that simulates the real-world surveillance conditions. The dataset contains still images of 25 subjects in portal 1 and 29 subjects in portal 2. In total, the dataset contains 64,204 facial ROIs accurately extracted from the images of 48 video sequences captured using three cameras, in two portals and with subjects entering and leaving the portals. For the comparison w.r.t. still-to-video scenario, we adopted the protocol proposed in [5] using a set of 5 randomly selected subjects of interest.

3.3.2 Protocol:

The main challenge in video-based FR is the lack of adequate amount of diversified training data to support training a deep model. Moreover, most of the video FR databases such as COX Face DB [31] and ChokePoint [79] contain a limited number of subjects and typically suffer from the lack of diversity in the video ROIs, specially diversity in different facial appearances with respect to the still ROIs. Following [17], we synthetically generate a video-like dataset from an existing dataset that contains a large number diverse subjects. In this

paper, motion blur and out-of-focus blur are emulated by adding noise to the original image. We further augment the artificially generated dataset by applying several transformations. For each artificially generated video ROI, we construct a set of images through the following transformations: shearing, mirroring, rotating, translating. These transformations help to enrich the artificial video dataset by simulating different viewpoints. Moreover, for each transformation we generate two images by applying two different levels of down-sampling followed by an up-sampling. The subsampling emulates different scales (distance from the camera) and also helps to embed the low-quality nature of the video facial ROIs.

In our experiments, HaarNet is trained on the Labeled Faces in the Wild (LFW) dataset [29]. In order to emulate video ROIs, an artificial video dataset containing roughly 8.2 million video face ROIs is generated. Additionally, the network is fine-tuned using the COX Face DB in order to embed the camera field of view information in the network. So far, the network has no knowledge about the subjects of interest enrolled to the system. In order to embed this knowledge, final fine-tuning round is employed over still ROIs of subjects. During this phase, another artificial video dataset is generated using only the still face ROIs by following the aforementioned process. The objective of the fine-tuning process is to train the network in order to acquire knowledge about similarities and dissimilarities among the subjects of interest based on their still and synthesized video ROIs. Fig. 3.6 presents some of augmented images generated for the fine-tuning stage.

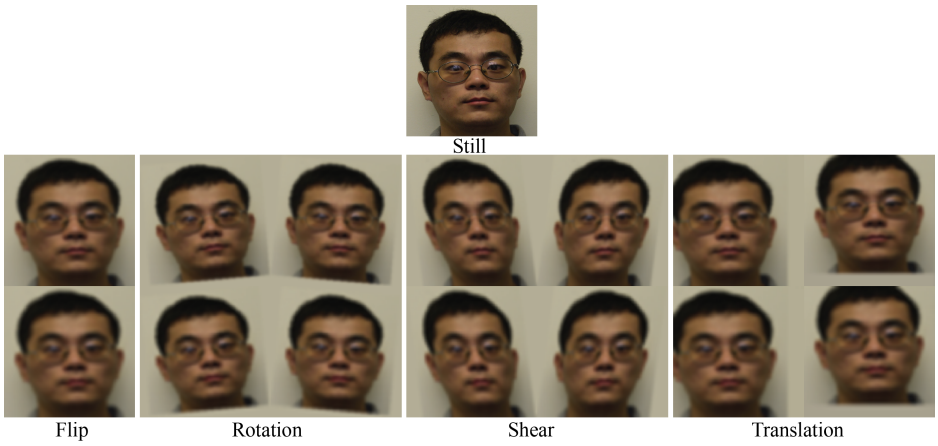


Fig. 3.6 Sample of augmented facial images generated from a Chokepoint still for the fine-tuning stage. The first row represents facial ROIs generated by one level of sub-sampling, while the second row represents images generate by two levels of subsampling followed by upsampling.

For COX Face DB, experiments conducted using the list of training and testing images provided as suggested by [31]. Thus, we used 300 subjects for training and 700 subjects for testing over a course of independent 10 replications, where the training and testing subjects

are randomly selected for each replication. Training is performed using all the still and video facial ROIs of the 300 subjects, while for testing, the high resolution still images from the remaining 700 subjects are used for enrollment as the gallery set and the probe set contains the facial ROIs of the three video clips from the corresponding 700 subjects. Therefore, each probe is matched against all the gallery images and rank-1 recognition performance is reported for still-to-video FR scenario. Moreover, for video-to-still FR, the gallery set and probe sets are swapped and each still image is compared against all the video facial ROIs of those 700 subjects. Furthermore, for fine-tuning, we used the still images of the 700 test subjects to generate the artificial video facial ROI dataset. This allows the network to gain knowledge about the subjects of interest. However, for the sake of fairness in comparisons, we provide the results of our framework with and without this final fine-tuning stage.

In the experiment with ChokePoint dataset, the instructions from [5] is followed in order to perform still-to-video FR. In this experiment, 5 subjects are selected randomly to be enrolled in the system. On the other hand, the probe set contains all video ROIs of these subjects along with 10 unknown subjects appeared in the operational scene, while their still images are not included in the gallery. In the final experiment, all the video ROIs in the ChokePoint dataset are used as probe set and the gallery contains the 27 high-quality controlled images. This experiment is similar to the aforementioned experiment on the COX Face DB with more video facial ROIs per person.

In order to have a consistent neural network, we resized all the facial ROIs from these two datasets to 192x192 pixels. Moreover, the LFW dataset was used to train the network. First, the trunk was trained for 30 epochs using a softmax, then each branch is trained for 20 epochs using a softmax loss. Subsequently, the complete network is assembled and trained by adopting a softmax loss function for 15 epochs. Finally, HaarNet is trained using the proposed regularized triplet loss for extra 15 epochs. The similarity measure network is then trained using the face representations obtained from the HaarNet on the LFW dataset. Thereafter, the similarity measure network is trained on the 300 training subjects from the COX Face DB for another 5 epochs. On the top of all these training stages, there is an additional fine-tuning stage using the artificially simulated video images based on the 700 images of the gallery, where we only fine-tune the final classification layer. The network trained on COX Face DB was used to assess on the ChokePoint dataset with an exception that the fine-tuning is performed using the simulated images generated from the still images of the ChokePoint dataset. Noted that in this experiment, the network has no knowledge about the background subjects and thus, this experiment would be a more realistic challenge than the protocol suggested for experimenting on the COX Face DB. The parameters and their corresponding values of the proposed triplet loss function are presented in Table 3.

Table 3.3 Parameters of the regularized triplet-loss function used during the training process.

Parameter	α	β	γ	δ_1	δ_2	δ_3
Value	0.2	0.3	0.2	0.5	0.3	0.2

3.3.3 Performance metrics:

In the experiments, rank-1 recognition is reported to compare the performance of the proposed HaarNet against the state-of-the-art video FR systems in a face identification scenario, while ROC curve is presented to perform a comparison under a face verification scenario. Rank-1 recognition is computed based on the highest response in the gallery (among enrolled subjects) for the given probe ROI. The rank-1 recognition and ROC curve of the HaarNet are compared against point-to-set correlation learning (PSCL) [31], learning euclidean-to-riemannian metric (LERM) [32], and TBE-CNN [17] on COX Face Database and ensemble-based method (EBM) [5], and [50] on the ChokePoint dataset.

Receiver Operating Characteristic (ROC) curve and the area under the ROC curve are more appropriate way for comparing methods in open-set authentication scenarios as found in video surveillance applications [31]. The ROC space is defined as False Negative Rate (FNR) along x-axis and True Positive Rate (TPR) along the y-axis. TPR is the ratio of correctly classified facial ROIs as a target subject in the gallery over number of all probes with a corresponding ROI in the gallery. On the other hand, FNR is the ratio of incorrectly labeled probes as one of the target subjects over the number of non-target probes. Area Under the ROC Curve (AUC) is a well-known global measure of detection performance and can be interpreted as the probability of the correct classification over the range of TPR and FPR [5].

3.3.4 Results

Table 3.4 presents Rank-1 accuracy of the proposed HaarNet and baseline systems on the COX Face DB.

Table 3.4 Rank-1 accuracy for still-to-video FR over the COX Face DB.

FR systems	**Video1**	**Video2**	**Video3**
PSCL [31]	36.39±1.61	30.87±1.77	50.96±1.44
LERM [32]	49.07±1.53	44.16±0.94	63.83±1.58
TBE-CNN [17]	88.24±0.45	87.86±0.85	95.74±0.67
HaarNet	89.31±0.94	87.90±0.60	97.01±1.65
HaarNet + FT	98.86±0.37	97.58±0.77	98.97±0.15

As shown in Table 3.4, the proposed method significantly outperforms hand-crafted feature extraction methods. By exploiting Haar-like features along with the novel triplet-loss function, the HaarNet can provide higher level of performance compared with the existing deep learning methods. Table 3.4 also shows an additional improvement in rank-1 accuracy after the proposed fine-tuning (HaarNet + FT). The presented results confirm that most of the existing still-to-video FR methods fail to convey the knowledge embedded in the still images. However, the proposed fine-tuning stage efficiently encodes the still images in the gallery to learn the similarities and dissimilarities among the subjects of interest. Moreover, by learning the facial appearance of the subjects of interest, the proposed data augmentation proved to be effective in reducing false negatives.

Table 3.5 shows the rank-1 accuracy for video-to-still FR in comparison with state-of-the-art FR methods. In this scenario, each still image is compared against all the video sequences. Due to existence of multiple video facial ROIs in the gallery, a higher accuracy than still-to-video FR scenario is expected. As shown in Table 3.5, the proposed HaarNet with fine-tuning surpasses the state-of-the-art methods for video-to-still FR.

Table 3.5 Rank-1 recognition for video-to-still FR over the COX Face DB.

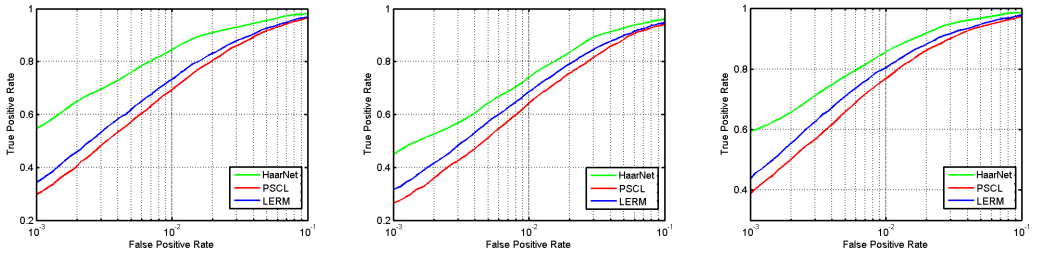
FR systems	**Video1**	**Video2**	**Video3**
PSCL [31]	38.60±1.39	33.20±1.77	53.26±0.80
LERM [32]	45.71±2.05	42.80±1.86	58.37±3.31
TBE-CNN [17]	93.57±0.65	93.96±0.51	98.96±0.17
HaarNet	92.73±1.93	93.57±1.62	97.48±1.54
HaarNet + FT	98.26±0.49	95.27±0.12	99.26±0.69

Amongst the state-of-the-art methods, TBE-CNN is the most competitive one after the proposed HaarNet. However, as shown in Table 3.6, HaarNet has a significantly lower computational complexity. Since both the TBE-CNN and HaarNet are based on GoogLeNet, the trunk network requires 5,798K parameters, while HaarNet contains 3 branches and TBE-CNN considers 7 branches for each face landmark, respectively. Thus, the proposed HaarNet is more efficient in terms of the number of parameters.

Table 3.6 The comparison of complexity (number of parameters that need to be estimated) by TBE-CNN and HaarNet architectures.

FR systems	**Number of parameters**		
	Trunk	**Branch**	**Trunk + Branch**
TBE-CNN [17]	5,798K	5,798K x 7	46.4M
HaarNet	5,798K	(3,338K x 2) + 654K	13.1M

Fig. 3.7 shows ROC curves for HaarNet, as well as, for PSCL [31] and LERM [32] for each camera, separately. As shown in this figure, the AUC accuracy for HaarNet is larger than others.



(a) camera 1 (b) camera 2 (c) camera 3

Fig. 3.7 ROC curves of HaarNet and baseline FR methods for videos of each camera in the Cox Face DB.

For evaluation on the Chokepoint dataset, we adapted the network trained on COX Face DB and tested it without any modifications on the Chokepoint dataset. Then, we fine-tuned the network using simulated video ROIs augmented from the still images of the five subjects of interest. The performance of the HaarNet against EBM [5] is presented in Table 3.7, where area under precision-recall (AUPR) curve is considered as the performance metric. AUPR is used to measure the performance under the imbalanced data circumstances, where the space is defined by TPR (recall) and precision. Precision is the ratio of true positives over the sum of true positives and false positives.

Table 3.7 Average AUPR for videos of the Chokepoint.

FR systems	**EBM [5]**	**HaarNet**	**HaarNet + FT**
AUPR	99.24±0.38	95.57±1.12	99.36±0.59

It is worth noting that, EBM [5] implements a complex individual-specific ensemble of classifiers for each subject of interest using multiple face representation, while HaarNet benefits from a deep specialized neural network.

The final experiment is performed using the the protocol adopted by [50], where the training is performed on a separate dataset (in our case, COX Face DB) and tested on all of the face images in the Chokepoint dataset. Therefore, all video ROIs are considered as probes and all still ROIs are registered in the gallery. However, the rank-1 accuracy rate documented in [50] for still-to-video FR is 62.7%, whereas we could reach up to 84.92% before fine-tuning. Moreover, by performing the aforementioned fine-tuning stage, HaarNet could achieve 96.12% rank-1 accuracy on all the probe images in the dataset.

3.4 Conclusion

This paper presents a deep neural network that can learn face representations for each target individual for accurate video-based FR systems. The proposed HaarNet architecture employs an ensemble of DCNN in order to obtain a discriminative embedding of the facial ROI. In particular, the network utilizes a trunk that shares weights with branches and each branch is trained to compute features similar to Haar-like features. The trunk is specialized for matching the global appearance of the face, while the branches embed informative features, such as pose, and asymmetrical facial features of the subjects. In order to effectively train the proposed deep architecture, a novel regularized triplet-loss function was proposed to generate face embedding with high similarity among intra-class samples, while maximizing the inter-class variations. In order to address the single training sample issue, synthetic facial images were generated form still images of the subjects of interest using different transformations, such as shearing, rotation, translation, and subsampling. Finally, the network was fine-tuned over the simulated video ROIs in order to utilize the knowledge existing in still images in the gallery set for higher recognition accuracy.

Several experiments were conducted to evaluate the performance of the proposed HaarNet under different real-world scenarios, such as still-to-video FR. The results obtained over COX Face DB and Chokepoint data indicate a convincingly higher level of accuracy of HaarNet, yet a lower complexity against state-of-the-art FR systems, even when the gallery set contains a large number of subjects. In order to achieve a higher level of performance, future research should 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 would lead to a robust spatio-temporal suitable for real-world video surveillance applications.

Chapter 4

Convolutional NNs for Face Recognition in Video Surveillance Using a Single Training Sample Per Person

Abstract

In video surveillance, face recognition (FR) systems seek to detect individuals of interest appearing over a distributed network of cameras. Still-to-video FR systems match faces captured in videos under challenging conditions (pose, illumination, etc.) against facial models designed using a single reference still per individual. Although CNNs can achieve among the highest levels of accuracy in many real-world FR applications, state-of-the-art CNNs that are suitable for still-to-video FR, like trunk-branch ensemble CNNs, represent complex solutions for real-time applications. In this paper, an efficient CNN architecture is proposed for accurate still-to-video FR. The CCM-CNN is based on new cross-correlation matching (CCM) and triplet-loss optimization methods that provide discriminant face representations. The matching pipeline exploits a matrix Hadamard product followed by a fully connected layer inspired by adaptive weighted cross-correlation. The triplet-based training approach is proposed to optimize the CCM-CNN parameters such that the inter-class variations are reduced, while enhancing robustenss to intra-class variations. Finally, to improve the robustness of facial models, the network is fine-tuned using unlabeled still and video faces of non-target individuals in the operational domain. Experiments on videos from the COX Face and Chokepoint datasets indicate that, although TBE-CNN and HaarNet can provide a higher level of accuracy, the CCM-CNN achieves comparable accuracy with significantly lower time and memory complexity. It may represent the better trade-off between accuracy and complexity for real-time FR.

4.1 Introduction

Face recognition (FR) is widely used in applications of law enforcement, forensics, biometrics and surveillance. In video surveillance applications, FR systems seek to recognize target individuals of interest appearing in unconstrained scenes based on their facial appearance [24, 34, 80]. Each face captured over distributed network of video cameras is segmented into a region of interest (ROI), and the pattern extracted from the ROI is matched against the facial models designed a priori for target individuals. Captured in uncontrolled conditions,

these faces may vary considerably according to pose, illumination, occlusion, blur, scale, expression, camera inter-operability, etc. [8, 17, 50]. When a person appears in a camera field of view, their face is initially detected and tracked over multiple frames, and the matching scores of each face model are integrated along a facial trajectory for robust spatio-temporal FR [**?**]. The computational complexity is therefore an important consideration because of the growing number of cameras, and the processing time of state-of-the-art face detection, tracking and matching algorithms.

Watch-list screening is a common yet challenging application in video surveillance. In this case, still-to-video FR systems are employed, where the facial model of target individuals are often designed using a single reference image or mugshot captured from a still camera under controlled conditions [8]. In pattern recognition literature, this challenging situation is referred to as a single sample per person (SSPP) problem [**?**]. Accordingly, the performance of still-to-video FR systems typically declines in complex real-world environments due mostly to the lack robustness of facial models to intra-class variations [8, 17]. To improve matching robustness, several approaches have been proposed to generate synthetic target samples, to extract multiple representations, and to exploit auxiliary data to enlarge the training set [24, 32, 31]. For instance, the reference face has been synthesized through morphing and 3D reconstructions to produce additional target facial images under various capture conditions [25**?**]. Classification systems based on different descriptors and local patch extraction methods have also used to generate multiple diverse face representations [8**?**]. Sparse representation-based classification methods have also been proposed that train auxiliary (variational) dictionaries to improve robustness [16, 51, 80].

Although the aforementioned methods can improve performance, current systems for still-to-video FR provide a low-level of accuracy in real-world watch-list screening applications [8, 31, 71]. Recently, deep convolutional Neural Networks (CNNs) have shown to achieve a high-level of accuracy in many FR applications, where effective facial representations are learned directly from large-scale datasets [11, 17]. For SSPP problems, triplet-based loss has recently been exploited in [17, 52, 54, 59] to learn a face embedding, where the loss seeks to discriminate the positive pair of matching facial ROIs from the negative non-matching facial ROI. In addition, branch-based CNNs like the Trunk-Branch Ensemble CNN (TBE-CNN) [17] and HaarNet [52] can further improve robustness to variations in facial appearance. The trunk network extracts features from the global appearance of faces (holistic representation), while branch networks effectively embed asymmetrical and complex facial traits (local overlapping/non-overlapping patch representations) to handle the pose and occlusion variations. For instance, HaarNet employes 3 branch networks based on Haar-like

features, along with a regularized triplet-loss function. However, these specialized CNNs represent complex solutions for real-time FR [**?**].

In this paper, an efficient CNN architecture is proposed for accurate still-to-video FR from a reference facial ROI per target individual. Based on a novel pair-wise cross-correlation matching (CCM) and a robust facial representation learned through triplet-loss optimization, the proposed CCM-CNN architecture is fast and compact (requires few network branches, layers and parameters). The contributions of this paper are threefold. First, the matching pipeline exploits a matrix Hadamard product followed by a fully connected layer that simulates the adaptive weighted cross-correlation technique [26]. Second, a novel triplet-based approach is proposed to optimize the representations of the triplet containing the positive, negative video ROIs and the corresponding still ROI. In particular, the similarity between the representations of positive faces in video ROIs and the still ROI is enhanced, while reducing the similarity between negative video ROIs and the still ROI, as well as, positive and negative representations. Finally, to improve robustness of facial models, fine-tuning of CCM-CNN incorporates knowledge of target individuals using synthetically-generated video ROIs based on the reference still faces. The accuracy and complexity of the proposed CCM-CNN is compared with state-of-the-art FR systems on videos from the COX Face and Chokepoint datasets.

4.2 Proposed system

The proposed network iterates over a batch of triplets containing the still ROI along with corresponding positive and negative video ROIs to learn robust representations of the triplets (see Figure 4.1). The proposed system consists of two major components including feature extraction and cross-correlation matching. Feature extraction pipeline extracts distinctive features from each ROI such that these features are similar for two images from the same person under different conditions like illumination, viewpoint, and facial expression. The cross-correlation matching component takes features extracted from the ROI and computes the likelihood of the faces belonging to the same person.

4.2.1 Feature extraction

To obtain a discriminative matrix representation of the facial ROI and perform cross-correlation matching, a customized version of [46] is adopted. Despite the differences in the domain of the target still and non-target video face ROIs, the proposed network is able

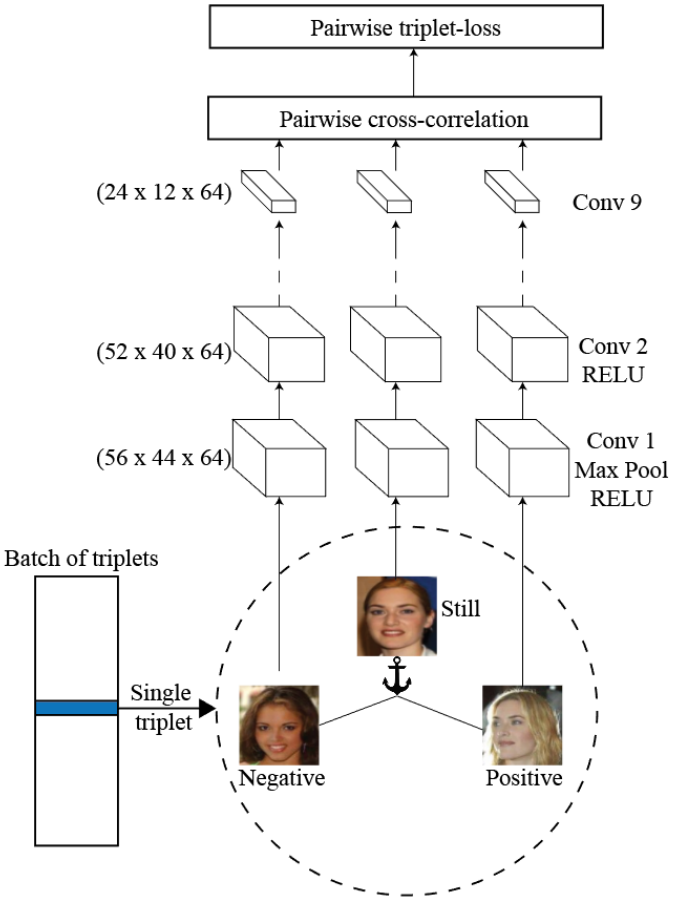


Fig. 4.1 The block diagram of the proposed video-based FR system illustrating pairwise triplet-loss training.

to effectively extract robust features. The block diagram of the feature extraction is presented in Figure 4.1.

As shown in Figure 4.1, feature extraction is carried out by a Siamese network consisting 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 features and to avoid losing informative data for the matching. 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. Nevertheless, most of the state-of-the-art systems for SSPP rely on accurate face alignment and do not consider any possible displacement which highly affects the local matching [31].

It is worth mentioning that all three feature extraction pipelines shown in Figure 4.1 share the same set of parameters. This ensures that the features extracted from the two images are consistent and comparable. Each convolutional layer has 64 filters of size 5x5

without applying any padding. Thus, given the input size of 120x96, the output will be of size 24x12x64 features.

4.2.2 Cross-correlation matching

After extracting features from the still and video ROIs, a local matching method is employed to effectively compare these features and measure the matching similarity. The comparison in the proposed system has three stages: matrix dot product, fully connected neural network, and finally a softmax. There are several approaches to join the two branches of the deep network. One basic approach is to concatenate the two feature vectors and form a single super vector and pass it as an input to the fully connected network [24, 83]. However, in the proposed system, the resulting feature matrix is large and merging the two large matrices makes training the network more challenging due to over-fitting and complexity.

Therefore, the matrix dot product is exploited to simulate cross-correlation in the neural networks. Dot product of the two matrices provides a single three dimensional feature matrix that represents the similarity of the two images. Then, this matrix is vectorized to obtain a one-dimensional feature vector of size 18,432. This feature vector is then fed into a two-layer fully connected neural network that classifies the input vector as either a match or a non-match. Furthermore, a softmax layer is applied to obtain a score for each of the two classes (match and non-match).

4.3 Pairwise triplet-loss training

A multi-stage training approach is considered to efficiently train the proposed network suitable for the SSPP problems. To that end, a pre-training stage is performed on a general face dataset where more training data is available and later the network is fine-tuned on a dataset designed to Video-based FR.

4.3.1 Pre-training

During the pre-training, the network is trained as a general face matching system. Thus, it has no prior knowledge about the subjects of interest and the focus is to train mainly the feature extraction pipeline. To that end, a pool of matching and non-matching images is employed from the Labeled Faces in the Wild [30]. The images from this dataset are augmented to obtain roughly 1.3 million initial triplets. The trick to obtain a high accuracy is to train the network with a set of hard to classify triplets. In order to consistently update the training triplets, we followed the on-line sampling method proposed in [59] for 50 epochs.

Deep networks are typically trained by back-propagating the loss function calculated by comparing the output of the network with the ground truth label. In contrast with [59], we propose a pair-wise triplet-based optimization approach to effectively train the proposed network. In order to adapt the network for pairwise triplet-based optimization, the network is modified by incorporating additional feature extraction branches.

As shown in Figure 4.1, each batch contains several triplets, where the network is supposed to learn the classification for each of them. During the training, each branch is labeled by one of the elements of the triplet, where the main branch is responsible to process the still ROI and the positive (negative) branch extracts features from the positive (negative) sample of the triplet. Moreover, the matching pipeline is modified to benefit from the triplets by introducing an Euclidean loss layer followed by a SoftMax which computes similarity for each RIO pair in the triplet. The proposed loss layer is exploited to compute the final loss of the network as formalized in Eq (1).

$$Loss = \sqrt{(1 - S_{sp})^2 + S_{sn}^2 + S_{np}^2} \tag{4.1}$$

where S_{sp}, S_{sn}, and S_{np} are the similarity scores between still and positive, still and negative, and negative and positive samples of the triplet, respectively, computed using the aforementioned cross-correlation matching approach. Once the network is trained, during operations, the additional feature extraction pipeline is disassembled from the network and only the still and the positive (negative) branches are taken into account. The main branch extracts features from the gallery images, while the other branch extracts features from the probe images.

4.3.2 Fine-tuning

In the fine-tuning stage, the proposed network acquires knowledge about the similarities and dissimilarities between the subjects of interest to be enrolled in the system. So far, the network is pre-trained on face ROIs that are not expected to be seen during operations. In order to improve the facial model and take into account the gallery information to enhance the intra-class variations, the network is fine-tuned with video-like synthesized face images generated based on the high-quality still images. Thus, 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. Then, two levels of sub-sampling are applied to each of these images to obtain two images per transformation operation. While shearing, mirroring, rotating and translating is increasing the diversity in the viewpoint and facial appearance, sub-sampling encodes different distances from the camera, as well as, the quality

of face ROI. After sub-sampling, all images are up-scaled to the same size, where the still image resembles the low-quality video face ROIs.

For fine-tuning, similar to the pre-training stage, a triplet-based optimization approach is also employed. The same sampling and training strategies are applied to effectively fine-tune the network. In contrast with the pre-training, the focus of the fine-tuning stage is to learn dissimilarities between the subjects of interest and thus the parameters of the feature extraction pipelines are fixed. Fine-tuning in this case does not require being extensive and only several epochs on the augmented dataset can significantly boost the accuracy.

4.4 Experiments

4.4.1 Video datasets

The experiments are conducted on two challenging datasets specifically designed for video-based FR: COX Face DB [31], ChokePoint [79] datasets. Cox Face DB and Chokepoint datasets can be employed to emulate real-world still-to-video FR scenario, where their main characteristics are that they contain a high-quality still face images captured under controlled condition (with the same still camera), and low-quality surveillance videos for each subject captured under uncontrolled conditions (with surveillance cameras). Videos are captured over a distributed network of cameras that covers a range of variations (changes in, e.g., pose, illumination, blur, scale). The COX Face DB is constructed with participation of 1000 subjects. The dataset consists of one high quality still image and three uncontrolled video clips captured by three different off-the-shelf low-resolution cam-coders for each subject. The ChokePoint dataset contains still images of 25 subjects in portal 1 and 29 subjects in portal 2. In total, the dataset contains 64,204 face ROIs extracted from 48 video sequences captured using three cameras locating above the portals and four different monthly sessions, with subjects entering and leaving the scene.

4.4.2 Experimental setup

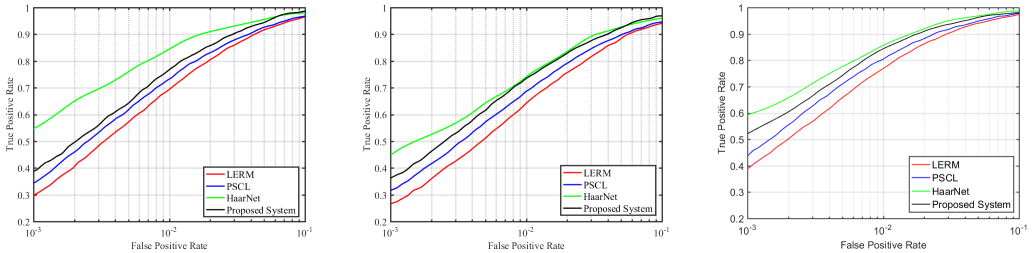
In order to fairly compare the results of the proposed network with sate-of-the-art systems, standard experimental setups suggested by [5], [31] and [50] are followed. For COX Face DB, the same training subjects are selected to train the feature extraction pipeline, where 300 subjects are considered for training and 700 subjects for testing over a course of 10 iterations with random selection of training and testing subjects for each iteration. During training, all the still and video face ROIs of the 300 subjects are adopted. On the other hand, the high-resolution still images from the rest of 700 subjects are used during testing as the gallery

set and the probe set contains the face ROIs of the video clips from the corresponding 700 subjects. Thus, each probe is compared against all the gallery images and rank-1 recognition is reported as the accuracy of still-to-video FR system. Furthermore, for fine-tuning, the still images of the 700 test subjects are used to perform augmentation of still faces to be similar to video face ROIs. This allows the network to gain knowledge about the probable appearance of people and contextual information within the surveillance environments. However, the results of the proposed network are provided with or without fine-tuning stage.

In the experiment over ChokePoint dataset, 5 subjects of interest are randomly selected and thus the gallery set contains only the still ROIs of these subjects. On the other hand, the probe set contains all video ROIs of these subjects along with videos of 10 unknown subjects appeared in the capturing scene. Moreover, the pre-trained network that was already trained on COX Face DB is utilized to operate on the ChokePoint dataset. Apart from that the fine-tuning is performed using the still images of the ChokePoint dataset.

Meanwhile, in order to design a consistent network, all the faces are scaled to 120x96 pixels. The proposed network is implemented using Torch 7.0 deep learning framework [13]. The training is performed for 30 epochs using the training data gathered from the COX Face DB. Also, for the fine-tuning purpose on the COX Face DB, the network is trained for an additional 5 epochs on the augmented faces synthesized from the still images. In order to fine-tune the network for ChokePoint dataset, the network is trained for 3 epochs on the simulated video faces generated from the still images from the same dataset. Rank-1 recognition accuracy and ROC curve of the proposed network is compared against (point-to-set correlation learning) PSCL [31], learning Euclidean to Riemannian metric (LERM) [32], VGG-Face [54], TBE-CNN [17] and HaarNet [52] on the COX Face DB and also ensemble-based method (EBM) [8], and [50] on the ChokePoint dataset.

4.4.3 Experimental results



(a) Camera1 (b) Camera2 (c) Camera3

Fig. 4.2 ROC curves of the proposed method and baseline FR methods for videos of each camera in the Cox Face DB.

The comparison of rank-1 accuracy of the proposed system for still-to-video FR against state-of-the-art video-based FR system over COX Face DB is presented in Table 4.1. Rank-1 accuracy is computed based on the highest response in the gallery set for the given probe face ROI.

Table 4.1 Rank-1 accuracy of the proposed network against state-of-the-art FR systems on COX Face DB.

FR systems	**Camera 1**	**Camera 2**	**Camera 3**
PSCL [31]	36.39 ± 1.6	30.87 ± 1.8	50.96 ± 1.4
LERM [32]	49.07 ± 1.5	44.16 ± 0.9	63.83 ± 1.6
VGG-Face [54]	69.61 ± 1.5	68.11 ± 0.9	76.01 ± 0.7
TLF-CNN	88.65 ± 1.1	87.82 ± 0.8	92.13 ± 0.9
TBE-CNN [17]	88.24 ± 0.4	87.86 ± 0.8	95.74 ± 0.7
HaarNet [52]	89.31 ± 0.9	87.90 ± 0.6	97.01 ± 1.7

As shown in Table 4.1, the proposed network significantly outperforms PSCL and LERM that exploit hand-crafted features. Moreover, the proposed system provides comparable Rank-1 accuracy to TBE-CNN and HaarNet. However, TBE-CNN and HaarNet employ an ensemble of CNNs to achieve a higher recognition accuracy. Despite the elegant and complex design of TBE-CNN and HaarNet, the proposed system can achieve competitive performance with a simpler design and training methodology.

The presented result supports the claim that most of the existing still-to-video FR systems lack in using the knowledge embedded in the still ROIs. Thereby, the proposed system efficiently takes advantage of the still images in the gallery set to enhance the intra-class variations, as well as, to keep the inter-class variations between the subjects of interest. Moreover, the proposed face augmentation proved to be effective in reducing false negative rates by learning the appearances of the face of the subjects in the gallery set.

Figure 4.2 shows ROC curves for each camera of Cox face DB for the proposed system, as well as, the ROC curves for PSCL [31], LERM [32] and HaarNet [52]. As shown in Figure 4.2, the area under the ROC curve (AUC) of the proposed system is better than PSCL and LERM, while it is slightly lower than HaarNet.

For comparison on the ChokePoint dataset, the trained network on COX Face DB is adopted and operated on the ChokePoint dataset without any modifications. Thus, the network is fine-tuned using simulated video face ROIs augmented from the still images of the subjects of interest and then the same operation is performed. The results and comparison with EBM [8] and HaarNet [52] are presented in Table 4.2, where the area under precision-recall (AUPR) curve is computed. AUPR is used to measure the performance under the

imbalanced data circumstances, where this space is defined by precision and TPR as Recall, where precision is the ratio of true positives over the sum of true positives and false positives.

As presented in Table 4.2, the proposed system can compete to more sophisticated EBM and HaarNet systems. It is worth noting that, EBM implements a complex individual-specific ensemble of classifiers for each subject of interest using multiple face representation.

The final experiment is conducted similar to the protocol adopted by [50], where the training is performed on a separate dataset (in this paper, it was trained the network over COX Face DB) and evaluated on all of the still faces in the ChokePoint dataset. Therefore, all video ROIs are considered as probes and all still images are preserved in the gallery. In order to have a fair comparison, the results of the proposed system is included before and after fine-tuning stage. The rank-1 accuracy documented in [50] for still-to-video FR scenario is 62.7%, whereas it is raised up to 70.1% before fine-tuning. Moreover, by employing the same fine-tuning approach, the proposed network can achieve 85.9% rank-1 accuracy.

Video-based FR systems typically require real-time operations. To that end, the proposed system is designed to be simple, yet accurate while maintaining the real-time aspect of the design. In order to confirm the feasibility of utilizing the proposed network in video surveillance applications, the complexity in terms of number of parameters and operating time is compared with other CNN-based systems in Table 4.2.

Table 4.2 Average AUPR for videos of the Chokepoint along with the comparison of complexity (number of parameters and operations).

FR systems	**Accuracy**	**Complexity**	
	AUPR	**No. operations**	**No. parameters**
ESRC-DA [51]	76.97±0.07	228M	N/A
EBM [8]	99.24±0.38	2.3M	N/A
VGG-Face [54]	69.86±1.25	31.7B	1.8B
TBE-CNN [17]	N/A	12.8B	46.4M
HaarNet [52]	99.36±0.59	3.5B	13.1M
TLF-CNN	98.87±0.63	33.3M	24.5K

Table 4.2 compares the number of parameters and operating time on an Intel(R) Core(TM) i7-37700M (3.40GHz) PC along with a GEFORCE GTX 1070 8GB, where the proposed system offers a significantly lower complexity.

4.5 Conclusion

This paper presents an efficient CNN architecture for video-based FR by simulating a weighted cross-correlation matching specialized for the SSPP problem. In the proposed network, a cascade of convolutional layers is employed to effectively extract discriminative feature maps from the still and video ROIs. These complex and non-linear representations provide robustness for face matching under variations in viewpoints of the cameras and facial appearances of the subjects. In addition, a novel triplet-loss optimization is utilized to efficiently obtain optimum parameters of CCM-CNN. Furthermore, to overcome the SSPP constraints, transfer learning is applied in order to embed knowledge about the face stills located in the gallery set. More importantly, the complexity of the proposed system is significantly lower than other CNN-based FR systems and satisfies the real-time requirements of the video surveillance applications.

Chapter 5

Video-Based Single Sample Face Recognition Using Face Frontalization via Autoencoders Deep Neural Networks

Abstract

Real-world video-based face recognition (FR) is a challenging task, where video faces are typically captured with low-quality surveillance cameras under unconstrained conditions, such as variations in pose, illumination, expression, etc. Still-to-video FR is involved with matching facial region of interest (ROI) of a target individual isolated in a single high-quality still against video ROIs. This paper presents a deep learning-based system to restrain the severe impacts of differences between still and video ROIs. In particular, canonical face representation convolutional neural network (CFR-CNN) is proposed based on an autoencoder to reconstruct a frontal well-illuminated face ROI with neutral expression from a non-frontal and blurred given input face. Thus, this frontalization network is trained using a novel weighted loss function that can generate robust face embeddings similar to the same subjects. Then, the face embeddings belonging to the pairs of still and video ROIs are accurately matched using a fully-connected classification network. Experimental results obtained over challenging Cox Face DB and Chokepoint datasets indicate that the proposed CFR-CNN can achieve convincing performance. The results also confirm its effectiveness and efficiency to perform as an accurate and real-time system, where the number of operations, network parameters and layers are significantly lower than state-of-the-art FR systems.

5.1 Introduction

Video-based face recognition (FR) systems as acquired in real-world scenarios (e.g., airports, portal control, shopping malls, etc.) attempt to detect the presence of target persons. Such systems typically are required to perform accurate and real-time FR over a network of video surveillance cameras under unconstrained environments [8, 34]. In the applications of video-based FR, e.g. still-to-video FR, faces captured from low-resolution video cameras are compared with facial models of target persons created from a limited number of faces captured from a high-quality still camera under controlled conditions [8, 17]. Thus, the unavailability of sufficient reference still faces for generation of a discriminative facial model can affect the performance of still-to-video FR adversely [27, 52]. In addition, perturbation factors

observed in unconstrained environments manipulate the appearance of faces significantly, because of variations in pose, illumination, expression, occlusion and blur [50].

Typically, to design a representative facial model in real-world still-to-video FR, only a single sample per person (SSPP) is available during enrollment of a target individual [8]. In the FR literature, there are different approaches that address the SSPP problems including extracting multiple face representations, face synthesizing and using auxiliary data [8, 25, 31, 32]. These approaches are mainly based on augmenting the number of target samples to compensate the lack of different profile views and to enhance the intra-class variations in the gallery set. Despite of their achievements to cope with the SSPP constraints, yet FR systems suffer from the significant performance gap compared to the human visual system [31, 71].

To improve the performance of FR with SSPP, robust convolution features have been extracted in [2] by sampling and detecting facial points using CNNs integrated with a joint and collaborative sparse representation based classification (SRC). Nevertheless, several recent techniques address FR with SSPP from the perspective of domain adaptation (DA), where the gallery set as the source domain contains a single labeled training sample with stable shooting conditions, and the probe set as the target domain consists of unlabeled video ROIs with unstable shooting conditions [7, 27, 51]. For example, in [51], an extended SRC with DA (ESRC-DA) was proposed using a generic face dataset. Dynamic classifier selection through DA (DCS-DA) was carried out within a multi-classifier system in [7] using multiple face representation. Moreover, a deep DA network with generating synthetic pose-free faces using a 3D face model has been introduced in [27] to tackle the SSPP constraints. Thereby, to improve the performance of FR with SSPP, robust convolution features have been extracted in [2] by sampling and detecting facial points using CNNs integrated with a joint and collaborative sparse representation based classification (SRC).

Learning effective feature representations directly from face images through deep networks has recently provided a successful tool for robust FR [11, 17, 54, 52, 59]. In addition, producing pose- and illumination-invariant features have been extensively studied using deep networks by generating different face images [82]. For instance, a facial component-based CNN has been learned in [88] to transform faces with different poses and illuminations to canonical frontal view and well-illuminated faces, where pose-robust 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, while preserving the identity [82, 87]. Moreover, a general fully convolutional architecture was employed in [22] to encode a desired attribute and combine it with the input image to generate target images as similar as the input image with

an altered visual attribute (a different illumination, facial appearance or new pose) without changing other aspects of a face. However, the aforementioned methods lack generating a robust face embedding to be utilized in video-based FR applications.

Autoencoder is a commonly used building block in deep neural networks, where it contains encoder and decoder modules. The former module maps 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 using some deterministic mapping functions [19]. Inspired from Denoising Autoencoder [73], several autoencoder networks have been built to extract robust features to remove the variances in face images [19, 37, 39]. These networks consider faces with different types of variations (e.g., illumination, pose, etc.) as noisy images. For instance, stacked progressive autoencoders (SPAE) composed of multiple shallow autoencoders was proposed in [37] to learn pose-robust features by smoothly mapping faces to near frontal views. Moreover, a supervised autoencoder has 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 [19]. In contrast with standard autoencoders, this network was designed to extract similar features corresponding to the same persons to facilitate robust FR coupling with the conventional SRC in order to predict the labels of probe ROIs.

In this paper, a frontalization network based on autoencoder CNNs is proposed to deal with the SSPP problem, as well as, existing differences in the source and target domains of still-to-video FR. The network is trained using a novel loss function designed for SSPP, where its goal is to reconstruct a frontal well-illuminated faces with neutral expression. In addition, the intermediate layers of the frontalization network are designed to generate a discriminative face embedding similar for the same subjects and robust to variations observed in unconstrained real-world environments. A separate fully-connected network is also trained to perform face classification using the face embeddings extracted from the frontalization network, and determine whether the pairs of still and video ROIs are a matching pair or not. The proposed CFR-CNN is compared against state-of-the-art FR systems over the challenging Cox Face DB [31] and Chokepoint [79] datasets, where it can achieve comparable results, with significantly lower design and operation complexities.

5.2 Proposed network

The proposed CFR-CNN consists of two major components: Frontalization Network and Classification network. The frontalization network (see Figure 5.1) is responsible to reconstruct a frontal face with neutral-expression based on a single low-quality video face ROI, as

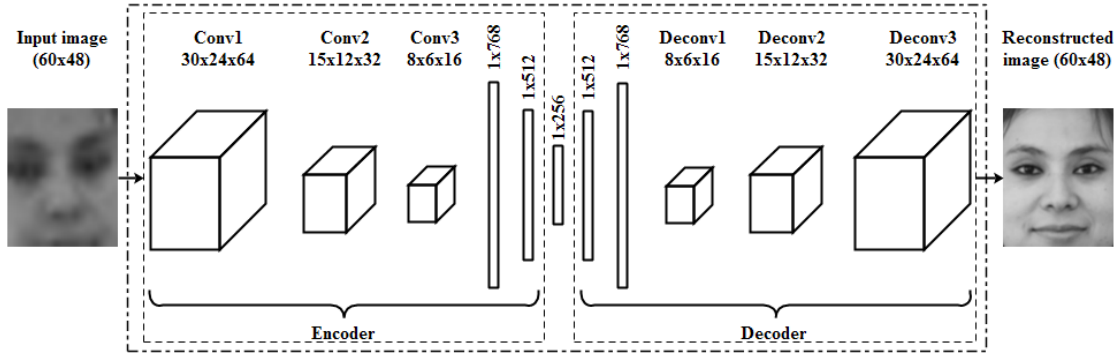


Fig. 5.1 The block diagram of the proposed frontalisation autoencoder network.

well as, to provide a discriminant face embedding. The classification network (see Figure 5.3) is designed to classify the face embeddings for a pair of (still, video) RIOs as matching or non-matching pairs. The capability of autoencoders as the frontalization network is to obtain a noise-free reconstruction for tackling view-point, pose and illumination variations. In order to eliminate such complexities and variations, we propose to employ autoencoders to produce a neutral frontal reconstruction of a face based on a single non-frontal low-quality video ROI. More specifically, fully-connected network is integrated with the convolutional autoencoder and the output of the intermediate layer is then utilized as a face embedding that is invariant to the different nuisance factors encountered in unconstrained surveillance environments.

The architecture of the proposed frontalization network is visualized in Figure 5.1, where the input image is a low-quality video ROI obtained from a surveillance camera and the output is a reconstructed neutral frontal image. This network consists of three convolutional layers each followed by a max-pooling layer to extract robust convolutional maps then a two layer fully-connected network 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 followed by un-pooling layers designed for generating the final reconstruction of the frontal face. In addition to the frontalization autoencoder network, the face matching is carried out by a fully-connected classification network as shown in Figure 5.3. This network is implemented to match the face representations of still and video ROIs.

5.2.1 Training frontalization network

In order to train the autoencoder to extract viewpoint and illumination invariant face embedding, a batch of video ROIs are fed into the network where the still images of the

corresponding persons are used as target reconstructions. Using still images that are captures under controlled conditions as target forces, the autoencoder network simultaneously learns frontalization and neutralization of faces. The parameters of this network are optimized by employing a novel weighted Mean Squared Error (MSE) criterion, where a T-shaped region suggested by [8] is considered as illustrated in Figure 5.2 to give higher significance to the face components like eyes, nose and mouth.

Fig. 5.2 T-shaped weight mask used for the proposed CFR-CNN loss function.

Thus, the weighted mean square loss function of the proposed CFR-CNN can be formulate as:

$$L_{CFR-CNN} = \sum_{i \in rows} \sum_{j \in cols} \tau_{i,j} \left\| X^2 - \hat{X}^2 \right\|$$
$$\tau_{i,j} = \begin{cases} \alpha & \text{if (i,j) belongs to T} \\ \beta & \text{if (i,j) otherwise} \end{cases} \tag{5.1}$$

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

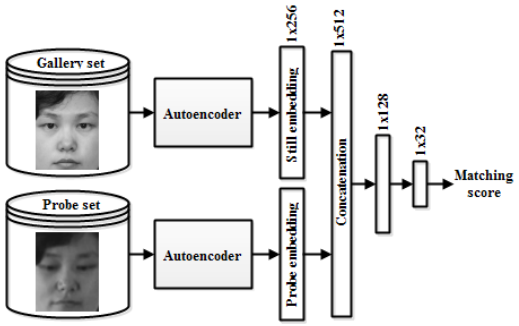


Fig. 5.3 The block diagram of the proposed classification network.

In order to appropriately train the frontalization network considering DA, COX Face DB is utilized which provides both low-quality video face ROIs from the source domain, as well as, the controlled high-quality still ground truths as the target domain for 1000 subjects.

Following the training protocol suggested by [31], 200 of the subjects are randomly sampled to train the frontalization network. The network is trained by feeding video ROIs as input and respective still ROIs as targets for 100 epochs using Adam optimization algorithm. The trained network not only is capable of reconstructing a high quality frontal image but also outputs a robust face embedding extracted from both ROIs. It thereby generates similar representations for the same identities. Figure 5.4 illustrates an example of 10 random video ROIs reconstructed by the frontalization network, where the odd rows visualizes the input video ROIs and the even rows present the reconstructed frontal images. As visualized in Figure 5.4, the network can successfully tackle the differences between the source and target domains and subsequently, generate a neutral frontal image for each given video ROI. While these reconstructed face ROIs might not be accurate, the face embeddings generated by the network can be utilized for robust FR task.

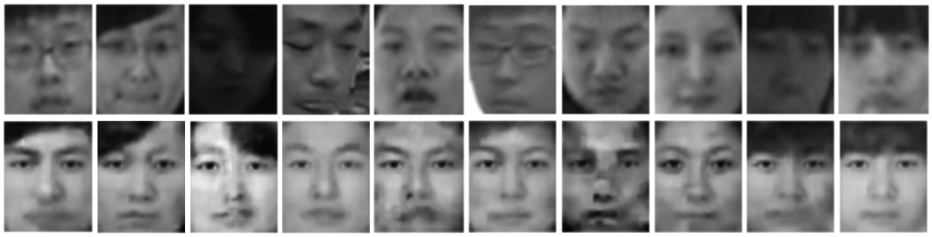


Fig. 5.4 Sample outputs of the frontalization network. The top row are the probe ROIs used as input and the bottom row are their corresponding reconstructed canonical faces.

5.2.2 Training Classification network

The fully-connected network is trained using a regular pairwise-matching scheme, where the face embeddings of still and video ROIs are fed into the network. The network then can learn to classify each pair of still and video ROIs as either matching or non-matching pairs. Consistent with [31], a randomly sampled set of 100 subjects is used to train the classification network and in total only 300 subjects are used for the training process. A training dataset is generated by pairing still and video ROIs and assigning a matching or non-matching labels to each pair. Furthermore, the frontalization network is applied to each face ROI to obtain a face embedding and each pair of face embeddings are fed to the classification network as input and the label as target. The network is trained for 20 epochs using Adam algorithm over roughly 10000 training samples by optimizing the cross-entropy criterion, where the network could achieve %89.01 accuracy over the validation dataset.

5.3 Experiments

5.3.1 Video datasets

Performance of the proposed system under the real-world still-to-video scenario is evaluated against the state-of-the-art systems using two challenging video-based FR datasets. Thus, Cox Face DB [31] and ChokePoint [79] are employed. These datasets are specifically constructed for video surveillance applications and are composed of high-quality still faces captured with still cameras under controlled conditions and low-quality video faces captured with off-th-shelf cam-coders under uncontrolled conditions. More specifically, Cox Face DB consists of one still and three video sequences of 1000 subjects captured from different viewpoints. Additionally, ChokePoint dataset is a benchmark for video surveillance application analysis under real-world scenarios. This dataset consists of video images of 25 subjects in portal one, and 29 subjects in portal 2 along with their corresponding controlled still images. In total, 64,204 face ROIs accurately extracted from 48 video sequences captured while subjects enter and leave the scene from these two portals.

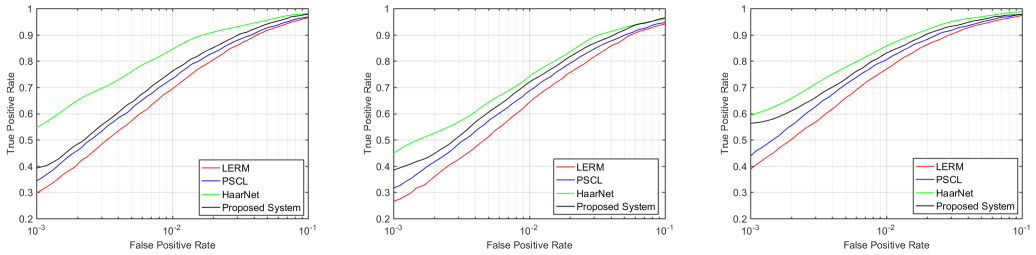
5.3.2 Experimental setup

Evaluation of the proposed system is performed by adopting experimental protocols suggested by [5], [31] and [50] on different datasets. For COX Face DB, a randomly set of 300 subjects are dedicated to training the autoencoder as well as the classifier. During evaluation, 700 subjects are utilized over a course of 10 iterations with random selection of training and testing subjects for each iteration. Thus, high-resolution still images from the 700 subjects are used as the gallery set and the probe set contains all face ROIs of the video clips from the corresponding 700 subjects. Thus, each probe is compared against all the gallery images and rank-1 recognition is reported as the accuracy of still-to-video FR system.

The same trained network is used without any further training for evaluation on Choke-Point where similar to [5], 5 subjects of interest are randomly selected and their still ROIs used as gallery images. On the other hand, all video ROIs of these subjects along with 10 unknown subjects appeared in the capturing scene are used as the probe set. This process is iterated 5 times, each time with random selection of the subjects of interest.

All images from both datasets re scaled to 60x48 pixels and the proposed system is implemented using Torch 7.0 deep learning framework [13].

Rank-1 recognition accuracy and ROC curve of the proposed network is compared against point-to-set correlation learning (PSCL) [31], learning Euclidean to Riemannian metric (LERM) [32], VGG-Face [54], Trunk-Branch Ensemble CNN (TBE-CNN) [17]



(a) Camera1 (b) Camera2 (c) Camera3

Fig. 5.5 ROC curves of the proposed method and baseline FR methods for videos of each camera in the Cox Face DB.

and HaarNet [52] on the COX Face DB, and also ensemble-based method (EBM) [8] and ESRC-DA [51] on the ChokePoint dataset. Additionally, the area under precision-recall curve (AUPR) is used to measure the performance under the imbalanced data circumstances on ChokePoint dataset where Precision-Recall space is defined by precision and true-positive-rate as Recall. Precision is the ratio of true positives over the sum of true positives and false positives.

5.3.3 Experimental results

Rank-1 accuracy of the proposed CFR-CNN is compared against the state-of-the-art FR systems over videos of Cox Face DB as shown in Table 5.1.

Table 5.1 Rank-1 accuracy of the proposed network against state-of-the-art FR systems on COX Face DB.

FR systems	**Camera 1**	**Camera 2**	**Camera 3**
PSCL [31]	36.39 ± 1.6	30.87 ± 1.8	50.96 ± 1.4
LERM [32]	49.07 ± 1.5	44.16 ± 0.9	63.83 ± 1.6
VGG-Face [54]	69.61 ± 1.5	68.11 ± 0.9	76.01 ± 0.7
CFR-CNN	85.32 ± 0.8	84.93 ± 1.2	91.52 ± 0.9
TBE-CNN [17]	88.24 ± 0.4	87.86 ± 0.8	95.74 ± 0.7
HaarNet [52]	89.31 ± 0.9	87.90 ± 0.6	97.01 ± 1.7

As observed in Table 5.1, PSCL, LERM and VGG-Face perform poorly, because they are not specifically designed for video-based FR. It is worth mentioning that, PSCL and LERM employed hand-crafted features as opposed to CNNs that can generate robust face representations. Amongst the CNN-based techniques that consider video-based FR, TBE-CNN and HaarNet provide higher performance. Although the proposed light-weight CFR-

CNN cannot outperform TBE-CNN and HaarNet, but it can achieve satisfactory accuracy with significantly lower complexity (see Table5.2).

The ROC curves for PSCL, LERM, HaarNet and CFR-CNN are depicted for each video of Cox Face DB in Figure 5.5, respectively. As demonstrated in Figure 5.5, CFR-CNN outperforms PSCL and LERM at transaction-level, while it can achieve comparable performance against HaarNet, specially over camera 2 and camera 3.

In addition, the proposed CFR-CNN is evaluated over videos of Chokepoint dataset according to AUPR values as presented in Table 5.2. More importantly, the complexity of the CFR-CNN in terms of number of operations, network parameters and layers is also compared in Table 5.2 against the state-of-the-art FR systems.

Table 5.2 Average AUPR performance for Chokepoint videos along with the comparison of complexity (number of operations, network parameters and layers).

FR systems	**Accuracy**	**Complexity**		
	AUPR	**No. operations**	**No. parameters**	**No. layers**
ESRC-DA [51]	76.97±0.07	228M	N/A	N.A.
EBM [8]	99.24±0.38	2.3M	N/A	N.A.
VGG-Face [54]	69.86±1.25	31.7B	1.8B	37
TBE-CNN [17]	N/A	12.8B	46.4M	144
HaarNet [52]	99.36±0.59	3.5B	13.1M	56
CFR-CNN	96.80±0.86	0.5M	2.5M	7

As shown in Table 5.2, CFR-CNN outperforms ESRC-DA and VGG-Face. Considering the elegant and complex design of EBM and HaarNet, CFR-CNN achieves slightly lower performance comparing with them. However, EBM was designed using an individual-specific ensemble of classifiers for each subject of interest and HaarNet is an ensemble of deep neural networks.

Since video-based FR systems in real-world scenarios are required to perform real-time, the number of operations to process a given probe is an important criterion. It can be seen in Table 5.2 that the proposed CFR-CNN needs significantly lower number of operations among other state-of-the-art FR systems. It confirms the feasibility of CFR-CNN to be operated on real-time with promising accuracy. Moreover, the number of network parameters and layers are also crucial factors in designing a deep CNN that can greatly affect the training time. Considering these criteria, the proposed CFR-CNN has the lowest design complexity and subsequently the shortest training time. In addition, a complex triplet-based loss function was employed to train TBE-CNN and HaarNet in order to learn a face embedding, where it aims to discriminate between the positive pair of two matching ROIs and the negative non-matching ROI.

Meanwhile, training data is typically limited in many video-based FR applications, where gathering sufficient training data to train a large network is costly and time consuming. TBE-CNN and HaarNet trained their networks on 2.6 and 1.3 million training samples, respectively, while the proposed CFR-CNN has been trained using only 136 thousands training samples.

5.4 Conclusion

This paper presents a deep learning-based solution for video-based FR by adopting an autoencoder to obtain a canonical face representation robust to existing variations in video surveillance unconstrained environments, such as changes in illumination, viewpoint, facial expression, etc. A novel frontalization autoencoder network is proposed to learn how to reconstruct a neutral frontal face from a low-quality video ROI and overcome the differences between the source and target domain in the context of DA. Furthermore, the intermediate face representations can be used as face embeddings to match the single still embedding against video face embeddings. The experimental results suggest that CFR-CNN is effective and highly efficient for video-based FR under the SSPP scenario. The results indicate that the proposed system is capable of learning a robust representation for face matching with significantly lower computational complexity. Despite the simple configuration and small dataset used for training, the proposed system can outperform most of the current state-of-the-art FR systems and is more suitable for real-time surveillance applications.

Chapter 6

Deep Feature Tracker: A Novel Application for Deep Convolutional Neural Networks

Abstract

Feature tracking is the building block of many applications such as visual odometry, augmented reality, and target tracking. Unfortunately, the state-of-the-art vision based tracking algorithms fail in surgical images due to the challenges imposed by the nature of such environments. In this paper, we proposed a novel and unified deep learning based approach that can learn how to track features reliably as well as learn how to detect such reliable features for the tracking purpose. The proposed network dubbed as Deep-PT, consists of a tracker network which is a convulutional neural network simulating cross correlation in terms of deep learning and two fully connected networks that operate on the output of intermediate layers of the tracker to detect features and predict trackability of the the detected points. The ability to detect features based on the capabilities of the tracker distinguishes the proposed method from previous algorithms used in this area and improves the robustness of the algorithms against dynamics of the scene. The network is trained using multiple datasets due to the lack of specialized dataset for feature tracking datasets and extensive comparisons are conducted to compare the accuracy of Deep-PT against recent pixel tracking algorithms. As the experiments suggest, the proposed deep architecture deliberately learns what to track and how to track and outperforms the state-of-the-art methods.

6.1 Introduction

Thanks to recent technological advances in robotic assisted surgery especially in minimally-invasive surgery (MIS), endoscopic cameras are nowadays widely used as a tool for diagnosis and cancer treatment procedures. During the MIS, the surgical instruments and the endoscope are inserted through tiny incisions and the surgery is performed remotely from a control console by utilizing video guidance provided by the endoscopic camera. Video-guided surgery has increased the need for translating the traditional computer vision algorithms for surgical vision environment and adapt them with unforeseen challenges available in such environments.

Compared to the traditional open-cavity surgery, in MIS the patients benefit from smaller incisions, less trauma, shorter hospitalization, less pain and more importantly lower infection risks [43]. Unfortunately, MIS poses major challenges for the surgeon who will experience a reduced awareness of the patient's anatomy due to narrow field of view of the endoscopic camera and lost depth perception [77]. As a consequence, the surgeon faces difficulty in locating and tracking critical anatomical structures such as blood vessels resulting in a higher risk of accidentally damaging an organ.

In this regard, computer-assisted navigation systems have been developed during the past years that promise to enhance the surgeon's perception of the environment by fusing the available pre-operative radiological data with the live endoscopic video. Detecting and tracking visual features in real-time is at the core of any such system to provide guidance and on-line decision-making assistance. Visual feature tracking finds a wide range of applications from target tracking [57, 44, 4] to tool tracking segmentation [9, 20], augmented reality [47], and deformation recovery [43].

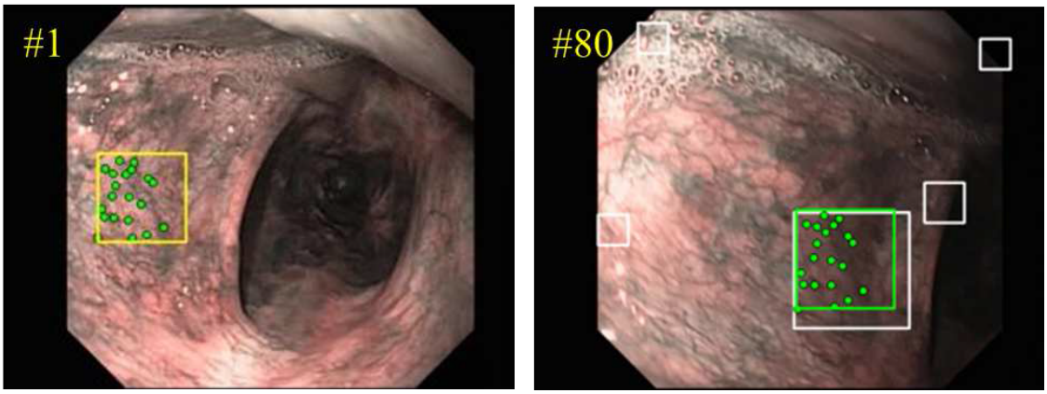


Fig. 6.1 Application of pixel based target tracking in biopsy. Left: The image where an optical biopsy site is selected. Right: The image where the site is tracked from previous frames using tracked keypoints. [81]

For example, Figure 6.1 shows a target tracking system where tracking of the area of interest is carried out by performing feature tracking on the surface of the organ. Tracking systems usually rely on an external feature detection system that detects a set of good features for tracking purpose.

As another example, Figure 6.2 shows a scenario where these tracked features can be used as anchor points for overlaying augmented reality on top of the image to give the surgeon a hint of depth perception. In this scenario the pre-operative radiological 3-D model is overlaid on top of the organ and the rendering is then updated by tracking the anchor points over time and aligning the 3-D model accordingly.

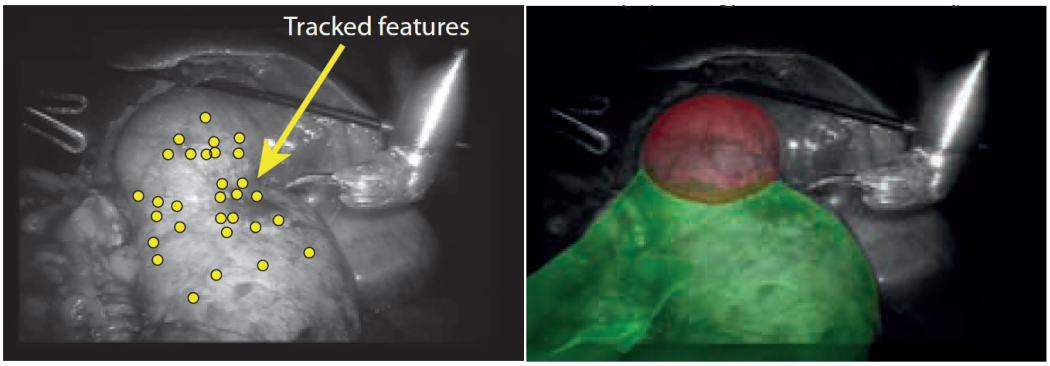


Fig. 6.2 Application of pixel based feature tracking in AR where the tracked pixels are used as anchor points to overlay a pre-operative 3-D model. Left: The tracked points visualized on the current frame. Right: The overlaid CT-scan model on top of the image. [56]

Despite recent efforts in adapted well-known feature detecting and tracking algorithms such as Kanade-Lucas-Tomasi (KLT) Tracker, most of the proposed prototypes [14, 49, 42, 58, 18] fail to provide a reliable and accurate long-term tracking under a surgical environment [43]. This is mostly due to the challenges posed by endoscopic imagery such as dynamic nature of the surgical environment, occlusions, sudden tissue deformations, specular highlights, image clutter caused by blood or smoke, and large texture-less areas [53]. As a result, off-the-shelf computer vision approaches simply fail when applied to the endoscopic images and usually require major revisions in order to make them applicable to such scenarios. Different approaches taken by scientist in order to address poor performance of of KLT includes exploiting Extended Kalman Filter(EKF) to utilize temporal information [18], on-line appearance learning and treating tracking as a classification problem [49], Thin Plate Spline (TPS) to track deforming surface [42], fusing intensity from stereo pair images for intensity matching [63], hierarchical feature matching [62].

Each of the aforementioned methods try to improve the accuracy of tracking by tackling the problem from a different perspective. However, the ultimate tracking system should be a self-contained framework that is able to overcome all shortcomings of the state-of-the-art methods. The goal of this thesis is to advance the reliability and robustness of surgical vision methods for endoscopic images by developing real-time algorithms to accurately detect and track reliable features under challenging and dynamic surgical environment.

6.2 Proposed Method

The main diagram for the invention is illustrated in Figure 6.3 and described in what follows. The framework has two major components. The first component, "Feature Detector" is responsible for detecting trackable features in the image. By trackable, we mean a feature than can be detected and recognized under small motion of camera and changes in the scene such as illumination. The second component, "Feature Tracker", takes the detected features and localized them in the next frame. In the first frame of the video sequence, the detector finds good features to track (initialization). During the tracking, if the number of the tracked features falls below a threshold (ε), then the "Feature Detector" is revoked to detect more features and add them to the list of tracked features (re-initialization).

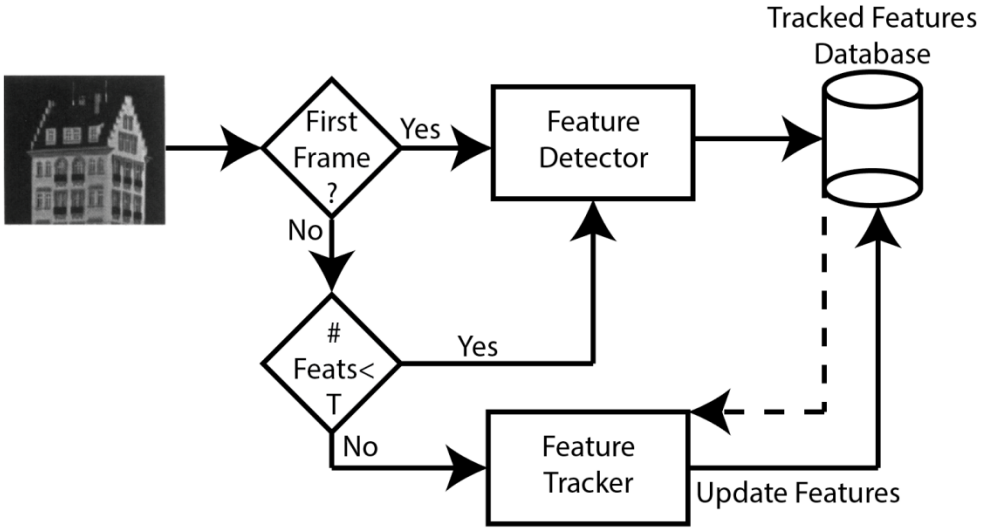


Fig. 6.3 Overall diagram for the Deep-PT. The method takes in input the live video from the single camera and detects and tracks features over time.

6.2.1 Feature Detection Network

The main diagram for the Feature Detector module is illustrated in Figure 6.4 and described below in detail.

The "Feature Detector" module uses a deep convolutional neural network to predict how good the given pixel is for the tracking purpose. It takes a patch around a pixel as input and spits out a trackability score. The 9 convolutional layers extract feature from the patch and a fully connected layer along with a softmax layer calculate a score for the given patch. This network sweeps through all the pixels of the image and evaluate each pixel location for

tracking. If the score is higher than a threshold, the location of the pixel (known as feature or or keypoint or interest point) is added to the database of features. One of the advantages of such feature detector is it's low computational burden as the convolutional layers are already applied to the image for tracking purpose. Moreover, if the feature detector is trained based on the capabilities of the tracker, then such unified tracking system can achieve higher accuracy and reliability.

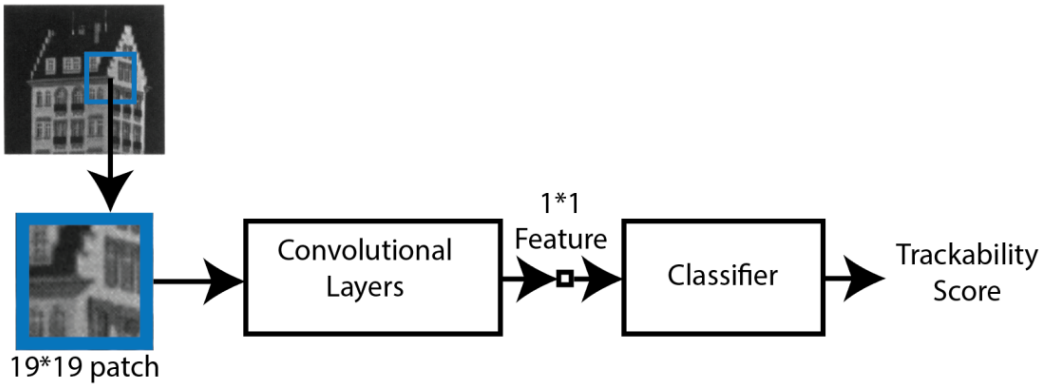


Fig. 6.4 Main diagram for the feature detection pipeline.

6.2.2 Feature Tracking Network

The main diagram for the Feature Detector module is illustrated in Figure 6.5 and described below in detail.

The "Feature Tracker" tracks each feature separately and does not consider any spatial correlation. It takes a small patch centered at the given location in the previous frame. Then, the same convolutional layers than the feature detector are applied to extract a representation for the patch. Also, a bigger patch (here 37*37) centered at the same position on the current frame will be passed through the same set of convolutional layers to extract the features. Once the deep representation of the patches are obtained, a matrix multiplication will join these two branches of the network and the location of the maximum in the resulted matrix determines the position of the feature in the current frame. The matrix multiplication resembles the traditional cross-correlation in patch-based matching. On the other hand, a similar fully connected network than the one in the feature detector is applied on the vectorized final matrix to determine tracking score of the feature. During the tracking, if this score is below a threshold for a specific feature, the same feature will not be tracked anymore. This may be caused by distortion, big change in viewpoint, or sudden deformation of the scene. Tracking

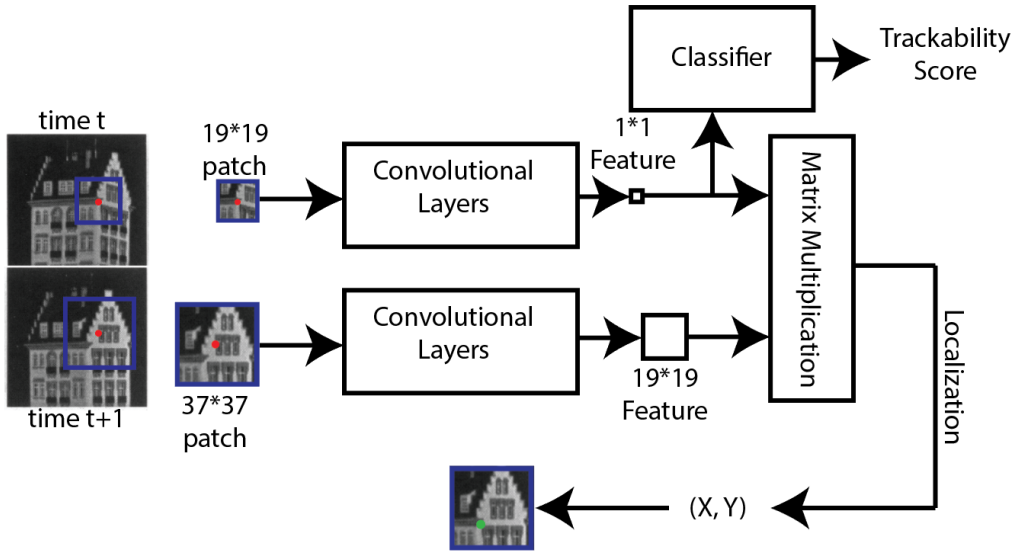


Fig. 6.5 Main diagram for the feature tracking pipeline.

score functionality allows the tracking framework to adapt itself with the dynamics of the scene and re-initialization gives it reliability to track more features once tracking is considered to be lost.

6.2.3 Training The Architecture

Training the proposed deep architecture requires a large dataset specific to feature tracking. Unfortunately, the lack of such training dataset that is specific to tracking made training even more difficult. Moreover, the network consists of multiple components that should be trained separately on a suitable dataset for each task. Therefore, the training is implemented in three stages: 1) training the tracker, 2) training the tracking score network, 3) training the feature detection network.

Training The Tracking Network

In order to train the tracking network, the tracking score network which is a fully connected is detached and the tracker is trained separately. For this purpose, We adopted the KITTI Flow 2012 dataset [21] which contains 389 pair of stereo images with ground truth suitable for stereo reconstruction and visual SLAM. The ground truth data provided by the dataset can be used to generate pairs of corresponding points for each pair of consecutive images.

In order to avoid training the network with texture-less areas, the training data is generated around Harris corners or SIFT keypoints with a small radius. This will ensure that the training

data does not contain any texture-less point such as sky or the road which may bias the tracker. Moreover, points that move more than 19 pixels from the previous frame are dismissed since those points don't satisfy our assumption that pixels don't move more than 19 pixels from a from the previous frame.

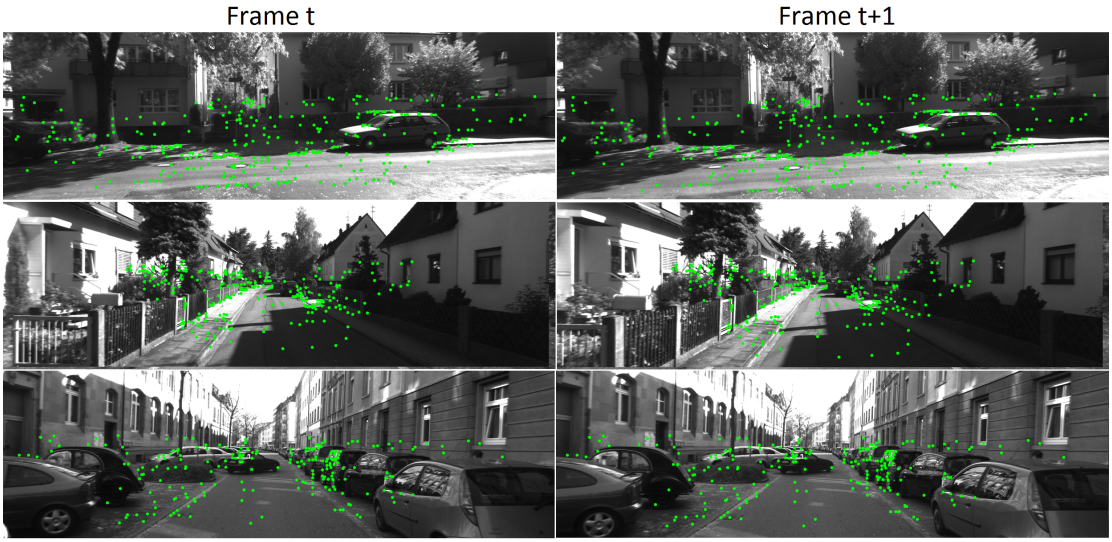


Fig. 6.6 Sample of training points generated for KITTTI Flow 2012 dataset. Each row presents a single pair of consecutive images with features marked with green dots.

The architecture is trained using an ad-hoc criterion where a 2-D Gaussian distribution with $\sigma = 3 pixels$ centered at the target position in the $frame t + 1$ to determine the loss. A small patch is extracted from the $frame t$ where the size of the patch is equal to the network's left branch receptive field. On the other hand, a bigger patch is extracted from the $frame t + 1$ where the patch extends to network's receptive size plus small window size of $37 pixels$ for the tracking and this patch is centered at the position of pixel in $frame t$. The inner-product layer of the network produces a score for each location in the patch taken from $frame t + 1$ and this allows us to compute a softmax for each pixel over all possible locations in that window. The parameters of the network are updated by minimizing cross entropy-loss with respect to the parameter set W give by:

$min_W \sum_i \sum_j P_{gt}(x_i, y_i) log P_i(x_i, y_i, W)$

Where $P_{gt}(x_i, y_i)$ is a 3×3 Gaussian filter centered around the ground truth and zero every where else to consider 3-pixel error metric. Also, $P_i(x_i, y_i, W)$ is the softmax probability distribution obtained by the forward pass using parameters W at position (i, j) in the window.

Roughly 100K points are used for training the tracker over the coarse of 200 epochs. Once the tracker is trained, it can be used to localize a feature in the next frame if the pixel

Table 6.1 Tracker's training parameters. Note that in addition to the learning rate decay, the learning rate is decreased by factor of 0.2 every 30 epochs after the epoch number 120.

Parameter	Learning Rate	Learning Rate Decay	Weight Decay	Momentum
Values	1e-2	1e-7	1e-4	0.9

moves in the 37×37 region. Table 6.1 tabulates the main training parameters used to train the tracker network with Adam algorithm.

Fig. 6.6 visualizes several images from the KITTI Flow 2012 dataset along with the generated ground truth points. The first column is a cropped region of the original image in the dataset with the location of each keypoint. The second column visualizes the next cropped frame with the same corresponding keypoints in the current frame.

Training The Tracking Score Network

Most applications that rely on tracking pixels, require a tracking score to detect when the tracking is lost or a specific feature is not reliable. In order to obtain such information from the tracker, a fully connected network is attached to the output of the matrix dot product layer that will generate a matching score for the two patches. In order to train this network, we adopted the UBC patch dataset [UBC] which is originally collected for local descriptor learning [78] by Winder et al. Fig. 6.7 visualizes some challenging images from this dataset where each patch is followed by several patches that represent a single 3-D point captured from different viewpoints.

Fig. 6.7 UBC Patches dataset [23] contains several viewpoints of each 3D point and is challenging due to different levels of rotation, translation and scale.

In order to be able to compare the network with state-of-the-art methods, the training and testing protocols suggested by [23] are applied. It's worth mentioning that the parameters of

the convolutional layers are not updated during training the tracking score network to make sure the accuracy of the tracker is not deteriorated. Table 6.2 tabulates the main training parameters used to train the tracking score network with Adam algorithm.

Table 6.2 Score Network's training parameters. Note that in addition to the learning rate decay, the learning rate is decreased by factor of 0.1 every 30 epochs after the epoch number 120.

Parameter	Learning Rate	Learning Rate Decay	Weight Decay	Momentum
Values	1e-3	1e-7	1e-5	0.85

Training Feature Detector

Recently, deep convolutional neural networks have shown significant improvement over the state-of-the-art interest point detectors especially for detecting facial keypoints [66]. In this paper, we propose to use a deep architecture for on-line keypoint detection. The proposed Network dubbed as "Feature Detector" uses the output of the left branch of the Feature Tracking Network to detect reliable features to track. Therefore, an additional fully connected network is attached to the output of the left branch of the network in order to classify each pixel as a keypoint or non-keypoint. Similar to the second stage of the training, during this stage of training, the parameters of the tracker are not updated as well.

Concerning training the network, we generated a train dataset by running roughly 100K points from KITTI Flow 2015 dataset [48] through the feature tracking network to obtain the ground truth labels for each pixel. To that end, pixels that were tracked correctly by the tracker are labeled as positive and otherwise negative and a balanced subset of these points are used to train the feature detection network. This ensures that the feature detector learns the behavior of the tracker on each point and can predict whether it will be reliably tracked or not. Such feature detection architecture can be used either to initialize the tracker or to re-initialize points in case the tracking is lost. The focus of this paper is mainly the feature tracker, thus, a comprehensive comparison of the proposed feature detector network against the state-of-the-art interest point detectors will be presented in another paper.

6.3 Experimental Results

This section presents extensive evaluation of the proposed unified feature detection and tracking framework. Different aspects of the proposed deep architecture is evaluated using

challenging datasets such as KITTI FLOW 2015 [48], MIS dataset [55], and UBC Patch dataset [UBC]. The KITTI FLOW 2015 dataset is used to evaluate the tracking capabilities of the dataset under a real-world scenario for autonomous driving. The MIS dataset provides a more challenges mainly encountered in surgical vision such as large texture-less areas, specular highlights, large deformations, close distance to the scene, motion blur, blood, and smoke [53]. On the other hand, the UBC patches dataset is employed to evaluate the feature tracker under a different application where the tracker is used to perform feature matching. Deep-PT is mainly compared against a modified version of the KLT-Tracker which is a widely used method for tracking in computer vision applications such as [35, 61, 41].

6.3.1 Evaluation on KITTI Flow 2015

The performance of the tracker is evaluated using KITTI Flow 2015 dataset over 30K points obtained by the following protocol. The KITTI dataset provides a semi-dense ground truth flow information for each pair of consecutive images. This ground truth data is used to generate roughly 30K pairs of corresponding points extracted around Harris corners and SIFT interest points in consecutive image pairs. Concerning comparison metrics, the tracking is compared by 1-pixel, 3-pixel, and 5-pixel accuracy where i-pixel accuracy means the ratio of correctly tracked pixels within "i" pixels of error over all pixel used for tracking.

The evaluation is performed by running the tracker specifically on these 30K points with the given ground truth and the results are compared against the most recent implementation of the KLT-Tracked algorithm with forward-backward error [36]. The forward-backward error ensures more reliable feature tracking by adopting a pyramidal approach for tracking both forward and backward in time. The points with high discrepancy in forward and backward tracking are marked as unreliable. Table. **??** tabulates the accuracy of the proposed tracker compared against forward-backward error KLT-tracker. The results presented in Table. **??** suggest a strong improvement over the state-of-the-art feature tracking methods.

Table 6.3 X-pixel tracking accuracy of Deep-PT and forward-backward KLT tracker in percentage.

Metric	1-pixel	2pixel	3-pixel
Deep-PT	%78.22	%88.78	%90.42
KLT	%53.93	%65.48	%70.61

Fig. 6.8 visualizes an example of tracking performed by our proposed method versus the KLT-tracker. In this figure, only a cropped region of the image is presented for convenience and green represents successful tracking of a point and red represents failure in tracking.

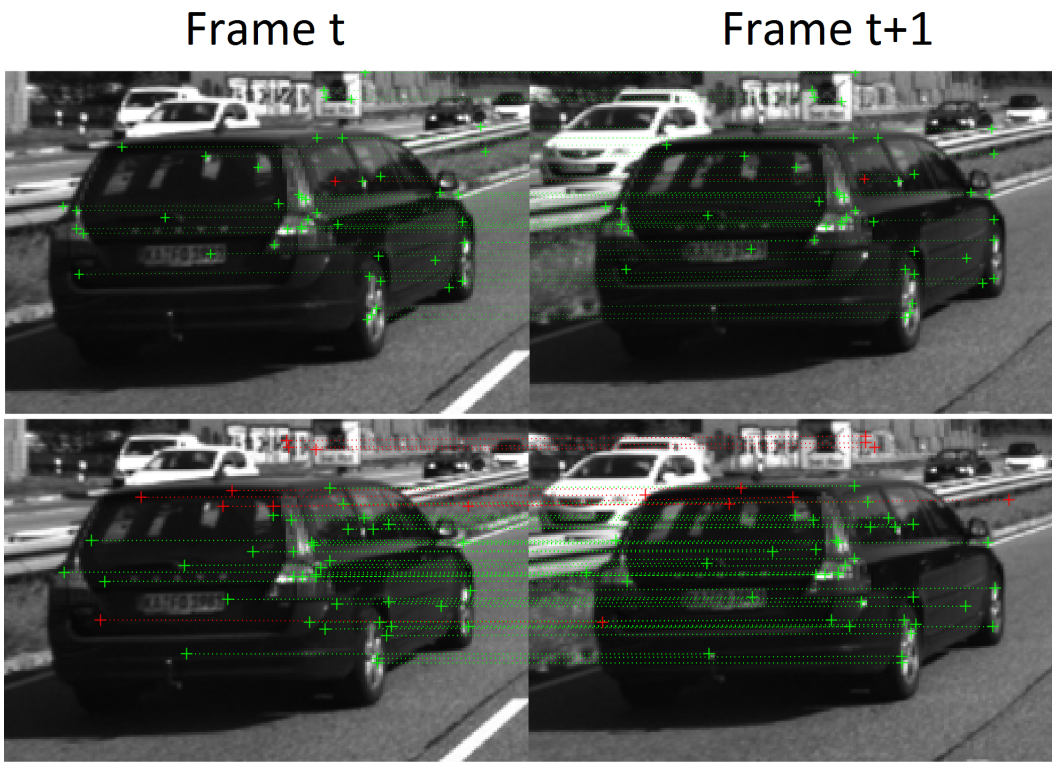


Fig. 6.8 Qualitative comparison of Deep-PT Vs. forward-backward KLT-tracker where the lines show correspondences. Top row: visualization of the tracking performed by the Deep-PT over a cropped region of an image from KITTI Flow dataset. Bottom row: visualization of the tracking performed on the same image by the KLT-tracker

The mis-tracked features detected by the tracking score network are not visualized here. As shown in Fig. 6.8, Deep-PT outperforms KLT tracker in effectively localizing features in the next frame. More specifically, the proposed method performs well on generic features and does not rely only on corner to predict the motion of a pixel. A closer look at Fig. 6.8 reveals that the only mis-tracked point in the first row is actually tracked correctly in that local area considering the shadow on the car moves backwards.

6.3.2 Evaluation on MIS dataset

While the KITTI Flow 2015 dataset provides a great ground truth data for our tracking purpose, it has limited types of motion and challenges. Thus, we propose to perform an experiment under a Minimally Invasive Surgical environment where the images are captured using an endoscopic camera of the da Vinci surgical platform [55]. Such dataset imposes more challenges, however, it lacks ground truth data fro tracking.

Table 6.4 Pixel back-projection error and inlier percentage for the MIS dataset.

	Average Error	% Inlier
Lowe's	4.66 ± 4.24	%34
AMA	2.49 ± 2.42	%40
Cho	3.56 ± 3.35	%39
HMA	2.84 ± 2.64	%39
Deep-PT	2.71 ± 2.81	%82

The quantitative evaluation the MIS dataset is performed by following the same protocol provided by [55]. For this purpose, the methods are compared using a back-projection error metric where the points in the current frame are back-projected to the previous frame using homography and the euclidean distance between the corresponding points is considered as error measure. Homography matrices are computed by considering the same planar patches obtained by [56]. Table 6.4 presents the back-projection error for the proposed method, Hierarchical Multi-Affine (HMA) [56] feature matching, Lowe's [45], Adaptive Multi-Affine (AMA) [62] and Cho [12]. As Table 6.4 suggests, Deep-PT provides more inlier points with a higher accuracy than the state-of-the-art methods in surgical environment.

Fig. 6.9 presents a pair of images from the MIS dataset where the feature points are visualized on each image. In Fig. 6.9, the correctly tracked features are visualized in green whereas the mis-tracked features that were not detected by the tracking score network are visualized in red. As suggested by Fig. 6.9, the proposed method performs better in such texture-less environments than the KLT-tracker.

6.3.3 Evaluation on UBC Patches dataset

So far the tracking capabilities of the proposed Deep-PT is evaluated and in this section we tend to evaluate the patch-matching competence of the proposed method against the state-of-the-art deep learning based methods. To that end, the UBC Patches dataset is employed to compare small patches. The trained feature score matching network is responsible to generate a matching score between two given patches.

In order to compare different algorithms fairly, we followed the protocol suggested by [23] and the error rate at %95 recall is reported in percentage. Table. 6.5 tabulates the comparison of the proposed method against MatchNet and other recent local descriptor learning algorithms. Considering that Deep-PT is not trained specifically to classify patches to matching and non-matching categories, the performance of the network is satisfactory. Additionally, Deep-PT utilizes only a small patch inside the 64×64 patches from the dataset

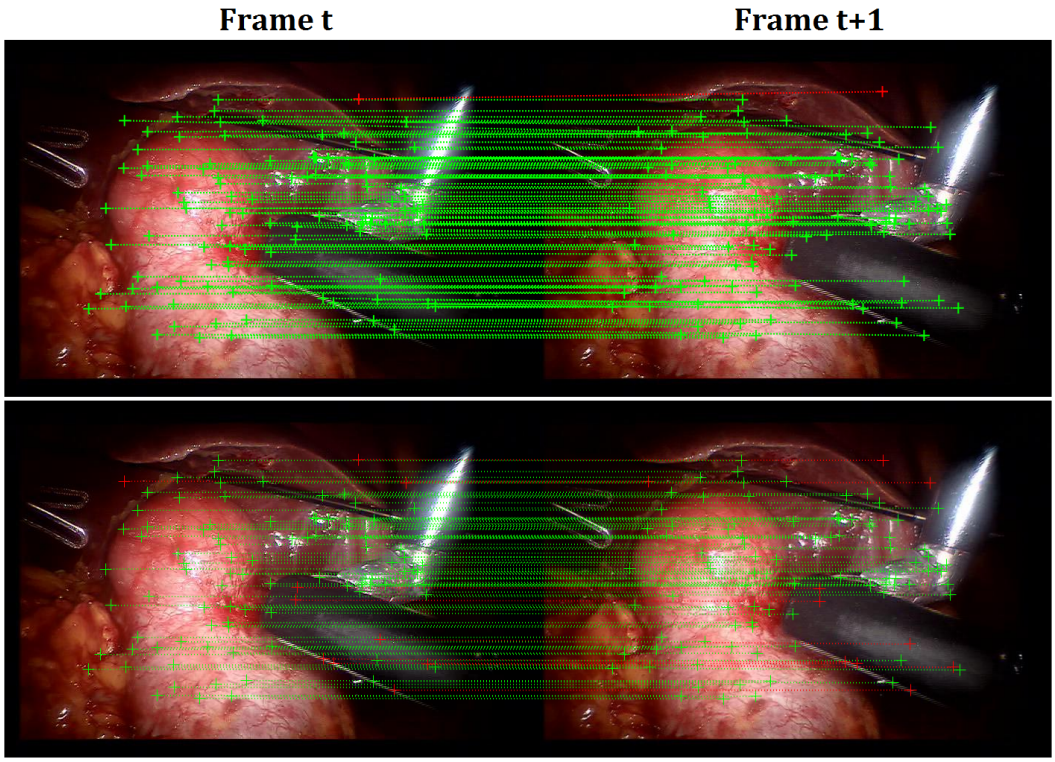


Fig. 6.9 Qualitative comparison of Deep-PT Vs. forward-backward KLT-tracker where the lines show correspondences. Top row: visualization of the tracking performed by the Deep-PT over a pair of consecutive frames from the MIS dataset. Bottom row: visualization of the tracking performed on the same images by the KLT-tracker

and training the network with the whole patches would noticeably increase the accuracy of matching.

Table 6.5 UBC matching results. Numbers are Error at %95 recall in percentage.

Training	Notredame	Liberty
Testing	Liberty	Notredame
Baseline: nSift+NNet [23]	%20.44	%14.35
Trzcinski et al [72]	%18.05	%14.15
Brown et al [10]	%16.85	*N.A.*
Simonyan et al [60]	%16.56	%9.88
MatchNet [23]	%9.82	%5.02
Deep-PT	%15.99	%12.79

6.4 Conclusion

This paper presented a novel unified deep learning based pixel tracking framework capable of detecting good features to track and re-initialize new features in case of failure in tracking. In that regard, Deep-PT intuitively simulates cross-correlation in deep learning to localize a pixel in the next time frame. The ability to detect features that are more suitable for the trained tracker differentiates the proposed methods from the state-of-the-art methods. Moreover, the results on KITTI Flow 2015 and MIS dataset suggests that in a real-world scenario, Deep-PT outperforms existing methods and can be generalized to any type of environment such as outdoors and surgical images. Additionally, extensive comparisons on UBC Patch dataset against patch-matching algorithms suggests that the network can be generalized to similar problems. Deep-PT is a reliable method for tracking features based on a learning method which enables it to track a variety of reliable types of features more accurately.

The proposed method is not perfect and has defects. More specifically Deep-PT fails in environments with highly repetitive texture patterns as suggested by experiments. The next step is to train the feature detection network to avoid such pitfalls. Moreover, a more extensive comparison of the feature detector with the state-of-the-art interest point detection algorithms will be performed. Additionally, a study on long-term tracking capabilities of the system should be explored and addressed in the future studies.

References

[UBC] Ubc phototour patches dataset. http://www.cs.ubc.ca/~mbrown/patchdata/patchdata.html. Accessed: 2017-03-22.

[2] (2017). Joint and collaborative representation with local adaptive convolution feature for face recognition with single sample per person. *Pattern Recognition*, 66:117 – 128.

[3] Ahmed, E., Jones, M., and Marks, T. K. (2015). An improved deep learning architecture for person re-identification. In *CVPR*.

[4] Amber, A., Iwahori, Y., Bhuyan, M., Woodham, R. J., and Kasugai, K. (2015). Feature point based polyp tracking in endoscopic videos. In *Applied Computing and Information Technology/2nd International Conference on Computational Science and Intelligence (ACIT-CSI), 2015 3rd International Conference on*, pages 299–304. IEEE.

[5] Bashbaghi, S., Granger, E., Sabourin, R., and Bilodeau, G. A. (2014). Watch-list screening using ensembles based on multiple face representations. In *ICPR*.

[6] Bashbaghi, S., Granger, E., Sabourin, R., and Bilodeau, G. A. (2015). Ensembles of exemplar-svms for video face recognition from a single sample per person. In *AVSS*.

[7] Bashbaghi, S., Granger, E., Sabourin, R., and Bilodeau, G.-A. (2017a). Dynamic selection of exemplar-svms for watch-list screening through domain adaptation. In *ICPRAM*.

[8] Bashbaghi, S., Granger, E., Sabourin, R., and Bilodeau, G.-A. (2017b). Robust watch-list screening using dynamic ensembles of svms based on multiple face representations. *Machine Vision and Applications*, 28(1):219–241.

[9] Bell, C. S., Obstein, K. L., and Valdastri, P. (2013). Image partitioning and illumination in image-based pose detection for teleoperated flexible endoscopes. *Artificial intelligence in medicine*, 59(3):185–196.

[10] Brown, M., Hua, G., and Winder, S. (2011). Discriminative learning of local image descriptors. *IEEE transactions on pattern analysis and machine intelligence*, 33(1):43–57.

[11] Chellappa, R., Chen, J., Ranjan, R., Sankaranarayanan, S., Kumar, A., Patel, V. M., and Castillo, C. D. (2016). Towards the design of an end-to-end automated system for image and video-based recognition. *CoRR*, abs/1601.07883.

[12] Cho, M., Lee, J., and Lee, K. M. (2009). Feature correspondence and deformable object matching via agglomerative correspondence clustering. In *Computer vision, 2009 IEEE 12th international conference on*, pages 1280–1287. IEEE.

[13] Collobert, R., Kavukcuoglu, K., and Farabet, C. (2011). Torch7: A matlab-like environment for machine learning. In *NIPS Workshops*.

[14] Davison, A. J. (2003). Real-time simultaneous localisation and mapping with a single camera. In *ICCV*, volume 3, pages 1403–1410.

[15] De-La-Torre, M., Granger, E., Radtke, P. V., Sabourin, R., and Gorodnichy, D. O. (2015). Partially-supervised learning from facial trajectories for face recognition in video surveillance. *Information Fusion*, 24:31 – 53.

[16] Deng, W., Hu, J., and Guo, J. (2012). Extended src: Undersampled face recognition via intraclass variant dictionary. *IEEE Trans on PAMI*, 34(9):1864–1870.

[17] Ding, C. and Tao, D. (2016). Trunk-branch ensemble convolutional neural networks for video-based face recognition. *CoRR*, abs/1607.05427.

[18] Figl, M., Rueckert, D., Hawkes, D., Casula, R., Hu, M., Pedro, O., Zhang, D. P., Penney, G., Bello, F., and Edwards, P. (2010). Image guidance for robotic minimally invasive coronary artery bypass. *Computerized Medical Imaging and Graphics*, 34(1):61–68.

[19] Gao, S., Zhang, Y., Jia, K., Lu, J., and Zhang, Y. (2015). Single sample face recognition via learning deep supervised autoencoders. *IEEE Transactions on Information Forensics and Security*, 10(10):2108–2118.

[20] Garcia Peraza Herrera, L., Li, W., Gruijthuijsen, C., Devreker, A., Attilakos, G., Deprest, J., Vander Poorten, E., Stoyanov, D., Vercauteren, T., and Ourselin, S. (2016). Real-time segmentation of non-rigid surgical tools based on deep learning and tracking. In *Lecture Notes in Computer Science*. Springer Verlag (Germany).

[21] Geiger, A., Lenz, P., and Urtasun, R. (2012). Are we ready for autonomous driving? the kitti vision benchmark suite. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3354–3361. IEEE.

[22] Ghodrati, A., Jia, X., Pedersoli, M., and Tuytelaars, T. (2016). Towards automatic image editing: Learning to see another you. In *BMVC*.

[23] Han, X., Leung, T., Jia, Y., Sukthankar, R., and Berg, A. C. (2015a). Matchnet: Unifying feature and metric learning for patch-based matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3279–3286.

[24] Han, X., Leung, T., Jia, Y., Sukthankar, R., and Berg, A. C. (2015b). Matchnet: Unifying feature and metric learning for patch-based matching. In *CVPR*.

[25] Hassner, T., Harel, S., Paz, E., and Enbar, R. (2015). Effective face frontalization in unconstrained images. In *CVPR*.

[26] Heo, Y. S., Lee, K. M., and Lee, S. U. (2011). Robust stereo matching using adaptive normalized cross-correlation. *IEEE Trans on PAMI*, 33(4):807–822.

[27] Hong, S., Im, W., Ryu, J., and Yang, H. S. (2017). Sspp-dan: Deep domain adaptation network for face recognition with single sample per person. *arXiv preprint arXiv:1702.04069*.

[28] Huang, G. B., Lee, H., and Learned-Miller, E. (2012). Learning hierarchical representations for face verification with convolutional deep belief networks. In *CVPR*.

[29] Huang, G. B., Ramesh, M., Berg, T., and Learned-Miller, E. (2007a). Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical report, Technical Report 07-49, University of Massachusetts, Amherst.

[30] Huang, G. B., Ramesh, M., Berg, T., and Learned-Miller, E. (2007b). Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49.

[31] Huang, Z., Shan, S., Wang, R., Zhang, H., Lao, S., Kuerban, A., and Chen, X. (2015). A benchmark and comparative study of video-based face recognition on cox face database. *IEEE Trans on Image Processing*, 24(12):5967–5981.

[32] Huang, Z., Wang, R., Shan, S., and Chen, X. (2014). Learning euclidean-to-riemannian metric for point-to-set classification. In *CVPR*.

[33] Jain, A. K., Nandakumar, K., and Ross, A. (2016a). 50 years of biometric research: Accomplishments, challenges, and opportunities. *Pattern Recognition Letters*, 79:80–105.

[34] Jain, A. K., Nandakumar, K., and Ross, A. (2016b). 50 years of biometric research: Accomplishments, challenges, and opportunities. *Pattern Recognition Letters*, 79:80 – 105.

[35] Ji, P., Li, H., Salzmann, M., and Zhong, Y. (2016). Robust multi-body feature tracker: a segmentation-free approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3843–3851.

[36] Kalal, Z., Mikolajczyk, K., and Matas, J. (2010). Forward-backward error: Automatic detection of tracking failures. In *Pattern recognition (ICPR), 2010 20th international conference on*, pages 2756–2759. IEEE.

[37] Kan, M., Shan, S., Chang, H., and Chen, X. (2014). Stacked progressive auto-encoders (spae) for face recognition across poses. In *CVPR*.

[38] Kim, M., Kumar, S., Pavlovic, V., and Rowley, H. (2008). Face tracking and recognition with visual constraints in real-world videos. In *CVPR*.

[39] Le, Q. V. (2013). Building high-level features using large scale unsupervised learning. In *ICASSP*.

[40] Li, W., Zhao, R., Xiao, T., and Wang, X. (2014). Deepreid: Deep filter pairing neural network for person re-identification. In *CVPR*.

[41] Lim, A., Ramesh, B., Yang, Y., Xiang, C., Gao, Z., and Lin, F. (2017). Real-time optical flow-based video stabilization for unmanned aerial vehicles. *arXiv preprint arXiv:1701.03572*.

[42] Lim, J. and Yang, M.-H. (2005). A direct method for modeling non-rigid motion with thin plate spline. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 1196–1202. IEEE.

[43] Lin, B., Sun, Y., Qian, X., Goldgof, D., Gitlin, R., and You, Y. (2015). Video-based 3d reconstruction, laparoscope localization and deformation recovery for abdominal minimally invasive surgery: a survey. *The International Journal of Medical Robotics and Computer Assisted Surgery*.

[44] Liu, J., Subramanian, K. R., and Yoo, T. S. (2013). An optical flow approach to tracking colonoscopy video. *Computerized Medical Imaging and Graphics*, 37(3):207–223.

[45] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110.

[46] Luo, W., Schwing, A. G., and Urtasun, R. (2016). Efficient deep learning for stereo matching. In *CVPR*.

[47] Marques, B., Plantefève, R., Roy, F., Haouchine, N., Jeanvoine, E., Peterlik, I., and Cotin, S. (2015). Framework for augmented reality in minimally invasive laparoscopic surgery. In *E-health Networking, Application & Services (HealthCom), 2015 17th International Conference on*, pages 22–27. IEEE.

[48] Menze, M. and Geiger, A. (2015). Object scene flow for autonomous vehicles. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3061–3070.

[49] Mountney, P. and Yang, G.-Z. (2008). Soft tissue tracking for minimally invasive surgery: Learning local deformation online. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 364–372. Springer.

[50] Mudunuri, S. P. and Biswas, S. (2016). Low resolution face recognition across variations in pose and illumination. *IEEE Trans on PAMI*, 38(5):1034–1040.

[51] Nourbakhsh, F., Granger, E., and Fumera, G. (2016). An extended sparse classification framework for domain adaptation in video surveillance. In *ACCV, Workshop on Human Identification for Surveillance*.

[52] Parchami, M., Bashbaghi, S., and Granger, E. (2017). Video-based face recognition using ensemble of haar-like deep convolutional neural networks. In *IJCNN*.

[53] Parchami, M., Cadeddu, J. A., and Mariottini, G.-L. (2014). Endoscopic stereo reconstruction: A comparative study. In *Engineering in Medicine and Biology Society (EMBC), 2014 36th Annual International Conference of the IEEE*, pages 2440–2443. IEEE.

[54] Parkhi, O. M., Vedaldi, A., and Zisserman, A. (2015). Deep face recognition. In *BMVC*.

[55] Puerto-Souza, G. A., Cadeddu, J. A., and Mariottini, G.-L. (2014). Toward long-term and accurate augmented-reality for monocular endoscopic videos. *IEEE Transactions on Biomedical Engineering*, 61(10):2609–2620.

[56] Puerto-Souza, G. A. and Mariottini, G. L. (2012). Hierarchical multi-affine (hma) algorithm for fast and accurate feature matching in minimally-invasive surgical images. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 2007–2012. IEEE.

[57] Pullens, H. J., Schwartz, M. P., Broeders, I., and van der Heijden, F. (2016). A real-time target tracking algorithm for a robotic flexible endoscopy platform. In *Computer-Assisted and Robotic Endoscopy: Second International Workshop, CARE 2015, Held in Conjunction with MICCAI 2015, Munich, Germany, October 5, 2015, Revised Selected Papers*, volume 9515, page 81. Springer.

[58] Richa, R., Poignet, P., and Liu, C. (2008). Efficient 3d tracking for motion compensation in beating heart surgery. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 684–691. Springer.

[59] Schroff, F., Kalenichenko, D., and Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. In *CVPR*.

[60] Simonyan, K., Vedaldi, A., and Zisserman, A. (2014). Learning local feature descriptors using convex optimisation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(8):1573–1585.

[61] Singha, J., Semwal, V. B., and Laskar, R. H. (2016). An accurate hand tracking system for complex background based on modified klt tracker. In *Region 10 Conference (TENCON), 2016 IEEE*, pages 3644–3647. IEEE.

[62] Souza, G. A. P., Adibi, M., Cadeddu, J. A., and Mariottini, G. L. (2011). Adaptive multi-affine (ama) feature-matching algorithm and its application to minimally-invasive surgery images. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 2371–2376. IEEE.

[63] Stoyanov, D., Mylonas, G. P., Deligianni, F., Darzi, A., and Yang, G. Z. (2005). Soft-tissue motion tracking and structure estimation for robotic assisted mis procedures. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 139–146. Springer.

[64] Sun, Y., Chen, Y., Wang, X., and Tang, X. (2014a). Deep learning face representation by joint identification-verification. In *Advances in neural information processing systems*, pages 1988–1996.

[65] Sun, Y., Chen, Y., Wang, X., and Tang, X. (2014b). Deep learning face representation by joint identification-verification. In *NIPS*.

[66] Sun, Y., Wang, X., and Tang, X. (2013a). Deep convolutional network cascade for facial point detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3476–3483.

[67] Sun, Y., Wang, X., and Tang, X. (2013b). Hybrid deep learning for face verification. In *ICCV*.

[68] Sun, Y., Wang, X., and Tang, X. (2014c). Deep learning face representation from predicting 10,000 classes. In *CVPR*.

[69] Sun, Y., Wang, X., and Tang, X. (2015). Deeply learned face representations are sparse, selective, and robust. In *CVPR*.

[70] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *CVPR*.

[71] Taigman, Y., Yang, M., Ranzato, M., and Wolf, L. (2014). Deepface: Closing the gap to human-level performance in face verification. In *CVPR*.

[72] Trzcinski, T., Christoudias, M., Lepetit, V., and Fua, P. (2012). Learning image descriptors with the boosting-trick. In *Advances in neural information processing systems*, pages 269–277.

[73] Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., and Manzagol, P.-A. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *JMLR*, 11:3371–3408.

[74] Viola, P. and Jones, M. J. (2004). Robust real-time face detection. *International Journal of Computer Vision*, 57(2).

[75] Wang, J., Song, Y., Leung, T., Rosenberg, C., Wang, J., Philbin, J., Chen, B., and Wu, Y. (2014). Learning fine-grained image similarity with deep ranking. In *CVPR*.

[76] Weinzaepfel, P., Revaud, J., Harchaoui, Z., and Schmid, C. (2013). Deepflow: Large displacement optical flow with deep matching. In *ICCV*.

[77] Wieringa, F. P., Bouma, H., Eendebak, P. T., van Basten, J.-P. A., Beerlage, H. P., Smits, G. A., and Bos, J. E. (2014). Improved depth perception with three-dimensional auxiliary display and computer generated three-dimensional panoramic overviews in robot-assisted laparoscopy. *Journal of Medical Imaging*, 1(1):015001–015001.

[78] Winder, S. A. and Brown, M. (2007). Learning local image descriptors. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE.

[79] Wong, Y., Chen, S., Mau, S., Sanderson, C., and Lovell, B. C. (2011). Patch-based probabilistic image quality assessment for face selection and improved video-based face recognition. In *CVPR WORKSHOPS*.

[80] Xu, H., Zheng, J., Alavi, A., and Chellappa, R. (2016). Learning a structured dictionary for video-based face recognition. In *WACV*.

[81] Ye, M., Giannarou, S., Meining, A., and Yang, G.-Z. (2016). Online tracking and retargeting with applications to optical biopsy in gastrointestinal endoscopic examinations. *Medical image analysis*, 30:144–157.

[82] Yim, J., Jung, H., Yoo, B., Choi, C., Park, D., and Kim, J. (2015). Rotating your face using multi-task deep neural network. In *CVPR*.

[83] Zbontar, J. and LeCun, Y. (2015). Computing the stereo matching cost with a convolutional neural network. In *CVPR*.

[84] Zhang, D., Xu, Y., and Zuo, W. (2016a). Sparse representation-based methods for face recognition. In *Discriminative Learning in Biometrics*, pages 199–214. Springer.

[85] Zhang, D., Xu, Y., and Zuo, W. (2016b). Sparse representation-based methods for face recognition. In *Discriminative Learning in Biometrics*, pages 199–214. Springer.

[86] Zhao, R., Ouyang, W., and Wang, X. (2014). Learning mid-level filters for person re-identification. In *CVPR*.

[87] Zhu, Z., Luo, P., Wang, X., and Tang, X. (2014a). Multi-view perceptron: a deep model for learning face identity and view representations. In *NIPS*.

[88] Zhu, Z., Luo, P., Wang, X., and Tang, X. (2014b). Recover canonical-view faces in the wild with deep neural networks. *arXiv preprint arXiv:1404.3543*.