Learning Boltzmann Distance Metric for Face Recognition

Truyen Tran
Department of Computing
Curtin University
Bentley, Western Australia, Australia
t.tran2@curtin.edu.au

Dinh Q. Phung and Svetha Venkatesh
School of Information Technology
Deakin University
Geelong, Victoria, Australia
{*dinh.phung,svetha.venkatesh*}*@deakin.edu.au*

***Abstract*—We introduce a new method for face recognition using a versatile probabilistic model known as Restricted Boltzmann Machine (RBM). In particular, we propose to regularise the standard data likelihood learning with an information-theoretic distance metric defined on intra-personal images. This results in an effective face representation which captures the regularities in the face space and minimises the intra-personal variations. In addition, our method allows easy incorporation of multiple feature sets with controllable level of sparsity. Our experiments on a high variation dataset show that the proposed method is competitive against other metric learning rivals. We also investigated the RBM method under a variety of settings, including fusing facial parts and utilising localised feature detectors under varying resolutions. In particular, the accuracy is boosted from 71.8% with the standard whole-face pixels to 99.2% with combination of facial parts, localised feature extractors and appropriate resolutions.**

***Keywords*-Face recognition; metric learning; Restricted Boltzmann Machines; information fusion.**

I. Introduction

Face recognition is often cast as a matching problem between a query face and faces in the database. The key is to prepare a representation that is rich enough to capture important facial properties under noisy measurements, and at the same time, supports matching under large intra-personal variations [1], [2], [3], [4], [5].

In this paper, we advocate the use of a versatile probabilistic model known as Restricted Boltzmann Machine (RBM) [6], [7] for face recognition. A RBM is a 2-layer Markov random field where the input layer represents facial features $\boldsymbol{x}$, and the hidden layer encodes binary factors of variations $\boldsymbol{h}$. As a modelling tool, the RBM is natural to capture the *regularity* in the facial space as well as the higher-order dependencies among features. As a data processing tool, the RBM transforms the features into a more robust (probabilistic) representation $P(\boldsymbol{h}|\boldsymbol{x})$ which can be used for classification and recognition. The model has recently been shown to be useful in a variety of vision tasks including object recognition [8], [9], learning image transformation [10] and generating facial expression [11]. However, the standard application of RBM for face recognition [12] can be limited since capturing the regularities in face data alone is not enough to separate a person from another if the intra-personal variations are high (e.g., due to poses and lighting conditions).

To that end, we propose a solution by regularising the objective function during training time so that the data likelihood $P(\boldsymbol{x})$ is maximised whilst the intra-personal distances $\mathcal{D}(\boldsymbol{x}, \boldsymbol{x}')$ are minimised. In other words, we attempt to learn a face representation that balances between the ability to explain the facial data well and the invariance to intra-personal variations. In particular, the representation is based on the posteriors of hidden factors given the facial features $P(\boldsymbol{h}|\boldsymbol{x})$. The regulariser is based on an information-theoretic distance metric known as symmetrized Kullback-Leibler divergence between the posteriors of intra-personal image pairs. While maximising the data likelihood is akin to capturing the facial variances in PCA [1], minimising the regulariser is equivalent to learning a non-linear *distance metric* between faces [13]. The use of intra-personal metric-regularised RBMs contributes to the existing face recognition literature with a novel probabilistic and non-linearity treatment.

We investigate the proposed RBM under various settings. Firstly, it may be more useful to fuse multiple sources of information rather than using just one. For example, the face recognition literature offers multiple facial representations (e.g., whole-face versus component-wise [14]) and a variety of localised feature detectors (e.g., local binary patterns (LBP) [4] and the Gabor filters [2]). Under RBMs, these information sources can be naturally integrated under shared representation with a controllable level of sparsity. Secondly, it has been conjectured that image resolutions can be critical for recognition as it is evident that human tends to focus on some small, informative and high-resolution areas while skimming over the whole face [11]. Our experiments with a database of high variations in pose and lighting conditions demonstrate that the proposed RBM is competitive against well-known face recognition methods. By taking into these settings into account, we can boost the accuracy from 71.2% under the standard whole-face pixel-based feature representation to 99.2% using the LBP feature representation of multiple facial parts.

We present the RBM and metric-based training for face recognition in the next section. The evaluation of the

proposed method is presented in Secion III. Section IV concludes the paper.

II. Restricted Boltzmann Machines with Metric-based Training

A. Restricted Boltzmann machines

Restricted Boltzmann Machine (RBM) [6], [15], [8], [7] for face recognition is a 2-layer probabilistic network in which the input layer represents facial features and the hidden layer represents binary factors of variation. Thus, a face is jointly generated from a set of *activated* hidden factors, which supposedly reflect structural information such as facial parts and variations due to expression, lighting conditions, poses and occlusions.

To be precise, let $\boldsymbol{x} \in \mathbb{R}^M$ be the vector of Gaussian features[1] and $\boldsymbol{h} \in \{0,1\}^K$ be the vector of hidden factors. The RBM is characterised by the Boltzmann distribution[2]

$$P(\boldsymbol{x}, \boldsymbol{h}) = \frac{1}{Z} \exp\left(-\frac{\boldsymbol{x}^\top \boldsymbol{x}}{2} + \boldsymbol{w}^\top \boldsymbol{x} + \boldsymbol{x}^\top \boldsymbol{U} \boldsymbol{h} + \boldsymbol{v}^\top \boldsymbol{h}\right)$$

where Z is the normalisation constant, and $\boldsymbol{w} \in \mathbb{R}^M, \boldsymbol{U} \in \mathbb{R}^{M \times K}, \boldsymbol{v} \in \mathbb{R}^K$ are model parameters. This Boltzmann machine is *restricted* in the sense that it limits direct interactions to those between layers. Given a set of active hidden factors, features are generated as

$$P(\boldsymbol{x}|\boldsymbol{h}) = \prod_i P(x_i|\boldsymbol{h}); \qquad P(x_i|\boldsymbol{h}) = \mathcal{N}(w_i + \boldsymbol{U}_{i\bullet}\boldsymbol{h}; 1)$$

where $\boldsymbol{U}_{i\bullet}$ is the row vector corresponding to the ith feature. On the other hand, given the facial features, the probability of activating hidden factors are

$$P(\boldsymbol{h}|\boldsymbol{x}) = \prod_k P(h_k|\boldsymbol{x}); \;\; P(h_k^1|\boldsymbol{x}) = \sigma\left[v_k + \boldsymbol{x}^\top \boldsymbol{U}_{\bullet k}\right] \quad (1)$$

where we have used h_k^1 as a shorthand for $h_k = 1$, σ is the sigmoid function $\sigma[z] = 1/1+e^{-z}$, and $\boldsymbol{U}_{\bullet k}$ is the column vector corresponding to the kth hidden factor.

The set of activation probabilities $\{P(h_k^1|\boldsymbol{x})\}_{k=1}^K$ can be seen as *probabilistic projections* of the facial features $\boldsymbol{x}$ onto the factor space. For recognition purposes, we can use either these probabilistic projections, which are bounded within the unit interval $(0, 1)$, or their linear counterparts $\left[\boldsymbol{x}^\top \boldsymbol{U} + \boldsymbol{v}\right]$ which are unbounded. For reconstruction, one can use $\boldsymbol{w} + \boldsymbol{U}\hat{\boldsymbol{h}}$ where $\hat{h}_k = P(h_k^1|\boldsymbol{x})$.

[1]We work with Gaussian features in this paper, but the RBM can encode different types, e.g., see [16].

[2]For simplicity, we assume that each feature, when conditioned on the hidden factors, has an unit variance. This can be approximated by appropriate normalisation over training data. For more complicated covariance modelling, we refer to the recent work of [9].

Remark 1: The RBM is somewhat similar to the PCA but with subtle differences: The PCA captures the data variance through orthogonal eigenvectors, thus it is linear and the subspace is continuous. On the other hand, the RBM focuses on explaining the data generation through a *discrete* set of hidden factors without any assumption of orthogonality and linearity. The key representation power comes from the space of exponentially many variations.

B. Facial metric learning

Standard training of RBMs maximises the data likelihood $P(\boldsymbol{x})$. As such, the estimated model captures *regularities* and *variations* in the human faces. However, this is *not* necessarily optimal for recognition purposes which often rely directly on the discriminative power to separate an *identity* from others. In other words, if two facial images are from the same person, their activation probabilities should be more similar than those from different persons. To be more concrete, let f and g are face indices, and $\mathcal{I}(f)$ is the identity of face f, then $P(h_k^1|\boldsymbol{x}^{(f)})$ should be close to $P(h_k^1|\boldsymbol{x}^{(g)})$ if $\mathcal{I}(f) = \mathcal{I}(g)$ for any factor k. To that end, we propose to maximise the regularised likelihood as follows $\mathcal{L}_{reg} =$

$$\sum_f \log P(\boldsymbol{x}^{(f)}) - \beta \sum_f \sum_{g \in \mathcal{I}(f)} \mathcal{D}\left(P(\boldsymbol{h}|\boldsymbol{x}^{(g)}), P(\boldsymbol{h}|\boldsymbol{x}^{(f)})\right) \quad (2)$$

where $\beta \geq 0$ is the coefficient controlling the regularisation effect, and $\mathcal{D}(P, Q) \geq 0$ is the distance metric between two distributions P and Q. Thus the new objective function attempts to balance between explaining the facial variations in the feature space and achieving intra-personal invariance in the posterior space.

Maximising the regularised likelihood $\mathcal{L}_{reg}$, however, is difficult due to the intractability of the data likelihood term $P(\boldsymbol{x}) = \sum_{\boldsymbol{h}} P(\boldsymbol{x}, \boldsymbol{h})$, which requires the summing over 2^K combinations of variation factors. In this paper, we resort to an efficient truncated sampling scheme known as Contrastive Divergence (CD) [15] in which the gradient of the log-likelihood is approximated by a very short Markov chain. More specifically, in one-step CD learning, we first sample the hidden factors from the training features as $\tilde{h}_k \sim P(h_k|\boldsymbol{x})$ and then reconstruct the features using $\tilde{\boldsymbol{x}}_i \sim \mathcal{N}(w_i + \boldsymbol{U}_{i\bullet}\tilde{\boldsymbol{h}}; 1)$. The gradient of the log-likelihood with respect to the (column) parameter vector $\boldsymbol{U}_{\bullet k}$ is approximated by[3]

$$\frac{\partial \log P(\boldsymbol{x})}{\partial \boldsymbol{U}_{\bullet k}} \approx P(h_k^1|\boldsymbol{x})\boldsymbol{x} - P(h_k^1|\tilde{\boldsymbol{x}})\tilde{\boldsymbol{x}}$$

For the gradient of the distance metric, according the chain rule

$$\frac{\partial \mathcal{D}(g, f)}{\partial \boldsymbol{U}_{\bullet k}} = \frac{\partial \mathcal{D}(g, f)}{\partial P(h_k^1|f)} \frac{\partial P(h_k^1|f)}{\partial \boldsymbol{U}_{\bullet k}} + \frac{\partial \mathcal{D}(g, f)}{\partial P(h_k^1|g)} \frac{\partial P(h_k^1|g)}{\partial \boldsymbol{U}_{\bullet k}}$$

[3]The derivatives with respect to $\boldsymbol{w}$ and $\boldsymbol{v}$ are omitted here for clarity.

where $\mathcal{D}(g,f)$ is a shorthand for $\mathcal{D}\left(P(\boldsymbol{h}|\boldsymbol{x}^{(g)}), P(\boldsymbol{h}|\boldsymbol{x}^{(f)})\right)$ and $P(h_k^1|f)$ is for $P(h_k^1|\boldsymbol{x}^{(f)})$. The derivative $\frac{\partial \mathcal{D}(g,f)}{\partial P(h_k^1|f)}$ depends on the choice of the distance measure $\mathcal{D}(g,f)$ (Sec. II-C) and is presented in Appendix A. Using Eq. 1, the derivative $\frac{\partial P(h_k^1|\boldsymbol{x})}{\partial \boldsymbol{U}_{\bullet k}}$ is then

$$\frac{\partial P(h_k^1|\boldsymbol{x})}{\partial \boldsymbol{U}_{\bullet k}} = P(h_k^1|\boldsymbol{x})\left(1 - P(h_k^1|\boldsymbol{x})\right)\boldsymbol{x}$$

Finally, stochastic gradient ascent is applied due to the approximate nature of the CD learning

$$\boldsymbol{U}_{\bullet k} \leftarrow \boldsymbol{U}_{\bullet k} + \nu \frac{\partial \mathcal{L}_{reg}}{\partial \boldsymbol{U}_{\bullet k}} \tag{3}$$

for some learning rate $\nu > 0$.

Remark 2: We wish to emphasize that the data likelihood and the regulariser in Eq. 2 can theoretically operate on different datasets, which may not overlap with the database used for recognition. For instance, we can use a large unlabelled dataset for the data likelihood, a smaller pairwise labelled dataset with *unknown identities* for regularisation, and another dataset with the known identities for recognition. We only require that these datasets share the same type of feature representation.

C. Information-theoretic distance measures

For the choice of distance function between two distributions $\mathcal{D}(P,Q)$ we typically expect that $\mathcal{D}(P,Q) = \mathcal{D}(Q,P)$ and $P = \arg\min_Q \mathcal{D}(P,Q)$. In this paper, we will focus on the symmetrized Kullback-Leibler (KL) divergence since it is natural in the space of activation probabilities.

For brevity, we use $\mathrm{KL}(g \parallel f)$ as a shorthand for $\mathrm{KL}\left(P(\boldsymbol{h}|\boldsymbol{x}^{(g)}) \parallel P(\boldsymbol{h}|\boldsymbol{x}^{(f)})\right)$. The KL divergence is given as

$$\begin{aligned}\mathrm{KL}(g \parallel f) &= \sum_{\boldsymbol{h}} P(\boldsymbol{h}|g) \log \frac{P(\boldsymbol{h}|g)}{P(\boldsymbol{h}|f)} \\ &= \sum_k \sum_{h_k \in \{0,1\}} P(h_k|g) \log \frac{P(h_k|g)}{P(h_k|f)} \end{aligned} \tag{4}$$

Here we have used the fact that $P(\boldsymbol{h}|\boldsymbol{x}) = \prod_k P(h_k|\boldsymbol{x})$.

However, since the KL-divergence is asymmetric (i.e. $\mathrm{KL}(g \parallel f) \neq \mathrm{KL}(f \parallel g)$), we will employ the symmetrized version, also known as the Jensen-Shannon divergence, $\mathcal{D}(g,f) =$

$$\begin{aligned}&= \frac{1}{2}\left(\mathrm{KL}(g \parallel f) + \mathrm{KL}(f \parallel g)\right) \\ &= \frac{1}{2}\sum_k \left(\left(P(h_k^1|g) - P(h_k^1|f)\right) \times \left(\boldsymbol{x}^{(g)} - \boldsymbol{x}^{(f)}\right)^\top \boldsymbol{U}_{\bullet k} \right)\end{aligned}$$

The details of the derivation will be presented in Appendix A.

Figure 1. Sub-images extracted from the faces. Top row: the whole faces; the second row: the left eyes; the third row: the right eyes; the bottom row: the mouths.

Remark 3: For a training set with $|\mathcal{I}|$ subjects and n images per identity, the training time is $\mathcal{O}\left(KM\,|\mathcal{I}|\,n^2\right)$. For most realistic datasets, n is quite small, and thus the algorithm may be considered as being linear in training size. For recognition, it takes $\mathcal{O}(KM)$ time per image comparison. To accelerate nearest retrieval, we may employ efficient bit-wise techniques such as those in [17] since our posteriors can be turned into binary vectors, i.e., $h_k^* = \arg\max_{h_k \in \{0,1\}} P(h_k^1|\boldsymbol{x})$.

D. Fusing multiple feature sets

Subspace approach in face recognition often relies on whole-face pixel-based features from relatively low resolution images. However, the existing literature also recognises the importance of rich feature extraction methods from different resolutions and local parts [18], [14]. For example, Fig. 1 suggests that higher resolution areas around the eyes and mouth are very informative. The question is therefore how to make use of these rich sources of information.

Fortunately, the RBM architecture allows natural fusion of heterogeneous features using *shared* factors of variation. In particular, let $\boldsymbol{x}_{1:C} = \{\boldsymbol{x}_1, \boldsymbol{x}_2 ..., \boldsymbol{x}_C\}$ be the collection of C feature sets, the activating probability in Eq. 1 can be modified as

$$P(h_k^1|\boldsymbol{x}_{1:C}) = \sigma\left[\sum_{c=1}^{C} \boldsymbol{x}_c^\top \boldsymbol{U}_{c\bullet k} + v_k\right] \tag{5}$$

where $\boldsymbol{U}_{c\bullet k}$ is the column vector with respect to the cth set and the kth hidden factor. Thus, features from different sets are fused together by appropriate coefficients $\boldsymbol{U}_{c\bullet k}$. To make the model more sparsely connected, we can vary the level of factor sharing among sets, e.g., by fixing $\boldsymbol{U}_{c\bullet k} = \boldsymbol{0}$ for some pairs (c,k).

III. Experiments

A. Settings

In order to test our recognition algorithm under strong variations, we capture facial images from 24 persons, each of whom has roughly 35 images under 7 poses and 5 lighting

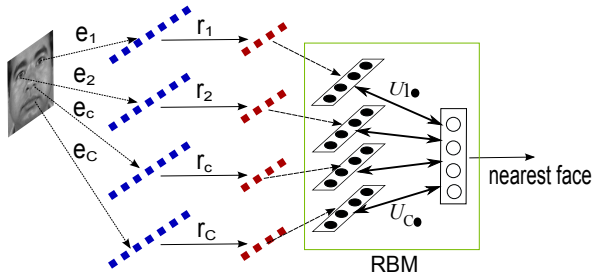

Figure 2. Schematic illustration of the RBM architecture for face recognition. For each face, we run C extractors $\{e_1, e_2, .., e_C\}$ to obtain C feature sets, each of which is passed through a dimensionality reduction method $\{r_c\}$. All the feature sets are then fed into the RBM model to produce a vector of posterior $P(h_1^1|\boldsymbol{x}_{1:C}), P(h_2^1|\boldsymbol{x}_{1:C}), ..., P(h_K^1|\boldsymbol{x}_{1:C})$. This vector is used for matching with the nearest face in the database.

Figure 3. 7 pose (row) and 5 lighting (column) conditions per person.

conditions (e.g., see Figure 3). Unless otherwise specified, the data is randomly split into a training set of 512 images and a test set of 330 images.

Our model architecture is depicted in Fig. 2. Image features obtained from C extractors[4] are first preprocessed by a dimensionality reduction module using PCA. These

[4]Each extractor may operate on a facial part, or may implement a specific feature extraction method.

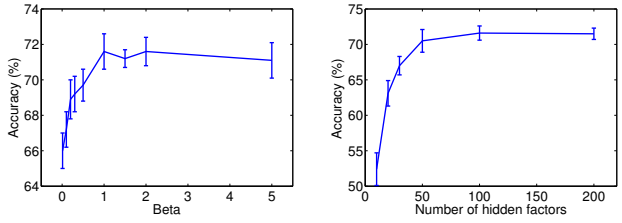

Figure 4. Performance as a function of the controlling parameter β (with $K = 200$) (Left), and the number of factor of variations (with $\beta = 1$) (Right). Facial images are preprocessed using PCA with top 50 eigenvectors.

	LDA	LPP	LDML	RBM	RBM/LDA
Fa	71.5	71.2	67.3	71.8±0.7	**76.5±1.2**
Le	65.5	50.9	53.3	64.0±1.0	63.4±1.0
Re	63.9	45.5	54.9	64.0±0.8	**68.2±0.5**
Mo	65.8	37.3	47.3	64.2±0.4	**67.1±0.8**
LRe	74.5	58.2	64.8	**78.4±1.1**	**79.6±1.3**
MoLRe	80.0	63.0	67.9	83.9±0.7	**89.4±0.6**
FaLRe	81.2	77.3	77.3	86.4±0.8	**89.2±0.7**
FaLReMo	83.9	79.7	81.5	88.9±0.6	**92.3±0.4**
(↓)	(43.5)	(29.5)	(43.3)	(60.6)	(**67.2**)

Table I
ACCURACY (%) WHEN FUSING FACIAL PARTS WITH RAW-PIXEL REPRESENTATION. FA = FACE, LE = LEFT EYE, RE = RIGHT EYE, MO = MOUTH, LRE = LEFT AND RIGHT EYES. RBM/LDA IS THE RBM WHOSE INPUT FEATURES ARE FIRST TRANSFORMED BY LDA. FOR THE PRE-PROCESSING STEP, 50 EIGENVECTORS ARE SELECTED FOR EACH FEATURE SUBSET. THE SYMBOL ↓ INDICATES THE *reduction* IN ERROR RATE WHEN COMBINING FEATURES COMPARED TO THE FACE FEATURES ALONE.

feature sets are then normalised to zero means and unit variances before fused into our RBM. For recognition, the person identity will be assigned to that of the nearest face in the training data according to the Euclidean distance on the activation probabilities.

For training, the parameters $\{\boldsymbol{U}_{c\bullet}\}_{c=1}^{C}$ are initialised randomly from Gaussian $\mathcal{N}(\mathbf{0}; \mathbf{0.1})$, while bias parameters $\boldsymbol{w}, \boldsymbol{v}$ are set to zeros initially. To reduce the training time, we divide training data into mini batches of $B = 100$ images, and update the parameter after each batch. The learning rate is fixed at $\nu = 0.005/B$ (Eq. 3) and the number of iterations is set at 50. All experiments with RBMs are repeated 10 times before results are reported.

Fig. 4 depicts the recognition performance of the proposed RBM under various hyper-parameters: the metric-learning coefficient β (Eq. 2) and the number of hidden factors K. It can be seen that metric learning effect is profound ($\beta = 0$ means no metric learning), suggesting that suppressing intra-personal variations is critical. The performance is quite stable against the number of hidden factors as long as the model is large enough (e.g., $K \geq 50$). For the rest of the paper, we select $\beta = 1$ and $K = 200$ unless specified otherwise.

B. *Fusion of facial parts*

In this experiment, we manually extract the regions around the eyes and the mouth, and use raw-pixels as features (we can employ automated detectors, e.g., see [14], in a more sophisticated setting). For each part, we run PCA with the top 50 eigenvectors for pre-processing. For comparison, we implement a recent metric learning method called logistic discriminant (LDML) [13], make use of the locality preserving projection[5] (LPP, a.k.a. Laplacian faces) [3] and the linear discriminant analysis (LDA)[19]. The LDML takes

[5]Code available at: http://www.zjucadcg.cn/dengcai

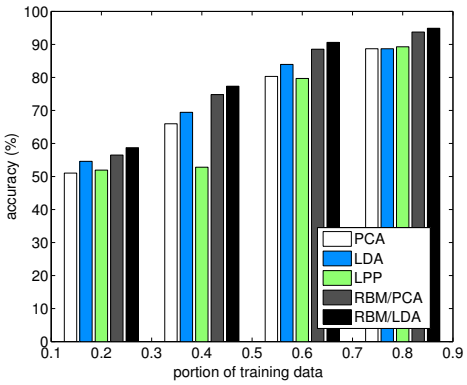

Figure 5. Accuracy as a function of training size with combination of parts and the whole face.

Face res	Pixel	LBP(*)	Gabor(**)
13×13	**92.3±0.4**	75.8±2.3(2×2)	90.4±0.6(/2)
19×19	91.2±0.6	83.0±1.3(3×3)	**95.9±0.6(/4)**
25×25	90.8±0.6	92.6±0.8(3×3)	**97.2±0.7(/4)**
38×38	91.6±0.7	94.8±0.8(5×5)	94.3±1.1(/4)
75×75	NA	**98.5±0.5(5×5)**	70.4±1.4(/8)
150×150	NA	**99.2±0.2(5×5)**	NA

Table II
PERFORMANCE OF RBM WITH RESPECT TO FEATURE REPRESENTATIONS, MULTIPLE PART FUSION UNDER DIFFERENT RESOLUTIONS. EXTRACTED FEATURES ARE FIRST PREPROCESSED BY LDA. THE FACE RESOLUTION IS LISTED FOR THE WHOLE FACE, AND THE COMPONENTS ARE PROPORTIONALLY RESIZED. (*) THE LBP DEPENDS ON HOW WE PARTITION THE IMAGE INTO SMALLER CELLS, E.G., 2×2 MEANS THERE ARE 4 CELLS. (**) THE GABOR FILTERS TYPICALLY PRODUCE BIG RESPONSE IMAGES WHICH ARE THEN DOWN SAMPLED BY A CERTAIN FACTOR, I.E., $(/8)$ FOR THE CASE OF 75×75 RESOLUTION.

every pair of images and estimates a parametric distance between them. For recognition of a given query face, the nearest training face according to the distance metric is used for recognition. The distance function is trained by maximising the pairwise likelihood under the logistic regression model to specify whether an image pair belongs to the same person. Thus, training LDML is expensive since its run-time complexity is quadratic in training size. Our method, on the other hand, is only quadratic in training size *per person* (which is quite small in most realistic datasets), and linear in number of subjects. The LPP, when the neighbourhood is defined on the intra-personal neighbours, can also be considered as a metric learning method. The main difference from our RBM is that the LDML and the LPP do not have the generative component to capture the regularity in the data.

Table I reports the results of RBMs against other methods. The RBM fares comparably with the LDA when there is a single source of information, and becomes competitive when there are multiple sources. It is interesting to see that the RBM can improve over the LDA since the LDA to some extent is also a metric learning method. We conjecture that the probabilistic and non-linear nature of the RBM may complement the LDA. When combining parts, the reduction of error rate is significant: we can reduce as much as 67% error by combining part-based features with whole-face features using the LDA as a pre-processing step for the RBM.

C. Feature extraction under different resolutions

We employ two feature representations: the local binary patterns (LBP) [20] and the Gabor features [2]. For the LBP we follow the recommendation in [4], in that each facial component or the whole face is partitioned into smaller cells, and then LBP is applied to each region to yield a histogram. Finally, all histograms are concatenated to form a long feature vector. The partition and concatenation processes implicitly encode the geometrical structure of the component or the face, which are critical to the recognition performance. Thus, there is a trade-off between the number of cells for preserving information richness and the length of vector for efficiency. For the Gabor representation, as suggested in [2] we employ a bank of 40 filters which account for 5 scales and 8 orientations.

Table II reports the results of the RBM with respect to resolution of the images. It can be seen that the raw-pixel representation is robust against image scaling. This may be explained by the fact that the pre-processing step based on LDA typically discovers the subspace of the faces, which depends on the overall structure of the face space rather than the details. Localized methods like LBP and Gabor filters, on the other hand, rely on the local details of the faces such as textures in the case of LBP and edges in the case of Gabor filters. The LBP is also depending on the number of partitions in the face, which means that larger faces will allow more cells.

IV. CONCLUSION

We have introduced a new method for face recognition using Restricted Boltzmann Machines. Our main contribution lies in the regularisation of the training objective function to reduce intra-personal variations. This is achieved by adding an information-theoretic divergence into the standard log-likelihood. The proposed model is flexible in incorporating multiple feature sets with easy controlling of sparsity level. Experiments on a dataset with strong variations in lighting and pose conditions have shown that our proposed method is competitive against other metric learning rivals. We also validated the method under a variety of settings, including fusing facial parts, using feature extraction techniques such as LBP and Gabor filters, and varying resolutions.

REFERENCES

[1] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of cognitive neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.

[2] 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.

[3] X. He, S. Yan, Y. Hu, P. Niyogi, and H.J. Zhang, "Face recognition using Laplacian faces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 3, pp. 328–340, 2005.

[4] T. Ahonen, A. Hadid, and M. Pietikainen, "Face description with local binary patterns: Application to face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2037, 2006.

[5] Q. Yin, X. Tang, and J. Sun, "An associate-predict model for face recognition," in *CVPR*. IEEE, 2011, pp. 497–504.

[6] P. Smolensky, "Information processing in dynamical systems: Foundations of harmony theory," *Parallel distributed processing: Explorations in the microstructure of cognition*, vol. 1, pp. 194–281, 1986.

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

[8] M. Welling, M. Rosen-Zvi, and G. Hinton, "Exponential family harmoniums with an application to information retrieval," in *Advances in NIPS*, vol. 17, pp. 1481–1488. 2005.

[9] M.A. Ranzato and G.E. Hinton, "Modeling pixel means and covariances using factorized third-order Boltzmann machines," in *CVPR*. IEEE, 2010, pp. 2551–2558.

[10] R. Memisevic and G.E. Hinton, "Learning to represent spatial transformations with factored higher-order boltzmann machines," *Neural Computation*, vol. 22, no. 6, pp. 1473–1492, 2010.

[11] H. Larochelle and G. Hinton, "Learning to combine foveal glimpses with a third-order Boltzmann machine," *Advances in Neural Information Processing*, vol. 23, 2010.

[12] Y.W. Teh and G.E. Hinton, "Rate-coded restricted Boltzmann machines for face recognition," *Advances in neural information processing systems*, pp. 908–914, 2001.

[13] M. Guillaumin, J. Verbeek, and C. Schmid, "Is that you? Metric learning approaches for face identification," in *ICCV*. IEEE, 2009, pp. 498–505.

[14] B. Heisele, T. Serre, and T. Poggio, "A component-based framework for face detection and identification," *International Journal of Computer Vision*, vol. 74, no. 2, pp. 167–181, 2007.

[15] G.E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Computation*, vol. 14, pp. 1771–1800, 2002.

[16] T. Tran, D.Q. Phung, and S. Venkatesh, "Mixed-variate restricted Boltzmann machines," in *Proc. of 3rd Asian Conference on Machine Learning*, Taoyuan, Taiwan, 2011.

[17] R. Salakhutdinov and G. Hinton, "Semantic hashing," *International Journal of Approximate Reasoning*, vol. 50, no. 7, pp. 969–978, 2009.

[18] R. Gottumukkal and V.K. Asari, "An improved face recognition technique based on modular PCA approach," *Pattern Recognition Letters*, vol. 25, no. 4, pp. 429–436, 2004.

[19] P.N. Belhumeur, J.P. Hespanha, D.J. Kriegman, et al., "Eigenfaces vs. Fisherfaces: recognition using class specific linear projection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711–720, 1997.

[20] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern recognition*, vol. 29, no. 1, pp. 51–59, 1996.

APPENDIX

For brevity, let $a_k(f) = \boldsymbol{x}^{(f)\top}\boldsymbol{U}_{\bullet k} + v_k$ and denote $Q_k(f) = P(h_k^1|\boldsymbol{x}^{(f)})$. Thus, $Q_k(f) = \sigma\left[a_k(f)\right]$ and expanding the KL-divergence $\mathrm{KL}\left(g \parallel f\right)$ in Eq. 4 yields $\mathrm{KL}\left(g \parallel f\right) =$

$$
\begin{aligned}
&= Q_k(g)\log\frac{Q_k(g)}{Q_k(f)} + (1-Q_k(g))\log\frac{1-Q_k(g)}{1-Q_k(f)} \\
&= Q_k(g)\log\frac{1+e^{-a_k(f)}}{1+e^{-a_k(g)}} + \\
&\qquad +(1-Q_k(g))\log\frac{(1+e^{-a_k(f)})e^{-a_k(g)}}{(1+e^{-a_k(g)})e^{-a_k(f)}}
\end{aligned}
$$

Rearranging terms we have $\mathrm{KL}\left(g \parallel f\right) =$

$$
\begin{aligned}
&= \log\frac{1+e^{-a_k(f)}}{1+e^{-a_k(g)}} + (1-Q_k(g))\log\frac{e^{-a_k(g)}}{e^{-a_k(f)}} \\
&= \log\frac{Q_k(g)}{Q_k(f)} + (1-Q_k(g))\left(a_k(f)-a_k(g)\right)
\end{aligned}
$$

Thus the symmetric KL-divergence $\mathcal{D}\left(g,f\right) = \frac{1}{2}(\mathrm{KL}\left(g \parallel f\right) + \mathrm{KL}\left(f \parallel g\right))$ is straightforward:

$$
\mathcal{D}\left(g,f\right) = \frac{1}{2}\left(Q_k(g)-Q_k(f)\right)\left(a_k(g)-a_k(f)\right)
$$

Replacing $a_k(g)$ and $Q_k(g)$ with their corresponding forms give us $\mathcal{D}\left(g,f\right) =$

$$
\frac{1}{2}\sum_k\left(\left(P(h_k^1|g)-P(h_k^1|f)\right)\times\left(\boldsymbol{x}^{(g)}-\boldsymbol{x}^{(f)}\right)^\top\boldsymbol{U}_{\bullet k}\right).
$$

The gradient of the symmetric divergence with respect to a distribution is then

$$
\frac{\partial\mathcal{D}\left(g,f\right)}{\partial Q_k(g)} = \frac{1}{2}\left(\frac{\partial\mathrm{KL}\left(g \parallel f\right)}{\partial Q_k(g)} + \frac{\partial\mathrm{KL}\left(f \parallel g\right)}{\partial Q_k(g)}\right)
$$

where

$$
\begin{aligned}
\frac{\partial\mathrm{KL}\left(g \parallel f\right)}{\partial Q_k(g)} &= \log\frac{Q_k(g)}{Q_k(f)} - \log\frac{1-Q_k(g)}{1-Q_k(f)} \\
\frac{\partial\mathrm{KL}\left(f \parallel g\right)}{\partial Q_k(g)} &= -\frac{1}{Q_k(g)} + \frac{1}{1-Q_k(g)}.
\end{aligned}
$$