J Inf Process Syst, Vol.16, No.1, pp.6~29, February 2020
https://doi.org/10.3745/JIPS.02.0129

ISSN 1976-913X (Print)
ISSN 2092-805X (Electronic)

JOURNAL OF INFORMATION PROCESSING SYSTEMS JIPS

Multimodal Biometrics Recognition from Facial Video with Missing Modalities Using Deep Learning

Sayan Maity*, Mohamed Abdel-Mottaleb**, and Shihab S. Asfour**

Abstract

Biometrics identification using multiple modalities has attracted the attention of many researchers as it produces more robust and trustworthy results than single modality biometrics. In this paper, we present a novel multimodal recognition system that trains a deep learning network to automatically learn features after extracting multiple biometric modalities from a single data source, i.e., facial video clips. Utilizing different modalities, i.e., left ear, left profile face, frontal face, right profile face, and right ear, present in the facial video clips, we train supervised denoising auto-encoders to automatically extract robust and non-redundant features. The automatically learned features are then used to train modality specific sparse classifiers to perform the multimodal recognition. Moreover, the proposed technique has proven robust when some of the above modalities were missing during the testing. The proposed system has three main components that are responsible for detection, which consists of modality specific detectors to automatically detect images of different modalities present in facial video clips; feature selection, which uses supervised denoising sparse auto-encoders network to capture discriminative representations that are robust to the illumination and pose variations; and classification, which consists of a set of modality specific sparse representation classifiers for unimodal recognition, followed by score level fusion of the recognition results of the available modalities. Experiments conducted on the constrained facial video dataset (WVU) and the unconstrained facial video dataset (HONDA/UCSD), resulted in a 99.17% and 97.14% Rank-1 recognition rates, respectively. The multimodal recognition accuracy demonstrates the superiority and robustness of the proposed approach irrespective of the illumination, non-planar movement, and pose variations present in the video clips even in the situation of missing modalities.

Keywords

Auto-encoder, Deep Learning, Multimodal Biometrics, Sparse Classification

1. Introduction

Several factors, e.g., changes in illumination and viewing direction, affect the accuracy and robustness of unimodal face biometrics [1-4]. To overcome these limitations, fusion of different modalities has been used in the literature to obtain robust and accurate recognition results.

There are several motivations for building robust multimodal biometric systems that extract multiple modalities from a particular source of biometrics, i.e., facial video clips. Firstly, acquiring facial video clips data is straight forward using conventional video cameras, which are ubiquitous. Secondly, the nature of data collection is non-intrusive and the ear, frontal, and profile face can appear in the same

Manuscript received April 26, 2018; first revision August 20, 2018; accepted March 23, 2019.

Corresponding Author: Sayan Maity (s.maity1@umail.miami.edu)
* Dept. of Industrial Engineering, University of Miami, Coral Gables, FL, USA (s.maity1@umail.miami.edu, ssasfour@miami.edu)
** Dept. of Electrical and Computer Engineering, University of Miami, Coral Gables, FL, USA (mottaleb@miami.edu)

video. Thirdly, in a multimodal biometric identification system, it is expected to encounter missing modalities when working with video data. Various modalities, e.g., frontal face, left ear, right ear, left profile face, and right profile face might exist in the training video clips. If the test data does not contain all the modalities during the classification, we should be able to perform multi-modal classification using the available modalities.

In this work, we proposed a novel multimodal biometrics methodology to efficiently recognize subjects from facial video surveillance data irrespective of the multiple constraints, such as illumination, pose variations, and non-planar movement existing in the face surveillance data. Unlike facial videos recorded under a constrained environment, facial video clips collected in unconstrained environments contain significant head pose variations due to non-planar movements. Moreover, detected frames of the same modality from unconstrained facial video clips contain a high degree of non-planar rotation variabilities compared with the constrained counterpart. This makes unconstrained facial video clips more challenging to adequately extract information for efficient recognition.

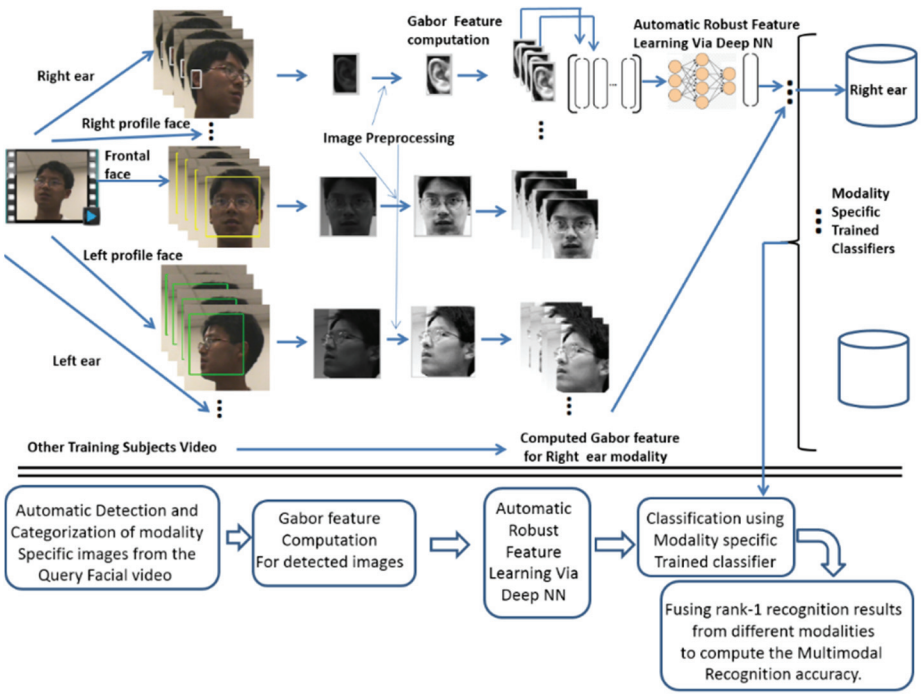


Fig. 1. System block diagram: multimodal biometrics recognition from facial video.

The proposed methodology, shown in Fig. 1, entails three distinct components to perform the task of efficient multimodal recognition from facial video clips. First, the automatic detection of modality specific regions from the video frames been performed by adopting the detection framework of Viola and Jones [5]. Unconstrained facial video clips contain significant head pose variations due to non-planar movements, and sudden changes in facial expressions. This results in an uneven number of detected modality specific video frames for the same subject in different video clips, and also a different number of modality specific images for different subject. From the aspect of building a robust and accurate model, it is always preferable to use the entire available training data. However, classification through sparse

representation (SRC) is vulnerable due to the existence of uneven count of modality specific training data for different subjects. Thus, to overcome the vulnerability of SRC while using all of the detected modality specific regions, in the model building phase we train supervised denoising sparse auto-encoder to construct a mapping function. This mapping function is used to automatically extract the discriminative features preserving the robustness to the possible variances using the uneven number of detected modality specific regions. Therefore, by applying deep learning network as the second component in the pipeline results in an equal number of training sample features for the different subjects. Finally, using the modality specific recognition results, score level multimodal fusion is performed to obtain the multimodal recognition result.

Due to the unavailability of proper datasets for multimodal recognition studies [6], often virtual multimodal databases are synthetically obtained by pairing modalities of different subjects from different databases. To the best of the authors' knowledge, the framework presented in this manuscript is the first work where multiple modalities are extracted from a single data source that belongs to the same subject. There are a very few studies in biometrics recognition literature that deal with substantial head pose variation in facial video clips. It may however be noted that majority of the previous studies were aimed to overcome the particular variabilities, e.g., expression, viewing angle, and illumination, in different facial images by applying individual transformations. The major contributions of the presented framework is the application of training a deep learning network for automatic feature learning in multimodal biometrics recognition using a single source of biometrics i.e., facial video data, irrespective of the various constraints, e.g., illumination, pose variations, and non-planar movement existing in the face surveillance data.

The rest of this manuscript is structured as follows: Section 2 analyses the related work. Section 3 details the modality specific frame detection from the facial video clips. Section 4 describes the automatic feature learning using supervised denoising sparse auto-encoder (deep learning). Section 5 presents the modality specific SRC and multimodal fusion. Section 6 provides the experimental results on the facial video datasets collected in constrained (WVU) [7], and the unconstrained (HONDA/UCSD) [8] environment, demonstrating performance of the presented framework. Finally, conclusion of the research with future potential to advance the proposed framework are described in Section 7.

2. Related Work

2.1 Multimodal Recognition

Research in face recognition domain been active during the past few decades [9-14]. Although majority of the study on face recognition is using 2D images or 3D data, there are few publications that address video-based face recognition [15-18]. In [15], face images extracted from the training video clips are used to build a dictionary where face images of the same subject with variations in illumination, viewing angle, and facial expression, reside on the same nonlinear sub-manifold. Later, the learned dictionary is used to recognize faces from query video clips. Lee et al. [16] proposed probabilistic appearance manifolds, a spatiotemporal manifold model, which computes the transition probabilities between the subspaces. Given a query video, the probabilistic appearance manifold algorithm locates the operating part of the manifold to identify the subject. In [17], a view synthesis method is proposed reconstructing 3D frontal face model using many non-frontal 2D face images obtained from training video frames. Later,

the synthesized frontal face image is used to match against the frontal face image extracted from the query video. In [18], decisions from multiple face matchers are adaptively fused in improving recognition accuracy using facial video. Lumini and Nanni [19] presented an article listing the state-of-the-art methodologies on fusing information in the multimodal biometric recognition.

The ear, though a comparatively new area of biometric research, owns multiple inherent characteristics, e.g., it nearly maintains its shape with aging, and is not at all affected by facial expressions, which makes its use beneficial [20]. Because of these advantages, several researchers built multimodal ear and face biometric systems [21-23]. In [24], the authors presented the advantage of using profile face, side view of the face including the ear, which provides discriminative information for human recognition. In [23], the authors presented a feature-fusion framework incorporating kernel Fisher discriminant analysis (KFDA) on 2D images, later utilize it for profile face- and ear-based recognition. Kisku et al. [22] presented a multimodal biometric framework to fuse 2D ear and facial biometrics using Dempster-Shafer decision theory. In [21], the authors incorporated eigen ear and face techniques to build a multimodal framework using 2D profile face and ear images. A sparse representation based multimodal biometric system is proposed in [6]. It fuses ear and face at the feature level, where the fusion weights are determined by computing the reliability of each modality.

2.2 Deep Learning in Biometrics

Recently, deep learning of artificial neural networks (ANN) has been used in several biometric authentication research studies. Ngiam et al. [25] presented a deep network based unsupervised feature learning for audio-visual speech classification. The features obtained from the audio and video data is used to learn the latent relationship of the lip pose and motions in the video with the articulated phonemes in the audio. Different variants of convolutional neural network [12,26,27] have been used to design face verification systems. A face verification framework using convolutional neural network based Siamese network is presented in [27]. In [12], a facial verification system using convolutional neural network was presented which considerably outperforms the existing systems on the LFW dataset. The above-cited research articles are proposed for face verification, whereas our proposed approach deals with multimodal recognition in which, given a test video, it identifies the subject among many.

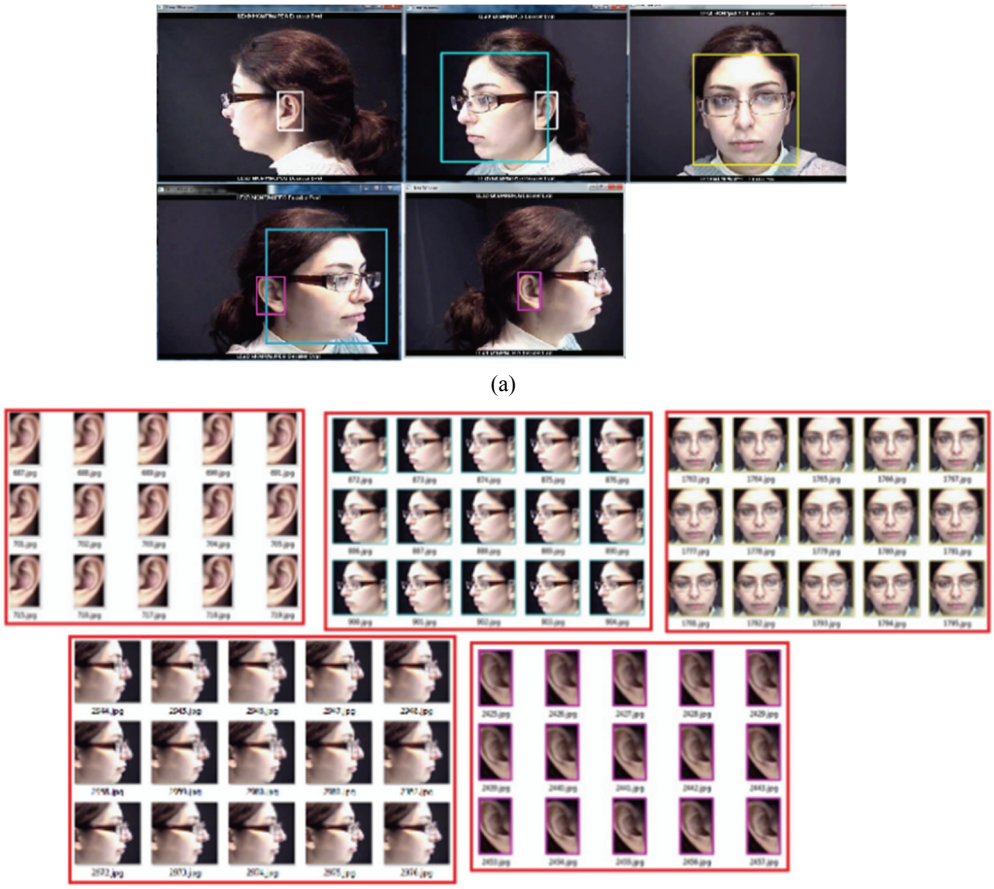
Goswami et al. [28] proposed MDLFace, a memorability-based frame selection technique that assists automatic selection of memorable frames for facial feature extraction and comparison, by using a deep learning algorithm. In [28], the deep learning algorithm is trained to identify the memorable faces, certain face images that can be more accurately remembered by human subjects compared to other faces, to resemble the human perception in face recognition. In [29], the multimodal biometrics (face, iris, fingerprint) anti-spoofing framework is presented using deep neural network. A stacked supervised auto-encoders based single sample face recognition technique is proposed in [30], which achieves considerably better accuracy compared to other DNN framework, such as Lambertian network.

3. Modality Specific Image Frame Detection

To perform multimodal biometric recognition, we first need to detect the image frames of the various modalities from the facial video. The facial video clips in the constrained dataset are collected in a controlled environment, where the camera rotates around the subject's head. The video frame sequences

begin with the left face profile, i.e., 0º, then proceed towards right face profile up to 180º rotation, contains image frames of various modalities, e.g., frontal face, left profile face, right profile face, left ear, and right ear, respectively. Sequences in the unconstrained facial video dataset contains uncontrolled and non-uniform head rotations and changing facial expressions. Thus, the appearance of a specific modality in a certain video frame of the unconstrained clip is random compared with the constrained video clips.

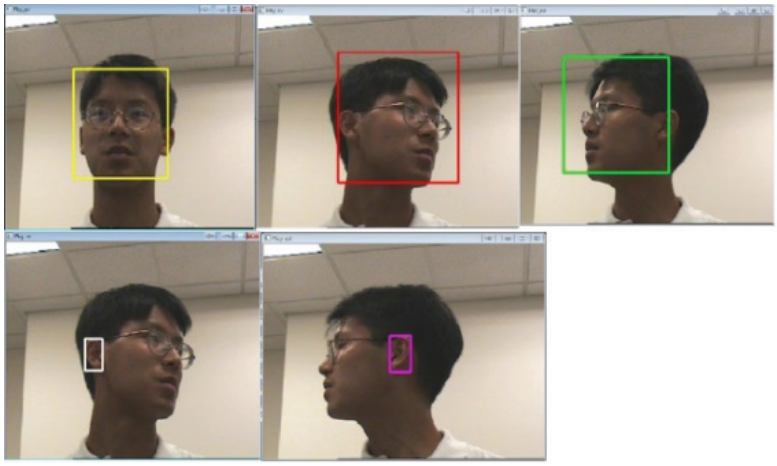
The algorithm was trained to detect the different modalities that appear in the facial video clips. To automate the detection process of the modality specific image frames, we adopted the AdaBoost detection framework [5]. We trained this detection framework to detect frontal and profile faces in the video frames, respectively, using manually cropped frontal face images from color FERET [31] database, and face profile images from the University of Notre Dame (UND) Collection J2 dataset. Moreover, it is trained using ear images in UND [32] color ear dataset to detect ear images in the video frames. By using these modality specific trained detectors, we can detect faces and ears in the video frames. The modality specific trained detectors are utilized in detecting face and the ear regions in the video frames. Figs. 2 and 3 consist examples of detection results from the constrained and unconstrained facial video dataset.



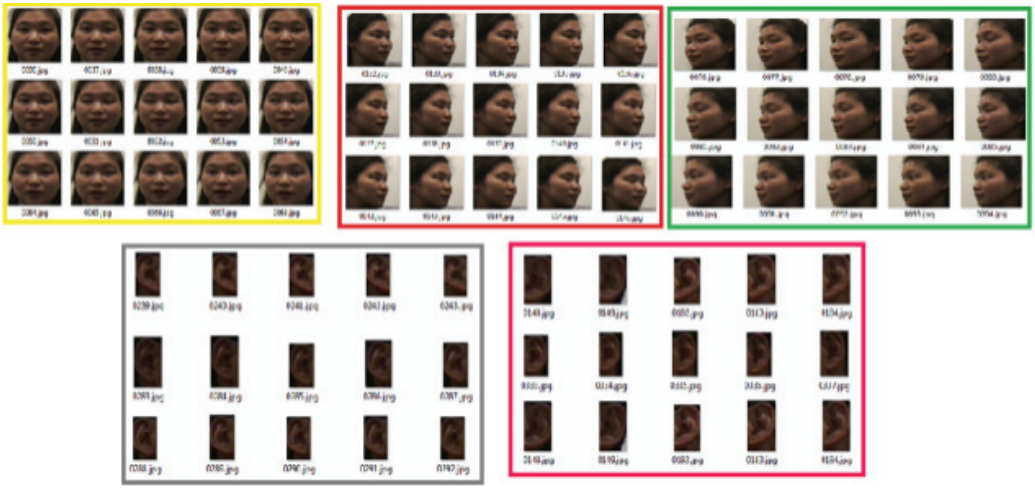
(a)

(b)

Fig. 2. Modality specific image frame detection for constrained facial video clips. (a) Automatic detection of image frames in WVU facial video clips using modality specific trained cascade classifier. (b) Categorized detected regions from WVU facial video clips into modality specific groups from a video sequence.



(a)



(b)

Fig. 3. Modality specific image frame detection for unconstrained facial video clips. (a) Automatic detection of image frames in HONDA facial video clips using modality specific trained cascade classifier. (b) Categorized detected regions from HONDA facial video clips into modality specific groups from a video sequence.

Performance of the modality specific detection for the constrained face video clip is highly accurate. However, due to the uncontrolled head movements and non-planar rotation present in the unconstrained dataset, the detection results are not as accurate and there are few false positives. The detection accuracies on the unconstrained facial database is listed in Table 1.

Before using these detected modality specific regions from the video frames for extracting features, some pre-processing steps are performed. The facial video clips recorded in the unconstrained environment contain variations in illumination and low contrast. The contrast of the images is enhanced through histogram equalization. Finally, all detected modality specific regions from the facial video clips were resized; ear images were resized to 110×70 pixels and faces (frontal and profile) were resized to 128×128 pixels.

Table 1. Detection accuracy for unconstrained video clips

Modality	Detection accuracy (%)
Frontal face	97.55
Left profile face	93.42
Right profile face	92.21
Left ear	98.77
Right ear	98.84

4. Automatic Feature Learning Using Deep Neural Network

Even though the modality specific sparse classifiers result in relatively high recognition accuracy on the constrained face video clips, the accuracy suffers in case of unconstrained video because the sparse classifier is vulnerable to the bias in the number of training images from different subjects. For example, subjects in the HONDA/UCSD dataset [8] randomly change their head pose. This results in a non-uniform number of detected modality specific video frames across different video clips, which is not ideal to perform classification through sparse representation.

In the subsequent sections we first describe the Gabor feature extraction technique. Then, we describe the supervised denoising sparse auto-encoders, which we use to automatically learn equal amount of feature vectors for each subject from the uneven quantity of modality specific detected regions.

4.1 Feature Extraction

Two-dimensional Gabor filters [33] are used in broad range of applications [34,35] to extract scale and rotation invariant feature vectors. In our feature extraction step, uniform down-sampled Gabor wavelets are computed for the detected regions using Eq. (1), as proposed in [36]:

$$\psi_{\mu,\nu}(z) = \frac{\|k_{\mu,\nu}\|^2}{s^2} e^{\left(\frac{-\|k_{\mu,\nu}\|^2 \|z\|^2}{2s^2}\right)} \left[e^{ik_{\mu,\nu}z} - e^{\frac{s^2}{2}}\right] \quad (1)$$

where $z = (x, y)$ represents each pixel in the 2D image, $k_{\mu,\nu}$ is the wave vector, which can be defined as $k_{\mu,\nu} = k_\nu e^{i\phi_u}$, $k_\nu = \frac{k_{max}}{f^\nu}$, k_{max} is the maximum frequency, and f is the spacing factor between kernels in the frequency domain, $\phi_u = \frac{\pi\mu}{2}$, and the value of s determines the ratio of the Gaussian window width to wavelength. Using Eq. (1), Gabor kernels can be generated from one filter using different scaling and rotation factors. In this paper, we used five scales, $\nu \in 0, \ldots, 4$ and eight orientations $\mu \in 0, \ldots, 7$. The other parameter values used are $s = 2\pi$, $k_{max} = \frac{\pi}{2}$, and $f = \sqrt{2}$.

Before computing the Gabor features, all detected ear regions are resized to the average size of all the ear images, i.e., 110×70 pixels, and all face images (frontal and profile) are resized to the average size of all the face images, i.e., 128×128 pixels. Gabor features are computed by convolving each Gabor wavelet with the detected 2D region, as follows:

$$C_{\mu,\nu}(z) = T(z) * \psi_{\mu,\nu}(z) \quad (2)$$

where $T(z)$ is the detected 2D region, and $z = (x, y)$ represents the pixel location. The feature vector is constructed out of $C_{\mu,\nu}$ by concatenating its rows.

4.2 Classical Sparse Auto-encoder

Deep learning is a suite of machine learning techniques, where multiple layers of information processing phases in hierarchical architectures are utilized for pattern analysis/classification. There are different deep learning architectures available in the literature. The available deep learning architectures can be categorized broadly into three major classes: convolution neural network (CNN), recurrent neural network (RNN), and deep auto-encoder. CNNs are neural network with local and global connectivity structure consist of multiple stages of feature extractors. CNNs are used in recognizing various images/scenes, video content analysis, natural language processing applications, etc. RNN contains feed-back connection, thus the activations can flow round in a loop. This phenomenon enables the networks to do temporal processing and learn sequences, e.g., perform sequence recognition/reproduction or temporal prediction/association. RNNs are used in speech recognition, video captioning, word prediction, translation applications, etc. Thus, we can see none of the CNN or RNN architectures are suitable for the automatic feature extraction. However, in the deep auto-encoder architecture the output target itself is the data input, usually pre-trained with deep belief network or using distorted training data to regularize the learning. In this subsection we describe the sparse auto-encoder algorithm [37], which is one of the approaches to learn features from unlabeled data automatically.

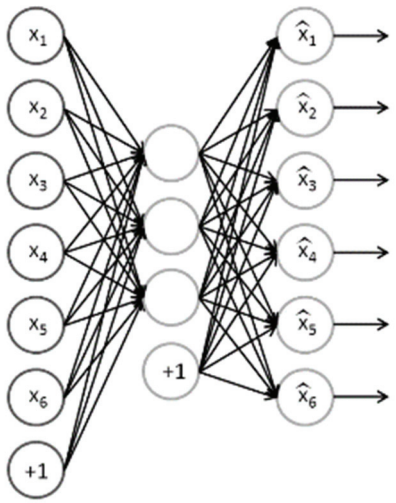


Fig. 4. Structure of an auto-encoder.

The application of ANN's to supervised learning [38] is well proven in variety of applications including speech recognition, computer vision such as self-driving car. An auto-encoder network is an unsupervised learning algorithm, one of the commonly used building blocks in deep neural networks, which applies backpropagation to set the target values to be equal to the inputs. The weights of each layer is adjusted by the reconstruction error between the input and the output of the network. As shown in Fig. 4, an auto-encoder tries to learn a function $x_i = \hat{x}_i$, where x_i belongs to unlabeled training samples

set $\{x_{(1)}, x_{(2)}, x_{(3)}, \ldots, x_{(n)}\}$, and $x_i \epsilon\ \mathbb{R}^n$. In other words, it is trying to learn an approximation to the identity function, to produce an output $\hat{x}$ that is similar to x, in two subsequent stages: (i) an encoder, maps the input *x* to the nodes in the hidden layer using some deterministic mapping function *f*: $h = f(x)$, then (ii) a decoder, maps the hidden nodes back to the original input space through another deterministic function *g*: $\hat{x} = g(h)$. For real-valued input the parameters of encoder and decoder can be learned by minimizing the reconstruction error $\|x - g(f(x))\|^2$. This simple auto-encoder often resembles learning a low-dimensional representation alike Principal Component Analysis (PCA) [39]. However, it has been proven in [40] that such a nonlinear auto-encoder is different from PCA, also training an auto-encoder results in minimizing the reconstruction error and maximizing a lower bound on the mutual information between the input and the learned representation.

In Fig. 4, the number of hidden units can be increased, i.e., the number of hidden nodes can be made even greater than the number of input nodes. In this case, we can learn some inherent structure of the data by imposing a sparsity constraint on the network. In other words, if we think of a neuron as being "active" if its output value is close to 1, or as being "inactive" if its output value is close to 0, we would like to constrain the neurons to be inactive most of the time. Recent research progress in biology reveals that the percentage of the activated neurons of human brain at a specific time is around 1% to 4% [41]. Thus, sparsity constraint on the activation of the hidden layer is frequently applied in the auto-encoder based neural networks. Recent research proven that that sparse auto-encoder usually achieves better performance than that trained without the sparsity constraint [37].

4.3 Denoising Auto-encoder

Denoising auto-encoder (DAE) [42] is a more generalized and robust version of the classical auto-encoder. Since it assumes that the input data contain noise, it is suitable for learning features from data with noise. In other words, DAE is trained to reconstruct a repaired or clean version of the input from a corrupted or noisy one. It is proven that compared to conventional auto-encoders, DAEs are capable to acquire Gabor-like edge detectors from image patches.

In [42], DAE is designed and effectively tested to address different real-world scenario where noise can corrupt the input data. The original input data $x \epsilon\ \mathbb{R}^n$ can be affected by (a) additive isotropic Gaussian noise $(\tilde{x}|x \sim \mathcal{N}(x, \sigma^2 I)$, (b) masking noise, i.e., a fraction of randomly chosen *x* is forced to 0, and (c) Salt-and-pepper noise, i.e., a fraction of randomly chosen *x* is forced to 0 or 1. The corrupted data is used as the input of the encoder, i.e., the encoding of DAE is obtained by a nonlinear transformation function:

$$h = f_e(\tilde{x}) = f_e(W\tilde{x} + b_e) \tag{3}$$

where $h \epsilon\ \mathbb{R}^y$ represent the output of the hidden layer and also be known as feature representation or code, *y* is the number of hidden layer units, $W \epsilon\ \mathbb{R}^{yXn}$ is weights for the input-to-hidden layer, b_e signifies the bias, stands for the hidden layer input, and f_e is the hidden layer activation function. The reconstruction of DAE or decoding is obtained by utilizing a mapping function g_d:

$$\hat{x} = g_d(h) = g_d(W'h + b_d) \tag{4}$$

where $\hat{x} \epsilon \mathbb{R}^z$ is the output of DAE, which is also the robust reconstruction of the corrupted original data $\tilde{x}$. The output layer contains equal number of nodes as the input layer. $W' = W^T$ known as tied weights. DAE incorporate reconstruction-oriented training, in other words training the network by imposing constraint on the output data $\hat{x}$ to reconstruct the noisy input data $\tilde{x}$. Thus, the objective function or cost function is the reconstruction error as follows:

$$\min_{W,W',b_e,b_d} \sum_{x \epsilon X} L(x, \hat{x}) \tag{5}$$

where L represents reconstruction error: when the values of input x range from 0 to 1 it's cross-entropy function, and squared error $L(x, \hat{x}) = \|x - \hat{x}\|^2$ for real-valued inputs, is used. Quantitative experiments show that even when the fraction of corrupted pixels, e.g., as corrupted by zero masking noises, reaches 55%, the recognition accuracy is still better or comparable with that of a network trained without corruptions.

4.4 Supervised Stacked Denoising Auto-encoder

To obtain feature values, those are not affected by changes in viewing angle, pose, illumination etc., from modality specific image regions, we adopted the supervised auto-encoder [30]. The supervised auto-encoder is trained using features extracted from image regions $(\hat{x_i})$ containing variations in illumination, viewing angle and pose whereas the features of selected image regions, (x_i), with similar illumination and without pose variations are utilized as the target. By minimizing the objective criterion given in Eq. (6) (subject to, the modality specific features of the same person are similar), the supervised auto-encoders learn to capture the modality specific robust representation.

$$min_{W,b_e,b_d} \frac{1}{N} \sum_i (\|x_i - g(f(\hat{x_i}))\|_2^2 + \lambda \|f(x_i) - f(\hat{x_i})\|_2^2) \tag{6}$$

where h represent output of the hidden layer, is defined as $h = f(x) = \tanh(Wx + b_e)$, $g = h(x) = \tanh(W^T h + b_d)$, N is the total number of training samples, and λ is the weight preservation term. The first term in Eq. (6) minimize the reconstruction error, i.e., after passing through the encoder and the decoder, the variations (illumination, viewing angle and pose) of the features extracted from the unconstrained images will be repaired. The second term in Eq. (6) enforces the similarity of modality specific features corresponding to the same person.

Stacking supervised DAEs to initialize a deep network follows the procedure of stacking restricted Boltzmann machines (RBMs) in deep belief networks [43-45]. It is worth noting that the corrupted/noisy input is only used for the initial denoising-training of each individual layer so that it may learn useful feature extractors. After training a first level DAE, the learned encoding function f_{e1} can be used on clean input for reconstruction. The resulting representation is used to train a second level DAE to learn a second level encoding function f_{e2}. This procedure can be repeated to stack the trained DAE layer-by-layer to form a stacked denoising auto-encoder (SDAE). Fig. 5 represent a conventional SDAE structure, this contains two encoding layers and two decoding layers. In the encoding section, the output of the first encoding layer works as the input data of the second encoding layer.

After training a stack of encoders its highest level output representation can be used as input to a stand-

alone supervised learning algorithm. A logistic regression (LR) layer was added on top of the encoders as the final output layer [46], which enable the deep neural network to perform supervised learning. By performing gradient descent on a supervised cost function, the supervised SDAE automatically learned fine-tuned network weights. Thus, the parameters of the entire SDAE network are attuned to minimize supervised target (e.g., class labels) prediction error. It is worthwhile to mention that SDAE is unsupervised where LR is supervised and the data contained labelled information can only be used in LR stage. The supervised SDAE network shown in Fig. 6, which represents a two-category classification architecture. As per [46], to produce the initial features the decoding part of SDAE is removed and only the encoding part of SDAE is retained. Additionally, the output layer of the entire network (LR layer), is added.

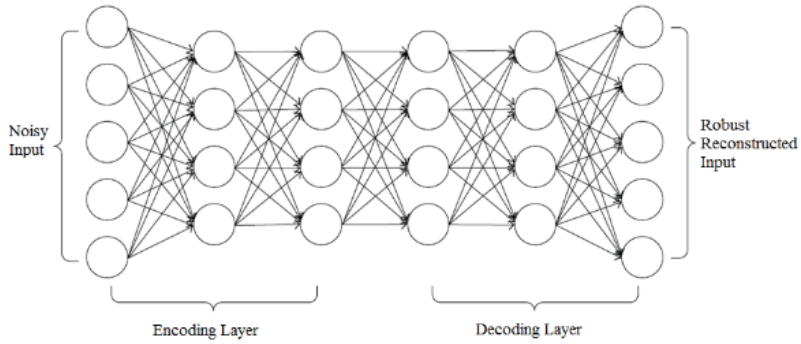


Fig. 5. Stacked denoising auto-encoder network.

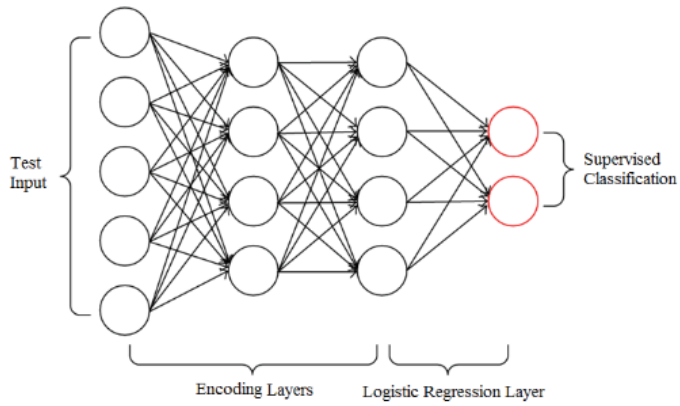


Fig. 6. Supervised stacked denoising auto-encoder.

4.5 Training the Deep Learning Network

In this subsection we will describe the constraints we faced while training the SDAE using the layer-wise greedy learning algorithm, and application of the supervised fine-tuning to minimize the error of predicting the supervised target.

Empirically, deep networks were generally found to be not better, and often worse, than conventional neural networks [42]. A reasonable explanation is that gradient-based optimization often get stuck near poor solutions.

An approach that has been explored and proved successful to train deep networks with more than two hidden layers, is based on constructively adding layers [47], using a supervised criterion at each stage. However, it requires having an extensive training dataset to achieve generalization and avoid overfitting. In our application, the technique of constructively adding layers did not perform well because of the relatively small number of training samples. Moreover, we need to initialize the weights in a region near a good local minima, to better generalize the internal representations of the data.

Thus, we adopt the two-stage training of the deep learning network, where we have a better initialization to begin with and a fine-tuned network weights that lead us to a more accurate high-level representation of the dataset. The steps of two-stage deep learning network training are as follows:

Step 1. SDAEs are used to train the initial network weights one layer at a time in a greedy fashion using deep belief network (DBN).

Step 2. The initial weights of the deep learning network are initialized using the learned parameters from DBN.

Step 3. Labelled training data are used as input, and their predicted classification labels obtained from the LR layer along with the initial weights are used to apply backpropagation on the SDAE.

Step 4. Back propagation is applied on the network to optimize the objective function (given in Eq. (5)), results in fine-tune the weights and bias for the entire network.

Step 5. Finally, the learned network weights and bias are used to extract image features to train the sparse classifier.

4.5.1 Layer-wise greedy learning

Deep multi-layer ANN's have numerous levels of non-linearities associated with them to efficiently symbolize the highly varying nonlinear functions through a compact representation. However, until recent days it was not obvious how to effectively train such deep networks since gradient-based optimization starting from random initialization often get stuck in local optima resulting in poor solutions. Hinton et al. [48], recently proposed a greedy layer-wise unsupervised learning algorithm for DBN, a generative model with numerous layers of hidden causal variables. Later on, in [46], a variant of the greedy layer-wise unsupervised learning is proposed to extend it to scenarios where inputs are continuous.

In a DBN, x is the input, g^i represent hidden variables in layer i, then the computation of probability and sampling can be represented by the joint distribution:

$$P(x, g^1, g^2, \dots, g^l) = P(x|g^1)P(g^1|g^2) \dots P(g^l - 2|g^l - 1)\, P(g^l - 1, g^l) \tag{7}$$

all the conditional layers $P(g^i|g^{i+1})$ represents the factorized conditional distributions. In Hinton et al. [48], the hidden layer g^i is used as a binary random vector with n^i elements, g_j^i

$$P(g^i|g^{i+1}) = \prod_{j=1}^{n^i} P\left(g_j^i|g^{i+1}\right) \tag{8}$$

where

$$P(g^i = 1|g^{i+1}) = sigm\,(b_j^i + \textstyle\sum_{k=1}^{n^{i+1}} W_{kj}^i g_k^{i+1}) \tag{9}$$

where $sigm(t) = \frac{1}{1+e^{-t}}$, the b_j^i are biases for unit j of layer i, and W^i is the weight matrix for layer i. If we set $g^0 = x$, the generative model for the first layer $P(x|g^1)$ will follow Eq. (7).

DBN can be utilized for generatively pre-training a deep neural network where the initial weights are

the learned weights [48]. A DBN can be efficiently trained in an unsupervised, layer-by-layer manner where the layers are typically made of RBM [49]. RBM is a generative stochastic ANN that can learn a probability distribution over the set of inputs.

It should be noted that 1-level DBN is equivalent to an RBM. The greedy layer-wise strategy to add multiple layers in the DBN follows this same methodology. Train the first layer as an RBM that models the raw input $x = g^0$ as its visible layer. Then, use the first layer to obtain the mean activations $P(g^1 = 1|g^0)$ of the input, which will be used as input data for the second layer. Train the second layer as an RBM $P(g^0, g^1)$, taking the transformed data (mean activations) as input to the visible layer of that RBM. Iterate the same steps to add the $(l + 1)^{th}$ level, after training the top-level RBM with l level DBN, such that, the distribution $P(g^l, g^l - 1)$ from the RBM associated with layers (l - 1) and l is kept as part of the DBN generative model. In training a single RBM, the following equation represent the weight updates using gradient ascent:

$$\Delta w_{ij}(t+1) = w_{ij}(t) + \eta \frac{\delta \log(p(v))}{\delta w_{ij}} \tag{10}$$

where $p(v)$ represent the probability of a visible vector and η is the learning rate, given by:

$$p(v) = \frac{1}{Z} \sum_h e^{-E(h,v)} \tag{11}$$

In Eq. (11), Z (used for normalizing) is the partition function and $E(h, v) = -h'Wv - b'v - c'h$, is the energy function assigned to the state-of-the-art network. Here, v stands for visible units and hidden layer activations h stands for hidden units. Computation of stepwise weight updates is explained in Algorithm 1, where b and c respectively represent the vector of biases for visible units and the hidden units.

Algorithm 1. Stepwise weight update of the DBN

1. Initialization: Visible units to training vector.
2. Update: hidden units in parallel given the visible units: $p(h_j = 1|V) = sigm(b^j + \sum_i v_i W_{ij})$
3. Update: visible units in parallel given the hidden units: $p(v_i = 1|H) = sigm(c^i + \sum_j h_j W_{ij})$ ("Reconstruction" step.)
4. Re-update the hidden units in parallel given the reconstructed visible units following the same equation as step 2.
5. Weight update by following: $\Delta w_{ij}\ \boldsymbol{\alpha}\ \langle v_j h_j \rangle_{data} - \langle v_j h_j \rangle_{reconstruction}$

4.5.2 Supervised fine-tuning

After all layers are pre-training completes the network perform the second phase of training for fine-tuning. This supervised fine-tuning is performed to minimize the overall prediction error of the entire deep learning network. To achieve this, a LR layer (or in generic scenario a soft-max regression classifier) is added on top of the network [46]. Later, train the entire network as we would train a multi-layer perceptron, where the encoding parts of each auto-encoder are used. This stage is supervised since now we use the target class during training.

The network represented in Fig. 6, symbolized a two-category classification problem, with two output classes, where the decoding part of SDAE is removed while the encoding part of SDAE is retained to produce the initial features. Also, the output layer of the whole network, known as LR layer, is added. Sigmoid function is incorporated as activation function in the LR layer:

$$h(x) = \frac{1}{e^{-Wx-b}} \tag{12}$$

where, x represents output of the last encoding (y^l) layer, in other words features are pre-trained by the SDAE network. The output of the sigmoid function ranges between 0 and 1, denotes the classification results in case of two-class classification problem. Thus, we can utilize the errors between the true labels and the predicted classification results associated with the training data points to fine-tune the whole network weights. The cost function can be defined following the cross-entropy function:

$$Cost = -\frac{1}{m}\left[\sum_{i=1}^{m} l^i \log\left(h(x^i)\right) + (1-l^i)\log\left(1-h(x^i)\right)\right] \tag{13}$$

where l^i represents the label (x^i) of the sample. We update the network weights by minimizing the cost function.

5. Modality Specific and Multimodal Recognition

The modality specific sub-dictionaries (d_j^i) contain feature vectors generated by deep learning network using the modality specific training data of each individual subject; where *i* represents the modality, $i \in 1,2,\dots,5$; and *j* stands for the number of training video sequence.

Later, we concatenate the modality specific learned sub-dictionaries (d_j^i) of all the subjects in the dataset to obtain the modality specific (e.g., frontal face, left profile face, right profile face, left ear, and right ear) dictionary D_i, as follows.

$$D_i = \left[d_1^i; d_2^i; \dots; d_j^i\right]; \forall i \in 1,2,\dots,5 \tag{14}$$

5.1 Sparse Representation for Classification

For each training video sequence, the modality specific sub-dictionaries $d_j^i \in \mathbb{R}^p$, are formed using the feature vectors generated by deep learning network utilizing the modality specific detected regions of the *j*th training video sequence, where p is the length of the feature vectors learned by the DNN. Similarly, features learned by the DNN, $y^i \in \mathbb{R}^p$, using modality specific detected regions in the test video, is then represented as a linear combination of the feature vectors learned from the training video sequences:

$$y^i = d_1^i * \alpha_1^i + d_2^i * \alpha_2^i + \dots + d_j^i * \alpha_j^i \tag{15}$$

where $\alpha_j^{i\prime}s$ are the coefficients corresponding to the training data of the *i*th modality in the *j*th training video sequence. Eq. (15) can be represented by using the concatenated modality specific dictionary D_i, defined in Eq. (14), as:

$$y^i = D_i x \in \mathbb{R}^p \tag{16}$$

where x is the coefficient vector, and the test data y^i belongs to the ith modality. In our approach we used Smoothed l_0 (SL0) [49] norm to solve Eq. (16). SL0 algorithm is utilized to obtain the sparsest solution of under determined systems of linear equations by directly minimizing the l_0 norm. SL0 has proven to be more efficient than l_0 *and* l_1 in space and time complexity [49].

Using majority voting on the sparse classification coefficients obtained from the individual sub-dictionaries for all the modality specific regions detected from a specific test video, the modality specific classification decisions are made. Later, the final classification of the subject present in the video sequence is made based upon the score level fusion of the modality specific classification. Some of the modalities may not be available in the video used for recognition, in these cases the available modalities will be used to generate the Rank-1 match. We tested the algorithm when all the modalities are available during the recognition phase and also all possible combinations of missing modalities, i.e., 1, 2 or 3 modalities are absent, detailed in the experimental section.

5.2 Multimodal Recognition

The recognition results from the five modalities—frontal face, left profile face, right profile face, left ear, and right ear—are combined using score-based fusion. Score-based fusion possess flexibility of fusing various modalities upon their availability. To prepare for fusion, the matching scores obtained from the different matchers are transformed into a common domain using a score normalization technique. Later, the weighted-sum technique is used to fuse the results at the score level. We have adopted the *Tanh* score normalization technique [50], which is both robust and efficient, defined as follows:

$$s_j^n = \frac{1}{2}\left\{tanh\left(0.01\left(\frac{s_j - \mu_{GH}}{\sigma_{GH}}\right)\right) + 1\right\} \tag{17}$$

where s_j and s_j^n are the match scores before and after normalization, respectively. μ_{GH} and σ_{GH} are the mean and standard deviation estimates of the genuine score distribution given by Hampel estimators [51], respectively. Hampel's estimators are based on the influence functions ψ which are odd function and can be defined for any x (matching score, s_j, in this paper) as follows:

$$\psi = \begin{cases} x, & 0 \le |x| < a, \\ a\ sign(x), & a \le |x| \le b, \\ \frac{a(r-|x|)}{r-b} sign(x), & b \le |x| \le r, \\ 0, & r \le |x|, \end{cases} \tag{18}$$

where

$$sign(x) = \begin{cases} +1, & if\ x \ge 0 \\ -1, & otherwise \end{cases} \tag{19}$$

In Eq. (18), the value of a, b, and r in ψ, reduces the influence of the scores at the tails of the distribution during the estimation of the location and scale parameters. The standardized match accuracy scores later fused utilizing weighted-sum technique:

$$S_p = \sum_{i=1}^{M} w_i * s_i^n \quad (20)$$

where w_i and s_i^n are the weight and normalized match score of the ith modality specific classifier, respectively, such that $\sum_{i=1}^{M} w_i = 1$. In this study, the weights $w_i, i = 1, 2, 3, 4, 5$; correspond for the frontal face, left profile face, right profile face, left ear, and right ear modalities, respectively. These weights can be obtained through brute force exploration or based on the separate performance of these classifiers [50]. Later, the weights for the modality specific classifiers in the score level fusion were determined by using a separate training set with the goal of maximizing the fused multimodal recognition accuracy.

6. Experimental Results

First, we outlined the constrained, WVU dataset [7] and the unconstrained, HONDA/UCSD [8] dataset contents. Then, we demonstrate the results of the modality specific and multi-modal recognition experiments on both datasets.

6.1 WVU Dataset

The WVU data set [7] contains video sequences obtained by a camera moving in a semicircle around the face, starting from the extreme left face profile, i.e. 0º, then proceed towards right face profile up to 180º rotation, for a total of 402 subjects. Video clips in the WVU database are collected at different times under the same environmental constraints, e.g., illumination and distance from the camera. Three of the subjects had their left and right ears fully occluded, and therefore, they were removed from the dataset. Fifty-nine subjects have two or more video sequences with widely varied appearance with and without facial hair, glasses, and long hair, which partially occluded the ear, while the remaining 340 subjects only have one video sequence.

To perform the multimodal recognition, we trained modality specific dictionaries for each of the five modalities using the training video sequences. In cases of missing modalities in the test video, we are still able to perform multimodal recognition using the available modalities. In order to evaluate our algorithm, we prepared two instances of datasets from the available video sequences in the WVU dataset.

6.1.1 Dataset-1

In dataset 1, we use one video sequence for each subject, which results into a total of 399 video sequences. The detected modality specific regions from the video sequence of each subject are separated for training and testing in a non-overlapping fashion. Detection of the left ear and the left profile face is performed between 0° to 30° rotations of the camera in the video. The detected regions in the first 100 frames were used for training and the detected regions in the next 100 frames were used for testing. Detection of frontal face is performed on frames between 75° to 105°, where the detected regions in the first 100 frames were used for training and the detected regions in the next 100 frames for testing. Detection of right ear and right profile face performed between 150° to 180°, where the detected regions in the last 100 frames in the video clip utilized in training and preceding 100 frames utilized during testing.

6.1.2 Dataset-2

In dataset 2, we use only the subjects who have more than one video sequence, which results into a total of 121 video clips, with one subject having three video sequences, one subject having four sequences, and the rest of the 57 subjects having two video sequences. The detected different modality specific regions from one video utilized during training, the regions detected from the second video applied in testing in cross-fold fashion. Detection of the left ear and the left profile face is performed between 0° to 30°, where detected regions in the first 200 frames are used. Detection of the frontal face is performed between 75° to 105° where detected regions in the first 200 frames are used. Detection of the right ear and the right profile face is performed between 150° to 180° where detected regions in the last 200 frames are used.

Table 2. State-of-the-art 2D multimodal (profile and frontal face, ear) rank-1 recognition accuracy comparison

	Kisku et al. [22]	**Pan et al. [23]**	**Boodoo and Subramanian [21]**	**This study**
Modalities	Ear & Frontal face	Ear & Profile face	Ear & Frontal face	Ear, Frontal face, and Profile face
Fusion performed	Decision level	Feature level	Decision level	Score level
Best reported Rank-1 accuracy (%)				
Ear	93.53	91.77	90.70	95.04
Frontal face	91.96	NA	94.70	97.52
Profile face	NA	93.46	NA	93.39
Fusion	95.53	96.84	96.00	99.17

Table 3. Recognition result of multimodal recognition with all possible combinations of 2, 3 and 4 modalities using dataset-1 and dataset-2 (WVU)

	Rank-1 (%)	
	Dataset 1	**Dataset 2**
Test done with combining any 2 modalities		
FF + Lt. PF/Rt. PF	97.52	91.23
FF + Lt. ear/Rt. ear	98.35	94.74
Lt. ear/Rt. ear + Lt. PF/Rt. PF	98.35	94.74
Test done with combining any 3 modalities		
FF + Lt. PF + Lt. ear	97.52	92.98
FF + Rt. PF + Rt. ear	97.52	91.23
FF + Lt. PF + Rt. PF	97.52	91.23
FF + Lt. ear + Rt. ear	98.35	94.74
FF + Rt. PF + Lt. ear	97.52	92.98
FF + Lt. PF + Rt. ear	97.52	91.23
Lt. PF + Lt. ear + Rt. PF	94.21	89.47
Rt. PF + Rt ear + Lt. PF	95.04	89.47
Rt. PF + Lt. ear + Rt. ear	98.35	94.74
Lt. PF + Lt. ear + Rt. ear	98.35	94.74
Test done with combining any 4 modalities		
FF + Rt. ear + Lt. ear + Lt. PF	98.35	94.74
FF + Lt. PF + Rt. PF + Lt. ear	97.52	92.98
Rt. ear + Lt. ear + Lt. PF + Rt. PF	98.35	94.74
FF + Rt. ear + Lt. ear + Rt. PF	98.35	94.74
FF + Lt. PF + Rt. PF + Rt. ear	98.35	91.23

FF=frontal face, PF=profile face, Rt.=right, Lt.=left.

To compute the multimodal Rank-1 recognition result, score level fusion is performed using majority voting of Rank-1 recognition accuracy from these five different modalities. The multimodal recognition accuracy of our approach is as follows: for dataset-1 at average we obtained 99.17% Rank-1 recognition accuracy, and in dataset-2, at average we obtained 96.49% Rank-1 recognition accuracy. The best Rank-1 recognition rates, using ear, frontal and profile face modalities for multimodal recognition, compared with the results reported in [21-23] is shown in Table 2. Recognition accuracies for all the modality specific and multimodal framework in this study outpaces the other multimodal recognition techniques that uses ear, frontal face and profile face.

Later, we performed experiments when all the modalities were available during the training and only some of the modalities were available during the testing. The accuracy of the recognition results using all possible combinations of the different modalities in the test video, for dataset-1 and dataset-2 been listed in Table 3, correspondingly. The results indicate that, among all possible combinations of different modalities, frontal face with ear, i.e., right and left ear modalities, have the best recognition rate.

6.2 HONDA/UCSD Dataset

The publicly available HONDA/UCSD dataset [8], contains facial video clips with non-planar head rotations as well as varying illumination. The dataset has two parts, dataset-1 and dataset-2, that consist of separate training and testing facial video clips of 20 and 15 unique subjects, respectively. HONDA/UCSD dataset consists of 89 facial video clips of 35 unique individuals, where each subject has two or more video clips. In our experiments, we used one facial video sequence for training and the rest for testing in cross-fold approach.

The AdaBoost detector results in a few false detections when applied to the HONDA/UCSD unconstrained dataset. To quantify how the false detections affect both the unimodal and multimodal recognition accuracies, comparison of the results in applying the trained detector on all the frames is performed, including the ones with false positives, with the performance while using the trained detector to only the frames with true positives. Tables 4 and 5 show the comparisons, where the multimodal recognition accuracy obtained when using all the frames is 97.14% (34 out of 35 subjects), and 100% when using only the true positive detected regions.

Table 4. Modality specific and multimodal Rank-1 recognition accuracy (%) using all detected regions

Gabor feature length	Frontal face	Left profile face	Right profile face	Left ear	Right ear	Multimodal
No feature reduction	91.43	71.43	71.43	85.71	85.71	88.57
1,000	91.43	71.43	74.29	88.57	88.57	97.14
500	88.57	68.57	68.57	85.71	82.86	91.42

Table 5. Modality specific and multimodal Rank-1 recognition accuracy (%) using only accurately detected regions

Gabor feature length	Frontal face	Left profile face	Right profile face	Left ear	Right ear	Multimodal
No feature reduction	91.43	80.00	82.86	91.43	91.43	94.29
1,000	97.14	82.86	82.86	94.29	94.29	100
500	91.43	81.19	80.00	91.43	91.43	94.29

The feature vectors automatically learned using the trained deep learning network resulted in length of 9,600 for frontal and profile face; 4,160 for ear. In order to decrease the computational complexity and to find out most effective feature vector length to maximize the recognition accuracy, the dimensionality of the feature vector is reduced to a lower dimension using PCA [13]. Using PCA, the number of features is reduced to 500 and 1,000. In Tables 4 and 5, the modality specific recognition accuracy obtained for the original feature vector and for the reduced feature vector of 500, 1,000 is shown. Feature vectors of length 1,000 resulted in best recognition accuracy for both modality specific and multimodal recognition.

Table 2 contains the best Rank-1 recognition rates, using ear, frontal and profile face modalities for multimodal recognition, compared with the results reported in [10-12].

6.3 Comparison with Baseline Algorithms

Due to the unavailability of proper datasets for multimodal recognition studies [6], often virtual multimodal databases are synthetically obtained by pairing modalities of different subjects from different databases. To the best of the authors' knowledge, the framework of extracting multiple modalities from a single data source that belongs to the same subject, have not been performed before in the state-of-the-art. Thus, we compare the performance of the proposed technique of learning automatic robust features using deep learning network and using sparse representation for classification with the following baseline algorithms due to their close relationships. It is also worth noting that all the comparisons are based on the same training/test set.

1) SRC [6] with extracted Gabor features.
2) SRC with K-SVD dictionary learning [52] and Gabor features. K-SVD is an iterative method that alternates between sparse coding of the examples based on the current dictionary and an update process for the dictionary atoms to better fit the training data. Therefore, SRC with K-SVD is better than conventional SRC to train models with variable number of training samples in the different classes.

Table 6. Comparison of multimodal recognition with the baseline algorithms

Modality	WVU	Honda/UCSD
SRC	95.24	51.43
SRC with K-SVD	97.52	68.57
This study	99.17	97.14

In Table 6, the comparisons of multimodal recognition accuracy of the baseline techniques and the proposed method are provided for both WVU and HONDA/UCSD datasets. The comparison shows that the proposed technique performs better on both the constrained and unconstrained datasets compared with the other baseline algorithms. However, we can see that the performance of the two baseline algorithms are relatively satisfactory on the constrained (WVU) dataset, but on the unconstrained (HONDA/UCSD) dataset, the performance of the two baseline algorithms is very poor. As expected SRC with K-SVD performs better than conventional SRC, but both have much lower performance than our proposed algorithm. We believe that the reason behind this is HONDA/UCSD database consists of faces with non-planar movements (shown in Fig. 7).

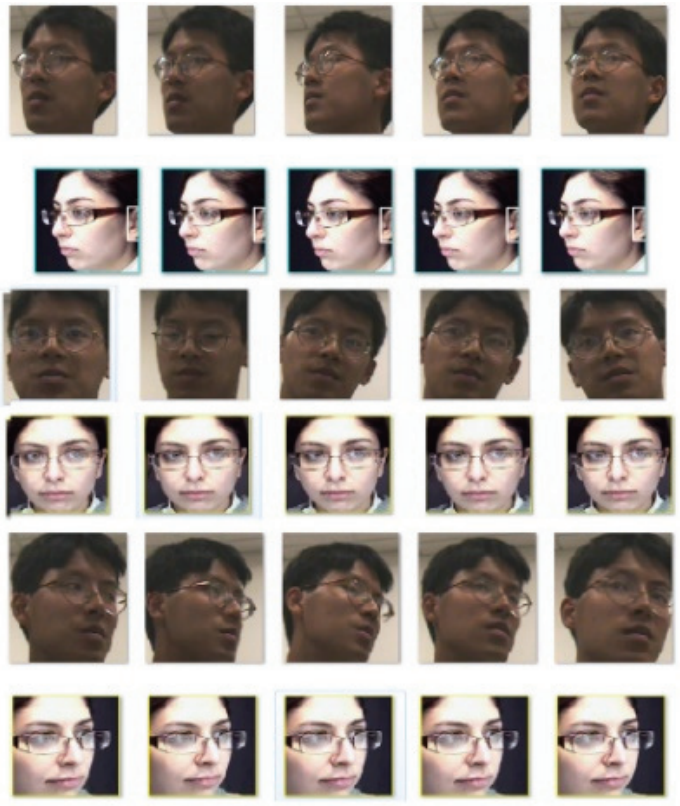


Fig. 7. Non-planar movement in HONDA dataset compared with WVU: left profile, left frontal, and right profile.

6.4 Parameter Selection for the Deep Neural Network

We have tested the performance of the proposed multimodal recognition framework against different parameters of the deep neural network. We varied the number of hidden layers from three to seven. By using five hidden layers we achieved the best performance. To incorporate the sparsity in the hidden layers, we also conducted experiments by changing the number of hidden nodes from two to five times of the input nodes. By using twice the hidden nodes of the input nodes in the five hidden layers we obtain the best accuracy of the multimodal recognition system. The pre-training learning rate of the DBN is used as 0.001 and the fine-tuning learning rate of the SADE is used as 0.1 to achieve the optimal performance. While training the SADE network in a Core i7-2600K CPU clocked at 3.40 GHz Windows PC using Theano Library (Python programming language) pre-training of the DBN takes approximately 600 minutes and the fine-tuning of the SADE network converged within 48 epochs in 560.2 minutes.

7. Conclusions

We proposed a system for multimodal recognition using only a particular biometrics source of data, face video surveillance. Using the AdaBoost detector, we automatically detect modality specific regions. We use Gabor features extracted from the detected regions to automatically learn robust and non-

redundant features by training a Supervised SDAE (deep learning) network. Classification through sparse representation is used for each modality. Then, the multimodal recognition is obtained through the fusion of the results from the modality specific recognition. We trained the algorithm using all modalities and tested the system when the test video clips contain all the modalities and when there are only some of the modalities are available. The results indicate that among all possible combinations of different modalities frontal face and ear, i.e., right or left ear, together produce the best recognition rate. In future additional biometrics modalities can be easily integrated to extend this generic framework. This framework will also be used to perform multimodal recognition using low-resolution video footages collected by the closed-circuit video surveillance system.

References

[1] F. Karray, J. A. Saleh, M. N. Arab, and M. Alemzadeh, "Multi modal biometric systems: a state of the rt survey," in *Proceedings of the 4th International Conference on Computational Intelligence, Robotics and Autonomous Systems* (CIRAS), Palmerston North, New Zealand, 2007.

[2] S. Cadavid, M. H. Mahoor, and M. Abdel-Mottaleb, "Multi-modal biometric modeling and recognition of the human face and ear," in *Proceedings of 2009 IEEE International Workshop on Safety, Security & Rescue Robotics (SSRR)*, Denver, CO, 2009, pp. 1-6.

[3] M. H. Mahoor and M. Abdel-Mottaleb, "A multimodal approach for face modeling and recognition," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 3, pp. 431-440, 2008.

[4] A. Ross and A. K. Jain, "Multimodal biometrics: an overview," in *Proceedings of 2004 12th European Signal Processing Conference*, Vienna, Austria, 2004. pp. 1221-1224.

[5] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, Kauai, HI, 2001, pp. 511-518.

[6] Z. Huang, Y. Liu, C. Li, M. Yang, and L. Chen, "A robust face and ear based multimodal biometric system using sparse representation," *Pattern Recognition*, vol. 46, no. 8, pp. 2156-2168, 2013.

[7] G. Fahmy, A. El-Sherbeeny, S. Mandala, M. Abdel-Mottaleb, and H. Ammar, "The effect of lighting direction/condition on the performance of face recognition algorithms," in *Proceedings of SPIE 6534: Biometric Technology for Human Identification III*. Bellingham, WA: International Society for Optics and Photonics, 2006.

[8] K. C. Lee, J. Ho, M. H. Yang, and D. Kriegman, "Visual tracking and recognition using probabilistic appearance manifolds," *Computer Vision and Image Understanding*, vol. 99, no. 3, pp. 303-331, 2005.

[9] A. M. Bronstein, M. M. Bronstein, and R. Kimmel, "Three-dimensional face recognition," *International Journal of Computer Vision*, vol. 64, no. 1, pp. 5-30, 2005.

[10] K. Etemad and R. Chellappa, "Discriminant analysis for recognition of human face images," *Journal of the Optical Society of America A*, vol. 14, no. 8, pp. 1724-1733, 1997.

[11] C. Lu and X. Tang, "Surpassing human-level face verification performance on LFW with GaussianFace," in *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, Austin, TX, 2015, pp. 3811-3819.

[12] Y. Taigman, M. Yang, M. A. Ranzato, and L. Wolf, "Deepface: closing the gap to human-level performance in face verification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, 2014, pp. 1701-1708.

[13] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71-86, 1991.

[14] P. P. Sarangi, B. P. Mishra, and S. Dehuri, "Multimodal biometric recognition using human ear and profile face," in *Proceedings of 2018 4th International Conference on Recent Advances in Information Technology (RAIT)*, Dhanbad, India, 2018, pp. 1-6.

[15] Y. C. Chen, V. M. Patel, P. J. Phillips, and R. Chellappa, "Dictionary-based face recognition from video," in *Computer Vision – ECCV 2012*. Heidelberg: Springer, 2012, pp. 766-779.

[16] K. C. Lee, J. Ho, M. H. Yang, and D. Kriegman, "Video-based face recognition using probabilistic appearance manifolds," in *Proceedings of 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Madison, WI, 2003, pp. 313-320.

[17] U. Park and A. K. Jain, "3D model-based face recognition in video," in *Advances in Biometrics*. Heidelberg: Springer, 2007, pp. 1085-1094.

[18] U. Park, A. K. Jain, and A. Ross, "Face recognition in video: adaptive fusion of multiple matchers," in *Proceedings of 2007 IEEE Conference on Computer Vision and Pattern Recognition*, Minneapolis, MN, 2007, pp. 1-8.

[19] A. Lumini and L. Nanni, "Overview of the combination of biometric matchers," *Information Fusion*, vol. 33, pp. 71-85, 2017.

[20] A. Abaza, A. Ross, C. Hebert, M. A. F. Harrison, and M. S. Nixon, "A survey on ear biometrics," *ACM Computing Surveys*, vol. 45, no. 2, pp. 1-35, 2013.

[21] N. B. Boodoo and R. K. Subramanian, "Robust multi biometric recognition using face and ear images," *International Journal of Computer Science and Information Security*, vol. 6, no. 2, pp. 164-169, 2009

[22] D. R. Kisku, J. K. Sing, and P. Gupta, "Multibiometrics belief fusion," CoRR, 2010; http://arxiv.org/abs/1002.2755.

[23] X. Pan, Y. Cao, X. Xu, Y. Lu, and Y. Zhao, "Ear and face based multimodal recognition based on KFDA," in *Proceedings of 2008 International Conference on Audio, Language and Image Processing*, Shanghai, China, 2008, pp. 965-969.

[24] S. El-Naggar, A. Abaza, and T. Bourlai, "A study on human recognition using auricle and side view face images," in *Surveillance in Action*. Cham: Springer, 2018, pp. 77-104.

[25] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, "Multimodal deep learning," in *Proceedings of the 28th International Conference on Machine Learning*, Bellevue, WA, 2011, pp. 689-696.

[26] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, pp. 1097-1105, 2012.

[27] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Proceedings of 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, San Diego, CA, 2005, pp. 539-546.

[28] G. Goswami, R. Bhardwaj, R. Singh, and M. Vatsa, "MDLFace: memorability augmented deep learning for video face recognition," in *Proceedings of IEEE International Joint Conference on Biometrics*, Clearwater, FL, 2014, pp. 1-7.

[29] D. Menotti, G. Chiachia, A. Pinto, W. R. Schwartz, H. Pedrini, A. X. Falcao, and A. Rocha, "Deep representations for iris, face, and fingerprint spoofing detection," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 864-879, 2015.

[30] S. Gao, Y. Zhang, K. Jia, J. Lu, and Y. Zhang, "Single sample face recognition via learning deep supervised autoencoders," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 10, pp. 2108-2118, 2015.

[31] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, "The FERET evaluation methodology for face-recognition algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 10, pp. 1090-1104, 2000.

[32] K. Chang, K. W. Bowyer, S. Sarkar, and B. Victor, "Comparison and combination of ear and face images in appearance-based biometrics," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1160-1165, 2003.

[33] C. Liu and H. Wechsler, "Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition," *IEEE Transactions on Image processing*, vol. 11, no. 4, pp. 467-476, 2002.

[34] R. Khorsandi, S. Cadavid, and M. Abdel-Mottaleb, "Ear recognition via sparse representation and Gabor filters," in *Proceedings of 2012 IEEE 5th International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, Arlington, VA, 2012, pp. 278-282.

[35] S. Urolagin, K. V. Prema, and N. S. Reddy, "Rotation invariant object recognition using Gabor filters," in *Proceedings of 2010 5th International Conference on Industrial and Information Systems*, Mangalore, India, 2010, pp. 404-407.

[36] M. Yang and L. Zhang, "Gabor feature based sparse representation for face recognition with Gabor occlusion dictionary," in *Computer Vision – ECCV 2010.* Heidelberg: Springer, 2010, pp. 448-461.

[37] Y. Bengio, "Learning deep architectures for AI," *Foundations and Trends in Machine Learning*, vol. 2, no. 1, pp. 1-127, 2009.

[38] D. E. Rumelhart and J. L. MacClelland, *Parallel Distributed Processing, Explorations in the Microstructure of Cognition*. Cambridge, MA: MIT Press, 1986.

[39] I. Jolliffe, "Principal Component Analysis," in *Encyclopedia of Statistics in Behavioral Science*. Hoboken, NJ: John Wiley & Sons, 2005.

[40] S. Rifai, P. Vincent, X. Muller, X. Glorot, and Y. Bengio, "Contractive auto-encoders: explicit invariance during feature extraction," in *Proceedings of the 28th International Conference on Machine Learning*, Bellevue, WA, 2011, pp. 833-840.

[41] P. Lennie, "The cost of cortical computation," *Current Biology*, vol. 13, no. 6, pp. 493-497, 2003.

[42] P. Vincent, H. Larochelle, Y. Bengio, and P. A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th International Conference on Machine Learning*, Helsinki, Finland, 2008, pp. 1096-1103.

[43] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504-507, 2006.

[44] Y. Bengio and Y. LeCun, "Scaling learning algorithms towards AI," in *Large-Scale Kernel Machines.* Cambridge, MA: MIT Press, 2007, pp. 321-360.

[45] M. Ranzato, C. Poultney, S. Chopra, and Y. L. Cun, "Efficient learning of sparse representations with an energy-based model," *Advances in Neural Information Processing Systems*, vol. 19, pp. 1137-1144, 2006.

[46] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," *Advances in Neural Information Processing Systems*, vol. 19, pp. 153-160, 2006.

[47] R. Lengelle and T. Denoeux, "Training MLPs layer by layer using an objective function for internal representations," *Neural Networks*, vol. 9, no. 1, pp. 83-97, 1996.

[48] G. E. Hinton, S. Osindero, and Y. W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527-1554, 2006.

[49] H. Mohimani, M. Babaie-Zadeh, and C. Jutten, "A fast approach for overcomplete sparse decomposition based on smoothed l_0 norm," *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 289-301, 2008.

[50] A. A. Ross, K. Nandakumar, and A. K. Jain, *Handbook of Multibiometrics*. New York, NY: Springer, 2006.

[51] F. R. Hampel, E. M. Ronchetti, P. J. Rousseeuw, and W. A. Stahel, *Robust Statistics: The Approach Based on Influence Functions*. New York, NY: John Wiley & Sons, 2011.

[52] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311-4322, 2006.

Sayan Maity **https://orcid.org/0000-0001-9249-4914**

He received his Ph.D. in Industrial Engineering from University of Miami. He received the M.E. in Electronics and Telecommunication Engineering from Jadavpur University, India in 2011. His research interests include biometrics, computer vision and pattern recognition.

Mohamed Abdel-Mottaleb **https://orcid.org/0000-0002-7749-4577**

He received the Ph.D. degree in computer science from the University of Maryland, College Park, in 1993. From 1993 to 2000, he was with Philips Research, Briarcliff Manor, NY, where he was a Principal Member of the Research Staff and a Project Leader. At Philips Research, he led several projects in image processing and content-based multimedia retrieval. He joined the University of Miami in 2001. He is currently a Professor and a Chairman of the Department of Electrical and Computer Engineering. He represented Philips in the standardization activity of ISO for MPEG-7, where some of his work was included in the standard. He holds 22 U.S. patents and over 30 international patents. He has authored over 120 journal and conference papers in the areas of image processing, computer vision, and content-based retrieval. His research focuses on 3D face and ear biometrics, dental biometrics, visual tracking, and human activity recognition. He is an Editorial Board Member of the Pattern Recognition journal.

Shihab S. Asfour **https://orcid.org/0000-0001-5052-9402**

He received his Ph.D. in Industrial Engineering from Texas Tech University, Lubbock, Texas. Dr. Asfour currently serves as the Associate Dean of the College of Engineering, since August 15, 2007, at the University of Miami. In addition, he has been serving as the Chairman of the Department of Industrial Engineering at the University of Miami since June 1, 1999. In addition to his appointment in Industrial Engineering, he holds the position of Professor in the Biomedical Engineering and the Orthopedics and Rehabilitation Departments. Dr. Asfour has published over 250 articles in national and international journals, proceedings and books. His publications have appeared in the Ergonomics, Human Factors, Spine, and IIE journals. He has also edited a two-volume book titled “Trends in Ergonomics/Human Factors IV” published by Elsevier Science Publishers in 1987, and the book titled “Computer Aided Ergonomics”, published by Taylor and Francis, 1990.