Information Sciences 609 (2022) 489–506

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Labeled projective dictionary pair learning: application to handwritten numbers recognition

Rasoul Ameri [a], Ali Alameer [b], Saideh Ferdowsi [c], Kianoush Nazarpour [d], Vahid Abolghasemi [c,*]

[a] Future Technology Research Center, College of Future, National Yunlin University of Science and Technology, Douliou, Taiwan, ROC
[b] School of Science, Engineering and Environment, University of Salford, UK
[c] School of Computer Science and Electronic Engineering, University of Essex, UK
[d] School of Informatics, University of Edinburgh, UK

ARTICLE INFO

Article history:
Received 17 October 2021
Received in revised form 14 July 2022
Accepted 15 July 2022
Available online 19 July 2022

Keywords:
Dictionary learning
Deep learning
Handwritten recognition
Histogram of oriented gradients
Image classification

ABSTRACT

Dictionary learning was introduced for sparse image representation. Today, it is a cornerstone of image classification. We propose a novel dictionary learning method to recognise images of handwritten numbers. Our focus is to maximise the sparse-representation and discrimination power of the class-specific dictionaries. We, for the first time, adopt a new feature space, i.e., histogram of oriented gradients (HOG), to generate dictionary columns (atoms). The HOG features robustly describe fine details of hand-writings. We design an objective function followed by a minimisation technique to simultaneously incorporate these features. The proposed cost function benefits from a novel class-label penalty term constraining the associated minimisation approach to obtain class-specific dictionaries. The results of applying the proposed method on various handwritten image databases in three different languages show enhanced classification performance ($\sim 98\%$) compared to other relevant methods. Moreover, we show that combination of HOG features with dictionary learning enhances the accuracy by 11% compared to when raw data are used. Finally, we demonstrate that our proposed approach achieves comparable results to that of existing deep learning models under the same experimental conditions but with a fraction of parameters.

1. Introduction

Handwritten number recognition is a sub-stage of the well-known Optical Character Recognition (OCR) technology. It is considered as a well-established pattern recognition problem which is traditionally addressed separately from handwritten alphabetical character recognition. This can be due to the importance of numeral values in our daily interactions, such as financial transactions, as well as complexity of mixed alphabetical-and-numeral character recognition. A typical OCR system includes: 1) text localisation on the input image, 2) line/character segmentation, and 3) character recognition [1]. The key role of OCR systems in our daily life is more revealing as nowadays we need to process various sources such as bank notes,

* Corresponding author.
E-mail addresses: D11023012@yuntech.edu.tw (R. Ameri), A.Alameer1@salford.ac.uk (A. Alameer), s.ferdowsi@essex.ac.uk (S. Ferdowsi), Kianoush.nazarpour@ed.ac.uk (K. Nazarpour), v.abolghasemi@essex.ac.uk (V. Abolghasemi).

https://doi.org/10.1016/j.ins.2022.07.070
0020-0255/

exam papers, and medical prescriptions, electronically. The importance of yielding high recognition accuracy of handwritten numbers in real-world applications, such as bio-metric authentication, financial systems, medical prescriptions, cell type classification, and postal mail sorting motivates us to design an efficient handwritten numbers recognition system. Another application where such systems are highly beneficial is in automated evaluation of homework, exam papers, which is becoming a necessity in e-learning platforms. Models with low accuracy are not suitable for real-world applications due to being unreliable [2,3]. Consequently, recognition of handwritten numbers (as opposed to printed numbers) from scanned documents has remained a challenging and interesting research theme within the field of computer vision and pattern recognition [4,5]. This becomes more challenging when handwritten numbers recognition in languages other than English (as addressed in this study) is of concern.

A generic handwritten recognition system uses machine learning to interpret and recognise the received handwritten text from different sources. Traditional recognition systems comprise two major stages: feature extraction and classification. The first stage transforms the input data into an informative and interpretable space while reducing the data dimensions. The second stage assigns class labels to the extracted features. Various techniques have been proposed for handwritten numbers classification where the main challenge is to learn efficient and comprehensive models capable of handling a diverse range of handwritten styles.

Pattern recognition is widely utilised in many applications from chest X-ray image classification to human recognition. Recent methods attempted to address the classification of handwritten numbers by combining neural networks, e.g. LeNet-5 and support vector machine (SVM), to gain better results. Other related methods, include utilising HOG and SVM to extract key features from input images followed by number classification. For example, in [6], HOG and Gabor filters were used as descriptors for feature extraction from Arabic words, leading to promising results using a k-nearest neighbour (kNN) classifier. HOG focuses on the structure of an object and can extract the gradient and orientation of edges in a given image. It was initially proposed for human detection and object localisation [7], however, it has shown great influence for feature extraction from text images and handwritten numbers [8]. Nevertheless, the efficacy of this powerful feature descriptor has not been thoroughly studied in this context, particularly for Chinese handwritten numbers.

The fast pacing developments of deep learning techniques have led to an increased tendency to embedding these in handwritten numbers recognition. A deep unsupervised network was proposed in [9] to learn invariant image representation from unlabeled data. The network architecture comprised a cascade of convolutional layers trained sequentially to represent multiple levels of features. In another work [10], Bengali handwritten number detection was performed using a deep structure called region proposal networks (RPN). Two major limitations of using deep learning methods for handwritten number recognition are the need for a relatively large annotated dataset and hardware requirements, e.g., GPUs (graphics processing units). Nevertheless, despite several works on Chinese handwritten *characters* recognition [11,12], there are no reported works on the performance of deep learning for Chinese handwritten *numbers*. This may be partly due to lack of a user-friendly and compact database of Chinese handwritten numbers. In this study, the recently published open-source Chinese handwritten dataset at Newcastle University[1] is considered.

Dictionary learning is another learning-based approach widely used for image representation, super-resolution, denoising and classification [13–15]. These have shown promising performance in image classification via providing a precise characterisation for any given image class using a dictionary matrix that describes key features of any sample in the same class [16]. Using a dictionary, all samples in a class can be represented as a sparse linear combination of the dictionary columns which allows a greater degree of discrimination. Nevertheless, finding a relevant and appropriate feature space for learning such dictionaries has remained a major challenge.

In this paper, we propose a novel method for handwritten numbers recognition based on dictionary learning. Our method is an extension of dictionary pair learning (DPL) [17] through using class labels and HOG features. It generates two types of synthesis and analysis dictionaries to classify handwritten numbers images. Our contributions in this paper are:

- Instead of using raw image pixels, we, for the first time, propose to use HOG descriptors to obtain class-specific dictionaries for handwritten recognition application. The motivation for opting for HOG is its robustness demonstrated in digit/character recognition applications. Although the significance of using statistical features in dictionary learning has already been implied [18] (for example in face recognition [19]), no HOG-based classification, built upon the DPL method, was proposed for recognition of handwritten numbers.
- We configure a novel penalty term by embedding class labels into our proposed cost function, which has a closed-form mathematical expression. This penalty term includes a label matrix and a coefficients matrix associated with each class.
- We provide mathematical derivations for minimising the cost function based on an alternating minimisation approach. The proposed technique provides an optimum trade-off between the grouping effect and the sparsity of coefficients without using ill-posed regularisers.
- To quantitatively evaluate the classification performance, we apply relevant deep neural network architectures in addition to other well-established dictionary learning methods to handwritten databases in three different languages.

[1] https://data.ncl.ac.uk/articles/dataset/Handwritten_Chinese_Numbers/10280831/1

2. Dictionary learning for image classification

Most conventional dictionary learning methods involve two key steps in which their performances are highly interdependent; sparse coding and dictionary update.[2] A naive way of building dictionaries is to stack all the training data into a matrix (the so-called dictionary). However, this approach leads to large and redundant dictionaries. Thus, many studies use machine learning techniques for obtaining dictionaries by extracting a low-dimensional feature domain from the training data. The main aim of these techniques is to determine a dictionary with approximately independent atoms. The utilised learning process, however, depends on the structure and nature of the input images. The efficiency of a dictionary is also dependent on the total number of coefficients containing in the associated sparse vectors. These sparse vectors together with dictionary atoms act as a coder for effective approximation of the data of interest. This idea can be extended from data representation to data classification, i.e., learning class-specific dictionaries.

Dictionary learning for classification has been utilised in face recognition [16] and brain signal separation [20]. A well-established work on dictionary learning for image classification is the sparse representation classifier (SRC) [16]. This technique uses sparse representation and learns dictionaries for classification of images in the pixel domain. It is increasingly being extended and used for a wide variety of image analysis, representation and classification tasks. To improve the classification accuracy, salient features were extracted from deformable objects, e.g., face images, in [21]. Then, two image enhancement and representation steps were combined to achieve a new sparse representation domain. In [22], a two-phase test sample representation method was proposed for face recognition. The method seeks to represent a test sample image as a linear combination of all the training samples looking for "M-Nearest Neighbours". In fact, a bank of training samples approach is fast and simple but not as accurate as those based on dictionary learning. A recently developed supervised dictionary learning approach constructs image classes using a shared dictionary and discirmintave class models [23]. One limitation associated with the above technique is its reduced capabilities in processing a large number of classes, i.e., large dictionary, due to the increased potentials of producing similar columns with high correlations. To efficiently scale up dictionary learning for increased number of classes, some researchers suggest to merge similar atoms in any given dictionary by optimising a customised objective function [24]. This procedure optimises the structure of the dictionary by reducing the mutual information among its atoms. In other words, this mechanism minimises the mutual information loss among the histogram of dictionary atoms across all the components of the data of interest. The main drawback of this approach is high computational cost of the feature merging stage which makes them impractical for large-scale classification problems.

Traditional dictionary learning methods merely rely on *synthesis* dictionaries in which the input data is in a sparse latent subspace. Meanwhile, K-singular value decomposition (K-SVD) is one of the well-recognised *synthesis*-based dictionary learning methods. A SVD-based method, which is called label consistent K-SVD (LC-KSVD), was introduced in [25], and later extended in [26]. LC-KSVD seeks the sparse coding problem by learning a discriminative dictionary using a modified cost function compared to that in K-SVD. Synthesis dictionaries can well preserve the local structures of the data. In contrast, *analysis* dictionaries rely on the assumption that the input data can be converted into a latent sparse subspace using the learned dictionary. Analysis dictionary can produce sparse representation of data via a simple data transformation, i.e., linear projection (simple dot product), without applying ℓ_0/ℓ_1 minimisation which is considered computationally expensive operations due to their non-convex nature.[3] For instance, an analysis discriminative dictionary learning has been proposed in [27] to process two-dimensional images. The method imposes a sparse $\ell_{2,1}$-norm constraint on the coding coefficients and attempts to learn dictionaries, representations, and linear classifiers as discriminant as possible. Recently, dictionary pair learning (DPL) approach was introduced, where both analysis (for generating discriminative code using linear projection), and synthesis (for image reconstruction) dictionaries were employed [17]. It benefits from an analysis-synthesis dictionary pair that avoids the need for utilising ℓ_0-norm or ℓ_1-norm minimisation. DPL has shown promising performance on face recognition application over state-of-the-art techniques. In [28], a discriminative sparse representation learning was proposed as an extension to DPL. Notably, this method preserves the local structures of the coding coefficients within each class by offering a structured reconstruction paradigm. In this study, the classification performance was evaluated on several face and scene datasets where promising results were reported. Another extension of DPL was proposed in [29] for pattern classification. The focus of this work is on classification of noisy images. The authors proposed a coding coefficient discriminant term to enhance discrimination power. To mitigate the influence of existing noise in input images, a low-rank constraint was introduced on each synthesis sub-dictionary. The authors evaluated the performance of their method using face and scene datasets.

Some recent works have addressed combination of dictionary learning and deep learning. Deep dictionary learning was proposed in [30] for building deeper architectures using the layers of dictionary learning. A method called deep micro-dictionary learning plus coding network was proposed in [31] which mainly includes standard deep neural network layers, such as pooling, fully, connected, and input/output. However, the deep learning architecture is augmented by replacing fundamental convolutional layers with a novel compound dictionary learning and coding layers. In [32], scalability and speed of deep learning were combined with dictionary learning to significantly reduce the number of parameters. This convolutional dictionary learning based auto-encoder was proposed for natural exponential-family distributions such as image denoising and neural spiking data analysis. A multi-layer dictionary learning network with added skip dense connections was proposed

[2] By definition, a sparse vector has few non-zero components. The quality of the learned dictionaries, i.e., the degree of the independence within the columns, directly affects the sparsity of the coefficients. Sparser coefficients with the smallest reconstruction error are preferred.

[3] Mathematically, p-norm of vector $\mathbf{x}$ is defined as $\|\mathbf{x}\|_p = \left(\sum |x_i|^p\right)^{1/p}$

for image classification [33]. The method offered a robust classification performance across several applications. Under a combined dictionary-and-deep learning approach in [34], a self-expressive adaptive locality preserving framework was proposed for object classification. The focus of this approach is to capture salient features from the samples. This approach used normalised block-diagonal coefficients to preserve the locality of the codes and salient features during the learning process. Promising results on face and natural scenes classification were reported. In [35], a transfer learning algorithm based on discriminative Fisher embedding and adaptive maximum mean discrepancy (AMMD) constraints was proposed. The method aimed to compensate for the drawback of general transfer learning algorithms where the interclass differences and intraclass similarities across domains are ignored. To this end, the label information of source domain and part of target domain were combined. Then, a model was constructed using atoms and profiles, which can adaptively minimise the distribution differences between source domain and target domain. The experiments on five public image dataets were implemented and promising results were achieved.

3. Proposed approach

In an earlier work, we applied DPL method to classify brain activities during hand movement tasks from electroencephalogram (EEG) signals [20]. In [20], some statistical features of the input data were used to obtain class-specific dictionaries. Later, we extended DPL method by injecting incoherence within the dictionary columns (the so-called InDPL method), where Chinese handwritten numbers were classified [36]. According to the literature and our investigations, selecting a proper feature space has a significant effect on the degree of discrimination in the learned dictionaries. The main novel ideas in this study are to 1) use HOG features (a strong image descriptor) as input to the dictionary learning process and 2) embed class labels in form of a mathematical constraint within a new cost function. The proposed method consists of four major steps: 1) pre-processing, 2) HOG feature extraction, 3) dictionary learning, and 4) classification. The first step is pre-processing, which includes binarisation, cropping, and resizing, and aims at enhancing the quality of raw images and preparing them for the next step. In the second step, the orientation histograms of edge intensity from pixels within local neighbourhoods are calculated to extract key HOG features. Then, the obtained HOG feature vectors are fed into the dictionary learning block to perform the necessary operations for obtaining the class-specific dictionaries. It is worth to mention that the entire procedure is carried out in two *training* and *testing* phases, and the classification task is performed only at testing phase on unseen image data. A self-explanatory representation of the proposed method outlining different steps in both phases is shown in Fig. 1.

3.1. Image databases

We used two Chinese handwriting numbers databases to analyse the effectiveness of our method. The first one is an open source database, published in association with our recent work in [36]. The database includes 15,000 images of handwritten numbers of size 64×64, written by 100 Chinese nationals studying at Newcastle University, UK. During data collection, the participants were asked to write 15 Chinese numbers given in Fig. 2a, 10 times. Another independent Chinese handwritten numbers database, which consists of 5,100 handwritten numbers from 34 persons, was also used to analyse this method. Each subject in this dataset wrote 10 times the 15 numbers illustrated in Fig. 2b.

In addition to the above Chinese databases, Arabic (MADBase[4]), English (USPS[5]), Persian (HODA[6]), and Urdu handwritten numbers (from MNIST-MIX[7]) were considered as case studies. MADBase consists of 70,000 numbers written by 700 persons that each person wrote 10 times each number from 0–9. Similarly, USPS database consists of 7,291 training samples and 2007 test samples of numbers 0–9 in form of grayscale images. Also, HODA database includes 60,000 training samples and 20,000 test sample. Urdu handwritten numbers taken from MNIST-MIX database includes approximately 6,600 training and 1,400 test images of size 28×28. Sample images of these databases are represented in Figs. 2c, 2d, 2e, and 2f.

3.2. Image pre-processing

In the pre-processing step, raw RGB images (e.g. Fig. 3A) are transformed into gray-scale space. In order to enhance the image contrast between background and foreground, a global image threshold is found using Otsu's technique [37]. This process is followed by converting the grayscale image into binary form, as shown in Fig. 3B. Then, the redundant background pixels, near image borders, are cropped by vertical and horizontal sweeping of the entire image. As a result of this process, the actual number is centered within a predefined bounding box as shown in Fig. 3C. In the final stage, the pre-processed images are down-sampled to 32×32 pixels which is applied to equalise the dimensions of input images and to avoid unnecessary computational burden (Fig. 3D).

[4] http://datacenter.aucegypt.edu/shazeem/
[5] https://git-disl.github.io/GTDLBench/datasets/usps_dataset/
[6] https://github.com/amir-saniyan/HodaDatasetReader
[7] https://github.com/jwwthu/MNIST-MIX

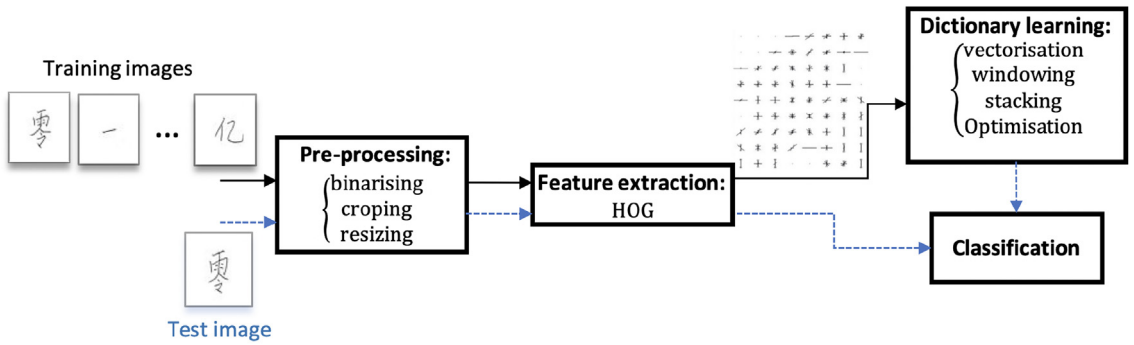
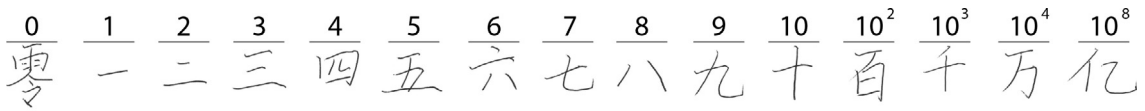
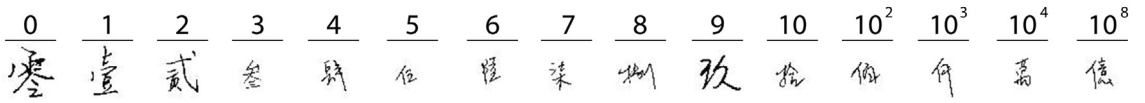


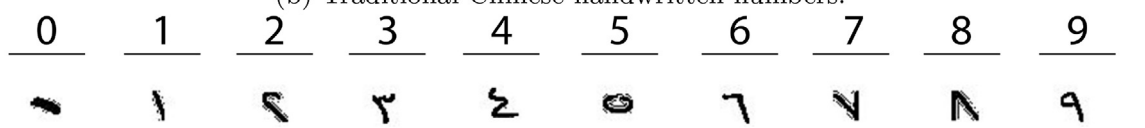
Fig. 1. Block diagram of the proposed method. Black solid and blue dashed lines, respectively, illustrate the flow of training and testing phases.



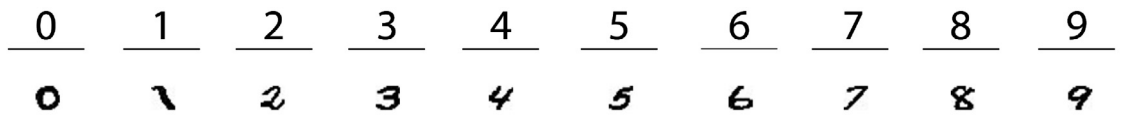
(a) Simplified Chinese handwritten numbers.



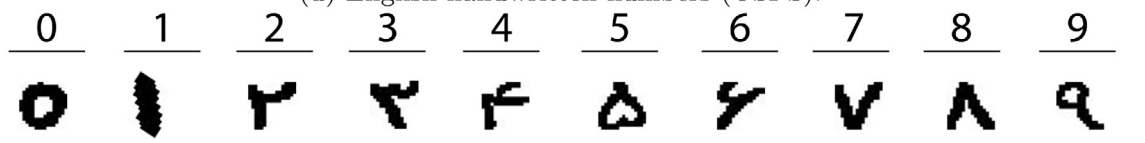
(b) Traditional Chinese handwritten numbers.



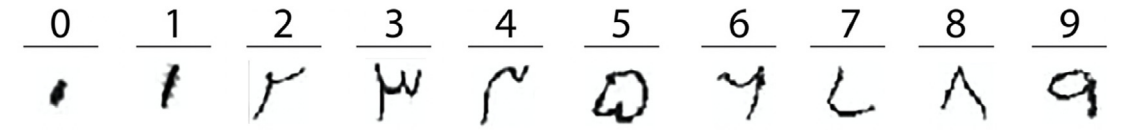
(c) Arabic handwritten numbers (MADBase).



(d) English handwritten numbers (USPS).



(e) Persian handwritten numbers (HODA).



(f) Urdu handwritten numbers (MNIST-MIX).

Fig. 2. Sample images from three different handwritten numbers databases and their equivalent English values.

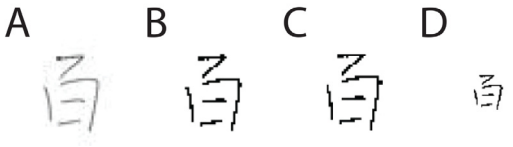


Fig. 3. Pre-processing sequence for an example Chinese handwritten image (number 100). (A) raw gray-scale image with size 64×64; (B) binarised image after applying Otsu's thresholding method; (C) cropped and centered image; (D) resized image to 32×32. Note that the images have been negated for better visualisation.

3.3. Feature extraction

The HOG descriptor focuses on the structure of an object and calculates the occurrences of gradient orientation in localised portions of the input image. By gradient, we mean small changes of pixels intensities in both vertical (v) and horizontal (h) directions [38]. The HOG feature extraction process includes three major steps: 1) calculating the gradient of every single pixel in both image directions (G_h, G_v) in a small neighbourhood: this is done by partitioning the input image into small square (3×3) patches and counting the occurrences of gradient directions using the central difference technique. 2) Determining the magnitude and orientation of each pixel value: the simplest way to do this is by following the Pythagoras theorem:

$$\begin{aligned} &\textit{Gradient Magnitude} = \sqrt{G_h^2 + G_v^2} \\ &\textit{Gradient Orientation} = \tan^{-1}(G_h/G_v) \end{aligned} \tag{1}$$

And 3) creating an image histogram to show the frequency distribution of any object in the image. This process is performed hierarchically by representing the values and directions of the HOG features for several image blocks. These blocks are then combined and normalised into a single output image according to the minimum and maximum contrasts in the image. The output is then vectorised and concatenated into the training matrix. In Fig. 4, an example for Chinese number '7' is illustrated. As seen from this figure, the HOG approach has extracted all existing directions and orientations.

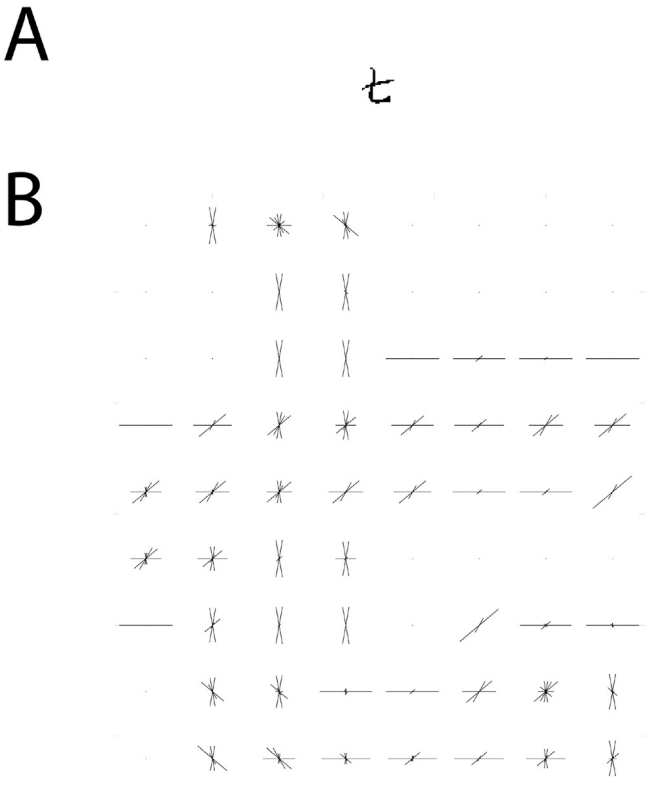


Fig. 4. The result of the HOG features for a sample number. (A) Input image (number 7); (B) the extracted HOG features.

3.4. Labeled projective dictionary pair learning

In a handwritten recognition problem, the dictionaries are basically language-dependent. During the training phase, one dictionary is obtained per class (i.e., per handwritten number). For example, Chinese and English handwritten numbers require 15 and 10 dictionaries, respectively. Atoms of a dictionary represent a coarse-to-fine feature space dedicated to that specific class. Therefore, all possible directions, edges, curves, and shapes in a handwritten number, are translated into a dictionary matrix through an iterative process. A cost function is configured to confine the learning space of the dictionaries to their associated classes. The obtained dictionaries are finally used to classify a new test handwritten image. Dictionary pair learning technique was originally designed to obtain a synthesis dictionary as well as an analysis dictionary for every class during training phase [17]. Here, we enhance this approach in two ways. Firstly, we embed the HOG features (as described above) into the system hierarchy to create more representative and discriminant dictionaries. Secondly, we introduce the prior knowledge of the class labels as a new term in the proposed cost function. The proposed procedure is mathematically described in the following.

Let us present all the input pre-processed images which are collated for training by matrix $Y = [Y_1, Y_2, \cdots, Y_Q]$, in which $Y_q \in R^{n\times k}$ encompasses the samples of q-th class (out of total Q classes), n is the training vectors' length (i.e. vectorised images), and k represents the number of training vectors for q-th class. We define analysis dictionary by $P \in R^{m\times(Q\times n)}$where m and $Q \times n$are the number of rows and columns, respectively. The value of m is selected empirically which is set to $m = 240$ in our study. The Similarly, we have the synthesis dictionary denoted by $D \in R^{n\times(Q\times m)}$which contains structural information about different classes. Given the training matrix and analysis dictionary, the sparse coefficient matrix can be expressed as $A = PY$. Considering the aforementioned notations, a generic dictionary learning and classification problem can be concretely expressed via:

$$< P^*, D^* >= \text{argmin}_{P,D}\|Y - DPY\|_F^2 + \Psi(D, P, Y) \tag{2}$$

where the term $\|Y - DPY\|_F^2$denotes the reconstruction error. Frobenius norm of a matrix X is defined as $\|X\|_F = \left(\sum\sum |x_{ij}|^2\right)^{1/2}$. The crucial task here is to design an appropriate penalty function, Ψ, that drives (2) to a successful classification. We consider three important factors to successfully form the cost function with the following objectives: 1) obtaining a sparse representation of the coefficients PY; 2) learning class-specific dictionaries, and 3) minimising the classification error. In what follows, we propose a new design for Ψ to meet the aforementioned criteria, i.e. having a discrimination power in addition to minimising the classification error. Then, a recurrent alternating approach is proposed to minimise the proposed objective function and find suitable dictionaries for each class.

To further investigate the role of synthesis and analysis dictionaries, let us expand $D = \{D_1, D_2 \cdots, D_q, \cdots, D_Q\}$and $P = \{P_1, P_2, \cdots P_q, \cdots, P_Q\}$ where $D_q \in R^{n\times m}$and $P_q \in R^{m\times n}$ represent the sub-dictionaries associated to q-th class. In order to obtain class-specific dictionaries, and thus discriminate between the classes, $P_q$should merely convey features from q-th class. At the same time, it should contain no features from the rest of the classes ($q\prime$). This property can be mathematically expressed as:

$$P_q Y_{q\prime} \approx 0, \text{where } q\prime \neq q \text{ and } 1 \leqslant q\prime, q \leqslant Q. \tag{3}$$

where $Y_{q\prime}$ includes all samples but those from class q, and $P_q Y_{q\prime} \approx 0$ means that the analysis dictionary associated to class q should solely be able to represent samples from class q. Such discriminability can be reformulated by $\|P_i\overline{Y_i}\|_F^2$ and added to the reconstruction error in (4). The matrix $\overline{Y_i}$ denotes the complementary data matrix to Y_i, meaning that it encompasses samples of all classes except those from i-th class:

$$\sum_{i=1}^{Q}\|Y_i - D_i P_i Y_i\|_F^2 + \lambda_1\|P_i\overline{Y_i}\|_F^2. \tag{4}$$

Although (4) can enforce the synthesis dictionaries to be discriminant, it does not utilise this feature in the analysis dictionaries. Since the class labels are available during training phase, we propose to inject these information into the cost function on analysis dictionaries. To do this, a linear predictive term, i.e., $f(Y; W) = WY$, is added to (4) in order to enforce analysis dictionaries to provide a higher level of discrimination. This effectively injects the classification error into the minimisation problem (i.e. cost function). Let Hbe the binary label matrix corresponding to training samples Y, and Wdenotes classifier parameters. To estimate P^*, D^*, W^* the following minimisation problem is proposed:

$$\begin{aligned} \text{argmin}_{P,D,W} \quad & \sum_{i=1}^{Q}\|Y_i - D_i P_i Y_i\|_F^2 + \lambda_1\|P_i\overline{Y_i}\|_F^2 + \lambda_2\|H_i - W_i P_i Y_i\|_F^2 \\ & s.t.\ \|d_j\|_2^2 \leqslant 1 \text{ for } j = 1, 2, \cdots m \end{aligned} \tag{5}$$

where $H = \{H_1, H_2 \cdots, H_q, \cdots, H_Q\}$and $H_q \in R^{Q\times k}$stands for the matrix that keeps the labels for training sample $Y_q$which belong to q-th class. Also, d_j refers to j-th column of the corresponding dictionary D. Matrix H has a block-diagonal structure composed of sub-matrices H_q. All components of H_q are zero except q-th row which are all ones. This is to maximise the effect of q-th class, and minimise the influence of the remaining classes during the learning process. Additionally, H would always be a binary matrix regardless of the language used. It is merely generated based on the size and labels of the training set, and the total number of classes (Q). For clarity, sample snapshot of an H for a 15-class problem with 10 training samples in every class (i.e., $Q = 15$ and $k = 10$) is given in Fig. 5.

Eq. (5) is generally non-convex and cannot be simultaneously solved for all variables. However, if we replace $A = PY$ into (5), the objective function will be converted to (6), where P^*, D^*, W^*, and A^* can be calculated using an alternate minimisation technique:

$$\begin{aligned}&\text{argmin}_{P,D,W,A}\sum_{i=1}^{Q}\|Y_i - D_iA_i\|_F^2 + \lambda_1\|P_i\overline{Y}_i\|_F^2 + \lambda_2\|H_i - W_iA_i\|_F^2\\&+\lambda_3\|P_iY_i - A_i\|_F^2 s.t. \|d_j\|_2^2 \leqslant 1\end{aligned} \tag{6}$$

In this equation, $\lambda_1, \lambda_2$and $\lambda_3$are positive scalars as regularisation parameters which are set empirically. The constraint on dictionary columns, i.e., $\|d_j\|_2^2 \leqslant 1$, is considered to keep all the dictionary columns normalised during the algorithm iterations. This will constrain the energy of each atom to avoid the trivial solution, i.e., $D = 0$. In order to solve (6) in all variables, alternating direction method of multipliers (ADMM) can be adopted where only one variable is found at a time, while other variables are kept unchanged. Such procedure is implemented for all variables based on the following steps.

Step 1: In order to find A that satisfies (6), we only consider 1st, 3rd, and 4th terms, fixing D_i, W_i, P_i, and minimise:

$$A^* = \text{argmin}_A\sum_{i=1}^{Q}\|Y_i - D_iA_i\|_F^2 + \lambda_2\|H_i - W_iA_i\|_F^2 + \lambda_3\|P_iY_i - A_i\|_F^2. \tag{7}$$

Since (7) only involves Frobenius norms, its gradient, with respect to A_i, can be simply obtained. So, minimisation of (7) is achieved by tending the gradient to zero and estimating:

$$A^* = \left(D_i^TD_i + W_i^TW_i + \lambda_3 I\right)^{-1}\left(D_i^TY_i + \lambda_2 W_i^TH_i + \lambda_3 P_iY_i\right) \tag{8}$$

Step 2: The same settings, carried out in previous step, is applied for P_i where all variables except P is considered fixed and only 2nd and 4th terms in (6) are included in the calculations:

$$P^* = \text{argmin}_P\sum_{i=1}^{Q}\lambda_1\|P_i\overline{Y}_i\|_F^2 + \lambda_3\|P_iY_i - A_i\|_F^2. \tag{9}$$

And then by taking the gradient with respect to P_i and equating it to zero we get:

$$P^* = \left(\lambda_3 Y_iY_i^T + \lambda_1\overline{Y}_i\overline{Y}_i^T + \gamma I\right)^{-1}\left(\lambda_3 A_iY_i^T\right) \tag{10}$$

where γ is a very small positive scalar to avoid zero division.

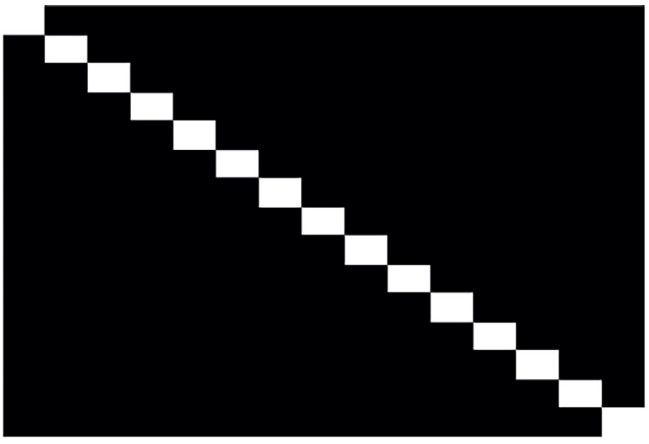


Fig. 5. Sample snapshot of binary matrix H for a 15-class problem with 10 training samples per each class. The size of this matrix is 15×150. White bars and black regions correspond to 1's and 0's, respectively.

Step 3: Since W_i appears only in the 3rd term in (6), its value can be simply estimated using the following expression:

$$W_i^* = \left(A_i A_i^T + \gamma I\right)^{-1}\left(H_i A_i^T\right). \tag{11}$$

Step 4: Finally, by fixing P, A, W and applying ADMM [39] we can estimate dictionaries D:

$$\begin{aligned} D^{(r+1)} &= \min_D \sum_{i=1}^{Q} \|Y_i - D_i A_i\|_F^2 + \rho\|D_i - S_i^{(r)} + T_i^{(r)}\|_F^2 \\ S^{(r+1)} &= \min_S \sum_{i=1}^{Q} \rho\|D_i^{(r+1)} - S_i^{(r)} + T_i^{(r)}\|_F^2 s.t. \|S_i\| \leqslant 1 \\ T^{(r+1)} &= T^{(r)} + D_i^{(r+1)} - S_i^{(r+1)}. \end{aligned} \tag{12}$$

The pseudo-code of the proposed method is summarised in Algorithm 1. The algorithm terminates after elapsing a fixed number of epochs or when the value of objective function reaches a pre-defined threshold and does not reduce further as the epochs evolve.

Algorithm 1: Pseudo-code of the proposed LpDPL method.

Input : Pre-processed handwritten images as training samples belong to Q classes $Y = [Y_1, ..., Y_2, ..., Y_Q]$.

Initialisation:
Generate random D_0 and P_0
Set m, λ_1, λ_2, λ_3, γ
Calculate A_0 in (8) and W_0 in (11), and set $t = 0$
while *not converged* **do**
  $t = t + 1$
  for $i = 1 : k$ **do**
    Update $A_k^{(t)}$ using (8)
    Update $P_k^{(t)}$ using (10)
    Update $W_k^{(t)}$ using(11)
    Update $D_k^{(t)}$ using (12)
  end
end
Output: P, D, W

3.5. Classification

After completion of the training phase with the labeled handwritten images, our model parameters including class-specific dictionaries P and D as well as the weights matrix (also referred as the transformation matrix) W are obtained. In testing phase, an unseen handwritten image is entered into the pipeline, shown in Fig. 1. Assume x to be the HOG output in vectorised form, the corresponding class can be simply found via:

$$Class(x) = \text{argmin}_i \|x - D_i P_i\|_F^2 + \|H_i - W_i P_i x\|_F^2.$$

Eq. (13) is executed for all $i = 1, \cdots, Q$ and the output is recorded as the predicted class of the input image.

4. Experimental results

In order to assess the effectiveness and performance of the proposed method we conducted extensive experiments with handwritten images in different languages. The classification results of the proposed method on two independent Chinese handwritten numbers databases, one Arabic and one English handwritten numbers database are presented and compared with state-of-the-art methods. Finally, the classification accuracy of the proposed method is compared and analysed with those obtained using conventional deep learning models. In all experiments, scalars $m, \lambda_1, \lambda_2$and $\lambda_3$were independently drawn based on performing 10-fold cross-validation on the training sub-set. We employed random initialisation for both $D$and $P$for every class. The obtained parameters through this procedure are used to initialise $A_0$and W_0as given by Eqs. (8) and (11).

4.1. Recognition performance

To investigate the robustness of the proposed method with Chinese handwritten database under various settings, we considered three different cross-validation procedures: between-subjects, within-subjects, and conventional. Between-subjects cross-validation is designed to explore how robust our learned model against each individual's handwriting style is. For this purpose, we considered all data from one subject in the database as test sub-set, while data of other subjects were considered as training sub-set. This process was sequentially repeated for every subject. In the within-subject cross-validation, the aim was to evaluate the recognition performance when the writings of all subjects are included during the training phase. To do this, we put aside one sample image from each subject for the testing phase while all the remaining samples were used for training. This process was repeated 10 times, and the results were averaged. Finally, in conventional cross-validation, no priority was given to the subjects and we performed random 10-fold cross-validation on all the handwritten images in the database.

In Table 1, the achieved classification scores for the proposed method with Chinese handwritten number database are provided. According to this table, LpDPL outperforms (by 3.8%, in average) the classification results reported in [36] where InDPL was applied under the same conditions. This is an indication that the proposed penalty terms here, i.e. classification labels as well as the proposed novel HOG features, significantly enhanced the performance of the class-specific dictionaries. To investigate the influence of using HOG features in the proposed method, we ran our method under the same experimental environment, i.e., parameters and data, without using the HOG features. The achieved classification accuracy reduced by $\sim 11\%$. This experiment highlighted the significant impact of HOG features in this context.

To compare the performance of the dictionary learning for classification against the classic k-nearest neighbour (kNN) approach, an experiment was conducted under three different conditions, i.e., conventional, within-subject and between-subject cross validations. Fig. 6 depicts average classification accuracy when LpDPL, DPL [17] and kNN were applied on Chinese handwritten numbers database. As seen from Fig. 6, both LpDPL and DPL significantly outperform kNN which confirms the strength of dictionary learning-based approaches for pattern classification. It is noteworthy to mention that HOG features were used for classification in both LpDPL and DPL for this experiment. Hence, LpDPL achieves slightly higher accuracy against DPL which implies the improvement due to adding class labels as a penalty term into the cost function.

To observe fine details of classification performance, the resultant confusion matrix, when LpDPL applied, is given in Fig. 7. The results are depicted for all databases and languages considered in our study. It can be noticed that the developed method performs consistently well on all databases (look at the diagonal elements). For most cases, the number of misclassified images remained below 10. The poorest classification occurred between Chinese numbers '10' and '1000' (Fig. 7a), respectively. This is believed to be because of the obvious semantical similarities exist between these numbers (Fig. 2a).

Table 1
Classification accuracy of LpDPL and InDPL [36] with Chinese handwritten database under three types of cross-validations, i.e., conventional, within-subject, and between-subjects.

Method	Conventional	Within-subject	Between-subjects
LpDPL	98.53%	98.56%	98.07%
InDPL [36]	93.00%	93.13%	97.53%

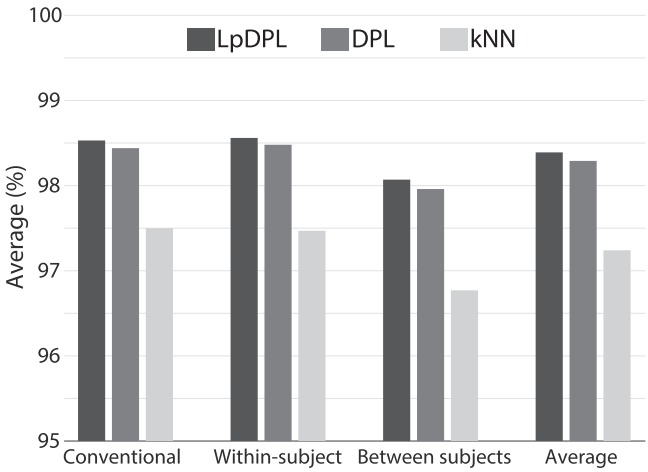
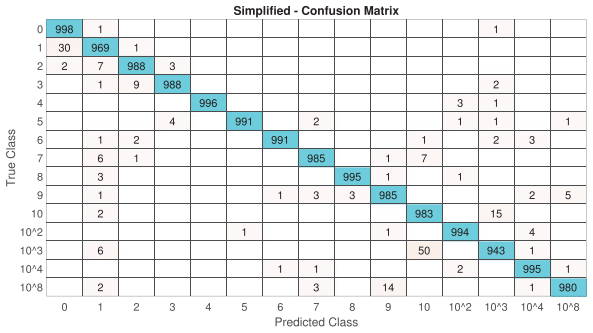
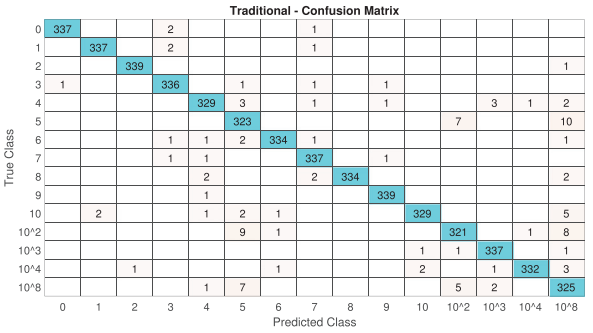


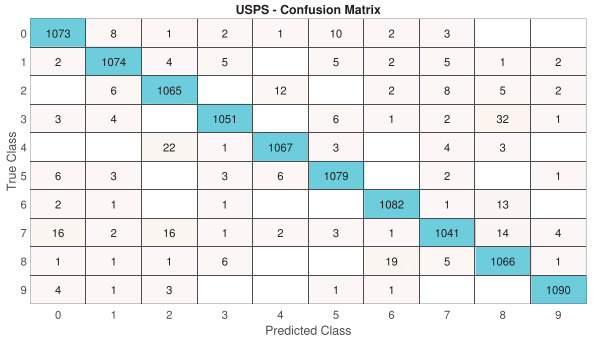
Fig. 6. Classification performance for three methods, i.e., LpDPL, DPL, and kNN, under three cross-validation settings.



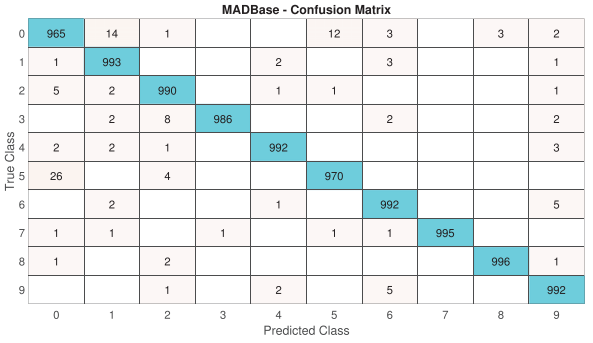
(a) Simplified-Chinese



(b) Traditional-Chinese



(c) USPS-English



(d) MADBase-Arabic

Fig. 7. Confusion matrix illustration as a result of applying LpDPL on different handwritten database under conventional cross-validation. The diagonal values indicate the number of correctly classified images, and off-diagonal elements represent incorrectly classified images corresponding to each target class.

From Fig. 7c, the lowest performance with English language (USPS database) is reported for numbers '3' and '8' due to their similarities.

We also compared the performance of the proposed method with existing dictionary leaning methods such as SRC [16], DLSI (dictionary learning with structure incoherence) [40], LCKSVD1 [26], LCKSVD2 [26], JDDRDL (joint discriminative dimensionality reduction and dictionary learning) [41], CRC (collaborative representation classification) [42], ESRC (extended sparse representation classification) [43], SSRC (superposed sparse representation classification) [44] and DPL (with the proposed HOG features) [17], under conventional cross validation setting. In this analysis, we used all four databases, i.e., two independent Chinese handwritten, one Arabic and one English sets. Also, four well-known evaluation metrics, i.e., Accuracy, recall, Precision and F1Score, are calculated and reported in this table. As seen from the results in Table 2, LpDPL outperforms other well-established techniques. Overall, the highest recorded performance was obtained for LpDPL with all databases. Among these methods, ESRC and SSRC have shown comparable performance with LpDPL. This reveals the effectiveness of the proposed hierarchy, i.e., the combination of the added penalty terms and HOG features.

As reviewed in Section I, HOG features have recently shown to be suitable descriptors for Arabic words too [6]. Therefore, we expected the proposed method performs well with Arabic handwritten numbers as it relies on HOG features for dictionary learning and classification. The results given in Table 2 support the effectiveness of using HOG features for dictionary learning with Arabic handwritten database. For further comparison, we also applied a convolutional neural network (CNN), named LeNet-5, to Arabic numbers MADBase [45]. It has scored 88% classification accuracy; this indicates that our method outperform the latter in a large margin (∼10%). Among all other methods listed in Table 2, LpDPL achieved higher accuracy. Our experimental results on different languages confirm the generalisation of the proposed method for handwritten databases in other languages. Another interesting finding that can be revealed by comparing results of Tables 1 and 2 is that HOG features has more tangible effects on Chinese numbers (than Arabic) which have complicated textures involving many line orientations. It is also noteworthy to mention that no significant changes in parameters were required for applying LpDPL to Arabic numbers.

To further evaluate system performance, we applied the proposed method on an English handwritten numbers database. According to the obtained results with USPS database, represented in Table 2, the proposed method outperforms existing relevant techniques where highest average accuracy of 97.17 % has achieved for LpDPL.

Table 2
Comparison of classification performance for popular dictionary learning methods on two Chinese handwritten databases; simplified and traditional, English (USPS), Arabic (MADBase), Persian (HODA), and Urdu (MNIST-MIX) databases.

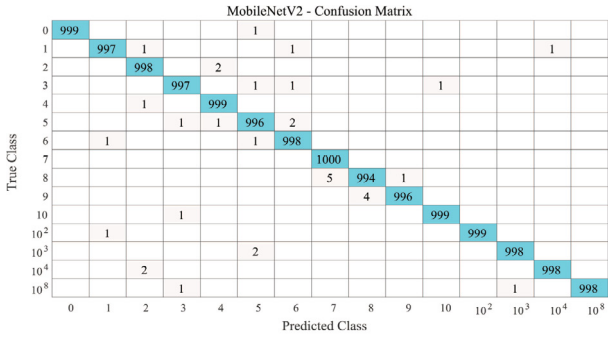
	Simplified database (Chinese)				Traditional database (Chinese)			
Method	Accuracy	Recall	Precision	F1Score	Accuracy	Recall	Precision	F1Score
SRC [16]	90.97%	90.93%	91.71%	91.32%	87.47%	87.47%	88.26%	87.86%
DLSI [40]	97.80%	97.79%	97.85%	97.82%	97.58%	97.58%	97.60%	97.59%
LCKSVD1 [25]	95.23%	95.23%	95.31%	95.26%	90.65%	90.65%	90.59%	90.62%
LCKSVD2 [26]	95.24%	95.23%	92.31%	95.27%	90.67%	90.67%	90.60%	90.62%
DPL [17]	98.36%	98.36%	98.37%	98.37%	97.82%	97.82%	97.84%	97.83%
JDDRDL [41]	97.19%	97.20%	97.25%	97.22%	95.08%	95.08%	95.08%	95.08%
CRC [42]	96.06%	96.06%	96.09%	96.08%	95.74%	95.74%	95.76%	95.75%
ESRC [43]	**98.86**%	**98.86**%	**98.86**%	**98.86**%	97.80%	97.80%	97.79%	97.80%
SSRC [44]	98.39%	98.39%	98.41%	98.40%	96.47%	96.47%	96.47%	96.47%
LpDPL	98.54%	98.54%	98.55%	98.55%	**97.82**%	**97.82**%	**97.86**%	**97.84**%
	USPS (English)				**MADBase (Arabic)**			
Method	Accuracy	Recall	Precision	F1Score	Accuracy	Recall	Precision	F1Score
SRC [16]	81.81%	81.80%	82.72%	82.26%	90.97%	90.93%	91.71%	91.32%
DLSI [40]	96.10%	96.10%	96.14%	96.12%	97.62%	97.62%	97.64%	97.63%
LCKSVD1 [25]	91.25%	91.25%	91.27%	91.26%	96.49%	96.49%	96.48%	96.49%
LCKSVD2 [26]	91.10%	91.10%	91.11%	91.10%	96.49%	96.49%	96.48%	96.49%
DPL [17]	96.68%	96.68%	96.72%	96.70%	98.21%	98.21%	98.22%	98.22%
JDDRDL [41]	95.36%	95.36%	95.54%	95.45%	97.85%	97.85%	95.86%	95.86%
CRC [42]	95.60%	95.60%	95.65%	95.63%	97.20%	97.20%	97.24%	97.22%
ESRC [43]	93.75%	93.75%	93.81%	93.78%	97.60%	97.60%	97.60%	97.60%
SSRC [44]	**97.24**%	**97.24**%	**97.24**%	**97.24**%	98.00%	98.00%	98.08%	98.04%
LpDPL	97.17%	97.16%	97.17%	97.17%	**98.71**%	**98.71**%	**98.71**%	**98.71**%
	HODA (Persian)				**MNIST-MIX (Urdu)**			
Method	Accuracy	Recall	Precision	F1Score	Accuracy	Recall	Precision	F1Score
SRC [16]	82.69%	82.69%	84.35%	83.51%	85.32%	83.33%	83.65%	83.48%
DLSI [40]	97.30%	97.30%	97.42%	97.36%	89.03%	88.94%	88.93%	88.93%
LSKSVD1 [25]	91.15%	91.15%	91.62%	91.39%	87.48%	88.45%	89.05%	88.75%
LSKSVD2 [26]	91.15%	91.15%	91.58%	91.37%	87.74%	88.04%	88.42%	88.22%
DPL [17]	98.46%	98.46%	98.46%	98.46%	95.23%	95.45%	95.23%	95.34%
JDDRDL [41]	95.38%	95.38%	95.46%	95.42%	92.87%	92.36%	91.28%	91.81%
CRC [42]	93.08%	93.08%	93.50%	93.29%	90.52%	90.52%	91.13%	90.82%
ESRC [43]	98.08%	98.08%	98.20%	98.13%	94.34%	95.16%	95.61%	95.38%
SSRC [44]	98.84%	98.84%	98.91%	98.88%	96.23%	96.23%	96.96%	96.59%
LpDPL	**98.85**%	**98.85**%	**98.89**%	**98.87**%	**98.87**%	**98.04**%	**98.11**%	**98.08**%

4.2. LpDPL versus deep learning

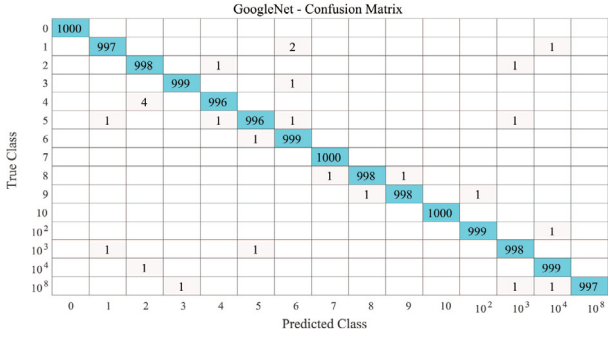
While the core mechanism of two learning-based methods, i.e., dictionary learning and deep learning, are different, it is worthwhile to analyse and compare their performances. Since deep learning is also widely used for classification problems, we considered several well-framed deep learning models, namely, MobileNetV2 [46], GoogLeNet [47], and SqueezeNet [48], to run with our Chinese handwritten database. The main limitation of using deep learning is the need for large training dataset. To maximise the performance of these models and present a fair comparison, we deployed fully-optimised versions of these platforms which were already trained on the large well-known ImageNet database [49]. Furthermore, a CNN-based method, with majority of parameters derived from the convolutional layer and the fully connected layer, was tested [50]. The results showed that the average classification accuracy when deep learning models were used are comparable with that of LpDPL (98.53%). More specifically, the following classification accuracies have been achieved: MobileNetV2 (98.55%), GoogleNet (99.83%), SqueezeNet (98.53%), and CNN (97.95%). Among these results, GoogleNet has recorded the highest accuracy. Moreover, the obtained results reveal that the proposed approach is more robust in recognising complex Chinese handwritten characters, e.g., number 9 and number 12; in comparison with the deep learning models. This can be observed by inspecting the confusion matrices provided in Fig. 8 for these three deep neural networks.

4.3. Optimisation performance

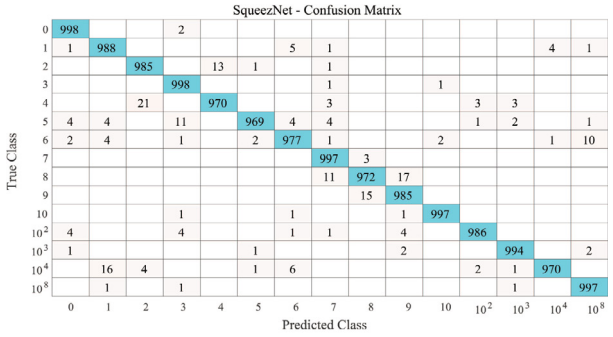
We investigate the optimisation performance of the proposed LpDPL method by reporting the results of two relevant experiments. Fig. 9a depicts the recorded trend in the value of objective function in (6) through 10 epochs. The proposed algorithm converges very fast while presenting a monotonic reduction rate in the value of objective function. One important parameter that affects the optimisation performance is the dictionary size m (number of columns in the synthesis dictionary D). To explore its impact, an analysis was conducted using the same experimental environment, however, with different val-



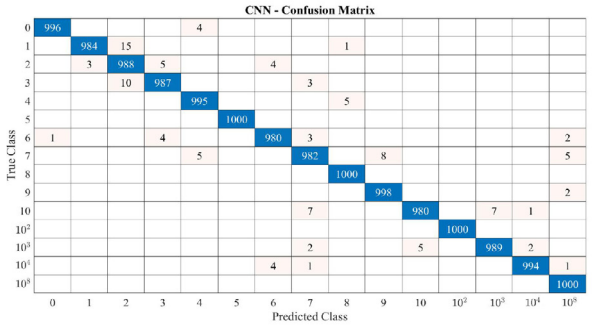
(a) MobileNetV2



(b) GoogleNet



(c) SqueezeNet



(d) CNN

Fig. 8. The confusion matrices obtained after applying three different deep learning models on Chinese handwritten database.

ues of m. The recorded classification accuracy for conventional 10-fold cross-validation using both DPL and LpDPL methods is illustrated in Fig. 9b. This figure shows that the highest and most consistent accuracy with LpDPL is achieved at around $m = 340$. Therefore, we adopt this value for dictionary size through all of the experiments in this study.

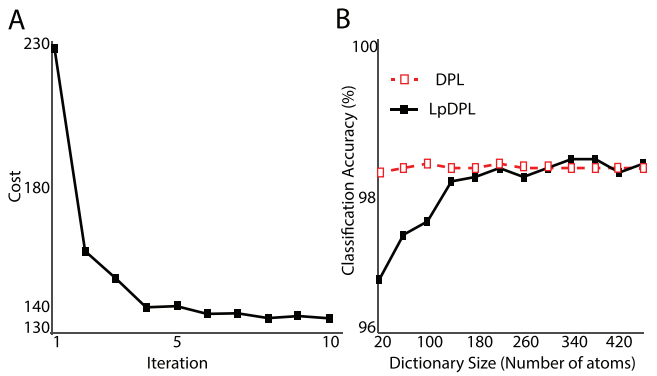


Fig. 9. Optimisation performance; A) value of objective function in Eq. (6) versus varying number of epochs; B) Variation in classification accuracy for DPL and LpDPL methods versus different dictionary sizes.

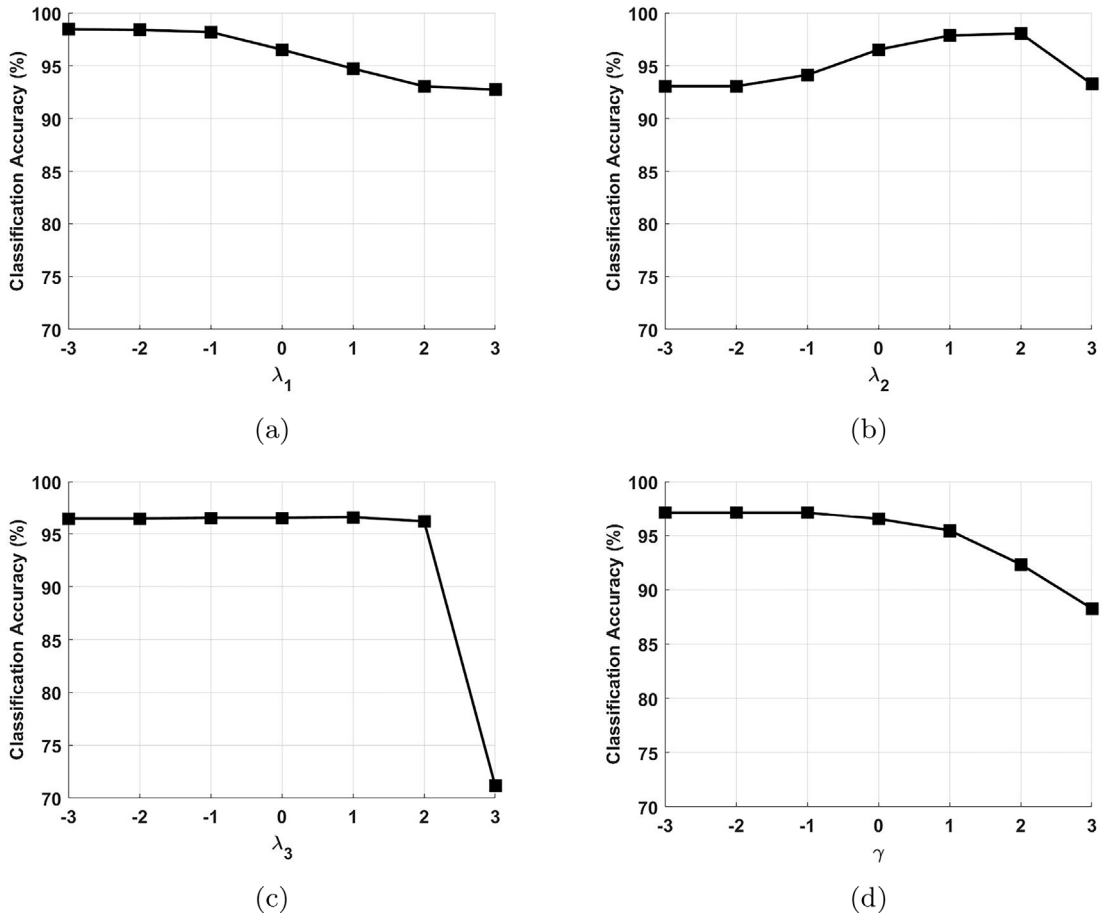


Fig. 10. Classification accuracy (%) of LpDPL versus variations of the parameters (a) λ_1, (b) λ_2, (c) λ_3, (d) γ.

According to DPL [17], the major computational burden in training phase of DPL is on updating the analysis dictionary P which requires an expensive matrix conversion via (10). In the proposed LpDPL algorithm, the same conditions hold, except an extra penalty term with the solution through (11). The added complexity due to this new term is $O\left(m^3 + Qmk + Qm^2\right)$ which is in the same order as computing other parameters, i.e., D and A, and hence negligible. In the testing phase, the classification using LpDPL and DPL has the same complexity $O(Qmk)$.

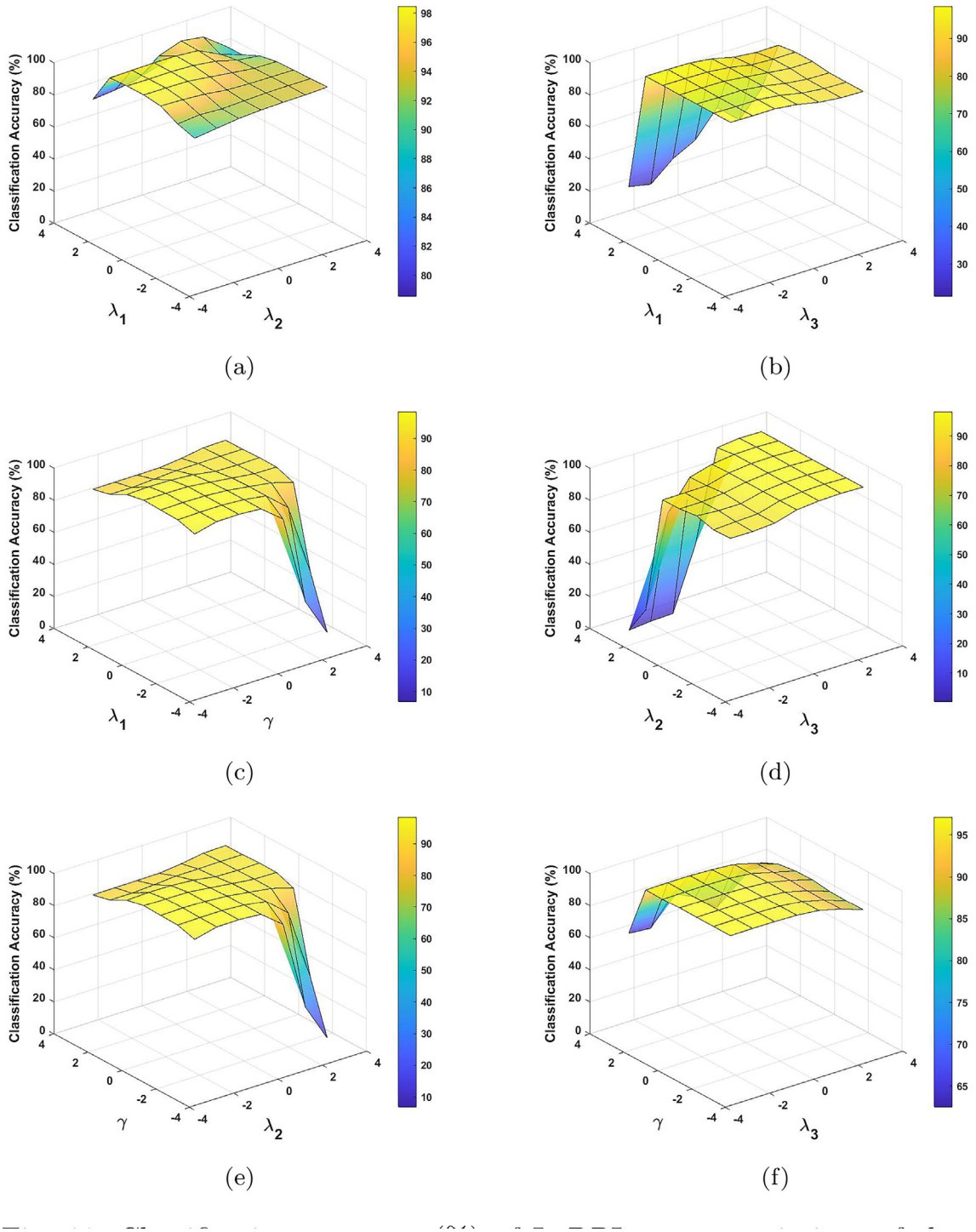


Fig. 11. Classification accuracy (%) of LpDPL versus variations of the parameters (a) λ_1, λ_2, (b) λ_1, λ_3, (c) λ_1, γ, (d) λ_2, λ_3, (e) γ, λ_2, and (f) γ, λ_3.

4.4. Parameter sensitivity study

In order to assess the robustness of the proposed method, we recorded the recognition performance of LpDPL over the variations of key parameters in Algorithm 1, i.e., $\lambda_1, \lambda_2, \lambda_3$ and γ. For this purpose, at each experiment, we fine-tuned the value of one parameter in the range $\left[10^{-3}, 10^{3}\right]$, while keeping other parameters fixed. Fig. 10 and Fig. 11 demonstrate the recog-

nition accuracy (%) of LpDPL versus variations of these parameters on simplified Chinese handwritten numbers. In particular, we have the following observations from these figures: When $\lambda_1 > 10^{-1}$, LpDPL slightly suffers a performance drop due to overweighting the discrimination factor. Interestingly, increasing contribution of class labels information (i.e. increasing λ_2) improves the performance. However, the performance drops for very large values, i.e., $\lambda_2 > 10^2$. LpDPL experiences a significant performance degradation when $\lambda_3 > 10^2$. We believe this is due to significant reduction of the contributions of discrimination power and class-label information when such a large λ_2 is selected. We also observe that when $\gamma > 10^0$, LpDPL's performance starts to drop (Fig. 11d). This is because γ is purposed to avoid zero division in (10). Therefore, a large γ leads to an inaccurate dictionary P. Overall, we observed that LpDPL is not sensitive to the parameters' variations within a broad range.

4.5. On real-time implementation

In this study, all the simulations related to dictionary learning were conducted and implemented in MATLAB (R2018a) environment on a machine equipped with core i7 processor with speed 2.20 GHz, and 8 GB of RAM. However, the deep learning experiments were implemented on a different machine equipped with Ubuntu 18.04 and MATLAB (R2019b) software, equipped with NVIDIA GeForce RTX 2080 Ti.

Dictionary learning step, which is regraded as an offline process, is solely implemented during the training phase. However, the feature extraction and classification steps are performed during both training (off-line) and the testing (in real-time) phases. The average computation times of performing these steps were very small, i.e., 0.64ms (for feature extraction) and 0.24ms (for classification) per image, without relying on costly GPUs. This allows the proposed method to run using lightweight and low-cost embedded hardware, such as a Raspberry PI, whereby this is not practical for deep learning-based methods.

5. Conclusions

In this paper, a novel dictionary learning technique, termed labeled projective dictionary pair learning, was proposed. The core advantage of using synthesis-analysis dictionary pair is to omit intractable ℓ_0/ℓ_1-norm calculation for sparse representation. More importantly, by utilising HOG features and adding available class labels as penalty term into the dictionary learning hierarchy, a robust pattern recognition model was achieved. We tested the proposed system with two Chinese handwritten numbers databases in addition to Arabic and English handwritten databases. The numerical results and comparison analyses with state-of-the-art methods verified robust classification performance of the proposed method. It is noteworthy to mention that no major differences in the model are required when different languages are used. When a new language is to be trained with LpDPL, dictionary size (n) and number of dictionaries (Q) should be adjusted depending on the images size and number of classes, respectively. The class labels (H) should also be selected according to the associated training set. Furthermore, as Figs. 10 and 11 suggest, the algorithm performs robustly against changes of various parameters. Hence, no significant variations are required with different languages. Unlike deep neural network models, our proposed technique runs locally on general-purpose computers without need for cloud servers or GPU devices; two standard resources which are essential to run deep learning models. Well-framed deep models used in study, i.e., SqueezeNet, GoogLeNet, and MobileNetV2, comprises of 1.24, 3.5, and 7 million tuned parameters, respectively, while the proposed method only requires 8 parameters to be fine-tuned.

There are three major research streams that we aim to pursue in the future: 1) combining deep learning and dictionary learning (particularly with DPL due to using a pair of synthesis-analysis dictionaries) for the purpose of handwritten numbers recognition, 2) exploring and extending the applicability of the proposed approach for a generic handwritten character recognition problem, and 3) optimising the implementation of the proposed method towards a real-time recognition system convertible to a mobile app.

Declarations

Funding Not Applicable.
Availability of data and material Not Applicable.
Code availability Not Applicable.
Ethics approval Not Applicable.
Consent to participate Not Applicable.
Consent for publication Not Applicable.

CRediT authorship contribution statement

Rasoul Ameri: Software, Methodology, Validation, Writing - original draft. **Ali Alameer:** Data curation, Formal analysis, Writing - review & editing. **Saideh Ferdowsi:** Conceptualization, Visualization, Writing - review & editing. **Kianoush Nazar-**

pour: Supervision, Formal analysis, Writing - review & editing. **Vahid Abolghasemi:** Conceptualization, Supervision, Project administration, Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Feng Wang, Huiqing Zhu, Wei Li, Kangshun Li, A hybrid convolution network for serial number recognition on banknotes, Inf. Sci. 512 (2020) 952–963.
[2] Ali Alameer, Ghazal Ghazaei, Patrick Degenaar, Kianoush Nazarpour, An elastic net-regularized hmax model of visual processing, in: 2nd IET International Conference on Intelligent Signal Processing 2015 (ISP), 2015, pp. 1–4.
[3] Ali Alameer, Ghazal Ghazaei, Patrick Degenaar, Jonathon A. Chambers, Kianoush Nazarpour, Object recognition with an elastic net-regularized hierarchical max model of the visual cortex, IEEE Signal Process. Lett. 23 (8) (2016) 1062–1066.
[4] Amirreza Fateh, Mansoor Fateh, Vahid Abolghasemi, Multilingual handwritten numeral recognition using a robust deep network joint with transfer learning, Inf. Sci. 581 (2021) 479–494.
[5] Miao Kang, Dominic Palmer-Brown, A modal learning adaptive function neural network applied to handwritten digit recognition. Inform. Sci., 178(20), 3802–3812, 2008. Special Issue on Industrial Applications of Neural Networks.
[6] Soufiane Hamida, Bouchaib Cherradi, Hassan Ouajji, Handwritten arabic words recognition system based on hog and gabor filter descriptors, in: 2020 1st International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET), 2020, pp. 1–4.
[7] Dinesh Elayaperumal, Young Hoon Joo, Robust visual object tracking using context-based spatial variation via multi-feature fusion, Inf. Sci. 577 (2021) 467–482.
[8] Amitava Choudhury, Hukam Singh Rana, Tanmay Bhowmik, Handwritten bengali numeral recognition using hog based feature extraction algorithm, in: 2018 5th International Conference on Signal Processing and Integrated Networks (SPIN), IEEE, 2018, pp. 687–690.
[9] Saleh Aly, Sultan Almotairi, Deep convolutional self-organizing map network for robust handwritten digit recognition, IEEE Access 8 (2020) 107035–107045.
[10] Shaharat Tajrean and Mohammad Abu Yousuf. Handwritten bengali number detection using region proposal network. In 2019 International Conference on Bangla Speech and Language Processing (ICBSLP), pages 1–6, 2019.
[11] Wenchao Wang, Jianshu Zhang, Jun Du, Zi-Rui Wang, and Yixing Zhu. Denseran for offline handwritten chinese character recognition. In 2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR), pages 104–109, 2018.
[12] Zhiyuan Li, Min Jin, Wu. Qi, Lu. Huaxiang, Deep template matching for offline handwritten chinese character recognition, J. Eng. 2020 (4) (2020) 120–124.
[13] Selen Ayas, Murat Ekinci, Single image super resolution using dictionary learning and sparse coding with multi-scale and multi-directional gabor feature representation, Inf. Sci. 512 (2020) 1264–1278.
[14] Bo Liu, Xiaodong Chen, Yanshan Xiao, Weibin Li, Laiwang Liu, Changdong Liu, An efficient dictionary-based multi-view learning method, Inf. Sci. 576 (2021) 157–172.
[15] Guoqing Zhang, Junchuan Yang, Yuhui Zheng, Zhiyuan Luo, Jinglin Zhang, Optimal discriminative feature and dictionary learning for image set classification, Inf. Sci. 547 (2021) 498–513.
[16] John Wright, Allen Y Yang, S. Arvind Ganesh, Shankar Sastry, Yi Ma, Robust face recognition via sparse representation, IEEE Trans. Pattern Anal. Mach. Intell. 31 (2) (2008) 210–227.
[17] Shuhang Gu, Lei Zhang, Wangmeng Zuo, and Xiangchu Feng. Projective dictionary pair learning for pattern classification. In Advances in neural information processing systems, pages 793–801, 2014.
[18] G. Madhuri and Atul Negi. Discriminative dictionary learning based on statistical methods. CoRR, abs/2111.09027, 2021.
[19] Yu.hua. Li, Chun Qi, Face recognition using hog feature and group sparse coding, in: 2013 IEEE International Conference on Image Processing, 2013, pp. 3350–3353.
[20] Rasool Ameri, Ali Pouyan, Vahid Abolghasemi, Projective dictionary pair learning for eeg signal classification in brain computer interface applications, Neurocomputing 218 (2016) 382–389.
[21] Xu. Yong, Bob Zhang, Zuofeng Zhong, Multiple representations and sparse representation for image classification, Pattern Recogn. Lett. 68 (2015) 9–14.
[22] Xu. Yong, David Zhang, Jian Yang, Jing-Yu Yang, A two-phase test sample sparse representation method for use with face recognition, IEEE Trans. Circuits Syst. Video Technol. 21 (9) (2011) 1255–1262.
[23] Julien Mairal, Francis Bach, Jean Ponce, Guillermo Sapiro, Andrew Zisserman, Supervised dictionary learning, in: Proceedings of the 21st International Conference on Neural Information Processing Systems, Curran Associates Inc., Red Hook, NY, USA, 2008.
[24] Brian Fulkerson, Andrea Vedaldi, Stefano Soatto, Localizing objects with smart dictionaries, in: European Conference on Computer Vision, Springer, 2008, pp. 179–192.
[25] Zhuolin Jiang, Zhe Lin, and Larry S Davis. Learning a discriminative dictionary for sparse coding via label consistent k-svd. In CVPR 2011, pages 1697–1704. IEEE, 2011.
[26] Zhuolin Jiang, Zhe Lin, Larry S. Davis, Label consistent k-svd: Learning a discriminative dictionary for recognition, IEEE Trans. Pattern Anal. Mach. Intell. 35 (11) (2013) 2651–2664.
[27] Zhao Zhang, Weiming Jiang, Jie Qin, Li Zhang, Fanzhang Li, Min Zhang, Shuicheng Yan, Jointly learning structured analysis discriminative dictionary and analysis multiclass classifier, IEEE Trans. Neural Networks Learn. Syst. 29 (8) (2018) 3798–3814.
[28] Yulin Sun, Zhao Zhang, Weiming Jiang, Zheng Zhang, Li Zhang, Shuicheng Yan, Meng Wang, Discriminative local sparse representation by robust adaptive dictionary pair learning, IEEE Trans. Neural Networks Learn. Syst. 31 (10) (2020) 4303–4317.
[29] Yuxi Wang, Du. Haishun, Yonghao Zhang, Yanyu Zhang, Efficient and robust discriminant dictionary pair learning for pattern classification, Digital Signal Processing 118 (2021) 103227.
[30] Snigdha Tariyal, Angshul Majumdar, Richa Singh, Mayank Vatsa, Deep dictionary learning, IEEE Access 4 (2016) 10096–10109.
[31] Hao Tang, Heng Wei, Wei Xiao, Wei Wang, Dan Xu, Yan Yan, and Nicu Sebe. Deep micro-dictionary learning and coding network. In 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), pages 386–395, 2019.
[32] Bahareh Tolooshams, Andrew H. Song, Simona Temereanca, Demba Ba, Convolutional dictionary learning based auto-encoders for natural exponential-family distributions, in: Proc. International Conference on Machine Learning (ICML), 2020.
[33] Zhao Zhang, Yulin Sun, Zheng Zhang, Yang Wang, Lin Wu, and Meng Wang. Mdpl-net: Multi-layer dictionary learning network with added skip dense connections. In 2020 IEEE International Conference on Data Mining (ICDM), pages 811–820, 2020.
[34] Zhao Zhang, Yulin Sun, Yang Wang, Zheng Zhang, Haijun Zhang, Guangcan Liu, Meng Wang, Twin-incoherent self-expressive locality-adaptive latent dictionary pair learning for classification, IEEE Trans. Neural Networks Learn. Syst. 32 (3) (2021) 947–961.
[35] Zhengming Li, Zheng Zhang, Jie Qin, Zhao Zhang, Ling Shao, Discriminative fisher embedding dictionary learning algorithm for object recognition, IEEE Trans. Neural Networks Learn. Syst. 31 (3) (2020) 786–800.

[36] Vahid Abolghasemi, Mingyang Chen, Ali Alameer, Saideh Ferdowsi, Jonathon Chambers, Kianoush Nazarpour, Incoherent dictionary pair learning: Application to a novel open-source database of chinese numbers, IEEE Signal Process. Lett. 25 (4) (2018) 472–476.
[37] Nobuyuki Otsu, A threshold selection method from gray-level histograms, IEEE Trans. Syst., Man, Cybern. 9 (1) (1979) 62–66.
[38] William T. Freeman and Michal Roth. Orientation histograms for hand gesture recognition. 1995.
[39] Tom Goldstein, Brendan O'Donoghue, Simon Setzer, Richard Baraniuk, Fast alternating direction optimization methods, SIAM J. Imaging Sci. 7 (3) (2014) 1588–1623.
[40] Ignacio Ramirez, Pablo Sprechmann, Guillermo Sapiro, Classification and clustering via dictionary learning with structured incoherence and shared features, in: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE, 2010, pp. 3501–3508.
[41] Zhizhao Feng, Meng Yang, Lei Zhang, Yan Liu, David Zhang, Joint discriminative dimensionality reduction and dictionary learning for face recognition, Pattern Recogn. 46 (8) (2013) 2134–2143.
[42] Lei Zhang, Meng Yang, Xiangchu Feng, Sparse representation or collaborative representation: Which helps face recognition?, in: 2011 International conference on computer vision, IEEE, 2011, pp 471–478.
[43] Weihong Deng, Hu. Jiani, Jun Guo, Extended src: Undersampled face recognition via intraclass variant dictionary, IEEE Trans. Pattern Anal. Mach. Intell. 34 (9) (2012) 1864–1870.
[44] Weihong Deng, Hu. Jiani, Jun Guo, In defense of sparsity based face recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2013, pp. 399–406.
[45] Ahmed El-Sawy, Hazem EL-Bakry, and Mohamed Loey. CNN for Handwritten Arabic Digits Recognition Based on LeNet-5. In Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2016, pages 566–575. Springer International Publishing, 2017.
[46] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1–9, 2015.
[47] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, Liang-Chieh Chen, Mobilenetv 2: Inverted residuals and linear bottlenecks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 4510–4520.
[48] Forrest N. Iandola, Matthew W. Moskewicz, Khalid Ashraf, Song Han, William J. Dally, Kurt Keutzer, Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <1MB model size, CoRR abs/1602.07360 (2016).
[49] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, Li Fei-Fei, Imagenet: A large-scale hierarchical image database, in: 2009 IEEE conference on computer vision and pattern recognition, 2009, pp. 248–255.
[50] Yue Yin, Wei Zhang, Sheng Hong, Jie Yang, Jian Xiong, Guan Gui, Deep learning-aided ocr techniques for chinese uppercase characters in the application of internet of things, IEEE Access 7 (2019) 47043–47049.