Hard negative generation for identity-disentangled facial expression recognition

Xiaofeng Liu\textsuperscript{a,b,c,e}, B.V.K. Vijaya Kumar\textsuperscript{c,d}, Ping Jia\textsuperscript{a,b}, Jane You\textsuperscript{a,h,e}

\textsuperscript{a} University of Chinese Academy of Sciences, Beijing, China
\textsuperscript{b} Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun, China
\textsuperscript{c} Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA
\textsuperscript{d} Carnegie Mellon University Africa, Kigali, Rwanda
\textsuperscript{e} Dept. of Computing, The Hong Kong Polytechnic University, Hong Kong, China

\textbf{A R T I C L E   I N F O}

Article history:
Received 22 April 2018
Revised 18 October 2018
Accepted 2 November 2018
Available online 3 November 2018

Keywords:
Hard negative generation
Adaptive metric learning
Face normalization
Facial expression recognition

\textbf{A B S T R A C T}

Various factors such as identity-specific attributes, pose, illumination and expression affect the appearance of face images. Disentangling the identity-specific factors is potentially beneficial for facial expression recognition (FER). Existing image-based FER systems either use hand-crafted or learned features to represent a single face image. In this paper, we propose a novel FER framework, named identity-disentangled facial expression recognition machine (IDFERM), in which we untangle the identity from a query sample by exploiting its difference from its references (e.g., its mined or generated frontal and neutral normalized faces). We demonstrate a possible ‘recognition via generation’ scheme which consists of a novel hard negative generation (HNG) network and a generalized radial metric learning (RML) network. For FER, generated normalized faces are used as hard negative samples for metric learning. The difficulty of threshold validation and anchor selection are alleviated in RML and its distance comparisons are fewer than those of traditional deep metric learning methods. The expression representations of RML achieve superior performance on the CK+, MMI and Oulu-CASIA datasets, given a single query image for testing.

© 2018 Elsevier Ltd. All rights reserved.

1. Introduction

Facial expression is the most natural and expressive nonverbal channel for humans to communicate their emotions [1]. Therefore, facial expression recognition (FER) has been an important and active topic for a wide range of applications including soft biometrics, digital entertainment, health care, robot systems and human-computer interaction (HCI). Ekman and Friesen postulated the universality of neutral (Ne) and six prototypical human facial expressions, namely, anger (An), disgust (Di), fear (Fe), happiness (Ha), sadness (Sa) and surprise (Su) [2].

The performances of the FER systems usually depend heavily on facial expression representations, which are affected by pose and illumination variations as well as facial morphology variations (i.e., identity-specific factors). As some facial expressions involve subtle facial muscle movements, the extracted expression-related information from different classes (in this paper, class refers to expression) can be overwhelmed by high-contrast identity-specific geometric or appearance features degrading FER performance. As illustrated schematically in Fig. 1, we want the intra-class distances of happy face images from different people to be smaller than the inter-class distances between face images of different expressions from the same subject. However, the nuisance identity factors often dominate the representation of the image in pixel space causing two images of the same subject with different expressions to be closer to each other than the same-expression images from two different subjects. This is because the extracted facial representation often contains identity-specific information that is irrelevant and may be counter-productive for the FER task. These identity-specific factors may degrade the FER performance on new identities unseen in the training data.

Aided by the advances in deep learning for computer vision [3], much progress has been made on extracting a set of features to represent a single facial expression image [4-5]. The hand-crafted features are constructed by exploiting domain knowledge of the specific relationships within pixels so that the features are invariant to some simple transformations (e.g., translation and scaling). More recently, feature-learning approaches are being investigated because of their ability to produce features that are tolerant to complex transformations. In the case of FER, identity-associated

---

\textsuperscript{e} Corresponding author.

E-mail addresses: luxiaofeng@cmu.edu (X. Liu), kumar@eco.cmu.edu (B.V.K. Vijaya Kumar), jia@comp.ac.cn (P. Jia), csyyjia@comp.polyu.hk (J. You).

https://doi.org/10.1016/j.patcog.2018.11.001
0031-3203/© 2018 Elsevier Ltd. All rights reserved.
因素可能落入这一类别。然而，由于紧耦合了各种误导因素，当我们试图减少与这些最先进的策略相关的身份敏感性，我们可能无法有效分离所有变化在面部表情中的变化。

我们的初步工作[6,7]提出了一种使用面部表情识别性能通过分离身份因素的方法来改善面部表情识别性能。通过利用通过面部表情识别方法中度量学习方法。这样可以将查询图像映射到非人脸数据库中包含其他面部表情图像的非人脸数据库中。查询图像映射到非人脸数据库中包含其他面部表情图像的非人脸数据库中。图像到负集。实际上，我们可能不需要比较查询面部表情与任何其他类别的面部表情。根据某些心理学和解剖学研究[8]，肌肉活动的不同的面部表情源于中性面作为图2所示。这是也是基本原理，面部表情不受影响的行动单位（AU）和面部表情编码系统（FACS）由Ekman[2]提出的。由于表达类是自然更具有分歧，从其他非中性面貌，对于面部表情，我们可能需要强调中性-表达距离多于需要大距离之间的表情类。中性面可以是理想的标准，可以改善度量学习的效率。然而，中性-表达面表情图像对可能不总是可用在实际的应用中和一些FER数据集。

上述见解表明，我们应当生成前额和中性归一化图像的查询图像和分离可计数的度量学习方法。我们提出的IDFERM方法就是基于这个目标。IDFERM包含两个主要部分，即硬的负生成（HNG）网络和径向度量学习（RML）网络。具体来说，给定一个查询图像，其归一化参考将会被合成使用HNG网络训练的通过表达归一化面部表情图像对的相同主体。然后，查询-参考对被输入到径向度量学习（RML）网络，该网络采用一种族向式卷积（Conv）层组和单一两分枝完全连接（FC）层框架来提取查询-参考对的度量。通过将面部表情图像从其生成参考和选择它们到它们的簇中心，度量学习可以将这些误导因素用于平衡内-外类的变化。

与其他基于图像的FER系统不同，该系统通过从单个输入图像提取表达-参考对来分离度量学习，该方法具有身份特定的因数。与基于视频的FER方法[9,10]或真实人脸表情图像对的方法[6,7,11]相比，我们的方法通过合成参考来解决现实世界极限，其中有些数据集可能不包含所有可能的面部表情样本。

在本文中，我们研究了几个概念并给出以下几点：

1）我们研究了不同表达类别的优先关系，并通过生成方案提出一种新型的人脸识别方法，作为替代传统人脸样本硬采样方案。
2）我们设计了端到端的IDFERM来提取身份-分离化表示，用于FER而无需实际的表达-中性对输入。
3）我们需要的度量距离比较适用于已适应RML的度量，该度量是数量级小于所需度量的。
4）我们通过使用新的架构和对更多数据集进行实验，测试中性面部表情样本。

在总结中，本文做出了以下贡献。

- 我们提出了一种改进的度量学习损失函数，它通过自适应学习参考阈值，解决了阈值验证和锚点选择的困难。
- 与基于身份的HNG方法相比，我们设计的RML只需要在较少迭代和计算中避免身份-不变FER。
- 我们联合优化使用HNG网络来生成逼真和身份保的开发数据集，通过结合现有知识从数据分布和领域知识来改善人脸表情图像。
- 使用数值实验，我们证明了该方法在CK+，MMI和Oulu-CASIA数据集上的有效性。
The rest of this paper is organized as follows. Section 2 briefly reviews related work in the literature. Section 3 introduces in detail the proposed normalized face generation with perceiver generative adversarial networks. Section 4 shows its application to FER with RML network. Section 5 reports the experimental results and ablation study of the auxiliary parts in HNG network. Finally, Section 6 provides our conclusions.

2. Related work

Despite receiving considerable research attention, FER remains very challenging [12]. Research developments in deep learning, especially the success of convolutional neural networks (CNN), have made high-accuracy image classification possible in recent years. Deep learning-based FER methods have emerged starting with Bengio et al. [13] who described the use of carefully designed CNN to learn expression features from raw pixels. Despite its popularity, current softmax loss-based approach does not explicitly reward intra-class compactness and inter-class separation, and identity-related factors remain major obstacles for FER. Machine recognition usually is based on similarity metrics, but those metrics may be more sensitive to identity than expressions. To decouple these two types of similarity and exploit the appearance information, substantial efforts have been dedicated to extracting features by learning [14]. Given that the expressions are formed by relaxing or contracting some facial muscles that result in temporally deformed facial features, identity-disentangled representations for FER normally separate a face with expression into a main component neutral face that encodes identity cues and an action component that encodes motion cues (such as movements of eye brows, cheeks, lips, eyelids and nose) which are related to the AUs and FACS [2].

FER is certainly not unique among computer vision applications that have to cope with nuisance factors causing variability in the data. Deep metric learning approaches have been shown to be successful for person and vehicle identification tasks [15–17], which also exhibit large intra-class variations. The initial work in this domain [18] involves training a Siamese network. Pairwise examples are fed into two symmetric sub-networks and the network is updated using contrastive loss function, i.e., their extracted representations should be close to each other if the inputs have the same class label, otherwise the distance between these representations should be large. One improvement is the triplet loss [19], in which, the inputs are triplets, each consisting of a query, a positive example and a negative example. An anchor is chosen from the query or positive examples, then the method requires the difference of the distance from the anchor point to the positive or query example and from the anchor point to the negative example to be larger than a fixed margin r. Recently, some variants of this offering faster and more stable convergence have been developed. The (N + 1)-tuplet loss [20] incorporated multiple negative examples while the coupled cluster loss (CCL) [15] incorporated multiple positive examples in a triplet. The center of positive examples c+ is set as the anchor in CCL. By comparing each example with c+ instead of each other, the number of distance evaluations needed are reduced significantly.

For the situation shown in Fig. 3, the triplet loss, (N+1)-tuplet loss and CCL are all 0, since the distances between the anchor and positive examples are indeed smaller than the distance between the anchor and negative examples for a margin r. This means the loss function will learn to neglect such non-trivial samples. We will need many more input iterations with properly selected anchors to correct this situation. The fixed threshold in the contrastive loss was also proven to be sub-optimal [21]. The difficulty of threshold validation and anchor selection have long been significant challenges until our initial work [6], which included an adaptive (N + M)-tuplet clusters loss function.

Also, the traditional online or offline mini-batch sample selection is a large additional computational burden and can result in poor local optima [22]. Generating all possible pairs or triplets would result in quadratic and cubic complexity, respectively and most of these pairs or triplets are not very useful for the training [6]. Our initial work [6] utilized identity-aware hard-negative mining and online positive mining for FER, but it still suffers from the dataset-sensitive and computationally-expensive example mining to provide nontrivial triplets.

Several approaches have been proposed for generative models. Conventional methods such as Principal Components Analysis (PCA), Independent Components Analysis (ICA), Gaussian Mixture Model (GMM), etc., have difficulty in modeling complex patterns of irregular distributions [23]. Recently, Restricted Boltzmann machines (RBM), Hidden Markov Model (HMM), Markov Random Field (MRF) etc., have been employed for modeling images of digits, texture patches, and well-aligned faces [24]. However, the limited ability of feature representations restricts further development. Since deep hierarchical architectures of the recent generative models are capable of capturing complex structure of data, generated images from these deep hierarchical structures are more realistic. The denoising auto-encoder (DAE) pairs a differentiable encoder and decoder, which encodes an image sample x to a latent representation z and then decodes the z back to another image ẑ [25]. For the normalized face generation task, pose and expression are regarded as the noise to be mitigated. The main limitation of this approach is that the squared pixel-wise reconstruction error would cause the generated samples to look blurry as they generate the mean image of the distribution. Generative Adversarial Network (GAN) [26] simultaneously trains two networks: a generative network Gen to synthesize images (maps latents z to image space), and a discriminative network Dis to discriminate between real training images from generated images. With the GAN, an expected image can be generated from a randomly sampled vector z from a certain distribution.

Normally, the GAN schemes are not well-matched to supervised recognition tasks. The GAN-generated results are expected to align with the central part of the data distribution, while the boundary
between classes in feature space is more important for classification. Limited research has been devoted to this topic. The semi-GAN [27] adds an extra task for a discriminator network to improve semi-supervised recognition task. The face rotator schemes proposed by Tran [28] generates a frontal face as the preprocessing for the face recognition network.

3. Hard negative generation

As we are trying to disentangle identity-related factors from a facial expression image \( x \), a reference neutral face image from the same subject is required to obtain the difference between the neutral face and \( x \) for FER. However, such a neutral face reference image is not always available in real-world application scenarios. Instead of mining several negative samples, we directly use the generated normalized face image as negative sample. The goal of the hard negative generation (HNG) network is to produce a photorealistic and identity-preserved normalized face image \( \tilde{x} \) from the probe image \( x \). The network architecture and loss functions are illustrated in Fig. 4. Our HNG network is composed of five major components: 1) an Encoder network \( Enc \), 2) a Decoder network \( Dec \), 3) a Discriminator network \( Dis \), 4) the Light CNN network and 5) the VGG-facenet. The function of \( Enc \) and \( Dec \) network is the same as that in denoising auto-encoder [25]. In DAE, the output is not required to be exactly the same as the input. For example, the denoising auto-encoder takes in an image that has been corrupted by some form of “noise”, and forced to output a denoised version of that image by requiring the output image to be similar to the original “clean” image. In our application, a face image with expression (input image \( x \)) can be regarded as a copy of neutral face image (target image \( y \)) that has been corrupted by expressions. The denoising (disentangling expression from a face image) is achieved by requiring our output \( \tilde{x} \) to be close to the target image \( y \), as the DAE requires its generated image to be close to a clean target image instead of the original noisy image. The \( Enc \) maps the input sample image \( x \) to a latent representation \( z \) through a learned distribution \( P(z|x) \), while the \( Dec \) generates predicted a facial image \( \tilde{x} \) corresponding to \( z \). The function of the \( Dec \) and \( Dis \) is the same as that in the GAN [26]. The \( Dec \) network tries to generate the real distribution by the loss of \( Dis \) which learns to distinguish between generated image \( \tilde{x} \) and real image in the guiding set \( g \). The guiding set contains all of the target images \( y \), which is a compilation of numerous frontal and neutral real face images. The input-target pairs \( \{x_i, y_i\} \) from multiple identities are required to learn the parameters \( \theta \) of the differentiable encoder \( \theta_{Enc} \) and decoder \( \theta_{Dec} \), where \( x \) is a face image with expression and \( y \) is the frontal neutral face image of that person. In our experimental setting, five different loss functions are used to combine the advantages of high quality GAN and stable auto-encoder which encodes the data into a latent space \( z \). In this section, we show how the multiple objective functions are employed for different parts to generate the facial reference images for FER.

3.1. Feature space perceptual loss

The squared error loss between the CNN feature representations is adopted to represent the feature-level perceptual loss. We make use of the independently-trained and fixed VGG-FaceNet [29] to model this semantic feature-level loss. Although it comprises 14 Conv layers and 3 FC layers, we omitted the deeper layers after the 5th Conv layer because their limited spatial resolution cannot support good image reconstruction performance. Denoted by \( \psi \), the feature map of the \( l \)th convolutional layer of VGG-FaceNet is used to extract the feature representations using the standard forward-propagation process. The semantic perceptual loss between two images \( \tilde{x} \) and \( y \) on the \( l \)th convolutional layer is defined as the following squared-error loss between the two feature maps.

\[
L_{feat}(\tilde{x}, y) = \frac{1}{W_l \times H_l} \sum_{n=1}^{W_l} \sum_{m=1}^{H_l} \|\psi_{l,n,m}(\tilde{x}) - \psi_{l,n,m}(y)\|^2_2 \tag{2}
\]

where \( W_l \) and \( H_l \) denote the width, height of the \( l \)th feature map and \( \psi_{l,n,m} \) is the value of the \( l \)th feature map at point \((n, m)\). In our experiment, \( l=5 \) based on empirical testing.

3.2. Symmetry loss

Symmetry is an inherent property of normal human faces. The symmetry loss of a face image takes the form:

\[
L_{sym}(\tilde{x}) = \frac{1}{W \times H^2} \sum_{n=1}^{W/2} \sum_{m=1}^{H} \|\tilde{x}_{n,m} - \tilde{x}_{W-(n-1),m}\| \tag{3}
\]

where \( W \) and \( H \) are the width and the height of the images and \((n, m)\) denotes the pixel of the generated image. The \(|\cdot|\) denotes the absolute value. For simplicity, when training our model with the symmetry loss, all the inputs are aligned and detected if the occluded parts are on the right side of image. If not, images are flipped so that the occluded parts are on the right side. Real-world
images may not exhibit the strict symmetry of pixel values. Considering the consistency of the pixel difference inside a local area, and the gradients at a point along all directions are largely preserved under different illuminations, minimizing a symmetry loss in the Laplacian space should emphasize human faces.

3.3. Adversarial loss

We introduce a discriminator Dis which serves as a supervisor to push the synthesized image to reside in the manifold of frontal neutral face images. It can reduce the blur effect and produce visually pleasing results.

The Dis aims to discriminate the predicted frontal neutral face image $\tilde{x}_t$ from real ones $x_i$ in the guiding set, and is trained concurrently with the transform network (Enc and Dec). The transform network tries to “trick” the Dis to classify the generated images as real. Formally, the discriminator is trained to minimize the following binary cross entropy loss:

$$L_{\text{GAN-Dis}}(g_i, \tilde{x}_t) = -\log(D_{\text{Dis}}(g_i)) - \log(1 - D_{\text{Dis}}(\tilde{x}_t))$$  \hspace{1cm} (4)

With respect to Dec, the parameters are trained by minimizing the following loss:

$$L_{\text{GAN-Dec}}(\tilde{x}_t) = -\log(D_{\text{Dis}}(\tilde{x}_t))$$  \hspace{1cm} (5)

3.4. Identity-Preserving loss

Synthesizing the frontal neutral face image while preserving the identity is a critical part of IDFERM. We introduce a direct supervision to reward the perceptual similarity between input and generated images using the face verification network. In our approach, we use the pre-trained Light CNN, a compact network that has only 4 convolution layers with Max-Feature-Map operations and 4 max-pooling layers [30]. In this work, the identity-preserving loss is defined based on the activations of the last two layers of the Light CNN:

$$L_{id} = \sum_{l=1}^{2} \frac{1}{W_l \times H_l} \sum_{n=1}^{W_l} \sum_{m=1}^{H_l} |\phi_{l,n,m}(\tilde{x}) - \phi_{l,n,m}(x)|$$  \hspace{1cm} (6)

where $W_l, H_l$ denotes the width and height of the $l^{th}$ layer, $\phi_{l,n,m}$ is the value of the feature map $(n, m)$ point and $|\cdot|$ denotes the absolute value.

3.5. Pixel-wise loss

Adversarial training is known to be sensitive to hyper parameters. Adding the following pixel-wise $L_1$ loss

$$L_{\text{pixel}} = \frac{1}{W \times H} \sum_{n=1}^{W} \sum_{m=1}^{H} |\tilde{x}_{n,m} - y_{n,m}|$$  \hspace{1cm} (7)

in the image space with a relatively small weight is one method to stabilize the training and accelerate the optimization. $\tilde{x}_{n,m}$ and $x_{n,m}$ are the pixel level gray values of the $(n, m)^{th}$ pixel.

Using judicious selection of aforementioned loss functions, we train the Enc, Dec and Dis simultaneously. The error signal from adversarial loss and symmetry loss are not back-propagated to Enc. Several tradeoff parameters constrained between $0$ and $1$ are used to balance the aforementioned loss functions. The weights $\lambda_1$ and $\lambda_2$ in Algorithm 1 are the tradeoff parameters for $L_{\text{feat}}, L_{id}$ and $L_{\text{pixel}}$ for the Enc and Dec. The parameter $\lambda_3$ is used to weight the $L_{\text{sym}}$ in Dec. As Dec also receives the error signal from the Dis, a parameter $\eta$ is used to weight the ability of fooling the discriminator.

We show some of the input-output pairs of our HNG network in Fig. 5. As the common quantitative metrics (e.g., log-likelihood of a set of validation samples) are often very informative for perceptual generative models [31], we provide a qualitative comparison of visual quality and a quantitative evaluation of identity-preservation in Section 5.

Unlike previous generative methods that utilize their intermediate features for the recognition tasks, the resulting expression- and pose- disentangled face image has potential for several downstream applications, such as facial expression or face recognition, and attribute estimation.

4. Radial metric learning

The proposed RML only requires the comparison of the representation of the query sample $f_i$ with the representation of its generated reference $\tilde{f}_j$ and its cluster center $C_{y}$. We introduce a distance $T$ from the query sample $x$ to control the relative boundary (T - $\frac{T}{2}$) and (T + $\frac{T}{2}$) for the intra-class center and generated references, respectively. The RML loss function is formulated as follows.

$$L(\{x_i\}_{i=1}^K, \{\tilde{x}_i\}_{i=1}^K, f) = \frac{1}{K} \sum_{i=1}^{K} \max(0, D(f, C_{y}) - T + \frac{T}{2})$$

Algorithm 1 Training the HNG network.

\begin{itemize}
  \item $\theta_{\text{Enc}}, \theta_{\text{Dec}}, \theta_{\text{Dis}} \leftarrow$ initialize network parameters
  \item \textbf{Repeat}
  \item \hspace{0.5cm} $X \leftarrow$ random mini-batch from dataset
  \item \hspace{0.5cm} $Z \leftarrow$ Enc($X$)
  \item \hspace{0.5cm} $\hat{X} \leftarrow$ Dec($Z$)
  \item \hspace{0.5cm} $L_{\text{feat}} = \frac{1}{2K} \sum_{i,m=1}^{K} \sum_{n=1}^{W} \sum_{m=1}^{H} \phi_{i,n,m}(X) - \phi_{i,n,m}(Y)$$
  \item \hspace{0.5cm} $L_{\text{sym}} = \frac{1}{2K} \sum_{i,m=1}^{K} \sum_{n=1}^{W} \sum_{m=1}^{H} |\tilde{x}_{i,n,m} - x_{i,n,m}|$
  \item \hspace{0.5cm} $L_{\text{GAN-Dis}} = -\log(D_{\text{Dis}}(g_i)) - \log(1 - D_{\text{Dis}}(\tilde{x}_t))$
  \item \hspace{0.5cm} $L_{\text{GAN-Dec}} = -\log(D_{\text{Dis}}(\tilde{x}_t))$
  \item \hspace{0.5cm} $L_{id} = \frac{1}{W \times H} \sum_{m=1}^{W} \sum_{m=1}^{H} |\phi_{l,n,m}(\tilde{x}) - \phi_{l,n,m}(x)|$
  \item \hspace{0.5cm} $L_{\text{pixel}} = \frac{1}{W \times H} \sum_{m=1}^{W} \sum_{m=1}^{H} |\tilde{x}_{n,m} - y_{n,m}|$
  \item \hspace{0.5cm} $\theta_{\text{Enc}} \leftarrow -\nabla_{\theta_{\text{Enc}}} L_{\text{Enc}}$
  \item \hspace{0.5cm} $\theta_{\text{Dec}} \leftarrow -\nabla_{\theta_{\text{Dec}}} L_{\text{Dec}}$
  \item \hspace{0.5cm} $\theta_{\text{Dis}} \leftarrow -\nabla_{\theta_{\text{Dis}}} L_{\text{Dis}}$
  \item \hspace{0.5cm} \textbf{Until deadline}
\end{itemize}
\[
+ \max \left( 0, \frac{\tau}{2} + T - D(\tilde{f}_i, C_y) \right)
\]

(8)

Only if the distances from all online mined examples \(f_i\) to its updated \(C_y\) are smaller than \((T - \frac{\tau}{2})\) and the distances from all the generated references \(\tilde{f}_i\) to its updated \(C_y\) are larger than \((T + \frac{\tau}{2})\), the loss \(L(\{x_i\}_i^K, \{\tilde{x}_i\}_i^K; f)\) can get a zero value. A simplified geometric interpretation of this is shown in Fig. 6.

By assigning different values for \(T\) and \(\tau\), we define a flexible learning task with adjustable difficulty for the network. We do not use the special case that requires inter-class variation to be zero (i.e., \(T = \tau/2\)) as the center loss [32] for the FER training set usually contains some unreliable labels [33]. However, these two hyper-parameters need manual tuning and validation. In here, we formulate the reference distance \(T\) to be a function \(S(\cdot, \cdot)\) which should be trained automatically, instead of a constant. Inspired by the Mahalanobis distance matrix \(M\) in Mahalanobis distance \(D\) (Eq. 9), which is a positive semi-definite (PSD) matrix and can be calculated via the linear fully connected layer as in [34], we try to automatically train both \(S\) and \(D\). Since the difference of the reference distance and the distance function need to be calculated in two terms in Eq. 8, a possible solution is to calculate \(S-D\) function via the linear FC layer.

\[
D(f_1, f_2) = \left\| f_1 - f_2 \right\|_M^2 = (f_1 - f_2)^\top M (f_1 - f_2)
\]

(9)

Since the metric \(M\) itself is quadratic, we assume that \(S\) has a simple quadratic form:

\[
S(f_1, f_2) = \frac{1}{2} f_1^\top \tilde{A} f_1 + \frac{1}{2} f_2^\top \tilde{B} f_2 + c^\top (f_1 - f_2) + b
\]

(10)

where \(\tilde{A}\) and \(\tilde{B}\) are both the \(d \times d\) real symmetric matrices (not necessarily positive semi-definite), \(c\) is a \(d\)-dimensional vector, and \(b\) is the bias term.

Then, a new quadratic expression \(H(f_1, f_2) = S(f_1, f_2) - D(f_1, f_2)\) is defined to combine the reference distance function \(S\) and the Mahalanobis distance metric function \(D\). Substituting \(S(f_1, f_2)\) and \(D(f_1, f_2)\) into \(H(f_1, f_2)\), we get:

\[
H(f_1, f_2) = \frac{1}{2} f_1^\top (\tilde{A} - 2M) f_1 + \frac{1}{2} f_2^\top (\tilde{A} - 2M) f_2 + f_1^\top \tilde{B} f_2 + c^\top (f_1 - f_2) + b
\]

\[
+ c^\top (f_1 - f_2) + b
\]

(11)

\[
H(f_1, f_2) = \frac{1}{2} f_1^\top A f_1 + \frac{1}{2} f_2^\top B f_2 + f_1^\top \tilde{B} f_2 + c^\top (f_1 - f_2) + b
\]

(12)

where \(A = (\tilde{A} - 2M)\) and \(B = (\tilde{B} + 2M)\). Suppose \(A\) is PSD and \(B\) is negative semi-definite (NSD), \(A\) and \(B\) can be factorized as \(L_A L_A^\top\) and \(L_B L_B^\top\). Then \(H(f_1, f_2)\) can be rewritten as follows:

\[
H(f_1, f_2) = \frac{1}{2} f_1^\top L_A L_A^\top f_1 + \frac{1}{2} f_2^\top L_B L_B^\top f_2 + f_1^\top L_B L_B^\top f_2 + c^\top (f_1 - f_2) + b
\]

\[
+ c^\top (f_1 - f_2) + b
\]

(13)

\[
H(f_1, f_2) = \frac{1}{2} (L_A f_1)^\top (L_A f_1) + \frac{1}{2} (L_B f_2)^\top (L_B f_2) + (L_B f_1)^\top (L_B f_2)
\]

\[
+ c^\top (f_1 - f_2) + b
\]

(14)

Motivated by the above, we propose a general, computationally feasible loss function. Following the notations in the preliminaries and denoting \((L_A, L_B, c^\top)\) as \(W\) which can be learned via the linear fully connected layer, we have:

\[
L_{W}(x_i)_i^K, (\tilde{x}_i)_i^K; f) = \frac{1}{K} \sum_{i=1}^{K} \left[ \max \left( 0, \frac{\tau}{2} - H(f_i, C_y) \right) + \max \left( 0, H(f_i, C_y) + \frac{\tau}{2} \right) \right]
\]

(15)

Moreover, we simplify \(\frac{\tau}{2}\) to be the constant \(1\), since changing it to any other positive value results only in the matrices being multiplied by a corresponding factor. Our hinge-loss like function is given as follows.

\[
L_{W}(x)_i^K, (\tilde{x}_i)_i^K; f) = \frac{1}{K} \sum_{i=1}^{K} \left[ \max(0, 1 - H(f_i, C_y)) + \max(0, H(f_i, C_y) + 1) \right]
\]

(16)

By doing this, the adaptive threshold can be seamlessly factorized into a linear-fully connected layer for end-to-end learning [34]. The RML loss can also be easily used as a drop-in replacement for the triplet loss and its variants, as well as used in tandem with other performance-boosting approaches and modules, including modified network architectures, pooling functions, data augmentations or activation functions.

For a training batch consisting of \(K\) query samples, the number of input passes required to evaluate the necessary embedding feature vectors in our application is \(K\), and the total number of distance comparisons can be \(2K\). Normally, \(K\) is much larger than 2. In contrast, triplet loss and \((N+1)\)-tuplet loss require \(O(K^2)\) comparisons, the contrast loss and CCL require \(O(K^2)\) comparisons, and the \((N+M)\)-tuplet cluster loss requires \(2(N+M)^2K\) comparisons after a strict example mining scheme using the special structure of some other datasets (i.e., each subject has all 6 expressions). Here \(N\) and \(M\) are the number of mined positive samples and the number of mined negative samples, respectively. Even for a dataset of a moderate size, it is computationally impractical to load all possible meaningful triplets into the processor memory for model training. With predefined anchors (i.e., \(C_y\) and \(\tilde{f}_i\)), we also alleviate the difficulty of anchor selection [6].

......The inception convolutional FER network and two-branch FC layer joint metric learning architecture proposed in our preliminary paper [6] are used in our framework in Fig. 4. The convolutional groups are made up of a \(1 \times 1, 3 \times 3\) and \(5 \times 5\) Conv layers in parallel.

Combining the metric learning loss and softmax loss is an intuitive idea to possibly achieve better performance [35]. However,
combining them directly on the last FC layer is sub-optimal. The basic idea of building two-branch FC layers after the deep convolution groups is to combine two losses at different levels of tasks. We learn the detailed features shared between the same expression class with the expression classification (EC) branch, while exploiting semantic representations via the metric learning (ML) branch to handle the significant appearance changes from different subjects. The connecting layer embeds the information learned from the expression label-based detail task to the identity label-based semantic task, and balances the weights in the two task streams. This type of combination can effectively alleviate the interference of identity-specific attributes. The inputs of connecting layer are the output vectors of the former FC layers—FC\textsubscript{2-2} and FC\textsubscript{2-3}, which have the same dimension denoted as $D_{\text{input}}$. The output of the connecting layer, denoted as FC\textsubscript{4} with dimension $D_{\text{output}}$, is the feature vector fed into the second layer of the ML branch. The connecting layer concatenates two input feature vectors into a larger vector and maps it into a $D_{\text{output}}$ dimension space:

$$
FC_{2-4} = \mathbf{P}[FC_{2-2}; FC_{2-3}] = \mathbf{P}_1[FC_{2-2}] + \mathbf{P}_2[FC_{2-3}]
$$

where $\mathbf{P}$ is a $(2D_{\text{input}} \times 2D_{\text{output}})$ matrix, $\mathbf{P}_1$ and $\mathbf{P}_2$ are $D_{\text{input}} \times D_{\text{output}}$ matrices.

Regarding the sampling strategy, every training image is used as a query example in an epoch. In practice, the softmax loss will only be calculated for the query examples. The relative importance of the two loss functions is managed by a weight $\alpha$. During the testing stage, this framework takes one query image and its generated reference image as input, and determines the classification result through the EC branch with the softmax loss function. Our disentangled feature learning scheme is described in Algorithm 2.

### 5. Numerical experiments

In this section, we compare the IDFERM with state-of-art methods on three benchmark datasets, i.e., CK+ [28], MMI and Oulu-CASIA datasets. Details of our training data are provided in Section 5.1, followed by our preprocessing methods in Section 5.2, and the implementation details in Section 5.3. In Section 5.4, we report a series of ablation experiments to analyze the function of our auxiliary networks. Numerical experiment results are shown in Section 5.5.

#### 5.1. Training data

Besides the FER datasets, a variety of large datasets of facial images for face recognition are publicly available online. We give some samples from the VGG-Face dataset Fig. 7. We use the VGG-Face dataset [29] to extend our data for neutral face generation. It contains approximately 2.6 million face images, but very few of them fit our requirements of neutral expression, front-facing, having no occlusion, and of sufficient resolution for face region. We use the Google Cloud Vision API to remove those images that look blurry, with high emotion score or eyeglasses or tilt or pan angles beyond 5°. These frontal and neutral face images are used as our target and guiding set samples. Their corresponding non-compliant images from the same subject are used as the inputs. All the samples are aligned and cropped to $64 \times 64$ Gy images. After filtering, we have about 12K target images ($< 0.5\%$ of the original set) and 50K input-target pairs. These data are used for pretraining to initialize the network parameters and then fine-tuned using the CMU Multi-PIE [36].

The CMU Multi-PIE itself is a facial expression dataset, but the facial expression labels it uses are slightly different from modern expression classification system. There are 4 expressions are useable (114,305 neutral images, 19,817 surprise images, 22,696 disgust images and 47,388 happy images), while the squint and scream are not regarded as expression now. Some samples are shown in Fig. 8. It contains more than 750,000 images of 337 people taken from fifteen directions, and in nineteen illumination conditions. There are four recording sessions in which subjects were instructed to display neutral, happy, disgust and surprise facial expressions. It is more close to our FER dataset in testing stage than the filtered VGG face dataset. We selected only the five groups of the nearly frontal view faces ($−45°$ to $+45°$). The neutral images from the 0° view are used as our target image and the guiding set.

---

**Algorithm 2** Disentangled feature learning algorithm.

**Input**
- $K$ face examples $\{x_i\}_{i=1}^K$ and their generated references $\{\tilde{x}_i\}_{i=1}^K$.

**Output**: The parameters of the FER network $\theta_{\text{FER}}$

1. While not converge do
2. Map examples to feature plane with CNN to get: $\{f_i\}_{i=1}^K$ and $\{\tilde{f}_i\}_{i=1}^K$.
3. Calculate the cluster centers $C_i$ for each class
4. $L_{\text{ML}} = \frac{1}{K} \sum_{i=1}^{K} \max(0, -H(f_i, C_i)) + 1) + \max(0, H(\tilde{f}_i, C_i) + 1)\)$
5. $L_{\text{softmax}} = -\log(e^{-\text{max}(f, i)}/\sum_{j} e^{-\text{max}(j)})$
6. Compute the joint loss $L_{\text{softmax}} + \alpha L_{\text{ML}}$
7. Compute the backpropagation error
8. Update the parameters
End while

---

**Fig. 7.** Samples from the VGG-Face dataset. Each row contains the face images of the same person.

**Fig. 8.** Samples from the CMU Multi-PIE dataset. Each row contains images of the same person.
5.2. Preprocessing

We follow the [6] to locate the 49 facial landmarks. Then, face alignment is done to reduce in-plane rotation and crop the region of interest based on the coordinates of these landmarks to a size of $64 \times 64$. An augmentation procedure is employed to increase the number of training images and alleviate the chance of over-fitting. We crop five $60 \times 60$ size patches from the center and four corners, flip them horizontally and transfer them to grayscale images. All the images are processed with the standard histogram equalization and linear plane fitting to remove unbalanced illumination. Finally, we normalize them to have zero mean and unit variance. In the testing phase, a single center crop with the size of $60 \times 60$ is used as input data.

5.3. Implementation details

We use $64 \times 64$ Gy images as the input-target pairs for the neutral face generation training. The filtered VGG-FaceNet and Multi-PIE images are used to pre-train the neutral face generation network. We construct the guiding set using the filtered VGG-FaceNet frontal neutral view and the 0° view neutral images from the Multi-PIE. Following the experimental protocol in [6], we pre-train our inception style convolutional groups, two branch FC layers on with 204,156 frontal view ($-45°$to $+45°$) face images selected from the CMU Multi-PIE dataset for 300 epochs, optimizing the joint loss using stochastic gradient descent with a momentum coefficient of 0.9. The initial network learning rate, batch size, and weight decay parameter are set to 0.1, 128, 0.0001, respectively based on optimizing the parameter choices using the validation set. If the training loss increased by more than 25% or the validation accuracy does not improve for ten epochs, the learning rate is halved and the previous network with the best loss is reloaded. We select the highest accuracy training epoch as our pre-trained model. In the fine-tuning stage, the mini-batch set size is fixed to two times the number of expression classes of the dataset. Random search is employed to select 2 images from each expression class to form the mini-batch set. The tuple-size is set to 12. In all our experiments, we set $\eta = 0.1$, $\lambda_1 = 3 \times 10^{-2}$, $\lambda_2 = 10^{-2}$, $\lambda_3 = 0.3$ determined by manual tuning. The weight of joint learning $\sigma = 1$. In the testing phase, only the convolutional groups and expression classification branch with softmax are used to recognize a single facial expression image.

The details of the encoder of HNG can be found in [7]. We fixed the latent vector dimension to be 256 and found this configuration to be sufficient for generating images for FER. A series of fractional-stride convolutions (FConv) transforms the 256-dim vector $z \in \mathbb{R}^{256}$ into a synthetic image $x \in \mathbb{R}^{64 \times 64}$, which is of the same size as $x$. To further incorporate the prior knowledge of the frontal neutral face's distribution into the training process, we introduce a discriminator $D$ to distinguish the generated face image from the real images in the guiding set.

The Leaky ReLU nonlinearities [37] are used in some Conv layers, where $LReLU(x) = \max(x, 0) + \sigma \min(x, 0)$. In our experiments, we set $\sigma = 0.1$. Optimizing this minimax objective function will continuously push the output of the generator to match the target distribution of the guiding set thus making the synthesized facial images to be more photorealistic. All the CNN architectures are implemented with the widely used deep learning tool “Tensorflow” [38].

5.4. Ablation study

The Light CNN and the first five layers of the VGG-FaceNet are used to embed the input, target or output images for the similarity measurements in different feature spaces. It is obvious that these two networks incur additional computation cost. We show in this section that they are needed.

The difference of our models trained with and without the $\mathcal{L}_{id}$ is subtle in visual appearance, as can be seen in Fig. 9, but its effect on improving the identity likeness of the generated faces can be measured by evaluating the similarity of the input-outputs pairs using VGG-FaceNet. Fig. 10 shows the distributions of $L_2$ distances between the embeddings of the facial expression images and their corresponding synthesized results, for models trained with and without this loss. Schroff et al. [18] consider two FaceNet embeddings to encode the same person if their $L_2$ distance is less than 1.242. All of the synthesized images using the identity-preserving loss pass this test using FaceNet, but about 2% of the images would be identified as a different subject by FaceNet when not using the identity-preserving loss. We investigated the effect of the weight of identity preserving loss and show the identity inconsistent percentage in Table 1.

The VGG-FaceNet is employed to calculate the feature level perceptual loss, which is expected to make the generated result to keep more perceptually important image attributes, for example sharp edges and textures. This loss was empirically given the largest weight in our experiments. In practice, without this part, we could never avoid the collapse of the adversarial training to generate the human face structure.

We also analyzed the effect of different hyper-parameter values in RML. The parameter $\sigma$ is used to balance the softmax loss and metric learning loss. We can see from the Fig. 11 (reproduced below) that the highest accuracy is achieved when $\sigma \epsilon [0.95, 1]$. As can be seen in Fig. 12 below, the networks were not sensitive to $\sigma \epsilon [0.075, 0.125]$.

5.5. Experimental results of FER

To evaluate the effectiveness of the proposed method, extensive experiments have been conducted on four well-known publicly available facial expressions datasets: CK + , MMI and Oulu-CASIA. Resulting IDFERM confusion matrices are shown in Fig. 13.

**CK + Dataset [39]:** We conducted both seven-class and eight-class expression recognition experiments (i.e., without or with neutral expression). In the setting without neutral sample, we directly compare to the most nontrivial hard negative samples (i.e., generated normalized face), which not only relaxes the requirements on the dataset (i.e., needing images of all different expressions of the same person) to extract the identity-disentangled expression representation but also reduces the number of comparisons in training stage. The training time of metric learning part is largely reduced as shown in Table 2. We note that [7] and IDFERM do need an additional training stage (around 6 hours) for normalized face generation, but the trained HNG network works for all of the FER datasets in our experiments without fine-tuning and can be regarded as a ready-made tool for several down-stream tasks.

In the testing stage, IDFERM recognizes a query facial expression image in about 50 ms, which is satisfactory for many applications. Video-based methods normally need a relatively longer sampling time (>0.25 s) to collect the whole expression change session.

From Table 3, we can see that the identity-disentangled representation with adaptive metric learning methods achieve higher accuracy than previous works. Therefore, comparing the query sample with its generated normalized face rather than all its other expressive query faces as in [6] is more efficient, which is consistent with the relationship of those expressions as analyzed in Fig. 2. However, just adding the generated neutral face images to original dataset enlarged the number of comparisons [7]. As an efficient hard negative mining scheme, the HNG offers the most nontrivial hard negative samples and the RML can efficiently utilizes them and outperforms the other methods.
The improved accuracy compared to the other methods is appealing in the image-based 7-class CK+, MMI and Oulu-CASIA setting which do not have real-neutral samples as training data. It also generalized well in dataset with neutral expressions as shown in Table 4. With the added generated samples, the accuracy of neutral class is improved to 99% as shown in Fig. 13(b).

Benefitting from the generated data, the proposed IDFERM outperforms our earlier approach [6] by 2.6% on MMI dataset.
and 1.25% on CK+ dataset. Using the same generated images as in [7], we achieve 0.87% and 0.86% improvements on MMI and CK+ datasets respectively, and the number of comparisons per training batch is reduced from $2(N+M)K$ (the $N$ and $M$ in MMI are 5 and 5 respectively [7]) to 2K. As a consequence, the training time of the earlier approach [7] for MMI dataset is 2 hours 8 mins, while the IDFERM needs only 30 mins on a single Titan X GPU as shown in Table 2. Considering the improved performance and reduced training time, the IDFERM is significantly more efficient than [7] to utilize the generated data.

**Fig. 11.** Facial recognition accuracy on CK+ (7-class) dataset as a function of $\alpha$, the parameter used to balance the softmax loss and metric learning loss.

**Fig. 12.** Facial recognition accuracy on CK+ (7-class) dataset as a function of parameter $\sigma$ in LReLU.

**Fig. 13.** Confusion matrix of the proposed IDFERM evaluated in the (a) CK+ seven-class, (b) CK+ eight-class, (c) MMI and (d) Oulu-CASIA database. The predicted labels and the ground truth labels are shown in the ordinate and abscissa, respectively.
Limited training data has long been a challenge for facial expression recognition. For example, [11] utilized the FER-2013 dataset for pre-training and then fine-tuned their facial expression recognition network in CK+/MMI datasets. The FER-2013 dataset is even larger than the Multi-PIE dataset. We chose the Multi-PIE pretraining for fair comparison with previous works [6]. We added a comparison of recognition accuracy with/without the pre-training, and show the results in Table 5. We can see that the pre-training can improve the performance consistently.

**MMI Dataset** [40]: This dataset consists of 213 sequences, 208 sequences from this data set containing frontal-view faces of 31 subjects were used in our experiment as in [6]. Since the actual location of the peak frame is not provided, we collect three frames in the middle of each image sequence and associate them with the labels, which results in 624 images in our experiments as in [6,11]. We divided the MMI dataset into 10 subsets for person-independent ten-fold cross validation. The sequence-level predictions are obtained by choosing the class with the highest average score of the three images. Consequently, 10-fold cross validation was conducted. This dataset could be suitable to measure the recognition performance in realistic situations when compared to other datasets.

With the identity-disentangled FER representation, the proposed methods achieve substantial improvements over the previous best performance in MMI dataset as shown in Table 3 and Fig. 13(c). The HNG can further boost the accuracy by incorporating the prior information of normalized face and the relationships of expressions within an applicable framework. Note that the image sequences in the MMI dataset contain a full temporal pattern of expressions, i.e., from neutral to apex, and then relaxed, and are especially favored by these methods exploiting temporal information.

**Oulu-CASIA VIS Dataset** [41]: This dataset consists of 480 image sequences of 80 individuals. This dataset is captured under the visible (VIS) normal illumination conditions and is a subset of Oulu-CASIA NIR-VIS dataset. Each individual poses six basic expressions as in MMI dataset. Only the last three frames are used for individual-independent 10-fold cross validation, and the total number of images is 1440 as in [5].

In Oulu-CASIA dataset, the IDFERM performs well in recognizing fear and happy expressions, while angry is the hardest expression, which is mostly confused with disgust as shown in Fig. 13(d). The performance results are shown in Table 6 and are similar to those on the CK+ and MMI datasets.

<table>
<thead>
<tr>
<th>Table 5</th>
<th>Comparison of the performance with/without pre-training using Multi-PIE on CK+ dataset.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2B(N + M) [6]</td>
</tr>
<tr>
<td>CK+(7-class)</td>
<td>97.03%</td>
</tr>
<tr>
<td>MMI</td>
<td>78.46%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 6</th>
<th>Performance comparison of the rank-1 recognition accuracy on the Oulu-CASIA VIS dataset in terms of the 6 expressions (without neutral expression).</th>
</tr>
</thead>
<tbody>
<tr>
<td>Methods</td>
<td>Oulu-CASIA VIS</td>
</tr>
<tr>
<td>STM-ExpLet(video) [9]</td>
<td>74.50%</td>
</tr>
<tr>
<td>DTAGN(video) [10]</td>
<td>81.46%</td>
</tr>
<tr>
<td>PPDN [48]</td>
<td>84.59%</td>
</tr>
<tr>
<td>FN2EN [5]</td>
<td>87.1%</td>
</tr>
<tr>
<td>IDFERM</td>
<td>88.25%</td>
</tr>
</tbody>
</table>

6. Conclusions

We proposed and investigated a novel recognition via generation scheme termed IDFERM to disentangle the identity factors from other factors that are responsible for facial expression. The anchor-selection and threshold-tuning problems present in previous approaches have been addressed in our proposed adaptive deep metric learning paradigm. The identity-preserving neutral face image generation is efficient for hard negative mining which requires fewer similarity comparisons. However, our gray-scale image processing can lead to information loss as the image quality is not emphasized in our framework as in conventional image generation methods. Also, the adversarial game at image-level is usually time-consuming. In future work, we intend to apply some commonly used visual quality assessment methods for the generated images on top of our model for better texture. Recent feature-level GAN’s backbone can be utilized to extend our framework for faster, more stable convergence training, and more complex data structure (e.g., color images). We also expect that the application of recognition via generation idea can facilitate several other closely related tasks, e.g., face recognition, person re-ID, and pose-invariant classification.

References


Xiaofeng Liu received the B.Eng. degree in automation and B.A. degree in communication from the University of Science and Technology of China, Hefei, China, in 2014. He was the research assistant in MSRA, CMU and NTU. He is currently pursuing his Ph.D. at the University of Chinese Academy of Sciences, Beijing and a Research Associate in the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh. He was a recipient of the Best Paper award of the 25th International Conference on International Security, and Behavior Analysis 2018. His research interests include machine learning, computer vision, and pattern recognition.

B.V.K. Vijaya Kumar is the LIA & Helen Whitaker Professor of Electrical and Computer Engineering at Carnegie Mellon University, Pittsburgh. He is also the Director of Carnegie Mellon University Africa in Rwanda. His research interests include computer vision, pattern recognition algorithms, and applications, coding, and signal processing of digital data storage systems. He has co-edited a book entitled Correlation Pattern Recognition, 22 book chapters, more than 400 conference papers, and more than 200 journal papers. He is also a co-inventor of 12 patents. He served as a topical editor of Applied Optics and as an associate editor of the IEEE Transactions on Information Forensics and Security. He has served on many conference action committees and was co-chair of several conference program committees. In 2003, he received the Eta Kappa Nu Award for Excellence in Teaching in the ECE Department, CMU and the Carnegie Institute of Technology's Dowd Fellowship for educational contributions and he was a co-recipient of the 2008 Outstanding Faculty Research Award in CMU’s College of Engineering. He is a fellow of IEEE, SPIE, OSAA, IAM and NAI.

Ping Jia received his B.Eng and MSc degree in computer science from the University of Science and Technology of China, Hefei and his Ph.D. from the Graduate University of the Chinese Academy of Sciences. He is the President of Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences. His current research interests include image processing, computer vision, and optical engineering.

Jane You received the Ph.D. degree from La Trobe University, Melbourne, VIC, Australia, in 1992. She is currently a Professor with the Department of Computing, Hong Kong Polytechnic University, Hong Kong, and the Chair of Department Research Committee. She has researched extensively in the fields of image processing, medical imaging, computer-aided diagnosis, and pattern recognition. She has been a Principal Investigator for one IIT project, three GRF projects, and many other joint grants since she joined PolyU in 1998. Prof. You was a recipient of three awards including Hong Kong Government Industrial Awards, the Special Prize and Gold Medal with Jurry's Commendation at the 39th International Exhibition of Inventions of Geneva in 2011 for her current work on retinal imaging, and the Second Place in an International Competition [SPIE Medical Imaging’2009 Retinopathy Online Challenge in ROC’2009]. Her research output on retinal imaging has been successfully led to technology transfer with clinical applications. She is an Associate Editor of Pattern Recognition and other journals.