Face Verification via Learned Representation on Feature-Rich Video Frames

Gaurav Goswami, *Student Member, IEEE,* Mayank Vatsa, *Senior Member, IEEE,* and Richa Singh, *Senior Member, IEEE*

***Abstract*—Abundance and availability of video capture devices such as mobile phones and surveillance cameras has instigated research in video face recognition which is highly pertinent in law enforcement applications. While the current approaches have reported high accuracies at equal error rates, performance at lower false accept rates requires significant improvement. In this research, we propose a novel face verification algorithm which starts with selecting feature-rich frames from a video sequence using discrete wavelet transform and entropy computation. Frame selection is followed by representation learning based feature extraction where three contributions are presented: (i) deep learning architecture which is a combination of stacked denoising sparse autoencoder (SDAE) and deep Boltzmann machine (DBM), (ii) formulation for joint representation in an autoencoder, and (iii) updating the loss function of DBM by including sparse and low rank regularization. Finally, a multilayer neural network is used as classifier to obtain the verification decision. The results are demonstrated on two publicly available databases, YouTube Faces and Point and Shoot Challenge. Experimental analysis suggests that (i) the proposed feature-richness based frame selection offers noticeable and consistent performance improvement compared to frontal only frames, random frames, or frame selection using perceptual no-reference image quality measures, and (ii) joint feature learning in SDAE and sparse and low rank regularization in DBM helps in improving face verification performance. On the benchmark Point and Shoot Challenge, the algorithm yields the verification accuracy of over 97% at 1% false accept rate whereas on the YouTube Faces database, over 95% verification accuracy is observed at equal error rate.**

***Index Terms*—Deep Learning, Autoencoder, Deep Boltzmann Machine, Face Recognition, Frame Selection**

I. Introduction

VIDEO face recognition has become highly significant in surveillance scenarios. For example, more than 80,000 people were identified and verified during the 2008 Beijing Olympics with the help of face recognition in videos [1]. With advancements in technology, video capturing devices are accessible to a large number of people in the form of portable electronic devices such as phones and tablets. In unconstrained scenarios, videos captured by such devices may also be used by law enforcement agencies. Therefore, there is a high motivation to utilize video data to perform accurate face recognition. Fig. 1 shows frames from four video clips in which the face regions have been detected and cropped. While a single frame from a video can only capture limited information, multiple frames capture a lot of information about the face pertaining to its appearance under the effect of common covariates such as pose, illumination, and expression. By utilizing the large variety of information present in a video, a robust and comprehensive representation of a face can be extracted and accuracy can be improved.

Gaurav Goswami, Mayank Vatsa, and Richa Singh are with IIIT-Delhi, India. E-mail: {gauravgs, mayank, rsingh}@iiitd.ac.in.

Fig. 1. A subset of frames illustrating the amount of information present in a video. A single video can capture a subject's face under different pose, expression, and illumination variations. While some frames can be highly useful for face recognition, others can be detrimental to performance. Images are frames from the PaSC database [2].

Video face recognition has been extensively studied and several algorithms have been proposed. Table I provides a review of some of the algorithms along with the summary of results reported on popular video face recognition databases. Video face recognition algorithms can broadly be classified into two types: (a) set-based and (b) sequence-based [26]. The set-based approaches consider a video as a set of images (frames) which are then modeled and matched using a variety of methodologies. These approaches may not utilize the temporal information contained in the video, i.e. the order of frames in the original video may not matter. On the other hand, sequence-based approaches are specifically designed to utilize temporal information of the video. These approaches model the video as a sequence of images and apply sequence classification techniques for recognition. Some of the recent techniques utilize large image dictionaries to characterize videos [8], while some others have focused on metric learning based approaches [10] or deep learning based approaches [11]. For comparison, the results are generally reported on benchmark databases such as the Honda UCSD database [27], YouTube face database (YTF) [3], and recently developed Point and Shoot Challenge (PaSC) database [2].

As shown in Table I, existing algorithms have attained high performance on YouTube video face database [3]. However, the protocol of this databases generally require reporting the results at equal error rate (EER) [28]. From implementation

TABLE I
REVIEW OF SELECTED PAPERS ON VIDEO FACE RECOGNITION THAT HAVE SHOWN RESULTS ON THE YTF AND PASC BENCHMARK FACE VIDEO DATABASES. RESULTS MARKED UNRESTRICTED DENOTE ALGORITHMS THAT HAVE USED EXTERNAL TRAINING DATA DURING TRAINING. THE ALGORITHMS FOLLOW THE STANDARD EXPERIMENTAL PROTOCOL DURING TESTING FOR BOTH DATABASES TO FACILITATE COMPARISON.

Authors	Algorithm	Database	Verification Accuracy
Wolf *et al.*, 2011 [3][1]	Matched background similarity L2 mean with LBP	YTF [3]	76.4%
Wolf and Levy, 2013 [4][1]	SVM-Minus similarity score with background similarity		78.9%
Li *et al.*, 2013 [5][1]	Probabilistic elastic matching		79.1%
Cui *et al.*, 2013 [6][2]	Spatial-temporal face region descriptor + Pairwise-constrained multiple metric learning		79.5%
Mendez-Vazquez *et al.*, 2013 [7][2]	Volume structured ordinal features		79.7%
Bhatt *et al.*, 2014 [8][1]	Clustering based re-ranking and fusion		80.7%
Hu *et al.*, 2014 [9][1]	Large margin multi-metric learning for face and kinship verification in the wild		81.3%
Hu *et al.*, 2014 [10][1]	Discriminative deep metric learning		82.3%
Taigman *et al.*, 2014 [11][1]	Nine-layer deep network		91.4 % (unrestricted)
Wang *et al.*, 2015 [12][1]	Discriminant Analysis on Riemannian manifold of Gaussian distributions		73.01 AUC
Khan *et al.*, 2015 [13][1]	Adaptive Sparse Dictionary		82.9%
Li *et al.*, 2015 [14][1]	Eigen-PEP for video face recognition		84.8%
Li *et al.*, 2015 [15][1]	Hierarchical-PEP for video face recognition		87.0%
Sun *et al.*, 2015 [16][1]	Semi-supervised convolutional neural network		93.2% (unrestricted)
Schroff *et al.*, 2015 [17][1]	Unified embedding learned using deep CNN		95.1% (unrestricted)
Parkhi *et al.*, 2015 [18][1]	Eleven-layer deep convolutional neural network with triplet loss based face embedding		97.3% (unrestricted)
Ding and Tao, 2016 [19][1]	Ensemble of Deep Convolutional Neural Networks		94.96% (unrestricted)
Yang *et al.*, 2016 [20][1]	GoogLeNet [21] features with aggregation		95.5% (unrestricted)
Tran *et al.*, 2016 [22][1]	3D Morphable Face Models regressed using a CNN		88.8% (unrestricted)
Beveridge *et al.*, 2013 [2][1]	Local region principal component analysis	PaSC [2]	8% (handheld) 10% (control)
Wang *et al.*, 2015 [12][1]	Discriminant Analysis on Riemannian manifold		18.3% (handheld) 18.7% (control)
Li *et al.*, 2015 [15][1]	Hierarchical-PEP for video face recognition		30.7%
Huang *et al.*, 2015 [23][1]	Projection Metric Learning on Grassmann Manifold		43.9% (handheld) 43.6% (control)
Huang *et al.*, 2015 [24][1]	Hybrid Euclidean-and-Riemannian Metric Learning		59% (handheld) 58% (control)
Ding and Tao, 2016 [19][1]	Ensemble of Deep Convolutional Neural Networks		95.9% (handheld-unrestricted) 96.2% (control-unrestricted)
Goswami et al. [25][1]	Memorability based frame selection and deep learning	PaSC	89% (handheld), 94% (controlled)
		YTF	88.6% (unrestricted)
Proposed[1]	Feature-richness based frame selection and deep learning (joint learning in autoencoder with sparse and low rank DBM)	YTF [3]	93.4%, 95.4% (unrestricted)
		PaSC [2]	93.1% (handheld), 97.2% (handheld-unrestricted), 95.9% (control), 98.1% (control-unrestricted)

[1]*denotes set based algorithms*, [2]*denotes sequence based algorithms*.

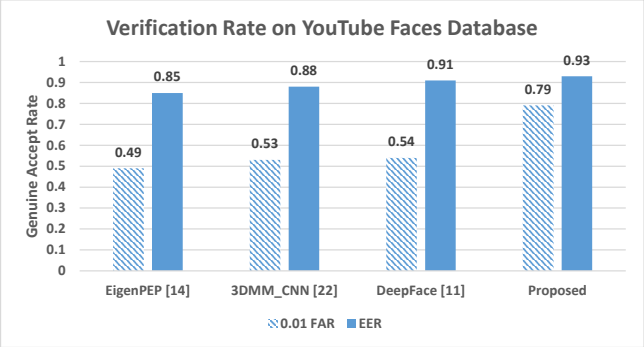


Fig. 2. Summarizing the performance of some of the best performing face verification algorithms on the YouTube faces database [3]. It is evident that there is a huge gap in the performance at low false accept rates as compared to performance at EER. We showcase that the proposed algorithm performs well even at a low false accept rate.

perspective, the algorithms are required to minimize false accept rate (FAR) or false reject rate (FRR). However, lower EER does not necessarily mean low FAR or FRR. Fig. 2 illustrates the performance of some of the existing algorithms on the YouTube Faces database [3]. It is observed that these algorithms attain very high accuracies at equal error rate, however, their performance at lower false accept rates is significantly lower. For example, DeepFace [11] yields over 91% verification accuracy at EER but only 54.1% at 1% false accept rate (FAR). For many security related applications, such as video surveillance, it is desirable to achieve high verification performance while minimizing the false accept rates. Therefore, it is our assertion that there is a significant scope of improvement in the performance of video face recognition and additional research is required, especially focusing at lower false accept rates.

A. Research Contributions

In general, video face verification involves matching using all the frames present in two videos. However, not all frames are equally informative and some frames might suffer from low image quality or extreme variations due to pose, expression, and illumination. Due to the presence of these covariates of face recognition, some frames may affect the inter-class and intra-class variations. In other words, it is highly probable that features extracted from such a frame might lead to incorrect

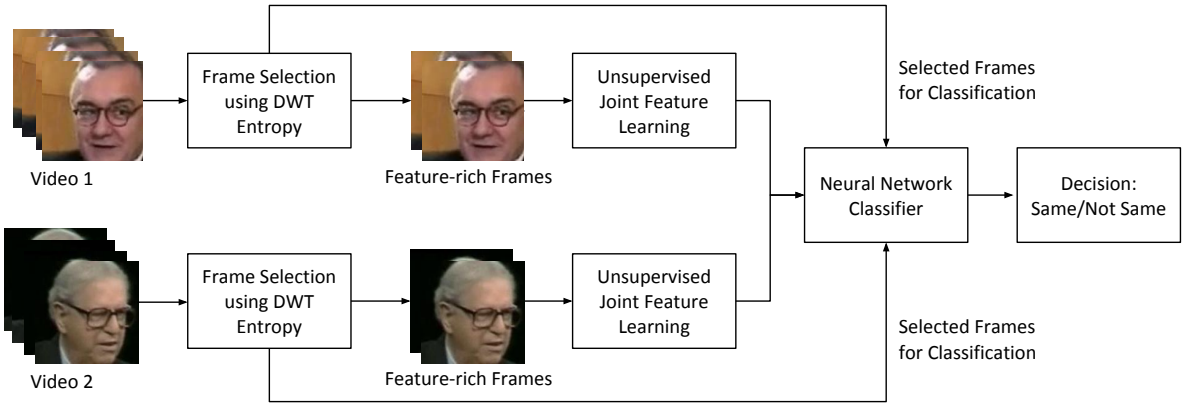


Fig. 3. Illustrating the steps involved in the proposed face recognition algorithm.

results. Therefore, it is important to select and utilize the high information content in a video carefully and efficiently which makes video data more challenging as well as rewarding for face recognition. To address some of these limitations and to improve overall performance, we propose a novel video face recognition algorithm, that utilizes frame selection process, followed by a deep learning architecture for feature extraction and matching as illustrated in Fig. 3[1].

The first contribution of this research is a novel algorithm for no-reference feature-richness based frame selection that quantifies feature-richness based on entropy [29] in the wavelet domain and enables better selection of frames for recognition as compared to traditional no-reference biometric quality measures [30], [31], [32]. The second contribution is designing a novel joint feature learning framework which can be utilized to combine intermediate features computed in a deep network. Deep learning architectures generally compute a series of intermediate features from input data and utilize the final layer of feature only for representation and classification. In the proposed deep architecture, we combine the intermediate representations computed by an autoencoder using a joint representation layer. This joint representation is utilized to retain the informative features of different granularities and is used as input to a Deep Boltzmann Machine (DBM) which interprets and enhances this combined information to create a *feature vector* for each input face. The proposed framework models the learned features as sparse and low-rank at the same time using ℓ_1-norm and trace-norm regularizations to improve the performance of the overall deep architecture. The learnt joint representation is input to a neural network for classification. The effectiveness of the proposed algorithm is evaluated on two large publicly available benchmark databases: the YouTube Faces (YTF) video [3] and Point and Shoot Challenge (PaSC) video [2].

II. Proposed Face Recognition Algorithm

The proposed algorithm is divided into three steps: (i) frame selection, (ii) deep learning based feature extraction, and (iii) face verification using learnt representations. An overview of the proposed algorithm is presented in Fig. 3.

A. Entropy based Frame Selection

Depending on the frame rate and duration, a video clip of $4-6$ seconds may contain 100-200 frames. Existing literature for video face recognition has either used all the frames, or processed some (randomly) selected frames, or have proposed algorithms for frame selection. Processing all the frames can result in inclusion of bad and redundant information. Liu *et al.* [33] proposed to partition the video into frame clusters and select the most representative frames from each cluster using Principal Component Analysis (PCA). Park *et al.* [34] proposed to select frames by estimating pose and motion blur information for each frame using Active Appearance Models (AAM) and selecting frames with controlled pose and minimal blur. Jillela *et al.* [35] utilized optical flow to create super-resolved frames by using short five frame sub-sequences while avoiding the sub-sequences which demonstrate high inter-frame motion.

The proposed algorithm presents a novel perspective towards frame selection by utilizing feature richness as the criteria. It is our assertion that quantifying the feature richness of an image helps in extracting the frames that have higher possibility of containing discriminatory features. In order to compute feature-richness, first the input (detected face) image I is preprocessed to a standard size and converted to grayscale. By performing face detection first and considering only the facial region, we ensure that other non-face content of the frame does not interfere with the proposed algorithm. The image is normalized using its mean and standard deviation. Thereafter, the discrete wavelet transform (DWT) of the pre-processed image I is computed as follows:

$$[I_{Ap}, I_{Ho}, I_{Vr}, I_{Dg}] = DWT(I) \tag{1}$$

Here, I_{Ap} captures the approximation coefficients of the image, whereas $[I_{Ho}, I_{Vr}, I_{Dg}]$ contain the detail coefficients in horizontal, vertical, and diagonal sub-bands respectively. The high and low pass filters used for decomposition depend on the type of mother wavelet used. In this research, we have

[1]A preliminary version of the proposed algorithm was published in IEEE IJCB, 2014 [25].

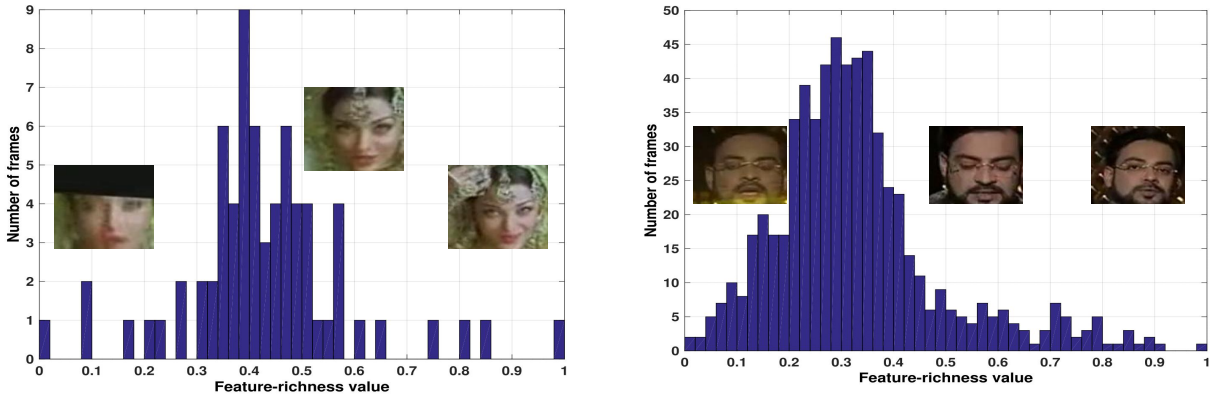


Fig. 4. Feature-richness distributions for two different videos. Some of the most feature-rich (values close to 1) and least feature-rich frames (values close to 0) are presented for illustration. We can see that the high fidelity frames are assigned a higher feature richness score and the poor frames which showcase artifacts such as occlusion and blur are assigned a low feature-richness score. Note that the total number of frames in the two videos is different.

utilized a bi-orthogonal mother wavelet which is symmetric and efficiently encodes edge features. The detail and approximation coefficients obtained using Eq. 1 represent the first level DWT coefficients. Another level of DWT is applied on the approximation band, I_{Ap}, as follows:

$$[I'_{Ap}, I'_{Ho}, I'_{Vr}, I'_{Dg}] = DWT(I_{Ap}); \tag{2}$$

Here, I'_{Ap} and $[I'_{Ho}, I'_{Vr}, I'_{Dg}]$ represent the second level DWT approximation and detail coefficients of input image I respectively. DWT is useful to enable multi-resolution analysis of the given image. While the first level DWT presents the coefficients for the finer details of the image, the second level DWT encodes the global features while focusing less on fine details. We have observed that with images of size 80×100 and below, the third level DWT is unable to preserve sufficient edge information and is not useful for frame selection. Therefore, in this research, we consider only two levels of DWT.

For an image region, entropy signifies the variation in pixel intensity values. To quantify the feature-richness of an image, entropy [29] is computed by using both levels of DWT coefficients. The local entropy of each DWT band is computed by dividing each band into 3×3 windows. On applying the algorithm to a DWT band instead of the image, the entropy value captures the local variations in high frequency and approximation subbands contained in the image. The entropy, $H(\kappa)$, of an image window κ is computed.

$$H(\kappa) = -\sum_{i=1}^{n} p(\kappa_i) log_2 p(\kappa_i) \tag{3}$$

where, n is the total number of pixel values, and $p(\kappa_i)$ is the value of the probability mass function for κ_i which represents the probability of pixel value κ_i appearing in the neighborhood. If the size of the window κ is $\mathcal{M}_\kappa \times \mathcal{N}_\kappa$ then

$$p(\kappa_i) = \frac{n_{\kappa_i}}{\mathcal{M}_\kappa \times \mathcal{N}_\kappa} \tag{4}$$

Here, n_{κ_i} denotes the number of pixels in the window with value κ_i. The entropy value of each window is combined to compute the feature-richness value of a band.

$$HF = \sum_{i=1}^{\omega} (|H_i|) \tag{5}$$

Here, HF denotes the feature-richness score of a DWT band, ω is the number of windows in the band and H_i denotes the entropy of the i^{th} window. The final score of image I, $HF(I)$, is obtained by aggregating the feature-richness values of individual bands.

$$\begin{aligned} HF(I) = HF(I'_{Ap}) + HF(I'_{Ho}) + HF(I'_{Vr}) + HF(I'_{Dg}) \\ + HF(I_{Ho}) + HF(I_{Vr}) + HF(I_{Dg}) \end{aligned} \tag{6}$$

Given a video $\mathcal{V}$, the feature-richness score of a frame f_i is represented as $HF(f_i)$. Since the score of each frame depends on the distribution of intensity values in a frame, it is important to normalize the scores across the frames in one video. Let m_i represent the feature-richness value corresponding to the i^{th} frame f_i, it is obtained using min-max normalization.

$$m_i = \frac{HF(f_i) - \min(\mathbf{HF})}{\max(\mathbf{HF}) - \min(\mathbf{HF})} \tag{7}$$

where, **HF** denotes all the feature-richness scores for the video $\mathcal{V}$ and $\min(\mathbf{HF})$ and $\max(\mathbf{HF})$ denote the minimum and maximum values in **HF**, respectively. Higher values of m signify a more feature-rich frame. Fig. 4 shows the feature-richness distribution for two videos of different individuals from the YouTube Faces database [3] along with sample frames of high, average, and low feature-richness values. Once the score of each frame is computed, *adaptive* frame selection is performed to determine the optimum set of frames to represent a video.

Let σ_m denote the standard deviation and μ_m denote the mean pertaining to the set of feature-richness values of the video $\mathcal{V}$. In order to decide which frames are selected for verification, φ_i is computed corresponding to each frame f_i.

$$\varphi_i = \left\{ \begin{array}{ll} 1, & \text{if } m_i \geq \mu_m + \frac{\sigma_m}{2} \\ 0, & \text{otherwise} \end{array} \right\} \tag{8}$$

To perform adaptive frame selection, each frame with $\varphi = 1$ is selected from a given video. These frames are utilized for feature extraction using the deep learning architecture described in the next section.

B. Deep Learning Framework for Feature Extraction

Once the feature-rich frames are obtained, the next step involved feature extraction and matching. Several state-of-the-art algorithms in recent literature use convolutional neural network. In this paper, we propose a stacked denoising autoencoders (SDAE) and Deep Boltzmann Machine (DBM) based algorithm that can yield good results with limited training data while simultaneously being able to utilize more training data to further improve performance. First, we briefly present an overview of SDAE and DBM followed by the proposed architecture.

1) Stacked Denoising Autoencoder and Deep Boltzmann Machines: An *autoencoder* [36], [37] maps the data $\mathbf{x} \in \mathbb{R}^{\alpha}$ into feature (latent representation) $\mathbf{f}$ using a deterministic (encoder) function g_{Θ} such that,

$$\mathbf{f} = g_{\Theta}(\mathbf{x}) = s(\mathbf{w} \cdot \mathbf{x} + \mathbf{\Delta}) \tag{9}$$

where, $\Theta = \{\mathbf{w}, \mathbf{\Delta}\}$ is the parameter set, s represents the sigmoid, $\mathbf{w}$ is the $\alpha' \times \alpha$ weight matrix, and $\mathbf{\Delta}$ is the offset vector of size α'. Feature $\mathbf{f}$ can be mapped to feature vector $\hat{\mathbf{x}}$ of dimensionality α using a decoder function $g'_{\Theta'}$ such that,

$$\hat{\mathbf{x}} = g'_{\Theta'}(\mathbf{f}) = s(\mathbf{w}' \cdot \mathbf{f} + \mathbf{\Delta}') \tag{10}$$

Here, $\Theta' = \{\mathbf{w}', \mathbf{\Delta}'\}$ is the decoder parameter set such that $\arg\min_{\mathbf{w},\mathbf{w}'} ||\mathbf{x} - \hat{\mathbf{x}}||_2^2$. The parameters are optimized by utilizing the unsupervised training data. Denoising autoencoder [37], a variant of autoencoder, operates on the noisy input data $\mathbf{x}_n$ and attempts to reconstruct $\hat{\mathbf{x}}$ such that $\mathbf{f} = g_{\Theta}(\hat{\mathbf{x}}_n) = s(\mathbf{w} \cdot \mathbf{x}_n + \Delta)$. It is observed that this variant is robust to noisy data and has good generalizability. Further, adding sparsity constraint helps in learning useful features and the cost function is updated as,

$$||\mathbf{x} - \hat{\mathbf{x}}||_2^2 + \beta \sum_j KL(\rho \parallel \hat{\rho}_j) \tag{11}$$

where, ρ is the sparsity parameter, $\hat{\rho}_j$ is the average activation of the j^{th} hidden unit, $KL(\rho \parallel \hat{\rho}_j) = \rho \log \frac{\rho}{\hat{\rho}_j} + (1-\rho) \log \frac{1-\rho}{1-\hat{\rho}_j}$ is the KL-divergence, and β is the sparsity penalty term. KL divergence measures the difference between a true probability distribution and its approximation. By setting the value of ρ to a small value (such as 0.05), the number of data points for which the j^{th} unit is activated can be forced to be low, which introduces sparsity of features. Smaller values of ρ and larger values of β promote more sparse features. However, a higher value of β conversely reduces the importance of accurate reconstruction. The values of ρ and β are learnt during the training and validation stages to achieve a trade-off between reconstruction performance and learning more generalizable features. If the autoencoders are stacked in a layered manner, they are called as stacked autoencoders and form a deep learning architecture to discover "*patterns*" in the input data.

Deep Boltzmann Machine is an undirected graphical model, a deep network architecture, with symmetrically coupled binary units [38]. It is designed by layer-wise training of Restricted Boltzmann Machine (RBM) and stacking them together in an undirected manner. A RBM has stochastic visible and hidden variables which are connected and the energy function is defined as:

$$E(v, h; \theta) = -\sum_{i=1}^{D}\sum_{j=1}^{F} W_{ij} v_i h_j - \sum_{i=1}^{D} b_i v_i - \sum_{j=1}^{F} a_j h_j \tag{12}$$

Here, $\mathbf{v} \in \{0,1\}^D$ denotes the visible variables and $\mathbf{h} \in \{0,1\}^F$ denotes the hidden variables, respectively. The model parameters are denoted by $\theta = \{\mathbf{a}, \mathbf{b}, \mathbf{W}\}$. W_{ij} denotes the weight of the connection between the i^{th} visible unit and j^{th} hidden unit and b_i and a_j denote the bias terms of the model. For real valued visible variables such as image pixel intensities, generally, Gaussian-Bernoulli RBMs are utilized and the energy is defined as:

$$E(v, h; \theta) = -\sum_{i=1}^{D} \frac{v_i}{\sigma_i} \sum_{j=1}^{F} W_{ij} h_j - \sum_{i=1}^{D} \frac{(v_i - b_i)^2}{2\sigma^2} - \sum_{j=1}^{F} a_j h_j \tag{13}$$

Here, $\mathbf{v} \in \mathbb{R}^D$ denotes the real-valued visible vector and $\theta = \{\mathbf{a}, \mathbf{b}, \mathbf{W}, \sigma\}$ are the model parameters. A single Gaussian-Bernoulli RBM can learn a representation of the input data. However, multiple such RBMs can be stacked in a layer-wise manner to learn increasingly complex representations of data in the form of a DBM. In this research, a three layer DBM is utilized with a greedy learning approach [39]. A three layer DBM comprised of Gaussian-Bernoulli RBMs can learn complex representations of a real-valued input vector $\mathbf{v} \in \mathbb{R}^D$ using a sequence of layers of hidden units $\mathbf{h}^{(1)}$, $\mathbf{h}^{(2)}$, and $\mathbf{h}^{(3)}$. The first layer connects the visible units to the first layer of hidden units. Thereafter, subsequent layers connect the hidden units of one layer to the hidden units of the other, causing the hidden units of a layer to act as the visible units for the next layer and so on. The energy of this DBM can be defined as:

$$\begin{aligned} E(\mathbf{v}, \mathbf{h}; \theta) = &-\sum_{i=1}^{D}\sum_{j=1}^{F_1} W_{ij}^{(1)} \frac{v_i}{\sigma_i} h_j^{(1)} - \sum_{j=1}^{F_1}\sum_{l=1}^{F_2} W_{jl}^{(2)} h_j^{(1)} h_l^{(2)} \\ &- \sum_{l=1}^{F_2}\sum_{m=1}^{F_3} W_{lm}^{(3)} h_l^{(2)} h_m^{(3)} - \sum_{i=1}^{D} \frac{(v_i - b_i)^2}{2\sigma^2} \\ &- \sum_{j=1}^{F_1} a_j^{(1)} h_j^{(1)} - \sum_{l=1}^{F_2} a_l^{(2)} h_l^{(2)} - \sum_{m=1}^{F_3} a_m^{(3)} h_m^{(3)} \end{aligned} \tag{14}$$

Here, D, F_1, F_2, F_3 are the number of units and visible and hidden layers, and $\theta = \{\mathbf{W}^{(1)}, \mathbf{W}^{(2)}, \mathbf{W}^{(3)}, \mathbf{b}, \mathbf{a}^{(1)}, \mathbf{a}^{(2)}, \mathbf{a}^{(3)}, \sigma\}$ is the set of model parameters representing visible-to-hidden and hidden-to-hidden symmetric connection weights, bias terms, and the Gaussian distribution

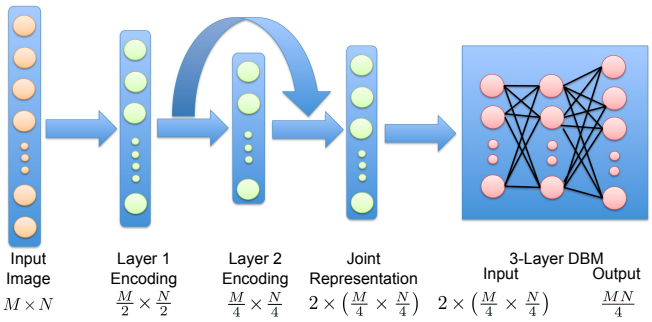


Fig. 5. Proposed deep learning architecture for facial representation: from input layer (image), two hidden layer representation are computed using SDAE encoding function. A joint representation is then obtained which combines the information from two SDAE encoding layers. Using joint representation as input, a DBM is used for computing a final feature vector.

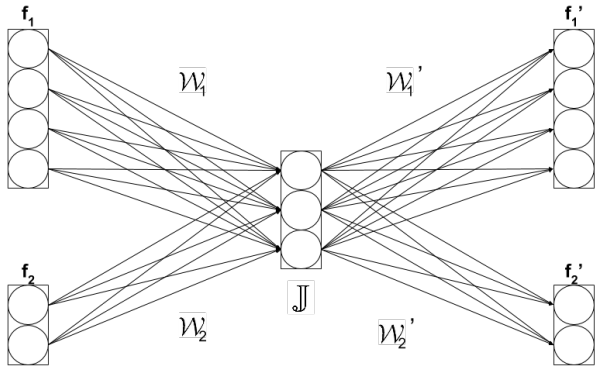


Fig. 6. Joint learning framework: features learned from the first and second levels of autoencoder, i.e., $\mathbf{f_1}$ and $\mathbf{f_2}$ are given as input to DBM to learn the joint representation $\mathbb{J}$.

standard deviation, respectively. The probability assigned by this model to a visible vector $\mathbf{v}$ is given by the Boltzmann distribution:

$$P(\mathbf{v};\theta) = \frac{1}{Z(\theta)} \sum_{\mathbf{h}} exp\left(-E\left(\mathbf{v}, \mathbf{h}^{(1)}, \mathbf{h}^{(2)}, \mathbf{h}^{(3)}; \theta\right)\right). \tag{15}$$

Here, $Z(\theta)$ is the normalizing constant. If only $\mathbf{W}^{(1)}$ is considered, the derivative of the log-likelihood with respect to the model parameters is:

$$\frac{\delta logP(\mathbf{v};\theta)}{\delta \mathbf{W}^{(1)}} = \mathbb{E}_{P_{data}}[\mathbf{v}\mathbf{h}^{(1)^T}] - \mathbb{E}_{P_{model}}[\mathbf{v}\mathbf{h}^{(1)^T}] \tag{16}$$

Here, $\mathbb{E}_{P_{data}}[\cdot]$ denotes the expectation with respect to the data distribution and $\mathbb{E}_{P_{model}}[\cdot]$ is the expectation with respect to the distribution defined by the DBM as in Eq. (15). Similar derivatives are obtained for $\mathbf{W}^{(1)}$ and $\mathbf{W}^{(2)}$, with the product $\mathbf{v}\mathbf{h}^{(1)}$ replaced by $\mathbf{h^{(1)}h^{(2)}}$ and $\mathbf{h^{(2)}h^{(3)}}$ respectively.

2) Unsupervised Joint Feature Learning: SDAE and DBM both individually learn the *useful* (intermediate) representation of input data. While the SDAE learns two layers of image-level features that can be best utilized to reconstruct the original input, in this paper, we propose a joint representation layer that learns the important features from each constituent layer. This joint layer representation combines two different levels of granularities in features to obtain a better representation. Further, this joint feature is used as input to a DBM to obtain the final representation. While SDAE and joint representation are robust to noise in the input data, DBM learns the internal complex representations probabilistically. Therefore, it is our assertion that the proposed architecture should be able to produce a robust representation compared to using SDAE or DBM in isolation. Further, DBM is able to interpret the features learned by the joint representation and combine each of its components as required to obtain an enhanced higher level discriminative representation, especially after fine-tuning.

Let the size of the input data be $M \times N$; in the proposed architecture, each layer of SDAE is one-fourth the size of its previous layer. Layer-by-layer greedy approach [40] with stochastic gradient descent is utilized to train the SDAE followed by fine-tuning with back-propagation method. Intermediate representations obtained using the 2-hidden layer SDAE are further combined to obtain a joint representation as illustrated in Fig. 5. The two layers of size $\frac{M}{2} \times \frac{N}{2}$ and $\frac{M}{4} \times \frac{N}{4}$ are utilized as input and one joint layer of size $2 \times \left(\frac{M}{4} \times \frac{N}{4}\right)$ is learned. Let $\mathbf{f_1}$ be the representation learned by the first layer of SDAE and $\mathbf{f_2}$ be the feature learned by the second layer of SDAE, the joint representation $\mathbb{J}$ can be learned using Eq. (17).

$$\mathbb{J} = \mathcal{G}(\mathbf{f_1}, \mathbf{f_2}) \tag{17}$$

Here, $\mathcal{G}$ is the joint learning function to obtain $\mathbb{J}$. In this research, using encoder-decoder approach, we defined the cost function associated with Eq. (16) as:

$$\underset{\Phi}{\operatorname{argmin}}(\| \mathbf{f_1} - \mathbf{f_1'} \|_2^2 + \| \mathbf{f_2} - \mathbf{f_2'} \|_2^2 + \mathcal{R}) \tag{18}$$

where, Φ represents the set of all the variables to be learnt and $\mathcal{R}$ is a regularizer. For ease of explanation, we first present the formulation with linear activation. Eq. (17) can be written as,

$$\mathbb{J} = \mathcal{W}_1\mathbf{f_1} + \mathcal{W}_2\mathbf{f_2} \tag{19}$$

Using Eq. (18), the associated cost can be written as,

$$\underset{\Phi}{\operatorname{argmin}}(\| \mathbf{f_1} - \mathcal{W}_1'\mathcal{W}_1\mathbf{f_1} - \mathcal{W}_1'\mathcal{W}_2\mathbf{f_2} \|_2^2 + \| \mathbf{f_2} - \mathcal{W}_2'\mathcal{W}_2\mathbf{f_2} - \mathcal{W}_2'\mathcal{W}_1\mathbf{f_1} \|_2^2 + \mathcal{R}) \tag{20}$$

As shown in Fig. 6, this approach learns the weights $\Phi = \{\mathcal{W}_1, \mathcal{W}_2, \mathcal{W}_1', \mathcal{W}_2'\}$ to obtain the joint representation $\mathbb{J}$. In a similar fashion, non-linear cost function can be written as (for simplicity, bias terms are omitted),

$$\underset{\Phi}{\operatorname{argmin}}(\| \mathbf{f_1} - s(\mathcal{W}_1'[s(\mathcal{W}_1\mathbf{f_1})]) - s(\mathcal{W}_1'[s(\mathcal{W}_2\mathbf{f_2})]) \|_2^2 + \| \mathbf{f_2} - s(\mathcal{W}_2'[s(\mathcal{W}_2\mathbf{f_2})]) - s(\mathcal{W}_2'[s(\mathcal{W}_1\mathbf{f_1})]) \|_2^2 + \mathcal{R}) \tag{21}$$

Adding ℓ_2-norm regularization term on $\mathcal{W}_1, \mathcal{W}_2$ and *dropout* [41] on the joint representation network, Eq. (21) can be written as,

$$\underset{\Phi}{\operatorname{argmin}} \big(\| \mathbf{f_1} - s(\mathcal{W}'_{\mathbf{1}}[s(\mathcal{W}_{\mathbf{1}}\mathbf{f_1})]) - s(\mathcal{W}'_{\mathbf{1}}[s(\mathcal{W}_{\mathbf{2}}\mathbf{f_2})]) \|_2^2 + \| \mathbf{f_2} - s(\mathcal{W}'_{\mathbf{2}}[s(\mathcal{W}_{\mathbf{2}}\mathbf{f_2})]) - s(\mathcal{W}'_{\mathbf{2}}[s(\mathcal{W}_{\mathbf{1}}\mathbf{f_1})]) \|_2^2 + (\lambda_1 \| \mathcal{W}_{\mathbf{1}} \|_2^2 + \lambda_2 \| \mathcal{W}_{\mathbf{2}} \|_2^2) \big)_{dropout} \quad (22)$$

The joint representation combines *abstract* and *low-level* features obtained from SDAE encoding layers and is used as input to a three hidden layer DBM, i.e. $\mathbb{J}$ acts as the visible vector. Similar to Eq. (14), the energy of this DBM is represented as:

$$E(\mathbb{J}, \mathbf{h}; \theta) = -\sum_{i=1}^{D}\sum_{j=1}^{F_1} W_{ij}^{(1)} \frac{\mathbb{J}_i}{\sigma_i} h_j^{(1)} - \sum_{j=1}^{F_1}\sum_{l=1}^{F_2} W_{jl}^{(2)} h_j^{(1)} h_l^{(2)} - \sum_{l=1}^{F_2}\sum_{m=1}^{F_3} W_{lm}^{(3)} h_l^{(2)} h_m^{(3)} - \sum_{i=1}^{D} \frac{(\mathbb{J}_i - b_i)^2}{2\sigma^2} - \sum_{j=1}^{F_1} a_j^{(1)} h_j^{(1)} - \sum_{l=1}^{F_2} a_l^{(2)} h_l^{(2)} - \sum_{m=1}^{F_3} a_m^{(3)} h_m^{(3)} \quad (23)$$

Inspired from [42], [43], we believe that the learned weight matrix can be modeled as sparse and low rank at the same time and therefore, a regularization approach incorporating both of these can improve feature learning. Hence, we extend the loss function of DBM (RBM) by introducing trace-norm regularization technique.

Let $\mathcal{L}$ be the loss function of RBM (DBM) with the energy function defined in Eq. (23). Along with ℓ_1-norm, trace-norm is added to the loss function as follows:

$$\mathcal{L}_{new} = \mathcal{L} + \mathcal{A} \| W \|_1 + \mathcal{B} \| W \|_\tau \quad (24)$$

where $\| \cdot \|_1$ is the ℓ_1-norm, and $\| \cdot \|_\tau$ is the trace-norm, and $\mathcal{A}$, $\mathcal{B}$ are the regularization parameters which control sparsity and low-rankness. In general, elastic net regularization ($\| \cdot \|_1 + \| \cdot \|_2$) [44] may be used; however in this formulation, we propose to utilize trace-norm in conjunction with ℓ_1-norm for learning representation in RBM (DBM). While ℓ_1-norm induces sparsity in the weight matrix, trace-norm induces features to have low-rankness. The weight matrix learned by the updated loss function has the benefits of both the regularizations and as shown in experimental results, improves the overall verification performance.

The size of the first two layers of the DBM is set to $2 \times \left(\frac{M}{4} \times \frac{N}{4}\right)$ and the final layer is set to $\frac{MN}{4}$. A pre-training approach [39] combined with generative fine-tuning [45] is followed to train the DBM. The final hidden layer provides a complex representation of the input which can be utilized for classification.

C. Face Verification using Feature Richness and Deep Learning based Representation

As shown in Fig. 3, the proposed framework utilizes the frame selection, feature extraction, and classification architecture for video based face recognition. During training, the stack of SDAE joint representation and DBM is utilized for facial representation. Let $I_{gallery}$ and I_{probe} be the two detected, preprocessed and geometrically normalized face images to be matched. These images are resized to $M \times N$ (in our experiments, it is 80×100) and converted into vector form. The trained architecture is used to extract the features from $I_{gallery}$ and I_{probe}, respectively. According to the previous discussion, the input to the feature extraction module is the MN size image vector and the output is a vector of length $\left(\frac{MN}{4}\right)$. Features are extracted for each selected frame in a video and given as input to a five layer neural network (one input layer - 3 hidden layers - one output layer) for classification (verification). The neural network classifier is trained to verify a pair of input images (frames) input as a concatenated feature vector of size $\frac{MN}{2}$, using all the frames in the training videos. The output of the network is a scalar match score.

During testing, the most feature-rich frames are selected from each of the gallery and probe videos, and matched using the proposed feature extraction and matching algorithm. The output of neural network (classifier) is undecimated and match scores are computed. The videos to be matched may have significant variations in quality and feature-richness. It has been shown in literature that if the images are of very different quality, then the matching performance may deteriorate [46]. Therefore, we perform a post-processing step to select frame-pairs with similar feature-richness and discard the remaining pairs. Let $\mathcal{V}_1$ and $\mathcal{V}_2$ be the two videos to be matched, a pair-wise feature-richness value is computed for each possible frame-pair using the algorithm explained in Section II-A.

$$\left[m_{1,1}m_{1,2}; m_{2,1}m_{2,2}; ..., m_{i,1}m_{j,2}; ..., m_{\mathcal{N}_{1,1}}m_{\mathcal{N}_{2,2}} \right] \quad (25)$$

$m_{i,1}m_{j,2}$ denotes the product of feature-richness value associated with the pair formed by the i^{th} frame from $\mathcal{V}_1$ and the j^{th} frame from $\mathcal{V}_2$. $\mathcal{N}_1$ and $\mathcal{N}_2$ denote the total number of selected frames from $\mathcal{V}_1$ and $\mathcal{V}_2$ respectively. Let $\sigma_m^{'}$ be the standard deviation and $\mu_m^{'}$ be the mean pertaining to the set of the pair-wise feature-richness values for all pairs possible between $\mathcal{V}_1$ and $\mathcal{V}_2$. To finally select the pairs for decision making, following equation is utilized:

$$\Upsilon_{i,j} = \left\{ \begin{array}{ll} 1, & \text{if } m_{i,1}m_{j,2} \geq \mu_m^{'} + \frac{\sigma_m^{'}}{2} \\ 0, & \text{otherwise} \end{array} \right\} \quad (26)$$

If the combined score of a pair $f_{i,1}f_{j,2}$ is more than the threshold, i.e., if $\Upsilon_{i,j} = 1$, then this pair is considered for computing the match score. While pairs with $\Upsilon_{i,j} < 1$ are not considered for verification, other selected frame-pairs are weighted according to the joint feature-richness value. For frame-pair $f_{i,1}f_{j,2}$, this weight is computed as $\Upsilon_{i,j}m_{i,1}m_{j,2}$. A pair where both participating frames are highly feature-rich is assigned a higher weight compared to other combinations. Here, facial coordinates obtained during face detection are used to ensure that frontal-only and semi-profile images are not matched with profile faces (i.e, when pose variations are very large). The final match score is computed in the form of a weighted sum of scores obtained from each participating

TABLE II
DETAILS OF THE YOUTUBE AND PASC DATABASES.

Database	No. of		Avg. no. of	
	Subjects	Videos	Videos per subject	Frames per video
YouTube Faces	1595	3425	2.15	181.3
PaSC (Handheld)	265	1401	4 to 7	234.8
PaSC (Control)	265	1401	4 to 7	239.0

frame-pair. The undecimated/unthresholded network (classifier) output of these pairs are combined using weighted sum rule [28] and a verification threshold is applied to provide the final decision of accept or reject (same or not same) at a fixed false accept rate.

III. RESULTS AND ANALYSIS

In order to evaluate the efficacy of the proposed algorithm, face verification[2] experiments are performed on two popular video benchmark databases: YouTube Faces [3] and the Point and Shoot Challenge [2]. Three different experiments are performed to demonstrate the efficacy of the proposed algorithm.

- compare the performance of state-of-the-art results reported on these databases with the proposed algorithm,
- evaluate the effectiveness of individual components of the proposed algorithm, and
- evaluate the generalization capability by evaluating the performance with cross database training and testing sets.

A. Database and Experimental Protocol

The YouTube Faces database contains 3,425 videos downloaded from YouTube belonging to 1,595 individuals. The PaSC database contains 1,401 handheld and 1,401 control (high resolution) videos pertaining to 265 individuals. Videos in the PaSC database capture individuals in various indoor and outdoor locations while performing a predefined activity. The details of both the databases are summarized in Table II. Both YouTube Faces database [3] and PaSC database [2] have predefined experimental protocols. For the YouTube faces database, we have followed the restricted protocol which consists of 10 splits, each containing 250 genuine and 250 impostor pairs. No information outside of these splits is used during any stage of the evaluation. The results are reported with 10 fold cross validation, 9 splits are used for training and one split for testing.

The PaSC database contains videos from a handheld camera of low resolution and a control camera of high resolution. The handheld-to-handheld experiment evaluates the accuracy of an algorithm when matching videos of low resolution, whereas the control-to-control evaluates the accuracy for high resolution videos. The experiments are performed for both handheld-to-handheld and control-to-control protocols of video face recognition. Training is performed on a separate set of training videos provided with the database and the signature sets already provided with the PaSC database are used to select the pairs for testing.

For both databases, the training data is divided into two parts: first part is utilized as unlabeled data for training the proposed joint representation model and second part is used for supervised training. Data augmentation with different image processing operations such as mirror/flip, color/grayscale, and jittering, are also applied to increase the training database size. After training, the proposed algorithm is evaluated on the testing data. The metadata and annotations provided with each database are used to perform face detection and pose detection (to determine pose) as applicable. Receiver Operating Characteristic (ROC) curves are computed for each experiment and the verification accuracies are reported at multiple false accept rates.

B. Experimental Results

1) Results on YouTube Faces Database: ROC curves of existing algorithms and the proposed face recognition algorithm on the YouTube faces database are shown in Fig. 7. It is evident that the proposed algorithm not only achieves high accuracy at low FARs, but also achieves state-of-the-art performance of 0.93 GAR at equal error rate, without outside training data. We next analyzed selected top-performing algorithms to understand their performance at 0.01 FAR which is more pragmatic with respect to real world scenarios. As shown in Fig. 8, the proposed algorithm substantially outperforms these algorithms at lower FARs. At 0.01 FAR, the proposed algorithm achieves GAR of 0.79 whereas, the next best GAR is 0.54 by DeepFace[3]. It is our assertion that selection of feature-rich frames and the proposed joint representation architecture helps to yield state-of-the-art face verification performance.

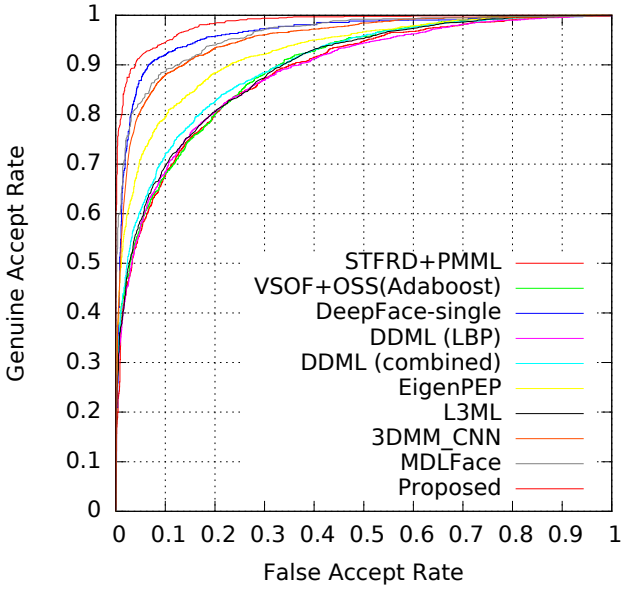


Fig. 7. ROC curves comparing the verification performance of the proposed algorithm with existing results reported on the YTF database webpage.

[2]In biometrics, recognition has two components: verification (1:1 matching) and identification (1:N) matching. In this paper, we have interchangeably used verification and recognition to report 1:1 matching performance.

[3]Since the ROC curve of FaceNet is not available, the results of FaceNet at different FARs could not be reported.

2) Results on PaSC Database: As explained in Section 3.1, Point and Shoot Challenge database has two protocols: handheld and control. Table III summarizes the results of the proposed algorithm along with existing results reported on both the protocols. Beveridge *et al.* [2] reported the performance of PittPatt and Local Region Principal Component Analysis (LRPCA) on both handheld and control subsets. The results show that at 0.01 FAR, the GAR of the proposed algorithm is more than twice of PittPatt. At 0.01 FAR, the proposed algorithm yields 0.93 and 0.96 GAR on the handheld and control subsets, respectively. Beveridge et al. [24], [47] have reported the results of the PaSC Video Face and Person Recognition Competitions. Table III shows the genuine accept rates of the algorithms reported in the competitions along with the results of the proposed algorithm. These results show that the proposed algorithm yields at least 34% higher verification accuracy than existing algorithms that have not utilized external data for training.

3) Impact of Frame Selection: Frame selection is an integral component of the proposed algorithm. The algorithm selects feature-rich frames from the given video and utilizes them for video to video matching. To evaluate the effectiveness of the proposed frame selection algorithm, multiple experiments are performed, including comparison with standard image quality measures.

Ideally, if the frames are selected optimally, then they should yield the best verification performance. To evaluate this, we have compared the verification performance of the proposed feature-rich frames with only frontal frames and when frames are selected randomly. Fig. 9 shows sample frames from the PaSC database. It illustrates randomly selected frames, frontal frames, most feature-rich frames and the least feature-rich frames as well. It can be observed that the most feature-rich frames are distinct in nature and of good quality whereas, the least feature-rich frames computed using the proposed frame selection algorithm do not contain very distinguishing information and are of poor quality. It is also interesting to note that the most feature-rich frames are not necessarily the frontal frames. The experiments are performed with both YouTube and PaSC databases and the results are presented in Fig. 10. It is evident that selecting the most feature-rich frames provides the best performance across all three protocols. Correlating these images with the accuracies re-emphasizes our hypothesis that frontal frames are not always optimal and hence do not necessarily provide the best verification results.

We also compare the performance of the proposed frame selection approach with frame selection based on no-reference image quality metrics namely BRISQUE [32], NIQE [30], and SSEQ [31]. The source codes provided by the respective authors have been utilized for each of these approaches. Similar to the proposed approach, frames are selected based on the quality measure and used in the proposed framework. We have also evaluated the performance of our preliminary frame selection approach [25] and the verification results obtained with each of the frame selection algorithms and the proposed face recognition algorithm are presented in Table IV. We observe that using any of the existing quality assessment algorithms results in a noticeable decline in the verification performance. On the YouTube faces database, the performance varies from 0.62 to 0.79 GAR, whereas on the handheld subset of the PaSC database the performance varies from 0.82 to 0.93 GAR by only changing the frame selection approach. The proposed feature-richness based frame selection approach consistently outperforms the quality based measures on all the protocols of both the databases. This experiment suggests that high image quality may not represent high feature richness and can affect the overall verification performance.

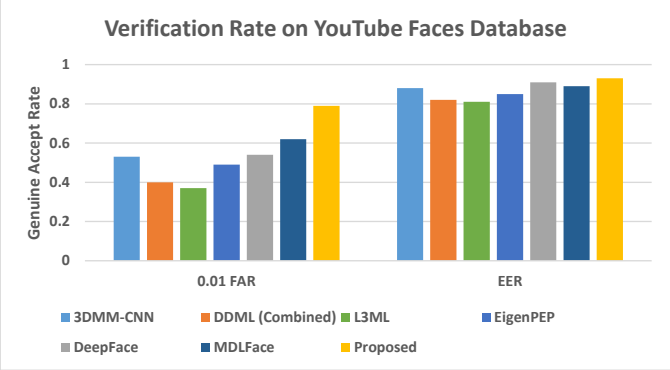


Fig. 8. Summarizing the verification performance of the proposed algorithm and state-of-the-art algorithms on the YouTube Faces database.

TABLE III
VERIFICATION ACCURACIES ON THE PASC DATABASE. RESULTS OF EXISTING ALGORITHMS ARE REPORTED FROM RESPECTIVE REFERENCES.

Algorithm	GAR at 0.01 FAR	
	Handheld	Control
ISV-GMM [47]	0.05	-
LBP-SIFT-WPCA-SILD [47]	0.09	-
PLDA-WPCA-LLR [47]	0.19	-
Eigen-PEP [47]	0.26	-
LRPCA Baseline [2]	0.08	0.10
PittPatt Baseline [2]	0.38	0.49
Surrey [24]	0.13	0.20
SIT [24]	0.31	0.35
Uni-Lj [24]	0.33	0.39
UTS [24]	0.38	0.48
CAS [24]	0.59	0.58
MDLFace [25]	0.89	0.94
Proposed	**0.93**	**0.96**

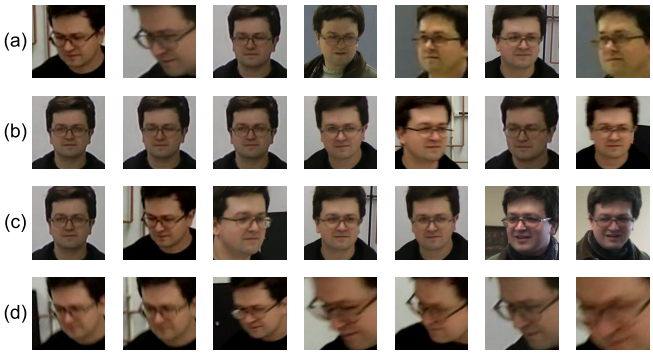
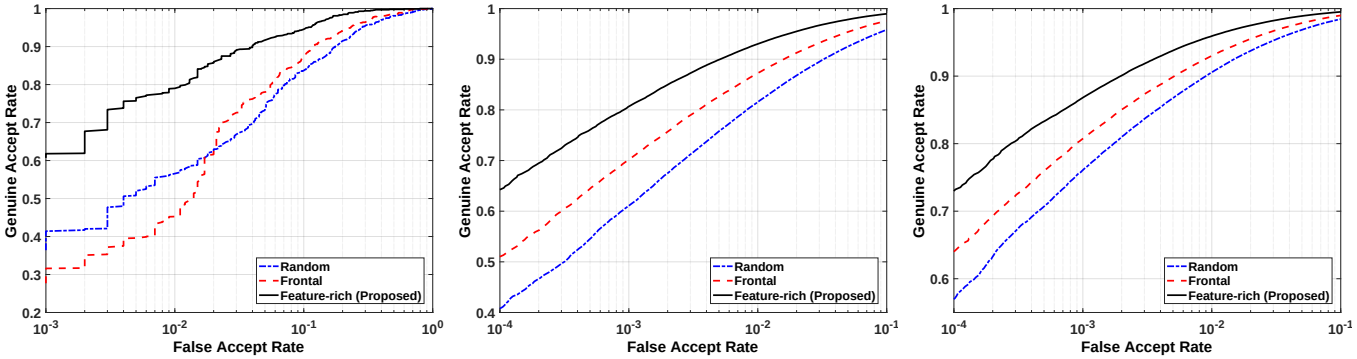


Fig. 9. Sample frames from the PaSC database: (a) random frames, (b) frontal frames, (c) most feature-rich frames, and (d) least feature-rich frames.



(a) YouTube Faces database (b) Handheld subset of PaSC database (c) Control subset of PaSC database

Fig. 10. ROC curves comparing the verification performance of the proposed algorithm with frame selection approaches on the two databases.

TABLE IV
COMPARING THE RESULTS OF THE PROPOSED FRAME SELECTION ALGORITHM WITH EXISTING IMAGE QUALITY ASSESSMENT ALGORITHMS AND RANDOM FRAME SELECTION.

Frame Selection	Algorithm	GAR at 0.01 FAR		
		YTF	PaSC Handheld	PaSC Control
All		0.74	0.89	0.92
Image Quality	BRISQUE [32]	0.62	0.82	0.84
	NIQE [30]	0.62	0.83	0.82
	SSEQ [31]	0.62	0.82	0.82
Memorability	MDLFace [25]	0.69	0.89	0.94
Proposed Feature Richness	25	0.75	0.91	0.94
	50	0.77	0.91	0.93
	Adaptive	**0.79**	**0.93**	**0.96**

TABLE V
ANALYZING THE PERFORMANCE OF INDIVIDUAL COMPONENTS OF THE PROPOSED ALGORITHM FOR FACE RECOGNITION.

Modified Architecture	GAR at 0.01 FAR		
	YouTube	PaSC	
		Handheld	Control
1 Layer DAE only	0.21	0.09	0.12
2 Layer SDAE only	0.39	0.28	0.39
DBM only	0.41	0.48	0.49
SDAE+DBM	0.61	0.87	0.93
Proposed: SDAE+DBM with joint representation	**0.79**	**0.93**	**0.96**

This is consistent with existing observations in *biometrics quality* literature [46]. We further analyze the performance of the proposed algorithm with fixed number of frames i.e., without adaptive approach, as well as without using any frame selection. As shown in Table IV, with all frames, top-25 and top-50 feature-rich frames, the verification accuracies are relatively lower. This shows the usefulness of the "adaptive" nature of the proposed algorithm. These experiments also validate our hypothesis that not all frames are useful for video face recognition.

4) Analysis of Deep Learning Architecture: Individual components of the proposed deep learning framework are experimentally evaluated to determine the efficacy of the algorithms. In this experiment, only one component is changed and the remaining components of the proposed framework are left unchanged and only the feature extractor module is varied across different experiments. These components include: (a) single layer denoising autoencoder, (b) two layer SDAE, (c) DBM, and (d) SDAE+DBM without the proposed joint representation layer.

Table V summarizes the GAR at 0.01 FAR for each of these components on both YouTube and PaSC databases (using feature-rich frames). From the results, it is evident that both SDAE and DBM are required in the proposed architecture to extract meaningful representation for face recognition. Using only DBM provides better performance than only using a 2-layer SDAE. However, neither DBM nor SDAE is able to achieve even 50% verification accuracies individually. A significant improvement is observed when SDAE and DBM are stacked sequentially. The proposed joint representation further improves the performance of the architecture, resulting in an improvement of up to 0.18 in GAR for the YouTube faces database. As mentioned previously, the joint representation combines different layers of feature granularity and from the results, it is evident that it is able to further improve upon the features learned by the deep architecture. This observation strengthens the requirement for the additional layer of learning after SDAE before the features are utilized by DBM.

An additional experiment is performed to evaluate the efficacy of the addition of trace-norm regularization. For this experiment, ℓ_2-norm, ℓ_1-norm, elastic net (ℓ_1+ℓ_2 norm), trace-norm (ℓ_τ) only, and ($\ell_1 + \ell_\tau$) are evaluated in the proposed framework (as shown in Eq. 24). For these regularizers, we observe that (ℓ_1+ℓ_τ) yields the best results followed by elastic net. Incorporating single norms i.e., ℓ_1-norm and ℓ_2-norm only, yield almost similar performance and are 1-2% (at 1% FAR) less than ($\ell_1 + \ell_\tau$) regularization.

The number of parameters in a deep neural network is determined by the weights and bias of each layer. The proposed algorithm involves a total of 22.5 million parameters whereas, other deep architectures such as Deepface [11] contain many more parameters (e.g. 120 million for Deepface). We observe that even with a relatively small number of parameters, the proposed algorithm achieves higher performance than Deepface. While architectures proposed in [18] and [17] perform

TABLE VI
GAR FOR CROSS DATABASE EXPERIMENTS AT 0.01 FAR.

Training Set	Testing Set		
	YTF	PaSC-Handheld	PaSC-Control
YTF	0.79	0.72	0.78
PaSC	0.43	0.93	0.96
PaSC + YTF	**0.83**	**0.96**	**0.97**

better on the YouTube database than the proposed algorithm, both involve a much higher number of parameters and have utilized large amounts (2.6 million and 200 million images respectively) of training data (the results are reported on the unrestricted setting of YouTube). It is to be noted that for these experiments, the proposed algorithm is not trained with external training data.

5) Cross Database Experiments: The generalizability of an algorithm can be evaluated in situations where the training and testing data belong to different databases, i.e, cross-database experiments. To evaluate the effectiveness of the proposed algorithm in cross database scenarios, we have performed three different experiments:

- Training and testing databases belong to the same database. For instance, training with YouTube faces train set and testing with YouTube faces test set.
- Training and testing databases belong to different database. For instance, training with YouTube faces train set and testing with PaSC test set.
- Training database is from multiple databases whereas, the testing is performed with a single database. For instance, training with both YouTube faces and PaSC train sets and testing on YouTube faces test set.

The results of all three experiments are presented in Table VI. On training with the YouTube Faces database and testing with the PaSC database, the proposed algorithm yields 0.72 GAR at 0.01 FAR which is considerably better than the results of many existing algorithms. On the other hand, the performance on the Youtube faces database suffers heavily when training data is taken only from the PaSC database. This may be due to the fact that the overall quality of faces in the Youtube video faces database is lower than the training set of the PaSC challenge database. Since the representation module has not seen low quality frames and noisy faces during training, it is unable to perform well on the YouTube database. On combining the training set from both the databases, i.e. PaSC + YTF training, the accuracies of both testing cases are improved. This is a well understood phenomena in deep learning - more training data is useful in improved representation and thereby achieving higher accuracies.

C. Comparison with Recent CNN based Algorithms

We next compare the performance of the proposed algorithm with some recently proposed CNN based algorithms on the benchmark protocols of the YTF and PaSC face databases. As shown in Table I, convolutional neural networks have demonstrated state-of-the-art results in deep learning based video face recognition; however, they generally use external data for training. Therefore, we have reported the results of the proposed algorithm in three settings: (i) without any external training data, (ii) using YTF and PaSC for training (as discussed in Section III-B), and (iii) using external training data of 2.48 million (with augmentation).

Table VII summarizes the results of the proposed and existing algorithms. Results of existing algorithms are reported directly from the associated publications, and the results for [18] are taken from [19]. Since we have not manually pruned the PaSC database for falsely detected faces, we report the corresponding performance values for [19]. We observe that even without utilizing any external data, the proposed algorithm is able to achieve comparable accuracies. Using large training data, the accuracy improves and with 2.48 million training data, the verification rate is higher compared to existing algorithms. In terms of computational requirements, on a 32 core server with Tesla K80 GPU with 512 GB RAM, the proposed algorithm requires approximately 29 hours to train with external data. Once the model is trained, it requires about 2 seconds to match two videos.

On analyzing the architectures, we observed that in order to optimize the network for a given problem, a deep CNN architecture requires a large number of layers, which results in a large number of parameters to optimize. This requires large number of training data so that all the parameters of the network can be estimated without overfitting. The proposed algorithm achieves comparable performance with a network of lesser depth (9 layers as compared to 22 layers in [20]) with relatively less training data. We also assert that the proposed architecture can be applied to solve other challenging problems where relatively less labeled data is available such as newborn face recognition [50].

IV. CONCLUSION

Verifying identities in videos has several applications in social media, surveillance, and law enforcement. Existing approaches have achieved high verification accuracies at equal error rate; however, achieving high performance at low false accept rate is still an arduous research challenge. In this research, a novel video face verification algorithm is proposed which utilizes frame selection and deep learning based feature representation. The proposed algorithm starts with adaptively selecting feature-rich frames from input videos using wavelet decomposition and entropy. The proposed deep learning architecture which combines SDAE joint representation with DBM is used to extract features from the selected frames. The extracted representations from two videos are matched using a feed forward neural network. The results are demonstrated on the challenging Point and Shoot Challenge and YouTube Faces databases. The comparison with state-of-the-art results on both the databases show that the proposed algorithm provides the best results on both the databases at low false accept rate, even with limited training data. Apart from the benchmark protocols of both the databases, several additional experiments have been performed to show the effectiveness of the proposed contributions: (i) joint feature learning in an autoencoder, (ii) sparse and low rank regularization in DBM, and (iii) combination of SDAE and DBM in the proposed architecture.

TABLE VII
COMPARING THE VERIFICATION ACCURACY OF RECENT CNN BASED METHODS WITH THE PROPOSED ALGORITHM.

Algorithm	External Training Data	Layers	YTF (at EER)	PaSC (at 1% FAR)	
				Control	Handheld
Trunk-Branch Ensemble CNNs with Batch Normalization [19]#	2.68 Million$	18 + 11 + 11*	94.9	98.0	97.0
VGG Face [18]+	2.62 Million	21	**97.4**	91.3	87.0
GoogLeNet [21] features with aggregation [20]	3 Million	22	95.5	-	-
CNN-3DMM Estimation [22]	0.49 Million	101	88.8	-	-
Proposed SDAE-DBM Joint Representation	No	9	93.4	95.9	93.1
	YTF + PaSC	9	95.0	96.6	96.1
	2.48 Million	9	95.4	**98.1**	**97.2**

*#Results on YTF are obtained from [19], results on PaSC are obtained from [48]. $2.68 million images obtained by augmenting 0.49M original images from the CASIA-WebFace database [49] using horizontal flipping and image jittering as explained in [19]. *The method uses a primary network with 18 layers and two secondary networks with 11 layers each. +PaSC results are obtained from [19].*

As a future research, we plan to extend the algorithm for "face recognition in crowd" with multiple subjects in each video.

ACKNOWLEDGMENT

This research is supported by MEITY, India. Gaurav Goswami is partly supported by IBM PhD fellowship.

REFERENCES

[1] "http://english.cas.cn/Ne/CASE/200808/t20080815_18764.shtml."
[2] J. Beveridge, P. Phillips, D. Bolme, B. Draper, G. Given, Y. M. Lui, M. Teli, H. Zhang, W. Scruggs, K. Bowyer, P. Flynn, and S. Cheng, "The challenge of face recognition from digital point-and-shoot cameras," in *IEEE Conference on Biometrics: Theory, Applications and Systems*, 2013.
[3] L. Wolf, T. Hassner and I. Maoz, "Face recognition in unconstrained videos with matched background similarity," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 529–534.
[4] L. Wolf and N. Levy, "The svm-minus similarity score for video face recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3523–3530.
[5] H. Li, G. Hua, Z. Lin, J. Brandt, and J. Yang, "Probabilistic elastic matching for pose variant face verification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3499–3506.
[6] Z. Cui, W. Li, D. Xu, S. Shan, and X. Chen, "Fusing robust face region descriptors via multiple metric learning for face recognition in the wild," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3554–3561.
[7] H. Mendez-Vazquez, Y. Martinez-Diaz, and Z. Chai, "Volume structured ordinal features with background similarity measure for video face recognition," in *International Conference on Biometrics*, 2013.
[8] H. S. Bhatt, R. Singh, and M. Vatsa, "On recognizing faces in videos using clustering based re-ranking and fusion," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 7, pp. 1056– 1068, 2014.
[9] J. Y. Junlin Hu, Jiwen Lu and Y.-P. Tan, "Large margin multi-metric learning for face and kinship verification in the wild," in *Asian Conference on Computer Vision*, 2014.
[10] J. Hu, J. Lu, and Y. Tan, "Discriminative deep metric learning for face verification in the wild," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1875 – 1882.
[11] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the Gap to Human-Level Performance in Face Verification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1701 – 1708.
[12] W. Wang, R. Wang, Z. Huang, S. Shan, and X. Chen, "Discriminant analysis on riemannian manifold of gaussian distributions for face recognition with image sets," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2015.
[13] N. Khan, X. Nan, A. Quddus, E. Rosales, and L. Guan, "On video based face recognition through adaptive sparse dictionary," in *IEEE International Conference and Workshops on Automatic Face and Gesture Recognition*, 2015.
[14] H. Li, G. Hua, X. Shen, Z. Lin, and J. Brandt, "Eigen-PEP for video face recognition," in *Asian Conference on Computer Vision*, 2014.
[15] H. Li and G. Hua, "Hierarchical-PEP model for real-world face recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2015.
[16] Y. Sun, X. Wang, and X. Tang, "Deeply learned face representations are sparse, selective, and robust," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2015.
[17] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2015.
[18] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *British Machine Vision Conference*, vol. 1, no. 3, 2015.
[19] C. Ding and D. Tao, "Trunk-Branch Ensemble Convolutional Neural Networks for Video-based Face Recognition," *ArXiv e-prints*, July 2016.
[20] J. Yang, P. Ren, D. Chen, F. Wen, H. Li, and G. Hua, "Neural Aggregation Network for Video Face Recognition," *ArXiv e-prints*, March 2016.
[21] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.
[22] A. T. Tran, T. Hassner, I. Masi, and G. Medioni, "Regressing robust and discriminative 3D morphable models with a very deep neural network," arXiv:1612.04904v1, Tech. Rep., 2016.
[23] Z. Huang, R. Wang, S. Shan, and X. Chen, "Projection metric learning on grassmann manifold with application to video based face recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2015.
[24] J. Beveridge, H. Zhang, B. Draper, P. Flynn, Z. Feng, P. Huber, J. Kittler, Z. Huang, S. Li, Y. Li, M. Kan, R. Wang, S. Shan, X. Chen, H. Li, G. Hua, V. Struc, J. Krizaj, C. Ding, D. Tao, and P. Phillips, "Report on the FG 2015 video person recognition evaluation," in *IEEE International Conference and Workshops on Automatic Face and Gesture Recognition*, 2015.
[25] G. Goswami, R. Bhardwaj, R. Singh, and M. Vatsa, "MDLFace: Memorability augmented deep learning for video face recognition," in *IEEE International Joint Conference on Biometrics*, 2014.
[26] J. R. Barr, K. W. Bowyer, P. J. Flynn, and S. Biswas, "Face recognition from video: A review," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 26, no. 05, 2012.
[27] K. Lee, J. Ho, M. 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.
[28] A. A. Ross, K. Nandakumar, and A. K. Jain, *Handbook of Multibiometrics*. Springer, 2006.
[29] A. Rrnyi, "On measures of entropy and information," in *Berkeley Symposium on Mathematical Statistics and Probability*, 1961, pp. 547–561.
[30] A. Mittal, R. Soundararajan, and A. C. Bovik, "Making a "completely blind" image quality analyzer," *IEEE Signal Processing Letters*, vol. 20, no. 3, pp. 209–212, 2013.
[31] L. Liu, B. Liu, H. Huang, and A. C. Bovik, "No-reference image quality assessment based on spatial and spectral entropies," *Signal Processing: Image Communication*, vol. 29, no. 8, pp. 856–863, 2014.
[32] A. Mittal, A. K. Moorthy, and A. C. Bovik, "No-reference image quality assessment in the spatial domain," *IEEE Transactions on Image Processing*, vol. 21, no. 12, pp. 4695–4708, 2012.
[33] W. Liu, Z. Li, and X. Tang, "Spatio-temporal embedding for statistical face recognition from video," in *European Conference on Computer Vision*, 2006, pp. 374–388.
[34] U. Park, A. K. Jain, and A. Ross, "Face recognition in video: Adaptive fusion of multiple matchers," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[35] R. R. Jillela and A. Ross, "Adaptive frame selection for improved face recognition in low-resolution videos," in *International Joint Conference on Neural Networks*, 2009, pp. 1439–1445.
[36] Y. Bengio, L. Pascal, P. Dan, and H. Larochelle, "Greedy layer-wise training of deep networks," in *Advances in Neural Information Processing Systems*, 2007, vol. 19, pp. 153–160.
[37] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *Journal of Machine Learning Research*, vol. 11, no. 12, pp. 3371–3408, 2010.
[38] R. Salakhutdinov and G. E. Hinton, "Deep boltzmann machines," in *International Conference on Artificial Intelligence and Statistics*, 2009, pp. 448–455.
[39] G. E. Hinton and R. Salakhutdinov, "A better way to pretrain deep boltzmann machines," in *Advances in Neural Information Processing Systems*, 2012, vol. 25, pp. 2447–2455.
[40] G. E. Hinton, and R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.
[41] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting." *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.
[42] Y. Peng, A. Ganesh, J. Wright, W. Xu, and Y. Ma, "RASL: Robust alignment by sparse and low-rank decomposition for linearly correlated images," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 763–770.
[43] P.-A. Savalle, E. Richard, and N. Vayatis, "Estimation of simultaneously sparse and low rank matrices." in *International Conference on Machine Learning*, 2012.
[44] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 67, no. 2, pp. 301–320, 2005.
[45] G. E. Hinton, "Learning multiple layers of representation," *Trends in Cognitive Sciences*, vol. 11, pp. 428–434, 2007.
[46] S. Bharadwaj, M. Vatsa, and R. Singh, "Biometric quality: a review of fingerprint, iris, and face," *EURASIP Journal on Image and Video Processing*, vol. 2014, no. 1, pp. 1–28, 2014.
[47] J. R. Beveridge, H. Zhang, P. J. Flynn, Y. Lee, V. E. Liong, J. Lu, M. de Assis Angeloni, T. de Freitas Pereira, H. Li, G. Hua, V. Struc, J. Krizaj, and P. J. Phillips, "The IJCB 2014 pasc video face and person recognition competition," in *IEEE International Joint Conference on Biometrics*, 2014.
[48] W. J. Scheirer, P. J. Flynn, C. Ding, G. Guo, V. Struc, M. A. Jazaery, K. Grm, S. Dobrisek, D. Tao, Y. Zhu, J. Brogan, S. Banerjee, A. Bharati, and B. RichardWebster, "Report on the BTAS 2016 video person recognition evaluation," in *IEEE 8th International Conference on Biometrics Theory, Applications and Systems*, 2016.
[49] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Learning face representation from scratch," *arXiv preprint arXiv:1411.7923*, 2014.
[50] S. Bharadwaj, H. S. Bhatt, M. Vatsa, and R. Singh, "Domain specific learning for newborn face recognition," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 7, pp. 1630–1641, July 2016.

Gaurav Goswami (S'12) received his Bachelor in Technology degree in Information Technology in 2012 from the Indraprastha Institute of Information Technology (IIIT) Delhi, India where he is currently pursuing a PhD. He has received the IBM PhD Fellowship. His main areas of interest are image processing, computer vision and their application in biometrics. He is the recipient of best poster award in IEEE BTAS, 2013 and the Best Doctoral Consortium presentation award in IJCB, 2014.

Mayank Vatsa (S'04 - M'09 - SM'14) received the Ph.D. degree in Computer Science from West Virginia University, Morgantown, USA, in 2008. He is currently an Associate Professor with the IIIT Delhi, India and Adjunct Associate Professor with the West Virginia University, USA. His areas of interest are biometrics, image processing, computer vision, and information fusion. He is a recipient of the AR Krishnaswamy Faculty Research Fellowship, the FAST Award by DST, India, and several best paper and best poster awards in international conferences. He has published more than 200 peer-reviewed papers. He is also the Vice President (Publications) of IEEE Biometrics Council, an Associate Editor of the IEEE ACCESS, and an Area Editor of Information Fusion (Elsevier). He served as the PC Co-Chair of ICB 2013, IJCB 2014, and ISBA 2017.

Richa Singh (S'04 - M'09 - SM'14) received the Ph.D. degree in Computer Science from West Virginia University, Morgantown, USA, in 2008. She is currently an Associate Professor with the IIIT Delhi, India and an Adjunct Associate Professor with the West Virginia University, USA. Her areas of interest are biometrics, pattern recognition, and machine learning. She is a recipient of the Kusum and Mohandas Pai Faculty Research Fellowship at the Indraprastha Institute of Information Technology, the FAST Award by DST, India, and several best paper and best poster awards in international conferences. She is also an Editorial Board Member of Information Fusion (Elsevier), and Associate Editor of IEEE Access and the EURASIP Journal on Image and Video Processing (Springer). She is serving as the General Co-Chair of ISBA2017.