

Hierarchical Sparse Autoencoder Using Linear Regression-based Features in Clustering for Handwritten Digit Recognition

Hai T. Phan An T. Duong Nam Do-Hoang Le Son T. Tran
Advanced Program in Computer University of Science John von Neumann Institute, University of Science
Science, University Of Science Ho Chi Minh City, Vietnam Vietnam National University Ho Chi Minh City, Vietnam
Ho Chi Minh City, Vietnam 0912019@student.hcmus.edu.vn Ho Chi Minh City, Vietnam ttson@fit.hcmus.edu.vn
pthai@apcs.vn nam.le.ict@jvn.edu.vn

Abstract—Recently, handwritten digit recognition using higher level features has got more promising results than conventional ones using intensity values, where the higher level features are considered as features of simple strokes in images. Although the state-of-the-art performance is very impressive, there is still room to improve better in both accuracy and computation complexity. In this paper, we propose a new feature based on linear regression to extract geometrical characteristics of handwritten digits. The linear regression-based features are utilized to cluster set of digit image in preprocessing. After that, each set of clustered digit images is inputted a hierarchical sparse autoencoder to extract higher level features automatically. Our method result achieves error rates lower than that of conventional method in the most of cases. The experiment shows that the efficiency of data clustering can get promising results.

Keywords - Higher level features, sparse autoencoder, hierarchical sparse autoencoder, handwritten digit recognition, linear regression-based features.

I. INTRODUCTION

Recognition algorithms provide some knowledge for computer perception and feature extraction plays an important role of recognition system. Some achievements of feature extraction such as SIFT [11], or HOG [2] and MFCCs [6] are utilized in most applications of computer vision and pattern recognition popularly. For instances, SIFT and HOG are utilized for image data and MFCCs is the features extraction of sound data. In pattern recognition, one of traditional researches is handwritten digit recognition. Many approaches such as K-Nearest neighbors, neural networks, and support vector machine are applied to classify representations from image data. Nevertheless, these approaches cannot obtain expected error rates without preprocessing [8]. Research [10] presents results of handwritten digit recognition by using state-of-the-art techniques on some popular databases of CENPARMI, CEDAR and MNIST. The meaning is the behaviors of performance in features extraction and classification techniques on those well-known databases. These approaches use representation of raw data for recognition. Recently, some researches [5], [17] focused on training deep, multi-layered networks proposed the methods using higher representation from raw data images and they have been recorded a significant improvement. Higher level features of data images are representations of input image by using simple strokes. Here higher representations

can be learned by a system where the knowledge of experts in some specific parts is not necessary [14], [3]. This is the reason why they are considered to higher-level features. Moreover, based on these properties, higher-level features can be applied for any kind of data including images [9], [14], audio [3], and texts [15], [16]. There are some proposed methods [13], [14], [15] using higher level features learning. However, these approaches do not take promising results for specific characteristics of raw data. In research of Olshausen [13], he proposed sparse coding algorithm to show that the ability of higher representations level learning from input signals is the simple-cells receptive fields in primary visual cortex of mammalian brain. Moreover, Honglak Lee presented method based on iteratively solving two convex optimization problems with high-dimensional images [14], [9] to get an efficient solving algorithm, or Jame Martens [12] proposed the improvement of sparse coding using weights optimizing. In this paper, we propose an efficient feature applied for clustering in preprocessing and utilize those results to cluster training images. Then the clustered sets of training image are inputted in hierarchical sparse autoencoder to solve handwritten digit recognition. The structure of paper is presented as follows.

Section II presents our proposed methods. Hierarchical sparse autoencoder is described in Section II-A. In Section II-B, we present a new feature based on linear regression-based features method (LR-based features). Those features will be automatically clustered into characteristic sets by using hierarchical sparse autoencoder. Section III describes experiments and results of our methods to show the efficiency of linear regression-based features method by making comparisons between conventional methods [5], K-Nearest Neighbor (KNN) methods [8] and some methods mentioned in [17]. Section IV is the part of discussions.

II. OUR PROPOSED METHODS

A. Hierarchical Sparse Autoencoder

Instead of training raw data like previous works, we approach hierarchically to take advantage of specific characteristics of data. Our method is inspired by Alexanders work in neuroscience. It shows that human brains have many different cortex areas to process the perceived outside world information. There are some specific cortex areas being active

to process the perceived signals. The interesting thing is that the brains active the same cortex areas when the signals are highly correlated with each other [7]. Furthermore, our observation also shows that handwritten digits are formed from primitive strokes such as straight and curve strokes. And some of handwritten digits have the correlated characteristics. In particular, the percentage of straight strokes to compose 1, 4 and 7 is more dominating than the percentage of curve strokes. In contrast, curve strokes are the key ingredient to compose 0, 2, 6 and 8 rather than straight strokes. Our approach has three main steps:

- Specific characteristics clustering.
- Higher representations learning.
- Double supervised learning.

The first step is specific characteristics clustering. A characteristic can be described as a d -dimensional vector. Let $\xi_1, \xi_2, \dots, \xi_k$ denote k correlated characteristic sets. Particularly, we can have ξ_i containing characteristics which are highly correlated with each other. In other word, the average distance of a d -dimensional characteristic vector in ξ_i to other characteristic vectors in ξ_i is smaller than the average distance of this vector to characteristic vectors in ξ_j ($i \neq j$). Thus, characteristics in ξ_i can be seen as specific characteristics for learning higher representations better. More specific in handwritten digits data, ξ_1 can be characteristic set containing characteristics which reveal that straight strokes are the key ingredient to compose digits. And it can be curve strokes are the key ingredient to compose digits for ξ_2 . As a result, handwritten digit samples, which have characteristics belonging to the same correlated characteristic set ξ_i , will also belong to the same set described as follow

$$E_i = \{x \in \mathbb{R}^n \mid \min \delta(\varphi(x), \xi_i)\} \quad (1)$$

where E_i is an entity set containing samples have characteristics which belong to ξ_i . x is a sample of our dataset. $\varphi(x) : \mathbb{R}^n \rightarrow \mathbb{R}^d$ is a mapping function which maps a sample to a characteristic (this function is described in detail in Section II-B). And $\delta(\varphi(x), \xi_i)$ can be computed as follow

$$\delta(\varphi(x), \xi_i) = \frac{1}{|\xi_i|} \sum_j^{|\xi_i|} \text{distance}(\varphi(x), c_j) \quad (2)$$

where c_j is a d -dimensional characteristic vector belonging to ξ_i . The distance $(\varphi(x), c_j)$ is a distance function such as Manhattan, Euclidean and Correlation. In our approach, we use K-means algorithm for this clustering step.

The second step is higher representations learning on entity sets. Suppose at the first step the dataset has been already divided into k entity sets corresponding to k correlated characteristic sets. At this step, we train separately each entity set with one sparse autoencoder to learn weight matrices of that sparse autoencoder. These weight matrices are used to compute higher representations of raw data. Higher representations of training data are used as features to train classifiers at supervised learning phase.

The third step is double supervised learning to recognize class labels. At the first level of supervised learning, higher representations in each cluster from the second step are inputted into one softmax regression model. In our work, we

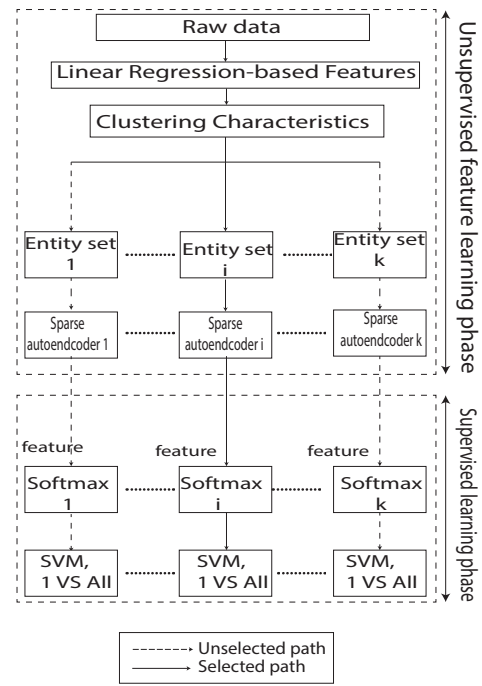


Fig. 1. Hierarchical Sparse Autoencoder

use softmax regression as classifiers after sparse autoencoder because softmax regression and sparse autoencoder can be combined to become a deep learning model. The output of softmax regression is probabilistic vectors instead of class labels of input data. A probabilistic vector shows the probability for an input data to belong to each class. These vectors are then concatenated with the corresponding higher representation vectors of input data to form the features for the next level of supervised learning. The purpose of using probabilistic vectors of softmax regression is to orient for prediction or augment information for the next learning level. At the second learning level, the concatenated features are trained with SVM using 1-vs-all method. Here we apply the 1-vs-all approach instead of one-vs-one method because it helps to reduce the number of SVM classifiers and to avoid the ambiguity of outputs by using the voting of SVM results. Therefore, it will need $k \times l$ SVM models at maximum for this training step (l is the number of classes). The complete process can be interpreted as the visualization in Fig. 1 where the raw data means training image, linear regression-based feature is a method to extract regression lines in skeleton image, the entity set is the clustered sample data based on the results of K-mean algorithm applied for normalized histogram of new features.

Those 3 steps described above focus on training phase of this approach. The testing phase here follows the flow of the model; however, it is quite different from the training phase. Concretely, a new testing sample can be assigned into 2 clusters at maximum. Of course the sample will belong to the cluster which has the minimum distance from the cluster centroid to the sample. In addition, the sample can also belong to one more cluster if the distance from it to the centroid of that cluster is not greater than 1.5 times the

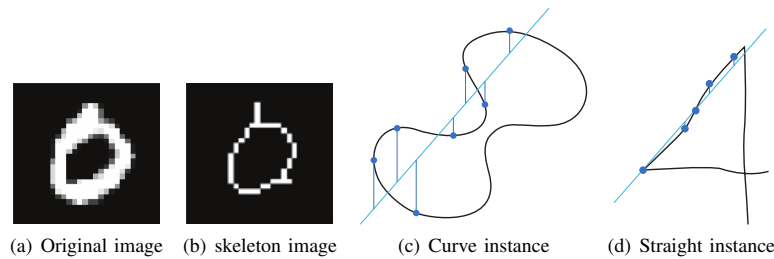


Fig. 2. Results of skeleton, and regression.

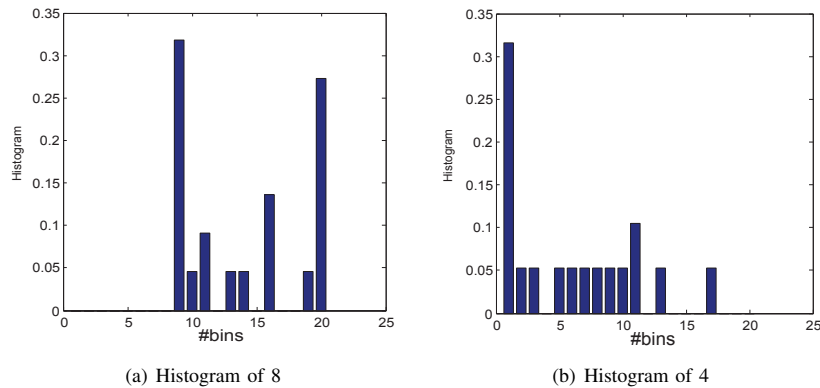


Fig. 3. Histogram results.

minimum distance to all clusters. After the new sample is tested with SVM models, the output of a SVM model is also a probabilistic vector. The number of probabilistic vectors is equal to the number of clusters that the sample belongs to. Finally, the maximum probability of all probabilistic vectors decides the class label of the testing sample.

B. Linear Regression-based Features

We present a new feature based on linear regression. From observations of digits (from 0 to 9) in some popular fonts as Times New Romans or Arial, we can see that there are three main kinds of characteristic. They are straight-line-segment digits of $\{1, 4, 7\}$, curve digits of $\{0, 3, 6, 8, 9\}$ and both straight and curve digits of $\{2, 5\}$. However, it is really difficult to classify the ten numbers of hand written digits based on these three kinds of characteristic. Our approach is based on statistical learning, where feature extraction must be considered carefully. Here we apply linear regression to approximate sampling points in skeleton image of sample data. As we know that linear regression constructs the best-fitting straight line for skeleton image of sample data by fitting a linear model. In one skeleton image, we have many regression lines and a set of mean square error (MSE) corresponding to them. It means that we can generally define characteristic sets of straight line segment, curve, and both straight-line segment and curve in the structure of ten digits by applying a cluster model to all histograms of MSE in one skeleton image.

Before applying linear regression, we take the skeleton of raw data. Fig. 2 b) shows the result of skeleton image. With a skeleton image, original characteristic of digit image is

reserved. In this case, for the leftmost corner pixel of skeleton image, we find 8-neighbor connected path takes priority with respect to counterclockwise. For each pixel in connected path, we consider n consecutive entries which are observed data for each regression line. For instances, if connected path have m pixels then there are $m - n$ regression lines. Here we assume that the value of n is always less than that of m .

To simplify notation for linear regression in this case, we denote the equivalent optimization problem

$$\underset{\theta}{\text{minimize}} \text{MSE} = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (3)$$

where X, Y are two vectors of coordinate points in the skeleton image, θ is a vector of coefficients, and n is the number of entries of observed data. After using least mean square method, we can calculate the values of parameter θ . The interpolation value of line regression is described by the following

$$\hat{Y}_i = \theta_0 + \theta_1 X_i \quad (4)$$

In practice, we set up ($n = \frac{\text{size}(X)}{2}$). We can see that mean square error (MSE) of regression lines in digits written by curve strokes is always greater than that of one digit written by straight strokes. This can help make distinct characteristic of digits. From that, for each raw image data, we obtain set of mean square errors represents for characteristics of digits. Here if m is the number of pixels in skeleton image, we will have $m - n$ values of MSE corresponding to $m - n$ regression lines in skeleton image. After linear regression calculation, we map the set of MSE in each skeleton image into histogram space