Field Effect Deep Networks for Image Recognition with Incomplete Data

SHENG-HUA ZHONG, Shenzhen University
YAN LIU, The Hong Kong Polytechnic University
KIEN A. HUA, University of Central Florida

Image recognition with incomplete data is a well-known hard problem in computer vision and machine learning. This article proposes a novel deep learning technique called Field Effect Bilinear Deep Networks (FEBDN) for this problem. To address the difficulties of recognizing incomplete data, we design a novel second-order deep architecture with the Field Effect Restricted Boltzmann Machine, which models the reliability of the delivered information according to the availability of the features. Based on this new architecture, we propose a new three-stage learning procedure with field effect bilinear initialization, field effect abstraction and estimation, and global fine-tuning with missing features adjustment. By integrating the reliability of features into the new learning procedure, the proposed FEBDN can jointly determine the classification boundary and estimate the missing features. FEBDN has demonstrated impressive performance on recognition and estimation tasks in various standard datasets.

CCS Concepts: • **Information systems** → **Image search**; • **Computing methodologies** → **Learning latent representations**

Additional Key Words and Phrases: Image recognition, incomplete data, missing features, deep learning

ACM Reference Format:
Sheng-hua Zhong, Yan Liu, and Kien A. Hua. 2016. Field effect deep networks for image recognition with incomplete data. ACM Trans. Multimedia Comput. Commun. Appl. 12, 4, Article 52 (August 2016), 22 pages.
DOI: http://dx.doi.org/10.1145/2957754

1. INTRODUCTION

Incomplete data, whose data values are partially observed [Liao et al. 2007], exists in a wide range of fields, including social sciences, computer vision, biological systems, and remote sensing [Williams et al. 2007; Li et al. 2015]. In general, features missing in real-world data result from measurement noise, corruption, or occlusion [Chechik et al. 2008]. Everybody has experiences of incomplete data, such as noisy photos, old broken posters, or ancient frescos. As we know, it is more difficult for computers to recognize meaningful patterns from incomplete data than complete data [Wu et al. 2015; Ding et al. 2015; Chen et al. 2015]. The interactive multimedia environment also requires spatial consistency [Wu et al. 2009; Natarajan et al. 2015]. If the distortion is very

Authors' addresses: S.-H. Zhong, College of Computer Science and Software Engineering, Shenzhen 518000, P.R. China; email: csshzhong@szu.edu.cn; Y. Liu (corresponding author), Department of Computing, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong 999077, P.R. China; email: csyliu@comp.polyu.edu.hk; K. A. Hua, Department of Electrical Engineering and Computer Science, University of Central Florida, Orlando, FL 32816-2362; email: kienhua@cs.ucf.edu.

DOI: http://dx.doi.org/10.1145/2957754

Fig. 1. Some incomplete images due to the occlusion of some important facial feature regions. The forehead is occluded in (a); the eyes are hidden behind sunglasses in (b) and (c); and the mouth is occluded in (d).

severe, even human beings cannot recognize these kinds of images correctly. Figure 1 provides some general cases of incomplete data in our daily lives. David Beckham is one of the most iconic athletes, and most fans have no difficulty in recognizing him from the four images in Figure 1. But it is not an easy task for many face recognition algorithms because some key facial features to identify people are not observable, such as the forehead, eyes, and mouth.

Current works on incomplete data can be roughly categorized into three classes based on the approaches to model the missing values [Dick et al. 2008]. The first kind of technique doesn't intend to estimate the missing values. For example, Chechik et al. [2006, 2008] proposed two methods to recognize the incomplete data directly without any completion of the missing features using a Geometric Margin (GEOM) learning framework [Chechik et al. 2006, 2008]. The second kind of technique fills the missing values based on the available information and then learns the decision function in a general way. Logistic Regression Classification based on Expectation Maximization (LRCEM) estimates the conditional density function using a Gaussian mixture model with expectation maximization [Williams et al. 2005]. Logistic Regression Classification based on Variational Bayesian Expectation Maximization (LRCVBEM) utilizes variational Bayesian expectation maximization to substitute expectation maximization [Williams et al. 2007]. The third kind of technique seeks the final decision boundary by estimating the missing value and constructing a predictive model jointly. Liao et al. [2007] proposed a statistical model named the Quadratically Gated Mixture of Experts (QGME) for multiclass nonlinear recognition. Dick et al. [2008] derived a generic joint optimization Weighted Infinite Imputation (WII) method, which learned the decision function and the distribution of imputations dependently. These experiments demonstrated significant performance improvements over the methods that separate estimation from classifier learning. Hence, this article intends to design a classifier to determine the decision boundary and estimate the missing values synchronously. Besides these three types of approaches, some methods for missing value imputation also can be extended for image recognition with incomplete data, such as the nonparametric Bayesian dictionary learning method [Zhou et al. 2012], the hybrid prediction model [Purwar et al. 2015], and self-organizing maps [Folguera et al. 2015].

Recently, there has been mounting neurophysiological evidence for considerable attentional modulation, which can enlighten us on incomplete image recognition. First, when an image is incomplete, humans can automatically adjust their attention to the available features and emphasize the contributions from them consciously and, actually, unconsciously. In comparison with the occluded part, the firing rates of the neurons will increase by preferring the available features [Ranzato et al. 2011]. Second, additive attention could lead to the occluded parts of the object becoming active, as the feedback from higher levels travels down the visual stream based on the feedback connections in the visual cortex [Taylor et al. 2006]. This process allows us to hallucinate occluded/undetected parts by filling in the missing features based on top-down knowledge from the model, which plays an important role in identifying

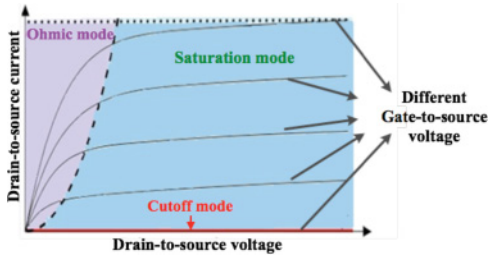


Fig. 2. The output characteristic curve and the operating mode of FET.

and completing objects when different portions are visible, or when parts are occluded or degraded [Aleman et al. 2003]. It means that humans can infer and estimate the incomplete parts by the reference data. In case the feature value is missing and no related information is useful to estimate this missing feature, humans will automatically neglect the specific missing feature.

Based on existing studies from both computer science and neurophysiology, to solve the problem of image recognition with incomplete data, we should answer three linked questions: (1) How do we represent the differences between the available features and missing features? (2) How do we use visible information to infer and estimate the occluded or degraded parts? (3) How do we unify incomplete image recognition and missing features estimation into a framework?

To address the first question, we construct the reliability function to model the quality of the features by reference to the characteristics of a Field Effect Transistor (FET). As a common electronic device, the FET has three possible operating modes, including the cutoff mode, the ohmic mode, and the saturation mode, as shown in Figure 2. As we described before, when the feature value is missing and no related information is available, humans will neglect the specific missing feature. In our model, the reliability function will go into the cutoff mode and the reliability of this missing feature is set to be zero. If some available information can be used to estimate the missing feature, humans will begin to estimate the missing feature and adjust the reliability. In our model, it is identical to the ohmic mode of the FET. In this mode, the missing features will be estimated in the process of the recognition while the reliability of the corresponding connection is updating. Given that the estimate is stable and no further information is available, humans will stop adjusting the estimation and the reliability will remain unchanged. In our model, it corresponds to the saturation mode of the FET. For the second question, we design a learning procedure based on deep learning techniques. Many experiments have demonstrated that deep learning techniques have the distinguishing ability of information abstraction in various real-world visual data analysis tasks [Schmidhuber 2014]. Deep learning methods also showed great potential to address the incomplete data recognition problem [Zhong et al. 2011]. For the third question, we integrate the features with their reliabilities into the three learning stages of the proposed model.

To conclude, in this article, we propose a deep learning model called Field Effect Bilinear Deep Networks (FEBDN) for image recognition with incomplete data. Inspired by the attentional modulation, we attempt to construct the reliability function to model the quality of features in reference to the characters of the FET. By integrating the features with their reliabilities into the three-stage learning of FEBDN, FEBDN constructs the optimal classification boundary and estimates the missing features synchronously. The remainder of this article is organized as follows. Related work on deep learning is reviewed in Section 2. A novel deep architecture with a new deep learning

algorithm is introduced in Section 3. Section 4 shows the performance of the proposed techniques in different tasks. Section 5 concludes this article and outlines our future work.

2. RELATED WORK ON DEEP LEARNING

Empirical validations in various real-world applications have shown that deep learning models perform impressively for applications in pattern recognition and machine learning [Atrey et al. 2010; LeCun et al. 2015], such as in image classification [Krizhevsky et al. 2012], image sentiment analysis [You et al. 2015], automatic localization in ultrasound [Chen et al. 2015], and document summarization [Zhong et al. 2015]. To the image classification task, in our previous work, we constructed a novel deep architecture of Bilinear Deep Belief Networks (BDBN) to preserve the natural tensor structure in information propagation [Zhong et al. 2011]. Based on the novel deep structure, the three-stage learning of BDBN was designed by referring to the procedure of object recognition of human beings, especially the "initial guess" part.

Although there are few deep learning techniques on incomplete data classification in the view of vision, some previous works can be applied in incomplete image classification and missing features estimation [Salakhutdinov et al. 2007; Sohn et al. 2013; Lee et al. 2011]. Imputing RBM was introduced to estimate the missing users' ratings of movies [Salakhutdinov et al. 2007]. Imputing RBM first initializes the missing feature as the mean value of the available features in the corresponding location. Then it computes the gradient of the log-probability of the data with respect to the missing features and updates their values. Conditional RBM can also be applied for incomplete data classification [Salakhutdinov et al. 2007]. It defines a binary vector to indicate whether the feature is missing and constructs a matrix that models the effect of the binary vector on features in the hidden layer. The learned matrix can increase the contribution of the available features. Point-wise Gated Boltzmann Machines (PGBM) was proposed to focus on separating the relevant patterns from irrelevant patterns [Sohn et al. 2013]. The switch units allow the model to estimate where the task-relevant patterns occur and make only those visible units contribute to the final prediction. Based on the switch design of PGBM, it also can be applied in the incomplete data classification. Convolutional Deep Belief Networks (CDBN) proposed a hierarchical (bottom-up and top-down) generative model for learning hierarchical representations from images [Lee et al. 2011]. With full Gibbs sampling, the bottom-up inputs combined with the context provided by the higher layer significantly improve the second-layer representation and can be utilized to infer the missing features in the input layer. sDBN, proposed by Tang et al. [2010], can be applied in incomplete image classification. They introduced a modified version of the DBN termed a sparse DBN. They also developed a probabilistic algorithm to denoise the noisy features.

3. FIELD EFFECT BILINEAR DEEP NETWORKS

In this section, we propose the FEBDN to model the reliability of the features and recognize the incomplete images. Section 3.1 presents the novel deep architecture of FEBDN. A three-stage learning procedure is presented in the following part. We also provide the algorithm and discuss the relations between FEBDN and some representative deep models in this section.

3.1. Framework of Field Effect Bilinear Deep Networks

Let X be a set of incomplete data samples as shown here:

$$X = [\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_k, \ldots, \mathbf{X}_K], \tag{1}$$

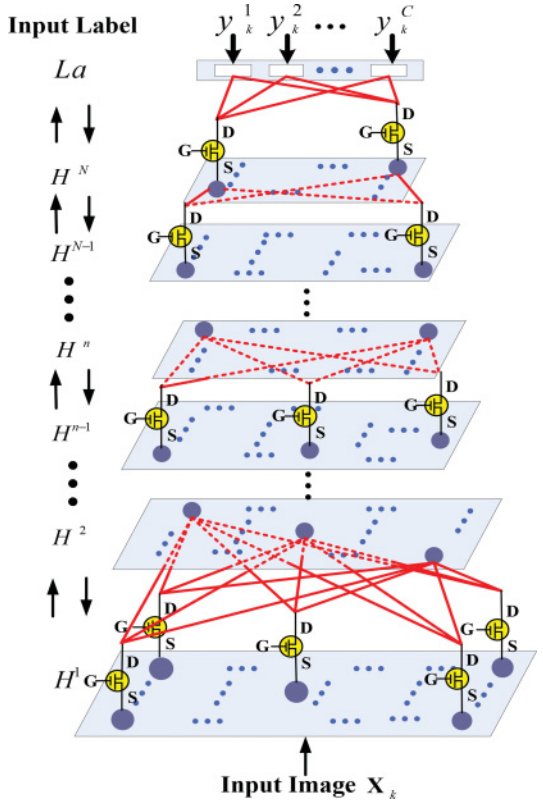


Fig. 3. Architecture of FEBDN. The small blue dots represent nodes in each respective layer.

where $\mathbf{X}_k \in \mathbb{R}^{I \times J}$ is a sample datum with missing features and K is the number of sample data. Let F_k denote the set of missing features of the sample $\mathbf{X}_k$; $(X_k)_{ij}$ is missing if $(X_k)_{ij} \in F_k$. Let Y be a set of labels corresponding to X:

$$Y = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_k, \ldots, \mathbf{y}_K], \tag{2}$$

and $\mathbf{y}_k$ is the label vector of $\mathbf{X}_k$ in $\mathbb{R}^C$, where C is the number of classes, $\mathbf{y}_k = [y_k^1, y_k^2, \ldots, y_k^c, \ldots, y_k^C]$:

$$y_k^c = \begin{cases} 1 & \text{if } \mathbf{X}_k \in c\text{th class} \\ 0 & \text{if } \mathbf{X}_k \notin c\text{th class} \end{cases}. \tag{3}$$

Based on the given training set, the goal in image recognition is to learn a mapping function from the image set X to the label set Y and then recognize the new data points according to the learned mapping function. To address the problem of incomplete image recognition, we propose a novel deep architecture as shown in Figure 3. A fully interconnected directed network includes the input layer H^1, hidden layer $H^2, \ldots, H^N$, and one label layer La at the top. In our model, the input layer and all hidden layers are constructed by a set of second-order planes, which are consistent with the natural tensor structure of images. The input layer H^1 has $I \times J$ units, and the size is equal to the dimension of the input features. We use the pixel values of sample datum $\mathbf{X}_k$ as the input features. In the top, the label layer has C units, which is equal to the number of classes. The search of the mapping function from X to Y is transformed to the problem

of finding the optimum parameter space θ^*. A set of new Field Effect RBM (FRBM) is proposed to connect the adjacent layers.

Furthermore, FRBM is utilized as the basic units of FEBDN instead of RBM. In DBN, RBM is used to abstract the embedding information by layer-wise reconstruction. Unfortunately, RBM cannot directly work when some features are missing and the corresponding units of the networks are empty. Inspired by common electronic circuits [Malik 1995], FRBM constructs the reliability-weighted connection by FET analogy between the current layer and the next layer. The reliability parameter curve of FRBM is designed to simulate the output characteristic curve and the operating mode of FET in Figure 2. In our model, the probability level of the estimates of the missing feature corresponds to the gate-to-source voltage of the FET. The similarity between the reference datum and incomplete datum corresponds to the drain-to-source voltage of the FET.

3.2. Initial Guess by Field Effect Bilinear Initialization

In this part, we introduce the Field Effect Bilinear Projection (FBP), which is utilized to extract the discriminant information from the image datasets with incomplete features. Given the training data points $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_K \in \mathbb{R}^{I\times J}$ with the missing features set F_1, $F_2, \ldots, F_K$, where $(X_k)_{ij}$ is missing if $(X_k)_{ij} \in F_k$, FBP aims to find two projection matrices $\mathbf{U} \in \mathbb{R}^{I\times P}$ and $\mathbf{V} \in \mathbb{R}^{J\times Q}$ such that the latent representation $\mathbf{TX}_1, \mathbf{TX}_2, \ldots, \mathbf{TX}_K \in \mathbb{R}^{I\times J}$ can be obtained by $\mathbf{TX}_k = \mathbf{U}^T\mathbf{X}_k\mathbf{V}$ $(k = 1, \ldots, K)$ from features with high reliability. Here, we define the reliability matrix $\mathbf{R}_k^F \in \mathbb{R}^{I\times J}$ of the features in $\mathbf{X}_k$. In the initial guess stage, the reliability matrix $\mathbf{R}_k^F$ is assigned as Equation (4), just as the cutoff mode of FET:

$$(R_k^F)_{ij} = \begin{cases} 0, & \text{if}(X_k)_{ij} \in F_k \\ 1, & \text{else} \end{cases}. \tag{4}$$

In order to preserve the discriminant information from features with high reliability in the learning procedure, the objective function of FBP could be represented as follows:

$$\begin{aligned} \arg\max_{\mathbf{U},\mathbf{V}} J(\mathbf{U},\mathbf{V}) = \sum_{s,t=1}^{K} (\alpha\mathbf{B}_{st} - (1-\alpha)\mathbf{W}_{st})||\mathbf{U}^T(\mathbf{X}_s. * \mathbf{R}_{st}^F - \mathbf{X}_t. * \mathbf{R}_{st}^F)\mathbf{V}||^2 \\ s.t.\, \mathbf{R}_{st}^F = \mathbf{R}_s^F. * \mathbf{R}_t^F, \mathbf{U}^T\mathbf{U} = \mathbf{I}_P, \mathbf{V}^T\mathbf{V} = \mathbf{I}_Q. \end{aligned} \tag{5}$$

In Equation (5), $\alpha \in [0, 1]$ is the parameter used to balance the between-class weights $\mathbf{B}_{st}$ and the within-class weights $\mathbf{W}_{st}$, which are defined as follows:

$$\mathbf{B}_{st} = \begin{cases} \frac{1}{n_d} - \frac{1}{n_c}, & \text{if } y_s^c = y_t^c = 1, \\ \frac{1}{n_d}, & \text{else,} \end{cases}, \quad \mathbf{W}_{st} = \begin{cases} \frac{1}{n_c}, & \text{if } y_s^c = y_t^c = 1, \\ 0, & \text{else,} \end{cases}, \tag{6}$$

where n_d is the number of data points in all classes and n_c is the number of data points in class c, where $c \in \{1, \ldots, C\}$. Different from Bilinear Discriminant Projection (BDP), which tries to preserve the discriminant information from all features [Zhong et al. 2011], we extract the discriminant information based on the features with high reliability. By simultaneously maximizing the distances between data points from different classes and minimizing the distance between data points from the same class, the discriminant information is preserved at the greatest extent in the projected feature space. Optimizing $J(\mathbf{U}, \mathbf{V})$ by solving $\mathbf{U}$ (or $\mathbf{V}$) with fixed $\mathbf{V}$ (or $\mathbf{U}$) is a convex optimization problem. Let $\mathbf{E}_{st} = \alpha\mathbf{B}_{st} - (1-\alpha)\mathbf{W}_{st}$, with the fixed $\mathbf{V}$. The optimal $\mathbf{U}$ is composed of the first P eigenvectors of the following eigendecomposition

problem:

$$\mathbf{D_V u} = \lambda \mathbf{u}, \tag{7}$$

where $\mathbf{D_V} = \sum_{st} \mathbf{E}_{st}(\mathbf{X}_s. * \mathbf{R}^F_{st} - \mathbf{X}_t. * \mathbf{R}^F_{st})\mathbf{V}\mathbf{V}^T(\mathbf{X}_s. * \mathbf{R}^F_{st} - \mathbf{X}_t. * \mathbf{R}^F_{st})^T$.

Similarly, with the fixed $\mathbf{U}$, the optimal $\mathbf{V}$ is composed of the first Q eigenvectors of the following eigendecomposition problem:

$$\mathbf{D_U v} = \lambda \mathbf{v}, \tag{8}$$

where $\mathbf{D_U} = \sum_{st} \mathbf{E}_{st}(\mathbf{X}_s. * \mathbf{R}^F_{st} - \mathbf{X}_t. * \mathbf{R}^F_{st})^T\mathbf{U}\mathbf{U}^T(\mathbf{X}_s. * \mathbf{R}^F_{st} - \mathbf{X}_t. * \mathbf{R}^F_{st})$.

Therefore, we can alternately optimize $\mathbf{U}$ (with a fixed $\mathbf{V}$) and $\mathbf{V}$ (with a fixed $\mathbf{U}$). The previously listed steps monotonically increase $J(\mathbf{U}, \mathbf{V})$, and since the function is upper bounded, it will converge to a critical point with transformation matrices $\mathbf{U}$ and $\mathbf{V}$.

In FBP, the sizes of P and Q are determined by the number of positive eigenvalues in $\mathbf{D_V}$ and $\mathbf{D_U}$, respectively, since adding the eigenvectors corresponding to the nonpositive eigenvalues will not increase the values, as shown in Equation (5). As a result, the original dimension $I \times J$ is automatically reduced to $P \times Q$.

3.3. Bidirectional Inference by Field Effect RBMs

In the human visual cortex, bidirectional inference includes bottom-up inference and top-down inference, and they are not separate processes. Thus, in our scheme, bottom-up inference and top-down inference are integrated together. The whole deep learning model with the parameter space is constructed based on the bottom-up inference from available features and estimated features. Simultaneously, the estimates of the missing features with their reliability parameters are obtained by the top-down inference.

The available features and the estimated features are input to the deep architecture as the state of the input layer H^1 to construct an FRBM with the first hidden layer H^2. The energy function of the state $(\mathbf{h}^1, \mathbf{h}^2)$ in the first Field Effect RBM is shown in Equation (9). Here, if the feature is available, the corresponding h^1_{ij} is the value of the available feature; if the feature is missing, the corresponding h^1_{ij} is the estimate of the feature:

$$E(\mathbf{h}^1, \mathbf{h}^2; \theta^1) = -\sum_{i=1,j=1}^{i\le I, j\le J} \sum_{p=1,q=1}^{p\le P^2, q\le Q^2} h^1_{ij} A^1_{ij,pq} R^{A,1}_{ij,pq} h^2_{pq} - \sum_{i=1,j=1}^{i\le I, j\le J} b^1_{ij} R^{b,1}_{ij} h^1_{ij} - \sum_{p=1,q=1}^{p\le P^2, q\le Q^2} c^1_{pq} h^2_{pq}, \tag{9}$$

In Equation (9), $I \times J$ is the number of units in H^1, while P^2 and Q^2 are the dimensions in hidden layer H^2. $\theta^1 = (\mathbf{A}^1, \mathbf{b}^1, \mathbf{c}^1)$ is the parameter space between the input layer H^1 and the first hidden layer H^2. $A^1_{ij,pq}$ is the symmetric interaction term between the input unit (i, j) in H^1 and the hidden unit (p, q) in H^2. b^1_{ij} is the $(i, j)^{th}$ bias of layer H^1 and c^1_{pq} is the $(p, q)^{th}$ bias of layer H^2. $\mathbf{R}^{\theta,1} = (\mathbf{R}^{\boldsymbol{A},1}, \mathbf{R}^{\boldsymbol{b},1})$ is the reliability parameter between the input layer H^1 and the first layer H^2 to control the reliability of the corresponding θ^1. Specifically, $R^{A,1}_{ij,pq}$ and $R^{b,1}_{ij}$ are used to represent the reliability of corresponding parameters $(\theta)_{ij}$ related to $(X_k)_{ij}$. To simplify the problem, reliability parameters $R^{\theta,1}_{ij} = (R^{A,1}_{ij,pq}, R^{b,1}_{ij})$ depend on the reliability of the estimates of missing feature $(X_k)_{ij} \in F_k$:

$$R^{A,1}_{ij,pq} = R^{A,1}_{ij,\bullet} = R^{b,1}_{ij} = \Re^{\theta,1}_{ij} \tag{10}$$

Therefore, the first RBM has the following joint distribution:

$$P(\mathbf{h}^1, \mathbf{h}^2; \theta^1) = \frac{\exp^{-E(\mathbf{h}^1, \mathbf{h}^2; \theta^1)}}{\sum_{\mathbf{h}^1} \sum_{\mathbf{h}^2} \exp^{-E(\mathbf{h}^1, \mathbf{h}^2; \theta^1)}}, \tag{11}$$

The log probability of the model assigned to $\mathbf{h}^1$ in H^1 is

$$\log P(\mathbf{h}^1) = \log \sum_{\mathbf{h}^2} \exp^{-E(\mathbf{h}^1, \mathbf{h}^2; \theta^1)} - \log \sum_{\mathbf{h}^1} \sum_{\mathbf{h}^2} \exp^{-E(\mathbf{h}^1, \mathbf{h}^2; \theta^1)}, \tag{12}$$

Similar to existing deep learning methods, we utilize the stochastic steepest ascent in the log probability of the training data to update the parameter space $\theta^1 = (\mathbf{A}^1, \mathbf{b}^1, \mathbf{c}^1)$:

$$A^1_{ij,pq} = \vartheta A^1_{ij,pq} + \Delta A^1_{ij,pq} R^{A,1}_{ij,pq} \tag{13}$$

$$\Delta A^1_{ij,pq} = \varepsilon_{\mathbf{A}}(<h^1_{ij}(0)h^2_{pq}(0)>_{data} - <h^1_{ij}(1)h^2_{pq}(1)>_{recon}), \tag{14}$$

where $\langle\cdot\rangle_{\boldsymbol{data}}$ denotes an expectation with respect to the data distribution and $\langle\cdot\rangle_{\boldsymbol{recon}}$ denotes the "reconstruction" distribution of data after one step. Other parameters in the θ^1 update function can be calculated in a similar manner:

$$b^1_{ij} = \vartheta b^1_{ij} + \Delta b^1_{ij} R^{b,1}_{ij} = \vartheta b^1_{ij} + \varepsilon_{\mathbf{b}}(h^1_{ij}(0) - h^1_{ij}(1)) R^{b,1}_{ij} \tag{15}$$

$$c^1_{pq} = \vartheta c^1_{pq} + \Delta c^1_{pq} = \vartheta c^1_{pq} + \varepsilon_{\mathbf{c}}(h^2_{pq}(0) - h^2_{pq}(1)), \tag{16}$$

where ϑ is the momentum and $\varepsilon_{\mathbf{A}}$, $\varepsilon_{\mathbf{b}}$, and $\varepsilon_{\mathbf{c}}$ are the learning rates of model parameters $\mathbf{A}$, $\mathbf{b}$, and $\mathbf{c}$.

As we described before, we find a Field Effect Bilinear Projection based on the reliable features that can automatically reduce the original dimension $I \times J$ to $P \times Q$ through the transformation matrices $\mathbf{U}^1$ and $\mathbf{V}^1$. As a result, in our model, the number of neurons in layer H^2 is determined by the row and column size of the transformation matrices $\mathbf{U}^1$and $\mathbf{V}^1$:

$$P^2 = row(\mathbf{U}^1), \quad Q^2 = column(\mathbf{V}^1). \tag{17}$$

We set the discriminative transformation parameters obtained from the Field Effect Bilinear Projection as the initial symmetrical connection weights by Equation (18):

$$A^1_{ij,pq}(0) = (\mathbf{U}^1_{ip})^T \mathbf{V}^1_{jq}. \tag{18}$$

The previous discussion is the construction of the first Field Effect RBM. Similar operations can be extended to deeper networks to construct the whole initial parameter space of the deep learning model in a straightforward way.

The estimate of the missing feature is obtained by the top-down inference. For the incomplete sample datum $\mathbf{X}_s (1 \le s \le K)$, we define $(f_s)^n_{pq}$ to denote the corresponding activation code in the hidden unit (p, q) of the layer $n (1 \le n \le N)$. The activation code $(f_s)^n_{pq}$ is obtained by Equation (19), where $\sigma(x)$ is the sigmoid function $\sigma(x) = 1/[1 + \exp(-x)]$, which also follows the activity of biological neurons:

$$(f_s)^n_{pq} = \sigma(h^{n-1}_{ij} A^{n-1}_{ij,pq} R^{A,n-1}_{ij,pq} + c^{n-1}_{pq}),\ n \ge 2. \tag{19}$$

The Euclidean distances sequence $\{g^n_{s,t}\}$ between the activation code of data points $\mathbf{X}_s$ and $\mathbf{X}_t$ in layer n is denoted as follows:

$$\{g^n_{s,t}\} = \left\| (f_s)^n_{pq} - (f_t)^n_{pq} \right\|, \quad 1 \le s, t \le K, s \ne t. \tag{20}$$

To the current data point $\mathbf{X}_s$, we sort the distance sequence $\{g^n_{s,t}\}$ in ascending order. The ranking position of the data point $\mathbf{X}_t$ in the sorted list is denoted as $L^n_{s,t}$. To infer and estimate the missing features $(X_s)_{ij}$ in $\mathbf{X}_s$, the nearest datum point $\mathbf{X}_{t^*}$ is selected out as the reference datum by Equation (21):

$$t^* = \arg\min_t \left[\sum_n \varepsilon_n L^n_{s,t} \right], \quad s.t.\ (X_s)_{ij} \in F_s, (X_t)_{ij} \notin F_t, \mathbf{y}_s = \mathbf{y}_t, \tag{21}$$

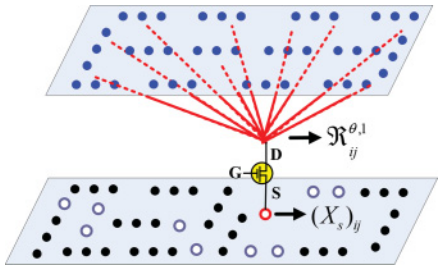


Fig. 4. Estimation of the missing features based on the node's reliability by top-down inference. The black/white dots represent available/missing features in the current layer. The blue dots represent reference features.

where ε_n is the weight of the activation codes in layer n. The higher-layer activation of the reference datum is utilized to infer and estimate the missing features of incomplete data just like Equation (22):

$$(X_s)_{ij} = \sigma\,[A^1_{ij,pq}(f_{t^*})^2_{pq} R^{A,1}_{ij,pq} + b^1_{ij}], \quad s.t.(X_s)_{ij} \in F_s. \tag{22}$$

This process is demonstrated in Figure 4.

To the missing features $(X_s)_{ij}$, the average ranking distance of the reference datum is defined as $(m_s)_{ij}$:

$$(m_s)_{ij} = \sum_n^N \varepsilon_n L^n_{s,t^*}/N. \tag{23}$$

Based on the distribution of all available features in the same category, the probability level of the estimated feature $(X_s)_{ij}$ is defined as $(z_s)_{ij}$ in Equation (24):

$$\begin{aligned}(z_s)_{ij} &= \int_{(\mu)_{ij}+\|(X_s)_{ij}-(\mu)_{ij}\|}^{\infty} \frac{2}{\sqrt{2\pi}(\lambda)_{ij}} \exp^{-\frac{[y-(\mu)_{ij}]^2}{2(\lambda)^2_{ij}}} \mathrm{d}y, \quad (\mu)_{ij} = \frac{1}{\sum_t t}\sum_t (X_t)_{ij}, \\ (\lambda)_{ij} &= \sqrt{\frac{1}{\sum_t t}\sum_t [(X_t)_{ij} - (\mu)_{ij}]^2}, \quad s.t.(X_s)_{ij} \in F_s, (X_t)_{ij} \notin F_t, \mathbf{y}_s = \mathbf{y}_t\end{aligned}, \tag{24}$$

where $(\mu)_{ij}$ and $(\lambda)_{ij}$ are the mean value and the standard deviation of all available features $(X_t)_{ij} \notin F (1 \le t \le K)$.

As we know, the gate-to-source voltage and the drain-to-source voltage are two deciding factors of the output characteristic curve and the operating mode of FET. In our model, we use the probability level of the estimated feature to correspond to the gate-to-source voltage. The similarity between the reference datum and incomplete datum corresponds to the drain-to-source voltage. The sigmoid curve is selected to construct the reliability function. The reliability parameter $\Re^{\theta,1}_{ij}$ is determined by the probability level of the estimated feature $(z_s)_{ij}$ and the average ranking distance of the reference datum $(m_s)_{ij}$ as in Equation (25):

$$\Re^{\theta,1}_{ij} = \frac{2(z_s)_{ij}}{1+\exp(-S/(m_s)_{ij})} - (z_s)_{ij}, \tag{25}$$

where S is the parameter to control the shape of the curve.

The maximum value of $\Re^{\theta,1}_{ij}$ is determined by $(z_s)_{ij}$, which is consistent with the role of the gate-to-source voltage in FET. With a fixed $(z_s)_{ij}$, $\Re^{\theta,1}_{ij}$ tends to $(z_s)_{ij}$ when $(m_s)_{ij}$ approaches zero, and $\Re^{\theta,1}_{ij}$ tends to zero when $(m_s)_{ij}$ approaches infinity. Hence, $(m_s)_{ij}$ is

the inverse of the drain-to-source voltage. If the estimate of the missing feature is near to the expectation of the distribution of the available features from the same category, $(z_s)_{ij}$ approaches 1, which means this estimate is reasonable. If the activation codes of the reference datum are close to those of the current data, $(m_s)_{ij}$ approaches zero, which means this reference datum is reliable to do the estimation. When both of the requirements are satisfied, the reliability parameter $\Re_{ij}^{\theta,1}$ will get a high value.

3.4. Postactivation by Global Fine-Tuning

Earlier, we used the bidirectional inference algorithm by Field Effect RBM to construct a deep model. In this part, we use backpropagation through the whole deep model to fine-tune the parameters $\theta = [\mathbf{A}, \mathbf{b}, \mathbf{c}]$ for optimal recognition and estimation. Unlike the usage of backpropagation in DBN, in our fine-tuning stage, the missing features in incomplete data are slightly re-estimated.

In the layer-by-layer bidirectional inference stage, a search has been performed for a sensible and good region in the whole parameter space. Therefore, before proceeding to fine-tuning, we have already constructed a good data concept extraction model, and most of the missing features are roughly inferred and estimated. Now, backpropagation is utilized to tune the entire parameter space of FEBDN. Two tasks are involved in the stage of fine-tuning: finding good local optimum parameters $\theta^* = [\mathbf{A}^*, \mathbf{b}^*, \mathbf{c}^*]$ to recognize the data effectively and adjusting the estimates of missing features elaborately.

To the first task, the learning algorithm is used to minimize the cross-entropy error $[-\sum_s \mathbf{y}_s \log \hat{\mathbf{y}}_s]$, where $\mathbf{y}_s$ and $\hat{\mathbf{y}}_s$ are the correct label and the output label value of sample datum $\mathbf{X}_s$, respectively.

Simultaneously, to the second task, the rough estimates of the missing features are slightly adjusted and updated by Equation (26) and Equation (27). Let OF_s denote the set of output estimates of the missing features in sample datum $\mathbf{X}_s$:

$$t^* = \arg\min_t \left[\sum_n \varepsilon_n L_{s,t}^n \right], \quad s.t.(X_s)_{ij} \in F_s, (X_t)_{ij} \notin F_t, \mathbf{y}_s = \mathbf{y}_t \tag{26}$$

$$(X_s)_{ij} = \sigma[A_{ij,pq}^1 (f_{t^*})_{pq}^1 + b_{ij}^1], \quad s.t.(X_s)_{ij} \in OF_s. \tag{27}$$

For the test data, similar activation codes in the higher layer of the same predicted category are utilized to infer and estimate the value of the missing features. Equation (26) is substituted by the following equation:

$$t^* = \arg\min_t \left[\sum_n \varepsilon_n L_{s,t}^n \right], \quad s.t.(X_s)_{ij} \in F_s, (X_t)_{ij} \notin F_t, \hat{\mathbf{y}}_s^c = \hat{\mathbf{y}}_t^c = 1. \tag{28}$$

3.5. Field Effect Bilinear Deep Networks Algorithm

The detailed procedure of FEBDN is described in Algorithm 1. Lines 2 to 17 consist of the stage of initial guess by field effect bilinear initialization. Lines 18 to 22 compose bidirectional inference. Lines 24 to 25 are the postactivation by global fine-tuning.

3.6. Discussion

In this part, we show the generalization of FEBDN by analyzing its relation with our previous work, BDBN [Zhong et al. 2011] and SBDBN [Zhong et al. 2012], and some representative existing works, including Imputing RBM [Salakhutdinov et al. 2007], sDBN [Tang et al. 2010], and PGBM [Sohn et al. 2013].

1) Relation with BDBN

In Section 2, we introduce our previous work, Bilinear Deep Belief Networks (BDBN). Unlike FEBDN, in all three learning stages, BDBN equally relies on all the features

ALGORITHM 1: Field Effect Bilinear Deep Networks

Input: Training data set X, Corresponding labels set Y, Missing features set F_k in $\mathbf{X}_k$

Number of layers N, Number of epochs M, Between-class weights $\mathbf{B}_{st}$, Parameter S

Within-class weights $\mathbf{W}_{st}$, Initial bias parameters $\mathbf{b}$ and $\mathbf{c}$, Momentum ϑ, Parameter α

Output: Optimal parameter space $\theta^* = [\mathbf{A}^*, \mathbf{b}^*, \mathbf{c}^*]$, Estimate of missing feature $(X_s)_{ij} \in OF_s$

for $m = 1,\ldots, M$ **do**
  for $n = 1,\ldots, N$ **do**
    if $n = 1$ $\quad T^n = X$
    else
      for $k = 1,\ldots, K$ **do**
        $\mathbf{T}_k^n = \sigma(\mathbf{T}_k^{n-1} A^{n-1} + c^{n-1})$
      end for
    end if
  while not convergent **do**
    $\mathbf{D}_V = \sum_{st} \mathbf{E}_{st} (\mathbf{X}_{s\cdot} * \mathbf{R}_{sl}^F - \mathbf{X}_{t\cdot} * \mathbf{R}_{tl}^F) \mathbf{V}\mathbf{V}^T (\mathbf{X}_{s\cdot} * \mathbf{R}_{sl}^F - \mathbf{X}_{t\cdot} * \mathbf{R}_{tl}^F)^T$, $\mathbf{D}_U = \sum_{st} \mathbf{E}_{st} (\mathbf{X}_{s\cdot} * \mathbf{R}_{sl}^F - \mathbf{X}_{t\cdot} * \mathbf{R}_{tl}^F)^T \mathbf{U}\mathbf{U}^T (\mathbf{X}_{s\cdot} * \mathbf{R}_{sl}^F - \mathbf{X}_{t\cdot} * \mathbf{R}_{tl}^F)$
    Fix $\mathbf{V}$, compute $\mathbf{U}$ by solving $\mathbf{D}_V \mathbf{u} = \lambda \mathbf{u}$, Fix $\mathbf{U}$, compute $\mathbf{V}$ by solving $\mathbf{D}_U \mathbf{v} = \lambda \mathbf{v}$
  end while
  Determine the size of next layer $P^{n+1} = row(\mathbf{U}^n)$, $Q^{n+1} = column(\mathbf{V}^n)$
  Compute initial connection weights $A_{ij,pq}^n(0) = (\mathbf{U}_{ip}^n)^T \mathbf{V}_{jq}^n$
  The energy in the current Field Effect RBM

$$E(\mathbf{h}^1, \mathbf{h}^2; \theta^1) = -\sum_{i=1,j=1}^{i\le I, j\le J} \sum_{p=1,q=1}^{p\le P^2, q\le Q^2} h_{ij}^1 A_{ij,pq}^1 R_{ij,pq}^{A,1} h_{pq}^2 - \sum_{i=1,j=1}^{i\le I, j\le J} b_{ij}^1 R_{ij}^{b,1} h_{ij}^1 - \sum_{p=1,q=1}^{p\le P^2, q\le Q^2} c_{pq}^1 h_{pq}^2$$

  Update the weights and biases $A_{ij,pq}^1 = \vartheta A_{ij,pq}^1 + \Delta A_{ij,pq}^1 R_{ij,pq}^{A,1}$, $b_{ij}^1 = \vartheta b_{ij}^1 + \Delta b_{ij}^1 R_{ij}^{b,1}$, $c_{pq}^n = \vartheta c_{pq}^n + \Delta c_{pq}^n$
  end for
  for $s = 1,\ldots, K$ **do**
  Obtain the estimates of missing features

$$t^* = \arg\min_t \left[\sum_n \varepsilon_n L_{s,t}^n \right], s.t.\ (X_s)_{ij} \in F_s, (X_t)_{ij} \notin F_t, \mathbf{y}_s = \mathbf{y}_t$$

$$(X_s)_{ij} = \sigma[A_{ij,pq}^1 (f_{t^*})_{pq}^2 R_{ij,pq}^{A,1} + b_{ij}^1],\ s.t. (X_s)_{ij} \in F_s$$

  Calculate gate-to-source, drain-to-source voltage

$$(z_s)_{ij} = \int_{(\mu)_{ij} + \|(X_s)_{ij} - (\mu)_{ij}\|}^{\infty} \frac{2}{\sqrt{2\pi}(\lambda)_{ij}} \exp^{\frac{[y-(\mu)_{ij}]^2}{2(\lambda)_{ij}^2}} \mathrm{d}y,\ (m_s)_{ij} = \frac{\sum_n^N \varepsilon_n L_{s,t}^{n,*}}{N}$$

  Update the reliability parameter of missing features $\Re_{ij}^{\theta,1} = \frac{2(z_s)_{ij}}{1 + \exp(-S/(m_s)_{ij})} - (z_s)_{ij}$
  end for
end for

Calculate optimal parameter space $\theta^* = \arg\min_\theta [-\sum_k \mathbf{y}_k \log \hat{\mathbf{y}}_k]$

Re-estimate the missing features $t^* = \arg\min_t [\sum_n \varepsilon_n L_{s,t}^n], s.t. (X_s)_{ij} \in F_s, (X_t)_{ij} \notin F_t, \mathbf{y}_s = \mathbf{y}_t$

$(X_s)_{ij} = \sigma[A_{ij,pq}^1 (f_{t^*})_{pq}^1 + b_{ij}^1],\ s.t. (X_s)_{ij} \in OF_s$

regardless of whether the features are missing. It can be viewed as one special version of FEBDN that only includes the saturation mode. And the reliability is "frozen" to be 1.

2) Relation with SBDBN

SBDBN is another special version of FEBDN, which is the opposite extreme of BDBN. Different from FEBDN, SBDBN does not adaptively adjust the reliability of the estimates for missing features. SBDBN only relies on fully exploiting the embedding information according to the available features rather than any completion of missing features. That is, there are only two modes in SBDBN: the cutoff mode and the saturation mode. For the available features, the reliability connections are set to be 1, and the reliability connections of the missing features are set to be 0.

3) Relation with PGBM

The PGBM algorithm focuses on separating the relevant patterns from irrelevant patterns. For the incomplete data classification task, the missing features can be considered as the irrelevant features. In this case, the switch units allow the model to make only those available units to contribute to the final prediction. It can be viewed as one special version of FEBDN, just as SBDBN, which only relies on fully exploiting the embedding information according to the available features rather than any completion of the missing features.

4) Relation with Imputing RBM

The idea of Imputing RBM is similar to FEBDN: treating the missing features as a set of variables and learning the values in the learning scheme. It utilizes the mean values of the available features as the initial values of the missing features; the whole parameter space will be constructed and updated based on the available features and the estimates of missing features regardless of whether the initial values are reliable. This setting will make the initial values dominate in the classification and estimation stage, which may impair the performance. In Section 4, we will demonstrate the performance of the FEBDN with Imputing RBM.

5) Relation with sDBN

From the comparisons of structures, both sDBN and FEBDN can be regarded as modified versions of DBN. In sDBN, the first layer is sparsely connected as the receptive field (RF). With these local connections, the model is more robust to noise or occlusion. sDBN uses the log probability to estimate which nodes should be unclamped. That is, sDBN could automatically identify the occluded regions in incomplete images. For feature estimation, sDBN presented a denoising algorithm that combines top-down and bottom-up inputs to "fill in" the missing features. In this process, a threshold is introduced to distinguish which nodes to unclamp, and only those available features are used to construct the recognition model. Different from sDBN, in bidirectional inference of FEBDN, all features with their corresponding reliabilities are involved in constructing the decision boundary and estimating the missing features. In Section 4, we will compare the recognition performance of the proposed FEBDN with sDBN.

4. EXPERIMENTS AND RESULTS

In this section, we demonstrate the performance of the proposed FEBDN on three standard datasets and a new dataset collected and constructed by our group. The first dataset is MNIST, a standard large database of handwritten digits, which is often used to illustrate the performance of deep models, and its subset has been used for performance comparison of incomplete data classification algorithms [LeCun et al. 1998]. The second dataset is CIFAR-10 [Krizhevsky and Hinton 2007], which is a standard large database of real-world natural images. The third standard dataset is the BioID face dataset, which is often used to illustrate the performance of face recognition [Jesorsky et al. 2001]. As we know, the more general case of incomplete data in our daily lives is that some key features in the data are not observable. Therefore, our

group tried to collect the fourth dataset, StarFace, with some incomplete face images due to occlusion of important facial feature regions.

For the common parameters of the deep learning techniques, we follow the classical setting of the standard Matlab toolbox for DBN [Hinton et al. 2006]. For example, the balance weight α is set as 0.5. And we used 50 iterations for pretraining and 200 iterations for fine-tuning with backpropagation. For other parameters in existing works, we follow their general settings in their papers. For example, the size receptive field is set as 7×7. In FEBDN, the weight ε_n in layer n is set as 1. For parameter S, FEBDN achieves better performance when it ranges from 5 to 10. For simplicity, we set it as 10 in our experiments. The classical setting for the size of the nodes in the first, the second, and the third hidden layer is 500, 500, and 2,000, respectively [Hinton et al. 2006].

We compare the performance of FEBDN with various state-of-the-art incomplete image recognition algorithms and representative deep learning models, including k-nearest neighbor estimation (k-NNE), SVM [Boser et al. 1992], LRCEM [Williams et al. 2005], GEOM [Chechik et al. 2008], QGME [Liao et al. 2007], WII [Dick et al. 2008], BDBN [Zhong et al. 2011], DBN [Hinton et al. 2006], Imputing RBM [Salakhutdinov et al. 2007], Conditional RBM [Salakhutdinov et al. 2007], PQBM [Sohn et al. 2013], CDBN [Lee et al. 2011], sDBN [Tang et al. 2010], SBDBN [Zhong et al. 2012], CNN [LeCun et al. 1998], and Field Effect Deep Neural Network (FEDNN). In k-NNE, the missing features were set with the mean value obtained from the nearest neighbors' instances. Neighborhood was measured using a Euclidean distance in the subspace relevant to each pair of samples. The number of neighbors was varied across one, three, five, 10, 15, and 20, and the best result is provided to make a comparison. In LRCEM, a Gaussian mixture model is learned by iterating between (1) learning a GMM model of the filled data and (2) refilling missing values using cluster means. In FEDNN, we combine the FET with a typical deep neural network to test whether FET can be combined with other deep learning methods to help recognize incomplete images.

4.1. Experiment on Handwritten Dataset MNIST

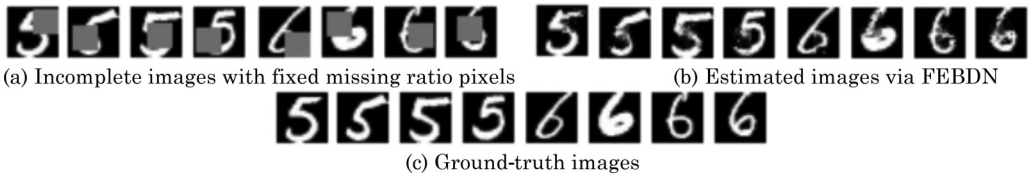
In this part, we explore the performance of FEBDN under a supervised learning scheme when features are missing at random. The first experiment in this dataset is used to demonstrate the effectiveness of FEBDN for recognition on incomplete images with a fixed missing ratio. In the second experiment, we demonstrate the incomplete image recognition when features are missing at random under different missing ratios. We test on the image dataset of handwritten digits, MNIST [LeCun et al. 1998]. MNIST is a standard database of handwritten digits containing 70,000 images with 10 classes. MNIST is widely used to compare deep learning performance [Salakhutdinov and Hinton 2007; Weston et al. 2008].

The first experiment on this dataset is used to demonstrate the effectiveness of FEBDN for recognition on incomplete images with a fixed missing ratio. We follow the same experimental setting of Chechik et al. [2008]: 1,200 images including 600 images of the digit 5 and 600 images of digit 6 are randomly selected from MNIST. These images are partitioned to 1,000 training data and 200 test data. We removed a square patch of pixels from each image that covered 25% of the total number of pixels. The location of the patch was uniformly sampled for each image.

We perform five random splits and report the average results over five trials. The recognition performance of FEBDN with other incomplete image recognition algorithms is shown in Table I. "Zero" means that the missing values were set to be 0. "Mean" means that the missing values were set as the average value of the features over all available data. From Table I, it can be observed that, compared with other

Table I. Average Recognition Accuracies (%) on Block Incomplete Digits 5 and 6 of MNIST

Model			Acc.
Proposed Model		FEBDN	**99**
Other Deep Model	Bilinear Deep Model	SBDBN	98.5
		BDBN (Zero)	97
		BDBN (Mean)	97.5
	Classical Deep Model	DBN (Zero)	96
		DBN (Mean)	96.5
Representative Model for Incomplete Data	Without Estimation	GEOM	95
	With Estimation	SVM (Zero)	95
		SVM (Mean)	95
		k-NNE	94
		LRCEM	95
	Joint Optimization	QGME	96.5
		WII	96



(a) Incomplete images with fixed missing ratio pixels (b) Estimated images via FEBDN

(c) Ground-truth images

Fig. 5. Samples of estimated images by FEBDN of the block missing features with fixed missing ratio.

state-of-the-art incomplete image recognition algorithms, deep learning models achieve better performance. This proves that the deep learning models have better recognition ability on incomplete data. Compared with two special versions of FEBDN, by fully exploiting the embedding information according to the available features, the recognition ability of SBDBN is better than BDBN. Relying on the field effect bilinear initialization and Field Effect RBMs, FEBDN obtains the best accuracy rate in all deep models. In Figure 5, some samples of the block incomplete images and the corresponding estimated images are demonstrated. Although some occluded blocks were located in the important parts in which digits 5 and 6 are similar, the estimated images obtained by FEBDN are correct.

In the second experiment, we demonstrate the incomplete image recognition when features are missing at random under different missing ratios: 60,000 images of 10 classes from MNIST are utilized as training data; the remaining 10,000 images are utilized for test. In this experiment, five random missing trials are performed and the average results over the five trials are reported. Some sample images with different missing ratios are shown in Figure 6. Although the image samples selected in Figure 6 are not difficult to recognize, when the missing ratio becomes higher, even humans cannot easily recognize these digits. Table II shows the performance comparison under different missing ratios. Obviously, FEBDN shows a higher incomplete recognition accuracy rate. Although recognition is adequately hard for a human when 80% of the features are missing, our algorithms demonstrate acceptable performance. In deep learning algorithms, the performance of CDBN+SVM is worse than others. Although CDBN can be utilized to estimate some missing features based on the shared weights in symmetrical parts, it is not proposed for the incomplete image classification. Thus, the random noise will have a bad influence on probabilistic max-pooling and higher-layer representations.

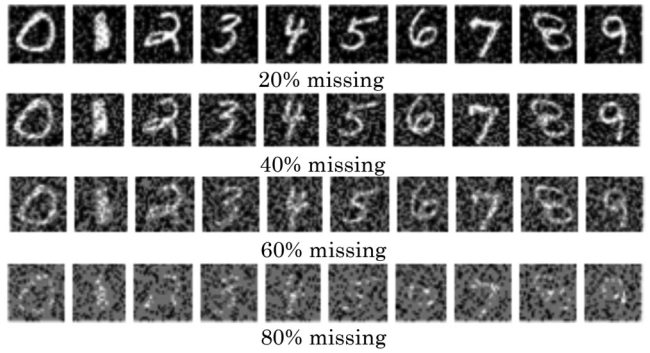


Fig. 6. Examples for different percentages of missing random pixels.

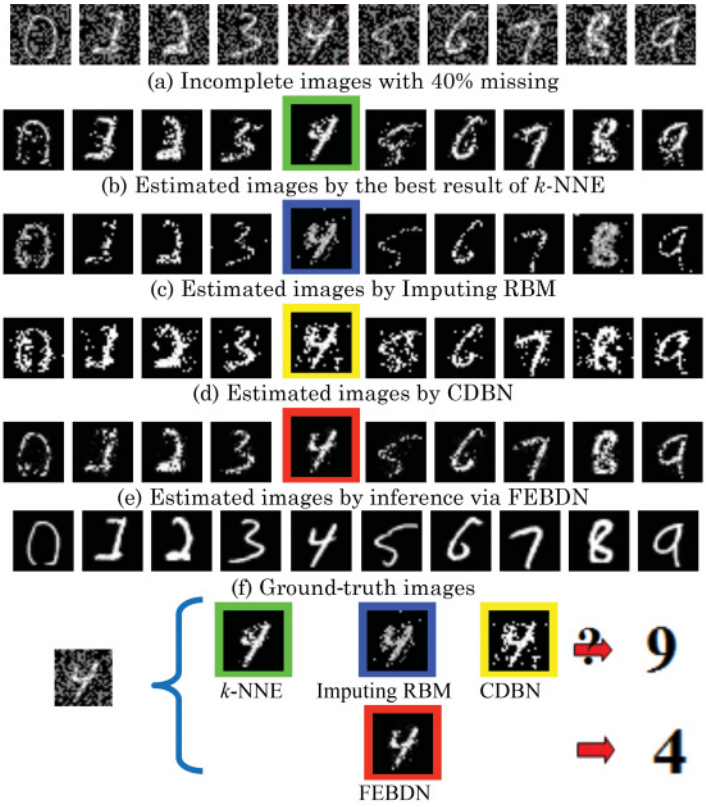
Table II. Average Recognition Accuracies (%) on Digits with Different Missing Ratio of MNIST

Alg. / Ratio	FEBDN	SBDBN	DBN(Zero)	DBN(Mean)	Conditional RBM	Imputing RBM
20%	**98.92**	97.45	96.99	96.8	92.82	97.42
40%	**98.34**	96.74	96.12	96.25	91.38	96.13
60%	**95.86**	94.4	92.78	94.23	88.2	92.96
80%	**84.98**	84.81	82.6	82.01	78.91	81.28
Alg. / Ratio	PGBM	CDBN+SVM	CNN(Zero)	CNN(Mean)	SVM(Zero)	SVM(Mean)
20%	97.42	84.46	96.86	96.88	87.84	87.25
40%	96.75	82.34	96.28	95.94	84.5	86.72
60%	94.37	80.03	94.52	93.53	80.96	82.7
80%	84.84	77.29	84.57	84.51	67.47	79.69

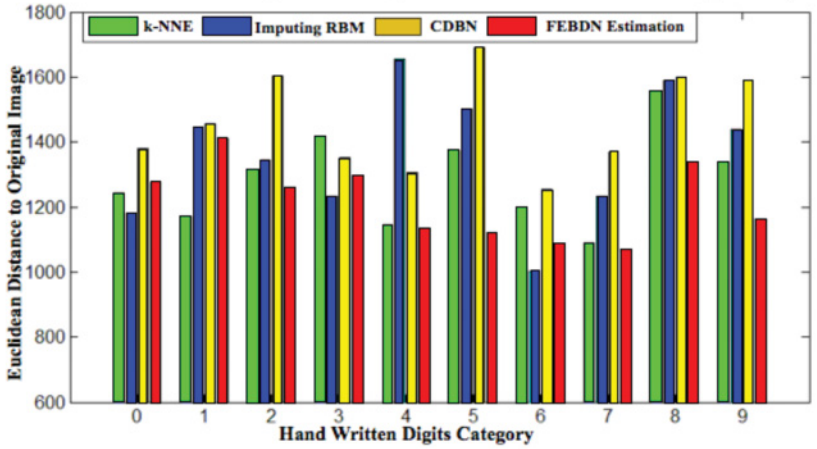
Additionally, to evaluate whether FEBDN has the ability to estimate the missing features, we compare our algorithm with the baseline estimation algorithm k-NNE and two other deep estimation algorithms: Imputing RBM and CDBN. In k-NNE, the number of neighbors was varied across one, three, five, 10, 15, and 20, and the result images with the shortest Euclidean distance were selected. Some samples are demonstrated in Figure 7. From Figure 7, we found that some handwritten digits are incorrectly estimated by k-NNE, Imputing RBM, and CDBN. For example, in Figure 7(g), by k-NNE, Imputing RBM, and CDBN, digit 4 is estimated just as digit 9. Although the distance between the ground-truth images and estimated images by both of the algorithms is not too far in Figure 7(h), the estimated images by k-NNE, Imputing RBM, and CDBN do not have enough discriminant information. In k-NNE and Imputing RBM, the estimates are based on the similarity of pixel-level features in the input layer and first hidden layer, respectively. In CDBN, the estimates rely on the shared weights from available features. But in FEBDN, the Field Effect RBMs help us to infer the missing features with better discriminant ability. This strategy helps our FEBDN achieve a better and effective estimate for the incomplete images.

4.2. Experiment on CIFAR-10

In this subsection, we further investigate the effect of FEBDN and other algorithms for image recognition on real-world natural images with missing features. The CIFAR-10 dataset [Krizhevsky and Hinton 2009] consists of 60,000 images with the a resolution of 32×32 from 10 classes (6,000 images per class), including 50,000 training images and 10,000 test images. This dataset includes 10 common categories, namely, "airplane," "automobile," "bird," "cat," "deer," "dog," "frog," "horse," "ship," and "truck." In



(g) Examples of the estimated images. The estimated images by k-NNE, Imputing RBM, CDBN, and FEBDN are shown with green, blue, yellow, and red boundaries, respectively.



(h) Mean squared error comparison from the estimated images with the ground-truth images.

Fig. 7. Samples of estimated images compared with 40% randomly missing ratio.

Table III. Average Recognition Accuracies (%) on CIFAR-10 with 20% Missing Ratio

Ratio \ Alg.	FEBDN	sDBN	FEDNN	Imputing RBM	PGBM	DBN (Zero)	DBN (Mean)	SVM (Zero)	SVM (Mean)
20%	**35.53**	34.91	35.21	32.87	33.41	31.60	33.23	28.38	29.27

Fig. 8. Sample images with missing important facial parts.

this experiment, we demonstrate the incomplete image recognition while the missing ratio of pixels is 20%. The gray values of the images are used as features. We perform five random splits and report the average results over five trials. The recognition accuracies on CIFAR-10 of FEBDN, sDBN, FEDNN, Imputing RBM, PGBM, DBN(Zero), DBN(Mean), SVM(Zero), and SVM(Mean) are shown in Table III. As we know, sDBN has the ability to identify the occluded regions in incomplete images before estimation. If this process is inaccurate, then it will have a direct effect on the accuracy. To ensure fair comparison among all these methods, the occluded regions are also marked in a preprocessing phase for sDBN. From Table III, it can be observed that FEBDN achieves a higher incomplete recognition accuracy rate, even compared with sDBN, Imputing RBM, and PGBM. The performance of FEDNN is better than sDBN. It evidences the idea that FET can be effectively combined with other methods to help recognize images with missing values.

4.3. Experiment on Face Image Dataset BioID

In this part, we explore the performance of FEBDN for face recognition on the dataset of BioID [Jesorsky et al. 2001]. The first experiment in this dataset is used to demonstrate the face recognition effectiveness of FEBDN when important facial features are missing. In the second experiment of this dataset, we verify the auto-encode ability of proposed FEBDN based on the incomplete similarity.

The BioID face dataset consists of 1,521 face images collected containing 23 subjects. The number of images in every category of BioID is varied, from 35 to 118. In our experiments, first, we select the categories with more than 50 face images as the subset that we work on. This subset includes 1,208 images in 14 categories. Then, just like the procedure on face datasets, the complete images are normalized (in scale and orientation) so that the two eyes are aligned at the same position. Finally, the facial areas are cropped and downsampled into the final images. The size of each final image in all of the experiments is 32×32 pixels, with 256 gray levels per pixel. The experiment in this dataset is used to demonstrate the face recognition effectiveness of FEBDN when important facial features are missing. In the preprocessing stage of every image, we removed a rectangle region of pixels automatically according to the pregiven coordinates of important facial features and generated five kinds of incomplete images. The locations of missing regions include the forehead, eyes, nose, mouth, and chin. Figure 8 demonstrates some sample images in this experiment.

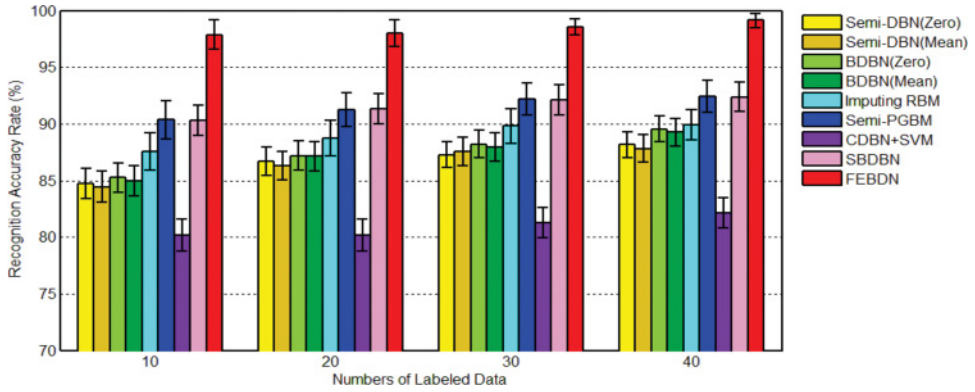


Fig. 9. Average recognition accuracy rates on test data in BioID with different numbers of labeled data.

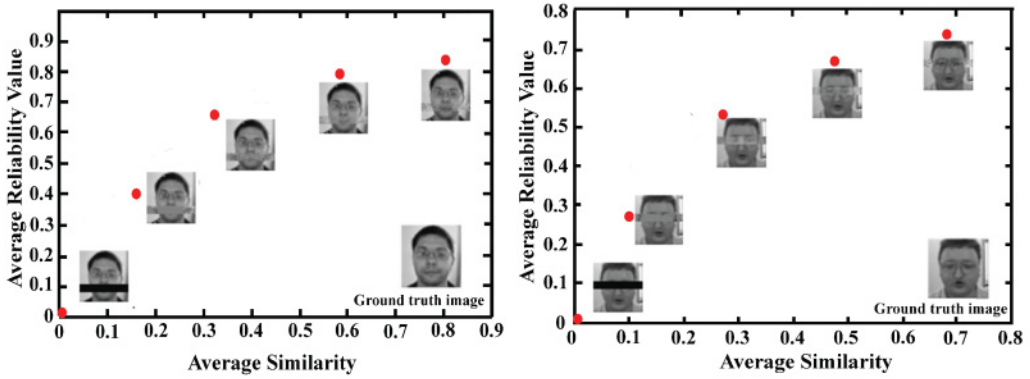


Fig. 10. The reliability update process with corresponding estimated images: (a) the estimated mouth and (b) the estimated eyes part.

In these experiments, deep learning models demonstrated better performance than other existing recognition models. In these experiments, we first compare the proposed FEBDN with other deep learning models under a semisupervised learning scheme. For this dataset, 250 images for each person with different missing regions are randomly selected to form the training set and the rest of the 2,540 images are utilized to form the test set. Different numbers of images of training data are randomly selected and labeled, while the other training data remain unlabeled. The number of selected labeled data in each category is equal to 10, 20, 30, and 40, respectively. We repeat each experiment for five random splits and report the average results over five trials. Figure 9 shows the face recognition accuracy rates of the test dataset. Although the recognition accuracies of Semi-DBN, BDBN, Imputing RBM, PGBM, and SBDBN are all higher than 80%, the recognition accuracy of FEBDN is the highest. As we know, Semi-DBN and BDBN equally trust on the available features and forecast unreliable features. The missing features located in the important facial regions will influence the recognition accuracies of them. It is obvious that the performance of PGBM and SBDBN is similar, and the performance of Imputing RBM is worse than them. These results are consistent with our previous discussion. Two examples of the average reliability update process with the estimated image are shown in Figure 10. With the aid of the bidirectional inference by Field Effect RBMs, the reliability of the estimates of the missing features is automatically and adaptively adjusted. With the increase in

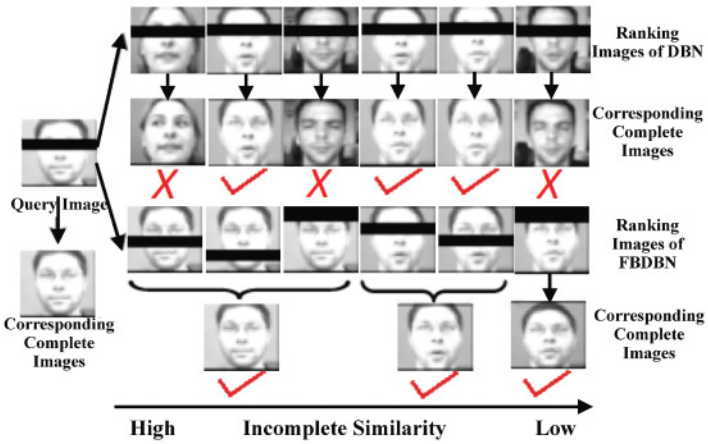


Fig. 11. Incomplete similarity comparison of DBN and FEBDN.

reliability, the estimated images are more and more similar with the ground-truth image without missing features. In this experiment, we also compare the recognition accuracies of FEBDN with some existing representative methods for each type of approach, including GEOM (without estimation), k-NNE (with estimation), and WII (joint optimization). When the number of selected labeled data in each category is equal to 40, the average accuracy of GEOM, k-NNE, and WII is 89.25%, 93.19%, and 91.06%, respectively. We could find that FEBDN is also better than these representative traditional algorithms.

In the second experiment of this dataset, we verify the autoencode ability of proposed FEBDN based on the incomplete similarity. In this experiment, we follow the parameter setting of the DBN encoder in Hinton et al. [2006] with 1,024, 1,000, 500, 250, and 30 numbers of nodes. All units are sigmoid except for the 30 linear units in the code layer. The whole parameter space is first learned by 40 labeled images per category and 110 unlabeled images per category. Then, for every query image in the training dataset, the low-dimension output code is compared with the output code of images in the test dataset. This incomplete similarity can be utilized to evaluate the high-level and low-dimension representation ability for incomplete data. We demonstrate one example of the incomplete similarity ranking results based on the Euclidean distance of the low-dimension output codes in Figure 11. It is obvious that in the ranking results of DBN, all the output codes with shorter distances are faces with missing eye parts. And most of them are not the images of the identical person. Contrary to DBN, in FEBDN, the output codes in the high-ranking level are the images of the same person with the same expression but missing different regions.

4.4. Experiment on Face Image Dataset StarFace

To further prove the effectiveness of the proposed FEBDN in real natural images, we collect and construct a new dataset, StarFace, from Google, including 120 face images of David Beckham, Victoria Beckham, Tom Cruise, and Julia Roberts and 5,000 face images from unknown people. Figure 12 shows some sample images in StarFace. We can see that some of the sample images are frontal with no occlusions, such as the first image of David Beckham and the first image of Victoria Beckham. It is also obvious that several important facial feature regions have been occluded in some samples, such as the second image of David Beckham, in which a hat has occluded his forehead, and the third image of David Beckham, in which his eyes are hidden behind sunglasses. In

Fig. 12. Sample images in StarFace.

Table IV. Comparisons of NDCG Scores in StarFace

	FEBDN	SBDBN	DBN (Zero)	DBN (Mean)
NDCG@10	**0.4904**	0.4208	0.3916	0.3787
NDCG@20	**0.4135**	0.3761	0.3289	0.3178

our experiment, for every occlusion region, we mark them as missing feature regions in the preprocessing stage. We evaluate the autoencode ability of the proposed FEBDN with its simplified version SBDBN and the classical DBN for face retrieval. For every class, we randomly select 50 images as training data, and 10 of them are labeled. In the remaining test data, we randomly select one image of the pop stars' faces with an important facial region missing as the query image. We measure the performance over five random splits and report the average results over the five trials. We calculated the Euclidean distance of the output codes between the query and other images in the test dataset to order the retrieved images. The mean value of Normalized Discounted Cumulative Gain (NDCG) is utilized to validate the retrieval results. NDCG measures the performance of a recommendation system based on the graded relevance of the recommended entities. This metric is commonly used in information retrieval. From the NDCG scores in Table IV, the FEBDN has better retrieval performance.

5. CONCLUSIONS

In this article, we propose a novel deep learning model, FEBDN, for image recognition with incomplete data. FEBDN has several attractive characteristics. First, incomplete image recognition is a classic challenge in computer vision and machine learning. But little work has been proposed to address this problem via deep learning methods. FEBDN is a deep learning model developed specifically for this problem. Second, inspired by attentional modulation in vision, we construct the reliability function to model the quality of features by reference to the output characteristic curve and operating modes of FET. Third, by integrating the reliability of features into the learning procedure, FEBDN can jointly determine the classification boundary and estimate the missing features. Moreover, in our experiments, FEBDN shows the distinguishing and robust ability to recognize incomplete data. FEBDN also achieves an effective estimate for the missing features in incomplete images. In the future, we will explore proposing a technique to identify the occluded regions in incomplete images and integrate it with the incomplete image classification technique. Due to the various origins of incomplete data, learning to find occlusion regions in incomplete data is a challenging problem. Most of the existing work for detection of occlusions is based on the information from consecutive frames, such as object boundaries, texture, or flow features in visual surveillance. Currently, we aim to address the recognition of incomplete data and the estimation of missing features. Based on these considerations, the occluded regions are marked missing in a preprocessing phase. We will investigate how to

combine the automatic occlusion region detection and incomplete recognition algorithms in a unified framework.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (No. 61502311, No. 61373122), Natural Science Foundation of Guangdong Province (No. 2016A030310053), Special Program for Applied Research on Super Computation of the NSFC-Guangdong Joint Fund (the second phase), the Science and Technology Innovation Commission of Shenzhen under Grant (No. JCYJ20150324141711640), and Shenzhen University research funding (201535).

REFERENCES

André Aleman, Koen B. E. Böcker, Ron Hijman, Edward H. F. de Haanb, and René S. Kahna. 2003. Cognitive basis of hallucinations in schizophrenia: Role of top-down information processing. *Schizophr. Res.* 64, 2–3, 178–185.

Pradeep K. Atrey, M. Anwar Hossain, Abdulmotaleb El Saddik, and Mohan S. Kankanhalli. 2010. Multimodal fusion for multimedia analysis: a survey. *Multimedia Syst*. 16, 345–379.

Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. 1992. A training algorithm for optimal margin classifiers. In *COLT*. ACM, New York, NY, 144–152.

Gal Chechik, Geremy Heitz, Gal Elidan, Pieter Abbeel, and Daphne Koller. 2006. Max-margin classification of incomplete data. In *NIPS*.

Gal Chechik, Geremy Heitz, Gal Elidan, Pieter Abbeel, and Daphne Koller. 2008. Max-margin classification of data with absent features. *J. Mach. Learn. Res.* 9, 1–21.

Hao Chen, Dong Ni, Jing Qin, Shengli Li, Xin Yang, Tianfu Wang, and Pheng Ann Heng. 2015. Standard plane localization in fetal ultrasound via domain transferred deep neural networks. *JBHI* 19, 5, 1627–1636.

Yanjiao Chen, Kaishun Wu, and Qian Zhang. 2015. From QoS to QoE: A tutorial on video quality assessment. *IEEE Commun. Surv. Tutorials* 17, 2, 1126–1165.

Uwe Dick, Peter Haider, and Tobias Scheffer. 2008. Learning from incomplete data with infinite imputations. In *ICML*. Citeseerx, Helsinki, Finland, 232–239.

Huijun Ding, Tan Lee, Ing Yann Soon, Chai Kiat Yeo, Peng Dai, and Guo Dan. 2015. Objective measures for quality assessment of noise-suppressed speech. *Speech Commun*. 71, 62–73.

Laura Folguera, Jure Zupan, Daniel Cicerone, and Jorge F. Magallanes. 2015. Self-organizing maps for imputation of missing data in incomplete data matrices. *Chemometr. Intell. Lab.* 143, 146–151.

Geoffrey E. Hinton and Roweis R. Salakhutdinov. 2006. Reducing the dimensionality of data with neural networks. *Science* 313, 5786, 504–507.

Oliver Jesorsky, Klaus J. Kirchberg, and Robert Frischholz. 2001. Robust face detection using the Hausdorff distance. In *AVBPA*. Springer-Verlag, London, UK, 90–95.

Alex Krizhevsky and Geoffrey E. Hinton. 2009. Learning multiple layers of features from tiny images. *Technical Report*. University of Toronto.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton. 2012. ImageNet classification with deep convolutional neural networks. In *NIPS*.

Honglak Lee, Roger Grosse, Rajesh Ranganath, and Andrew Y. Ng. 2011. Unsupervised learning of hierarchical representations with convolutional deep belief networks. *Commun. ACM*. 54, 10, 95–103.

Xuejun Liao, Hui Li, and Lawrence Carin. 2007. Quadratically gated mixture of experts for incomplete data classification. In *ICML*. ACM, New York, NY, 553–560.

Norbert R. Malik. 1995. *Electronic Circuits: Analysis, Simulation, and Design*. Prentice-Hall, Upper Saddle River, NJ.

Prabhu Natarajan, Pradeep K. Atrey, and Mohan Kankanhalli. 2015. Multi-camera coordination and control in surveillance systems: a survey. *ACM TOMM*. 11, 4, Article 57, 30.

Marc'aurelio Ranzato, Joshua M. Susskind, Volodymyr Mnih, and Geoffrey E. Hinton. 2011. On deep generative models with applications to recognition. In *CVPR*. 2857–2864.

Yann LeCun, Léeon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* 86, 11, 2278–2324.

Yann LeCun, Yoshua Bengio, and Geoffrey E. Hinton. 2015. Deep learning. *Nature* 521, 436–444.

Kun Li, Jingyu Yang, and Jianmin Jiang. 2015. Nonrigid structure from motion via sparse representation. *IEEE Trans. Cybern.* 45, 8, 1401–1413.

Archana Purwar and Sandeep Kumar Singh. 2015. Hybrid prediction model with missing value imputation for media data. *ESWA*. 42, 5621–5631.

Ruslan Salakhutdinov, Andriy Mnih, and Geoffrey Hinton. 2007. Restricted Boltzmann machines for collaborative filtering. In *ICML*. ACM, New York, NY, 791–798.

Ruslan Salakhutdinov and Geoffrey E. Hinton. 2007. Learning a nonlinear embedding by preserving class neighbourhood structure. In *AISTATS*. Omnipress, San Juan, Puerto Rico, 412–419.

Jürgen Schmidhuber. 2014. Deep learning in neural networks. *Technical Report*, 61, 85–117.

Kihyuk Sohn, Guanyu Zhou, Chansoo Lee, and Honglak Lee. 2013. Learning and selecting features jointly with point-wise gated boltzmann machines. In *ICML*. Citeseerx, Atlanta, GA, 217–225.

Charlie Tang and Chris Eliasmith. 2010. Deep networks for robust visual recognition. In *ICML*. ACM, 1055–1062.

Neill R. Taylor, Christo Panchev, Matthew Hartley, Stathis Kasderidis, and John G. Taylor. 2006. Occlusion, attention and object representations. In *ICANN*. Springer-Verlag, Athens, Greece, 592–601.

Jason Weston, Frédéric Ratle, and Ronan Collobert. 2008. Deep learning via semi-supervised embedding. In *ICML*. Springer, Berlin, 639–655.

David Williams, Xuejun Liao, Ya Xue, Lawrence Carin, and Balaji Krishnapuram. 2007. On classification with incomplete data. *IEEE TPAMI*. 29, 3, 427–436.

David Williams, Xuejun Liao, Ya Xue, and Lawrence Carin. 2005. Incomplete-data classification using logistic regression. In *ICML*. ACM, New York, NY, 972–979.

Hao-tian Wu, Jiwu Huang, and Yun-Qing Shi. 2015. A reversible data hiding method with contrast enhancement for medical images. *J. Vis. Commun. Image R.* 31, 146–153.

Wanmin Wu, Ahsan Arefin, Raoul Rivas, Klara Nahrstedt, and Renata M. Sheppard. 2009. Quality of experience in distributed interactive multimedia environments: toward a theoretical framework. In *ACM MM*. 1–10.

Xiaoshan Yang, Tianzhu Zhang, and Changsheng Xu. 2015. Boosted multifeature learning for cross-domain transfer. *ACM TOMM. Appl.* 11, 3, Article 35, 18.

Quanzeng You, Jiebo Luo, Hailin Jin, and Jianchao Yang. 2015. Robust image sentiment analysis using progressively trained and domain transferred deep networks. In *AAAI*.

Sheng-hua Zhong, Yan Liu, and Yang Liu. 2011. Bilinear deep learning for image classification. In *ACMMM*. ACM, New York, NY, 343–352.

Sheng-hua Zhong, Yan Liu, Fu-lai Chung, and Gangshan Wu. 2012. Semiconducting bilinear deep learning for incomplete image recognition. In *ICMR*. ACM, New York, NY, Article 32.

Sheng-hua Zhong, Yan Liu, Bin Li, and Jing Long. 2015. Query-oriented unsupervised multi-document summarization via deep learning model. *ESWA*. 42, 21, 8146–8155.

Mingyuan Zhou, Haojun Chen, John Paisley, Lu Ren, Lingbo Li, Zhengming Xing, David Dunson, Guillermo Sapiro, and Lawrence Carin. 2012. Nonparametric bayesian dictionary learning for analysis of noisy and incomplete images. *TIP*. 21, 1, 2012.

Received December 2015; revised May 2016; accepted May 2016