

High-Quality Wavelets Features Extraction for Handwritten Arabic Numerals Recognition

M. Suhail Akhtar[#], Hammad A. Qureshi^{*}, Mafawez Al-Harbi⁺, Hani Al-Quhayz⁺

[#] SEecs, National University of Sciences & Technology, Islamabad, Pakistan.
E-mail: 11msitmakhtar@seecs.edu.pk

^{*} College of Engineering & Technology, University of Derby, Derby, United Kingdom
E-mail: hqureshi@derby.ac.uk

⁺ College of Science-Zulfi, Majmaah University, Saudi Arabia
E-mail: ^{#4}h.quhayz@majmaah.edu.ksa

Abstract— Arabic handwritten digit recognition is the science of recognition and classification of handwritten Arabic digits. It has been a subject of research for many years with rich literature available on the subject. Handwritten digits written by different people are not of the same size, thickness, style, position or orientation. Hence, many different challenges have to overcome for resolving the problem of handwritten digit recognition. The variation in the digits is due to the writing styles of different people which can differ significantly. Automatic handwritten digit recognition has wide application such as automatic processing of bank cheques, postal addresses, and tax forms. A typical handwritten digit recognition application consists of three main stages namely features extraction, features selection, and classification. One of the most important problems is feature extraction. In this paper, a novel feature extraction approach for off-line handwritten digit recognition is presented. Wavelets-based analysis of image data is carried out for feature extraction, and then classification is performed using various classifiers. To further reduce the size of training data-set, high entropy subbands are selected. To increase the recognition rate, individual subbands providing high classification accuracies are selected from the over-complete tree. The features extracted are also normalized to standardize the range of independent variables before providing them to the classifier. Classification is carried out using k-NN and SVMs. The results show that the quality of extracted features is high as almost equivalently high classification accuracies are acquired for both classifiers, i.e. k-NNs and SVMs.

Keywords— wavelets analysis; wavelet packets analysis; handwritten digits recognition; k-NNs; SVMs.

I. INTRODUCTION

Handwritten numerals recognition has been a subject of research and study for many years now. Recognition of numerals of different languages namely English, Arabic, etc. has been studied. Handwritten digit recognition is the ability of a computer to interpret handwritten input from various sources such as paper documents, photographs, touch-screens and various other devices [1]. It is the method of recognizing and classifying handwritten digits 0 (0) to 9 (9) without human intervention. In general, handwritten digit recognition is classified into two categories off-line and on-line recognition [2]. However, we are concerned only with the off-line recognition of handwritten Arabic numerals in this work. Since the recent advancement in digitization and the rise of Big Data, handwritten digit recognition has again become an active area of research. The massive amount of digital data available today makes the automatic processing

of handwritten text even more critical. There is a great interest in this area due to many potential applications, mainly where a large number of documents must be analysed, such as postal mail sorting, banks cheque processing, and handwritten forms and scanned data analysis, etc. [3]. In addition to above, it is also applicable to license plate recognition, and security systems [4]. These applications require high recognition accuracy and speeds as well.

There are many software applications available today for text recognition including Adobe Acrobat, Omnipage, ABBYY FineReader, Readiris, etc. There is a considerable amount of opensource software as well which includes SimpleOCR, TopOCR, FreeOCR, etc. However, handwritten text recognition is still dependent upon the quality of input. Recognition works best for high quality printed text. The closer handwritten text is to the printed text the better recognition can be achieved [5]. Like much other real-world application, high accuracies are critical for

handwritten digits as well. Recently many new classifiers and feature extraction methods have been proposed and tested for handwritten numerals recognition. The techniques proposed and developed provide high classification accuracies [6]–[10]. However, there is still room for new techniques that are efficient and provide high-quality features. Experiments have shown that by using highly relevant and quality features, the accuracy and efficiency of many classifiers can be improved considerably [11].

Recent literature on the subject of digit recognition proposes deep learning, reinforcement learning and graph-based learning approaches [12]–[16]. Using such techniques very high accuracies are obtained, but these techniques suffer from the issue that it is not possible to explain the reasoning behind such high accuracies or which extracted features are the best for a given problem.

There are three major operations involved in digit recognition from a computational point of view, i.e. features extraction, features selection and classification [6]. The performance of the digit recognition technique dramatically depends upon the first stage, i.e. features extraction. In most of the classification problems, the focus is on improving the classification rates by using better classifiers. However, recently some effort has been made to improve the features extraction phase [3], [17]. For efficient feature extraction, we propose an algorithm based upon 1D wavelet packet transform. A total of eight different types of features, i.e. standard deviation, mean, mean absolute deviation, kurtosis, skewness, energy, entropy, and sum are extracted. Then the best features are selected by accuracy. Data normalization is carried out using z-score normalization. The objective of the normalization is to standardize the ranges of independent variables or features of data. The primary purpose of this normalization is to ensure the uniform contribution of all feature variables in the classification process. For classification, K-NN and SVM classifiers (with tuned parameters) are used.

Many studies on Arabic handwritten digit recognition have been carried out. Srinivas and Premchand used backpropagation to train the Elman Neural Network using an adaptive learning rate and performance parameters like accuracy and processing times were compared for different techniques [18]. Borji et al. compared standard classifiers such as the k-Nearest Neighbour (k-NN), Artificial Neural Networks and Support Vector Machines and the results were compared. A two-stage classifier was also designed for getting higher recognition rates. The recognition rate of SVM was found to be comparatively better than the k-NN (>10%) [6]. Alvarez et al. used the multi-layer neural network trained using the backpropagation algorithm. Kirsch masks were used for features extraction, and the recognition accuracy was 96.2% [6]. Goltsev and Gritsenko used two neural network classifiers (i.e., a modified 3-layer perceptron LiRA and a modular assembly neural network). A novel feature extraction method is used which analyses the connection of weights formed in the initial learning process. LiRA-features were acquired for image recognition, and the recognition accuracy of the modular assembly neural network was found to be better [19]. Lauer et al. introduced a trainable feature extractor based on the LeNet5 convolutional neural network architecture for classification

tasks. They used SVMs to increase the generalization capability of LeNet5. The system outperforms both SVM and LeNet5 which provide comparable performance to each other [20].

Singh and Khehra studied various features, i.e. coordinate bounding box, area, centroid, eccentricity, and equi-diameter while the backpropagation neural network classifier is used to recognize the digits and Gaussian filtering is used for quality improvement with an overall accuracy of 96.6% [7]. Romero et al. presented a novel pre-processing technique for digit recognition based on the directional continuous wavelet transform. The resulting transformation has four parameters (scale, angle, and position (x,y)) while a neural network is used for classification [21]. Enachescu et al. use a particular type of multi-layer neural network namely Convolutional Neural Network trained with backpropagation. Increased efficiency is acquired by pre-processing of the image, i.e. 96.76% [8]. Liu et al. proposed an improved normalization function and feature extraction strategy and performance comparison with other existing techniques is also presented [22]. Bortolozzi et al. compared various techniques for handwritten digit recognition like template matching, statistical and structural techniques and neural networks [23].

Trier et al. compared various feature extraction methods for character representation such as solid binary characters, character contours, and skeletons or grey level sub-images. They employ various techniques such as template matching, deformable templates, graph descriptions, zoning, projection histograms, unitary image transforms, contour profiles, geometric moment invariants, Zernike moments and Fourier descriptors. Orientation, size, and type of characters were found to play a significant role [24]. Liu et al. analyze various feature sets such as chain code feature, gradient feature, profile structure, and peripheral direction, etc. A recognition rate of 80% was acquired by combining eight classifiers with ten feature vectors on the test data set [9]. Kussul and Baidyk use the classifier Limited Receptive Area (LIRA) to classify MNIST database producing very high accuracies [14]. Kessab et al. presented a novel system of handwritten digit recognition using Multi-Layer Perceptron (MLP) and a novel method for features extraction, and a recognition rate of 80% is acquired [10].

Shi et al. [25] presented a handwritten digit recognition system using the gradient and curvature of the grey level character images. The experiments were conducted on different databases like IPTP CDROM1, NIST SD3, and SD7 with the recognition rates achieved between 98.25% and 99.49%. Indra et al. propose a similar curvature-based and distance matching technique for Bisindo alphabets [49]. Teow and Loe [26] recommend a handwritten digit recognition system based on a biological model. The features were extracted empirically by the model, which could linearly separate over a large training set (MNIST). They reported a high recognition rate of 99.41%. Decoste and Scholkopf [26], [27] suggested a handwritten digit recognition system where the prior knowledge regarding in variance of a classification task was included in the training procedure. The classifier used was Support Vector Machines (SVMs), and a low error rate of 0.56% was acquired on the MNIST dataset.

Simard et al. [28], expanded the training set of the MNIST database by adding new distorted data, and the convolutional neural network is used for classification purposes producing a high recognition rate of 99.60%. Ping et al. presented a new cascade ensemble classifier for handwritten digit recognition combined with discriminative features [29]. Yann LeCun et al. compared the performance of various handwritten classifier algorithms on handwritten digit databases taking in to account not only performance but memory requirements as well [30]. Jinhai et al. presented an approach combining statistical and structural information for the recognition of unconstrained handwritten digits. The experimental results show high performance using this approach in terms of speed and accuracy [31].

M.Hanmandlu et al. presented a novel approach aimed at handling the variability in the writing styles of isolated handwritten digits. A new feature extraction method, i.e. the sector based method was used for features extraction, and the backpropagation neural network was used for recognition [32]. Cruz et al. propose a novel system for features extraction and ensemble classifier is used for handwritten digit recognition. Six feature sets were extracted, and various combination of the schemes was tested using the neural network acquiring an overall accuracy of 99.68 % on MNIST database [33]. Sung-Bae Cho presents three neural network classifiers for handwritten digit recognition namely MLP, HMM/MLP hybrid classifier, and adaptive structure SOM. Experiments were conducted on the unconstrained handwritten digits database for the evaluation of the classifiers which produced accuracies of 97.35 %, 96.55% and 96.05 % respectively [34].

M. Karic et al. examined the concavity based structural features for the recognition of handwritten digits. For classification tasks, two support machines (SVMs), i.e. SVM with RBF kernel and SVM with polynomial kernel are used and are compared to k-NNs [3]. Bishnoi et al. propose a novel approach for off-line handwritten digit recognition classifying the digits into four regions, i.e. upper, lower, left and right. Images were identified by the curves in these four regions. This method was successfully used for various databases like NIST and MNIST [2]. Dan et al. propose a novel feature extraction method with backpropagation neural network for the recognition of digits. MNIST database was used, and PCA is used for features reduction to improve the performance of neural network [35].

Wu et al. acquire directional features while K-NN, Gaussian Mixture Models and SVMs are used for classification. An error rate of 1.19 % is achieved using 3-NN classifier [36]. Zhang et al. propose two types of wavelet features, i.e. kirsch edge enhancement based 2-D wavelets and 2-D complex wavelets. The hybrid feature sets are combined with the geometrical features for recognition [38]. We also use wavelet features in our analysis and produce comparable results. Hwang et al. used the RBF network for the recognition of handwritten digits obtaining high recognition rates [37].

The rest of the paper is organized as follows. In the next section, the proposed methodology is discussed in detail along with a brief discussion of various features extraction methods and classification techniques used. In Section IV, the experimental results are presented while Section V

contains the conclusion. The proposed solution to this problem provides high accuracies comparable to the other techniques reported in the literature. The proposed algorithm is developed and tested in Matlab, and the MNIST database is used in the analysis. The main contributions of the paper are summarized as follows:

- Finding 1: Acquisition of high accuracy subbands from the overcomplete tree for accurate handwritten numerals recognition.
- Finding 2: Determination of optimal parameters for SVM (RBF kernel) to achieve high accuracies.
- Finding 3: Algorithms based-upon selection of high entropy subbands from the over-complete tree for high classification accuracies are presented.

II. MATERIALS AND METHOD

In this paper, we present a novel wavelet analysis based technique for handwritten digits recognition. The purpose of this technique is to extract the most relevant features and acquire dimensionality reduction to increase the accuracy as well as the performance of the recognition system. The classification task is performed using the k-NN and SVM classifiers. A block diagram for the techniques included in this work is given in Figure 1.

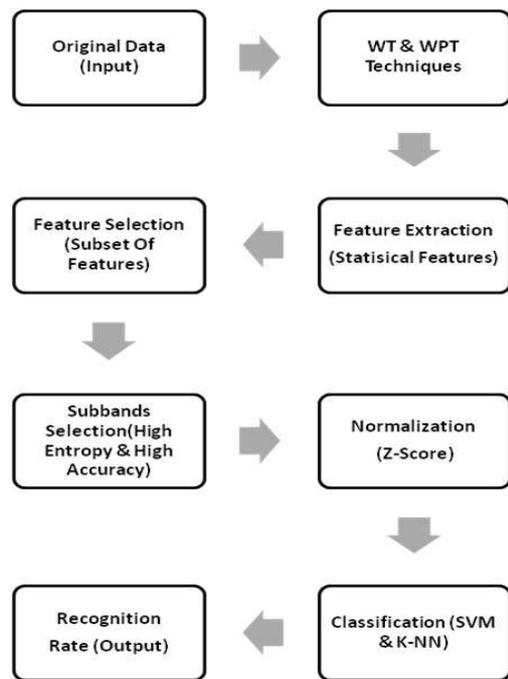


Fig. 1: Block diagram of various steps in the algorithm.

A brief description of the different steps in this work is given below.

A. Data Acquisition

The data used in the study is the famous MNIST database. As stated earlier, it is one of the most popular datasets for handwritten digits recognition. The dataset consists of 70,000 images containing the numerals 0 to 9 written by hand. Sixty thousand images are used for training while 10,000 are used for testing. However, in this work, we have reconfigured the dataset so that 48,000 images are used for

training while 22,000 are used for testing. This was done to comply with the requirements of competition at Kaggle.com. MNIST database is one of the most used datasets for handwritten digit recognition problem and acts as a benchmark for comparing the different techniques. Hence, we choose to use this dataset.

B. Wavelet Features Extraction

The next stage is the extraction of features using the wavelet transforms.

1) Wavelet Transform:

A wavelet is a small wave like a function that through convolution divides the data into different spatial-frequency resolutions. Two conditions are necessary for creating a wavelet. First it must be oscillatory, and secondly, its amplitude should be non-zero for only a short duration. Wavelet analysis is performed using the shifts and dilations of a prototype function $\psi(t)$ referred to as the mother wavelet. The function $\psi(t)$ also satisfies the condition,

$$\int_{-\infty}^{\infty} \psi(t) dt = 0 \quad (1)$$

To acquire the wavelet transform the mother wavelet is dilated by a factor of 'a' and translated by a scalar 'b' to obtain different spatial frequency wavelet subbands.

$$\Psi_{ab}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) \quad (2)$$

Wavelet transform allows studying of a signal both in time and frequency domain at the same time. It differs from the Fourier transform, as in Fourier transforms temporal/spatial information is hidden, while in the wavelet transform, frequency, as well as spatial information, is retained. Spatial-frequency analysis made possible using wavelet transforms multiresolution analysis of the input signal. This makes it possible to extract high-quality features from the data as the information in the signal or data is encoded into various subbands. Wavelet transform acts like a microscope analyzing a signal at various resolutions.

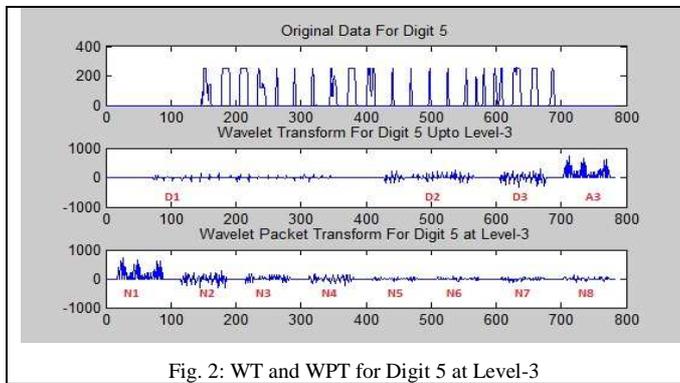


Fig. 2: WT and WPT for Digit 5 at Level-3

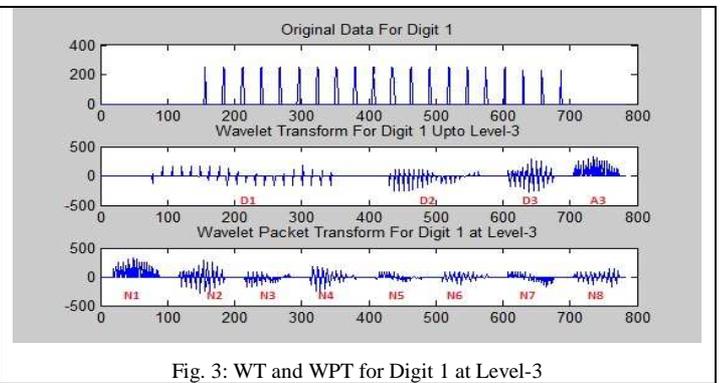


Fig. 3: WT and WPT for Digit 1 at Level-3

In wavelet transform, original data is decomposed into low-level approximation and high-level detail subbands. The approximation subband is further decomposed into approximation and detail subbands at each level. Figure 1 and Figure 2 show the original data, wavelet transform (all subbands) and wavelet packet transform (terminal subbands) for digit 5 and digit 1 respectively at level 3.

2) Wavelet Packet Transform

The wavelet packet method is a generalization of wavelet decomposition that offers richer data analysis. It differs from the discrete wavelet transform as in the wavelet packet transform, both the approximation and detail subbands are filtered using a scaling and wavelet function. In the orthogonal wavelet decomposition, the generic step decomposes the approximation coefficients into two parts.

As a result of the division, we get a vector of approximation coefficients and a vector of detail coefficients. Further decomposition of the approximation and detail subbands produces an overcomplete wavelet packet bases. Its terminal nodes are called full wavelet packet transform. A transform is 'over-complete' when it contains all the nodes at various levels. In case of full wavelet packet transform, an image is decomposed into its respective subbands, and each subband is further decomposed up to a certain predefined depth.

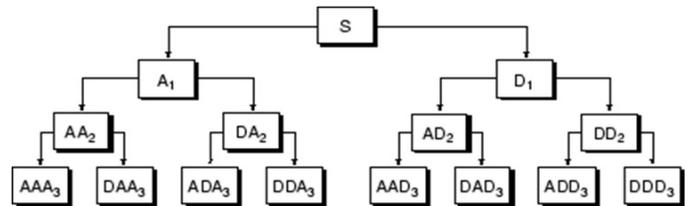


Fig. 4: Wavelet Packet Decomposition

In contrast to wavelet transform, in the wavelet packet transform, each detail coefficient vector is also decomposed into two parts similar to the approximation vector splitting as shown in Figure-3. Wavelet packets allow for more detailed analysis of the input signal as a wavelet packet decomposition tree and represent information of input signal in a more significant number of subbands and offer a rich library of wavelet bases. Hence, there is a potential for extracting higher quality and more representative features as will be seen in the results section.

C. Features Relevance

In this step, the feature sets that are used in training and recognition are extracted which play a major role in the recognition system. It is necessary for a good feature set to represent characteristics of a class that can distinguish it well

from other classes while retaining features that minimize differences within the class. Preferably, it is advisable that the dimensions of a feature vector should be kept as small as possible [3].

The primary purpose of the feature extraction process is to acquire the most relevant information. Naturally, it aims to extract a set of features, which maximize the recognition rate with a small subset of features. Feature extraction and selection leads to dimensionality reduction and results in better classification accuracies as it uses the most relevant information [17]. There are three main objectives of features selection [38].

- 1) Improve the clustering or classification performance.
- 2) Improve speed and cost-effectiveness of the solution.
- 3) Provide a better understanding of the data.

Various types of features exist for the recognition of handwritten digits which vary from structural to statistical features. The structural features are based on the geometric and topological properties, while statistical features are based on the statistical properties of the digit. A property of good features sets is that they maximize the differences between classes under study [3]. A wavelet transform is a good tool for the extraction of the useful feature as seen in the work of Chen et al. who uses a multi-wavelet orthonormal shell expansion on the character contours to obtain several levels of resolution and their mean values [27].

3) Statistical Features Extraction

Statistical features are extracted from each subband after applying wavelet transform and wavelet packet transform. A few most common features found in the literature are Standard Deviation, Energy, Entropy, Sum, Mean, Mean Absolute Deviation (MAD), Skewness and Kurtosis. Most of these features are statistical features as they are derived from the statistical distribution of the points. The discussion of the various features extracted is given below. The sum is defined as the sum of all the values X_i in a subband. The Formulae for the sum is:

$$S = \sum_{i=1}^N X_i \quad (3)$$

Standard Deviation (SD) is defined as the dispersion or spread out of data values from the average value. A low value of SD indicates that data points are very close to mean; however, a large value of SD indicates that there is high variation in the data. SD can be computed as:

$$\sigma = \sqrt{\frac{\sum_{i=0}^N (X_i - \mu)^2}{N}} \quad (4)$$

Where mean μ is defined as the average value of data. The formula for the mean is:

$$\mu = \frac{\sum_{i=1}^N X_i}{N} \quad (5)$$

Energy is defined as the sum of the square of the values. The energy of an image is the sum of squares of the pixel values. Energy is given by:

$$E = \sum_{i=1}^N X_i^2 \quad (6)$$

The entropy feature measures the variation or irregularity in data. For Entropy the formula is:

$$H(X_i) = - \sum_{i=1}^N P(X_i) \log(P(X_i)) \quad (7)$$

Skewness is a measurement of symmetry. Skewness is given by:

$$G = \frac{\sum_{i=1}^N (X_i - \mu)^3 / N}{\sigma^3} \quad (8)$$

Mean Absolute Deviation (MAD) is the robust measure of the variability of quantitative data taken from a univariate sample. As compared to the standard deviation, it is more resilient to outliers in a data set. In case of standard deviation, the distances from the mean are squared, so the large deviations are weighted more heavily, and outliers have more influence. MAD is given by:

$$\tau = (\sum_{i=1}^N (X_i - \mu)) / N \quad (9)$$

Other features were also tried in our analysis, but the above were found to be the best. A further subset of these features is used in our work as will be discussed in the Results and Discussion section.

D. Feature Selection

Feature selection is an essential step in OCR applications. Retaining informative and relevant features and removing the redundant ones are a recurring theme in pattern recognition. There are two main types of techniques for feature dimensionality reduction. One is a stepwise feature selection, which uses some criteria to select fewer features from the original feature set. The second type uses an optimal or sub-optimal transformation to acquire feature dimensionality reduction.

In our work, to select the best features, the classification accuracy for each feature is determined for all the subbands. By the classification accuracy, the best features are selected out of the eight features discussed below. Then these best features are passed to KNN and SVM classifiers and analysis of their results is carried out.

E. High Entropy Subbands Selection

To further reduce the feature set, the best subbands are acquired. The entropy feature is computed for all the subbands after applying the wavelet transforms. The top high entropy subbands are selected based upon a threshold which is defined later. Then the above-mentioned features are extracted for only the selected high entropy subbands.

F. Data normalization

The data normalization is a data transformation technique. In normalization, the value of an attribute is scaled, so that it may fall within a small specified range such as between 0.0 and 1.0. It is especially useful in classification algorithms. The purpose of this scaling process is to ensure that all the features variable contributes to the same extent in the classification process. Hence, it is used to standardize the range of independent variables or features of data [34], It

also speeds up the training process. In this work, z-score normalization is used for data normalization. The formula for z-score normalization is:

$$Z_i = \frac{x_i - \mu}{\sigma} \quad (10)$$

G. Classification

Classification is a process of assigning a new label to data based upon a model acquired from the training data. A classifier learns a model based upon the training data and then the test data is classified using the model acquired. The classification accuracy is determined as the number of tuples classified correctly. A good classifier can learn a model based upon the training data while maintaining the generalization property, i.e. it works well on the test data set as well [3]. In this work, the classification is carried out using k-NN and SVM classifiers.

1) *k*-Nearest Neighbour (*k*-NN)

k-NN is one of the simplest classifiers. It is included here for comparison with SVM results. Here the algorithm is used to find the nearest match for a given case to the known circumstances existing in memory. The value of k is used to denote how many instances of training data are used for decision making. It is most optimal to choose an odd value of K, so it eliminates a chance of tie between two sets. A distance metric is used to compute the distances between the various instances and to find the nearest neighbor.

The most widely used distance metric is the Euclidean distance. However, we experimented with various distance measure and found correlation distance (Pearson) to be one of the best for our problem. For the k nearest neighbor classifier we performed various experiments to find the best value of k. For the best number of nearest neighbors, we have to compare the classification accuracy of the nearest neighbor classifier for different values of k and each set is evaluated using 10-fold cross-validation.

2) Support Vector Machines (SVMs)

Support Vector Machines is a type of binary classifier. However, the classification accuracies provided by SVMs are usually high. We search for the best parameters as well. Various parameters of SVM can be considered to optimize the performance of this algorithm. Various kernel functions were used in our analysis namely the Radial Basis Function (RBF), Polynomial Kernel and the Sigmoid Kernel. The accuracy achieved from RBF is high as compared to other kernels, but it is expensive in terms of computation and storage [9], [37].

SVM is a discriminative classifier based on Vapniks structural risk minimization principle. In general, SVM solves a binary (two-class) classification problem. However multi-class classification can also be accomplished by combining multiple binary SVMs. The training of SVM creates a convex optimization problem [9]. We use a pre-implemented library of SVM called Libsvm. Good results have been acquired using SVMs in handwritten digit recognition [3], [39]. Figure 4 represents the methodology described above in detail.

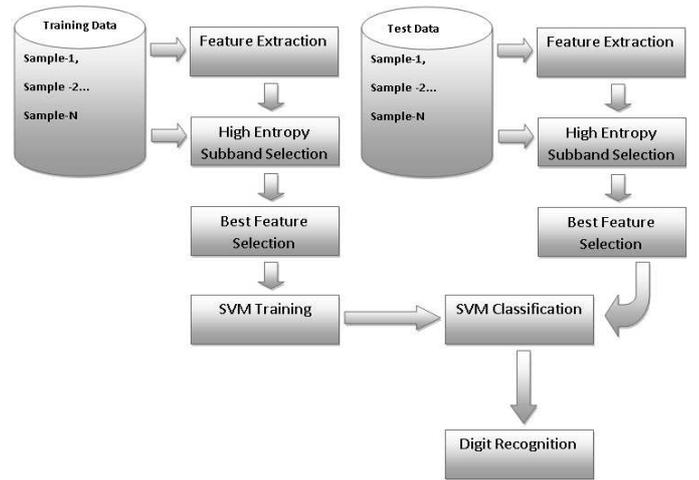


Fig. 4: The overall algorithm for the classification of digits

III. RESULTS AND DISCUSSION

This section presents the results of experiments conducted using the proposed method on the MNIST database. Amongst the popular handwritten digit databases such as CENPARMI, MNIST, and CEDAR, the MNIST has recently become more popular and is being used as a benchmark for the evaluation of new techniques and methods [24]. In our work, we have also used the MNIST database. It consists of training data and test data files. MNIST database consists of 60,000 images for training while 10000 images are kept separately for testing. In our work, we have increased the complexity of the task by dividing the data so that 48,000 images are used for training while 22,000 images are used for testing. Each data files contains greyscale images of handwritten digits, from zero through nine. Each image is 28X28 pixels forming 784 pixels. Each pixel has a single pixel-value (0-255) associated with it, which indicates the lightness or darkness of that pixel. In our work, each image is converted into a 1-D array by concatenating the image rows to produce a 1-D array of 784 values.

First, Simple Wavelet Transform (SWT) is applied to the training data. The application of SWT results in original data being decomposed into low-level approximation and high-level detail subbands. Now the approximation subband is further decomposed into approximation and detail subbands recursively until the desired level is reached. The training data set is divided into 90 percent for training and 10 percent for testing to obtain 10-fold cross-validation. To investigate the effect of the wavelet filter, various wavelets filters are applied on the original data up to the 8th level. The wavelet filters applied on the original data include Haar, Daubechies-1, Daubechies-2, Coiflets, and Symlets. Then the features mentioned above are extracted from the wavelet transformed data for each subband. The features are passed to k-NN for classification.

A. *k*-NN and Simple Wavelet Transform for Classification of MNIST Data

The average classification accuracy results are 80-84% using k-NN for 10-folds on the wavelet transformed training data set to the 7th level by using a Haar filter. The digit-wise accuracy results improve at the 8th level for the different

features included in the study as shown in Table-I. The best value of k is also computed and is found to be 9 in this instance.

TABLE I
OSS-VALIDATED CLASSIFICATION ACCURACY FOR EACH DIGIT USING K-NN (K=9) OVER FEATURES EXTRACTED USING HAAR WAVELET FILTERING

Digits	–	Level 8	–	–	Level 7	–
	Std Dev	MAD	Skew	Std Dev	MAD	Skew
	Acc. (%)					
0	98.5	98.5	96.8	96.3	96.1	94.9
1	99.1	99.1	98.4	98.7	98.6	98.2
2	89.8	89.5	76.3	70.1	67.7	68.1
3	91.7	91.7	77.6	81.7	80.9	69.5
4	92.7	92.5	89.8	86	83.1	82.7
5	88.1	87.3	70.3	72.5	70.5	64
6	96.9	97.1	94.8	92	91.5	84.3
7	93.6	93.1	91.5	85.5	83.6	89
8	86.6	86.1	73.1	76.9	75	69.3
9	88.4	88.4	88.3	80	78	82
Average	93	92.6	86	84	82.8	80.6

A comparison study was also carried out in order to choose the best wavelet filter by the recognition rate. Table II shows the classification accuracies obtained for various filters for the different digits. Haar and Daubechies filters performed 5-8% better in most instances as compared to Symlets and Coiflets. Another phenomenon that was observed was that the classification accuracies increased as the decomposition levels increased stabilizing at the 8th level. The comparison of recognition rates for various wavelet filters using all the features is shown in Table II. It should be noted that all the features given in the features section are concatenated and used for comparing wavelet filters.

TABLE II
ACCURACY COMPARISON FOR VARIOUS WAVELET FILTERS USING K-NN FOR ALL SIX FEATURES

Digits	Db-1	Db-2	Haar
0	94.6	95.9	96.5
1	99.0	97.6	97.9
2	90.8	86.7	88.0
3	79.7	83.0	80.1
4	92.6	91.2	92.6
5	75.9	74.9	76.4
6	93.6	92.3	92.5
7	91.0	92.6	94.3
8	91.2	93.0	90.7
9	85.3	87.6	88.2
Average	89.4	89.5	89.7

As Haar wavelets provide good results, they will be used in all our subsequent analysis. It can be seen from Table I and Table II that the overall accuracy for individual features is better for singular features as compared to combined features in Table II. Hence, it is important to investigate which features are better for our problem. Table III shows the results for the analysis of 6-features included in our study.

It can be seen from Table III that the top 3 features are Standard Deviation (SD), Mean Absolute Deviation (MAD) and the Sum. The higher order statistical features namely Skewness and Entropy do not seem to be performing as well. However, Entropy is better than Skewness as it approximates disorder in subbands, which is similar to SD and MAD. Although Sum and Energy are related Sum incorporates the sign of the coefficients while the significance of it is lost in Energy. Hence, Sum better approximates the variation in coefficient values. The best three features are used in all subsequent analysis. Next, we explore the results acquired using the Wavelet Packet Transforms.

TABLE III
ACCURACY RESULTS FOR VARIOUS FEATURES AT LEVEL 8 USING K-NN WITH K=9.

Digit	SD	MAD	Skew	Sum	Entropy	Energy
	Acc. (%)					
0	98.71	98.65	96.86	98.64	97.86	96.49
1	99.12	99.16	98.25	98.94	98.84	98.21
2	91.39	90.74	78.4	91.59	88.61	75.78
3	93.06	93.03	78.31	93.72	88.36	82.79
4	94	93.95	90.57	90.44	85.74	83.25
5	90.07	89.15	72.74	89.13	83.88	76.65
6	97.66	97.71	95.1	97.35	95.25	90.56
7	95.18	95.17	92.28	94.48	91.56	89.24
8	88.55	88.31	74.12	86.45	83.52	72.62
9	90.55	90.24	89.43	91.17	85.38	88.88
Overall	93.93	93.72	86.87	93.32	90.08	85.72

B. Entropy-based selected Terminal Subbands from the Wavelet Packet Tree

As described earlier, in wavelet packet decomposition, both approximation and detail subbands are decomposed into approximation and detail subbands at each level. Hence, a wavelet packet tree leads to an exponential rise in the number of subbands as the transform levels increase. To reduce the size of the data and extract the relevant information, only the selected (terminal) subbands are used in the analysis. For a selection of subbands, the analysis of the entropy feature is made for all the subbands. Entropy is used as an estimate of the information content of a subband and the top 120 subbands sorted on entropy are selected from the wavelet packet decomposition. The subbands having lower entropy are discarded. The overall average accuracy achieved for ten trials (by using 10-fold cross-validation) is 96.10 % for 8th level for high entropy bands (selected 120 subbands). The results of digit-wise accuracy can be seen in Table IV.

TABLE IV A
ACCURACY RESULTS FOR BEST FEATURES USING WAVELET PACKETS
TERMINAL 120 SUBBANDS AT LEVEL 8 USING K-NN WITH $k=9$

Digit	Total	Correct	Wrong	Accuracy (%)
0	455	452	3	99.27
1	458	455	3	99.31
2	392	373	19	95.22
3	448	428	20	95.56
4	438	418	20	95.42
5	354	330	24	93.29
6	413	407	6	98.57
7	421	405	16	96.25
8	397	371	26	93.33
9	424	397	27	93.59
Total	4200	4036	164	96.10

TABLE IV B
ACCURACY RESULTS FOR BEST FEATURES USING WAVELET PACKETS TERMINAL
HIGHEST ENTROPY 90 SUBBANDS AT LEVEL 8 USING K-NN WITH $k=9$.

Digit	Total	Correct	Wrong	Accuracy (%)
0	455	452	3	99.3
1	458	455	3	99.24
2	392	374	18	95.39
3	448	427	21	95.26
4	438	420	18	96.0
5	354	332	22	93.82
6	413	408	5	98.7
7	421	407	14	96.78
8	397	369	28	92.88
9	424	400	24	94.41
Total	4200	4042	158	96.24

TABLE V
ACCURACY RESULTS FOR THE TEST DATA USING THE BEST FEATURES FROM
TERMINAL HIGHEST ENTROPY 90 WAVELET PACKETS SUBBANDS AT LEVEL 8
USING K-NN ($k=9$).

Digit	Total	Correct	Wrong	Accuracy (%)
0	2827	2805	22	99.22
1	3391	3301	90	97.34
2	2733	2666	67	97.54
3	2785	2702	83	97.02
4	2726	2645	81	97.03
5	2528	2378	150	94.07
6	2767	2727	40	98.55
7	2917	2823	94	96.78
8	2528	2460	68	97.31
9	2798	2664	134	95.21
Total	28000	27171	829	97.04

The overall results improve slightly by further reducing the number of high entropy subbands, i.e., by selecting the top 90 high entropy subbands computed from the previous stage. The results are shown in Table V. The graphical representation of the accuracy of WPT data for topmost high entropy subbands (i.e., 80, 90, and 120) using KNN for level 8 is shown in Figure 5. Further reduction of subbands from 90 leads to a fall in accuracies as shown in Figure 5 below. All the previous analysis has been done on the training dataset using 10-fold cross-validation technique to acquire average accuracies. Next, the technique is applied on the test data set with the same steps i.e. applying the wavelet packet transform up to the 8th level, selecting high entropy subbands, extracting features and classifying using k-NN.

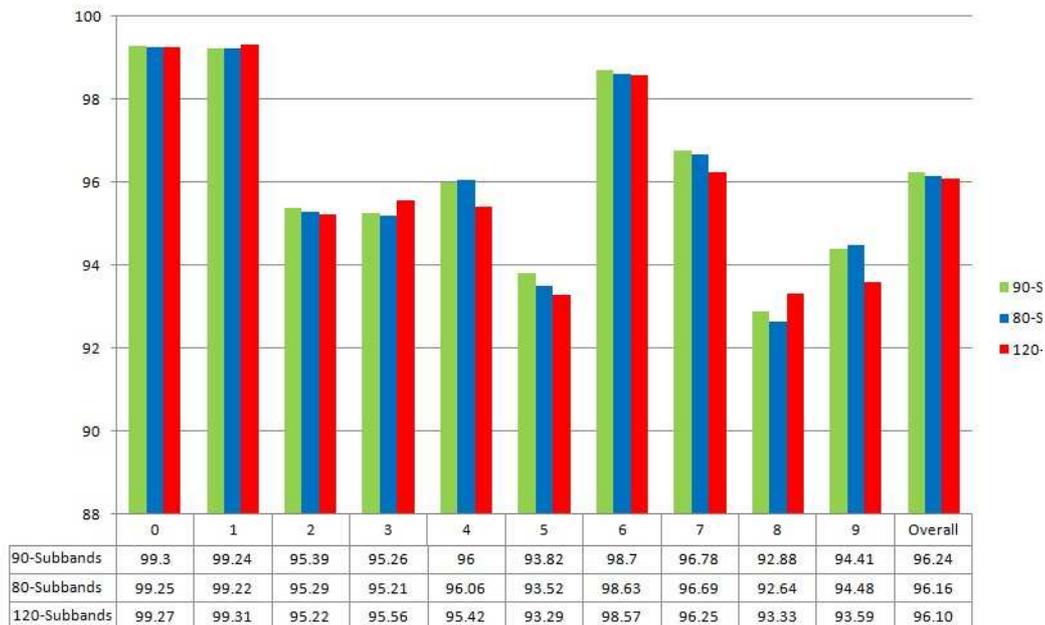


Fig. 5. Digit-Wise Accuracy for High Entropy Subbands 80, 90 and 120 at Level-8.

The overall average accuracy for ten trials is improved to 97.04% by using KNN for 8th level for high entropy bands (selected 120 high entropy subbands) with an entropy

threshold more significant than 10% of maximum. The digit-wise accuracy is shown in Table VI.

TABLE VI
BEST RESULTS ACQUIRED FOR OPTIMAL SUBBANDS USING K-NN AND SVMs

Type Of Data	Training Data Results		Test Data Results	
	KNN 10fold	SVM 10fold	KNN	SVM
90 High Entropy WPT Terminal Subbands at the 8 th level	96.24	96.29	97.04	96.18
90 High Entropy WPT Overcomplete tree subbands up to 8 th level	94.74	97.50	95.80	97

The best features extracted from the data are Standard Deviation (STD), Sum & Mean Absolute Deviation (MAD). The classification accuracy for digits 0,1,2,3,4,6,7 and 8 is higher than 96% and accuracy of digit 5 and 9 is approximately 94 and 95%.

C. Entropy-based selected Subbands from the Over-complete Wavelet Packet Tree

Next, we apply the same process as listed above for the over-complete wavelet packet tree i.e., we use all the subbands at all the levels for our analysis. We select the top 120 high entropy subbands and the top 90 high entropy subbands. We employ the technique on both the test as well as the training data. Moreover, we also use SVMs as they are considered to be more powerful classifiers as compared to k-NNs. Three different kernels i.e., linear, polynomial and RBF kernels were used. RBF kernels were found to be the best, and a parameter search is carried out to find the best parameters. The best results for top high entropy 90 subbands for the terminal as well as the overcomplete wavelet packet tree subbands are shown in Table VI. The results from Table VI show that comparative accuracies may be obtained using wavelets packet-based features for considerably different techniques namely k-NNs and SVMs. In our work, we have shown that if the features extracted are good such as the ones acquired using our techniques the results are comparable for the MNIST database. Some studies have shown SVMs to be poor as compared to k-NNs

for text categorization [12], but other studies show that comparable results may be obtained with good features extraction techniques [40]. Next, we investigate the accuracy of each subband to build an optimal classification algorithm.

D. Accuracy for Subband Selection in Over-complete Wavelet Packet Tree

Next, we determine the accuracy provided by each subband separately to assess the information content or usefulness of each subband. A wavelet packets transform of up to 8th level using the Haar filter is obtained, best features are extracted, and k-NN is used for classification. We use all the subbands from the over-complete tree in our analysis. The subband wise accuracy for 510 subbands is shown in Figure 6.

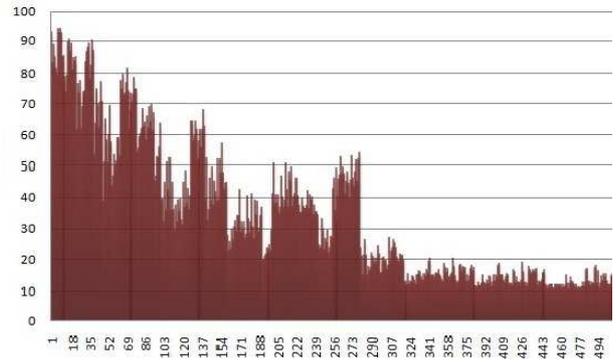


Fig. 6. Bands wise accuracy for frequency ordered subbands using the best features acquired from the previous stages

The subbands are frequency ordered with the lower frequency subbands at higher levels appearing first and the higher frequency subbands appearing later. The overall average accuracy is high for subbands from one to the 6th level, i.e., for the first 128 subbands. The accuracy decreases substantially for the later subbands representing higher frequencies. By the accuracy using KNN, selection of the best subbands is carried out, i.e., the subbands whose overall accuracy is greater than or equal to 90%. The subbands selected are 1, 7, 8, 16, and 37 as shown in Table VII.

TABLE VII
SUBBANDS-WISE ACCURACIES FOR EACH DIGIT USING SVMs.

Sub-band	Overall	0	1	2	3	4	5	6	7	8	9
1	93.0	97.3	98.9	91.8	94.9	92.7	90.5	98.1	95.9	87.7	90.6
3	88.9	97.7	98.9	90.1	90.7	85.5	84.7	95.9	90.7	86.9	83.0
6	79.1	95.8	97.4	76.7	78.6	74.7	68.5	89.8	82.0	77.8	70.7
7	94.0	97.8	98.5	94.0	92.6	93.0	90.5	97.5	97.5	86.5	92.6
8	92.7	98.1	99.3	90.1	93.0	91.9	87.8	96.6	96.4	87.5	90.4
15	89.3	97.4	98.0	91.4	87.8	86.2	80.6	97.4	91.7	85.4	83.7
16	90.5	98.5	98.6	91.3	90.2	88.2	85.4	96.1	92.5	88.2	84.2
18	89.1	97.8	98.2	88.5	90.1	85.9	83.3	96.0	89.5	86.3	82.6
19	81.0	96.0	98.0	79.4	78.5	77.2	69.3	92.2	82.9	79.3	71.1
20	84.4	97.4	98.1	82.3	86.2	80.1	73.4	93.4	84.7	85.2	72.4
21	84.0	96.8	98.1	80.0	85.7	83.8	72.4	93.5	86.1	84.4	73.3
22	84.9	96.2	98.4	82.5	85.9	84.5	78.8	95.8	88.9	77.7	80.7
25	73.2	91.5	97.4	63.3	71.1	70.5	66.4	92.9	79.1	64.8	72.3
26	77.2	92.8	97.4	76.7	78.8	71.8	74.0	93.0	81.2	69.1	73.3
29	73.6	90.3	98.3	65.5	66.2	72.6	59.4	92.7	81.6	65.2	73.3
32	86.6	97.4	98.2	87.3	82.2	86.3	76.2	95.3	90.4	84.8	80.0
34	88.9	97.7	98.4	81.9	89.2	85.2	82.5	97.0	89.0	87.0	80.7
37	90.2	97.3	98.5	86.9	88.3	88.0	85.4	97.1	93.4	82.3	86.7

These filtered subbands data is passed to SVM classifier, and the optimal configuration is obtained by setting the various parameter values. The kernel used here is the radial basis function. The digit-wise average accuracy which is achieved here for these five sub-bands is 98.38% as shown in Table VII. The results show that almost equivalent accuracies are obtained. It is also important to note that the accuracies are a bit lower than the ones reported by LeCun [11]. This is primarily because the training and test data was reconfigured so that the training has 42000 images and test data has 28,000 images. The default MNIST data configuration is 60000 images for training and 10000 for testing.

IV. CONCLUSIONS

This paper presents two good approaches for handwritten digit recognition, i.e. wavelet transforms, and wavelet packet transform using KNN and SVM classifiers. The overall average accuracy of KNN after wavelet packet transform is 96.24% on the ten-fold cross-validated data and 97.04% on the test data. However, the overall average accuracy using SVM is 96.29% on the ten-fold cross-validated data. The acquired results using KNN and SVM classifiers are comparable. It means the extracted features are good features. We further improved the results by determining the high accuracy subbands and then used them for classification. The reported accuracies are still comparable for both k-NN and SVMs (RBF) which are 97.5% and 98.38% respectively.

Although, deep learning techniques such as convolutional neural networks have demonstrated slightly higher accuracies in the literature, these techniques suffer from the issue that the results are acquired are difficult to explain. The internal working and construction of a deep neural network that enables it to achieve such high accuracies is an active area of research. However, our approach is useful in a way that allows for analysis of the results and the technique to better understand the reasons for the success of the technique. In our instance, we see that certain subbands extract more useful information to differentiate between different handwritten digits. Given the low error rate that is achieved by the proposed method on the MNIST data, we conclude that wavelet packets analysis is good for features extraction for handwritten digits. Moreover, the results acquired are comparable to the best techniques developed so far.

REFERENCES

- [1] R.-G. Mihalji, "Handwritten Digit Classification using Support Vector Machines," 2011.
- [2] D. K. Bishnoi and K. Lakhwani, "Advanced approaches of handwritten digit recognition using hybrid algorithm," *Int. J. Commun. Comput. Technol.*, vol. 1, no. 57, pp. 186–191.
- [3] M. Karić and G. Martinović, "Improving Offline Handwritten Digit Recognition Using Concavity-based Features," *Int. J. Comput. Commun. Control*, vol. 8, no. 2, pp. 206–219, 2013.
- [4] R. Kala, H. Vazirani, A. Shukla, and R. Tiwari, "Offline handwriting recognition using genetic algorithm," *arXiv Prepr. arXiv1004.3257*, 2010.
- [5] P. Dollár, Z. Tu, H. Tao, and S. Belongie, "Feature mining for image classification," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.
- [6] D. C. Álvarez, F. M. Rodríguez, and X. F. Hermida, "Printed and handwritten digits recognition using neural networks," in

Proceedings of ICSPAT-98 (International Conference on Signal Processing and Applications Technology), 1998.

- [7] V. Kapoor and P. Gupta, "Digit Recognition System by using Back Propagation Algorithm," *Int. J. Comput. Appl.*, vol. 83, no. 8, 2013.
- [8] C. Enachescu and C.-D. Miron, "Handwritten digits recognition Using neural computing," *Sci. Bull. the "Petru Maior" Univ. Targu Mures*, vol. 6, p. 17, 2009.
- [9] C.-L. Liu, K. Nakashima, H. Sako, and H. Fujisawa, "Handwritten digit recognition: benchmarking of state-of-the-art techniques," *Pattern Recognit.*, vol. 36, no. 10, pp. 2271–2285, 2003.
- [10] B. El Kessab, C. Daoui, B. Bouikhalene, M. Fakir, and K. Moro, "Extraction method of handwritten digit recognition tested on the mnist database," *Int. J. Adv. Sci. Technol.*, vol. 50, pp. 99–110, 2013.
- [11] Y. LeCun, C. Cortes, and C. J. C. Burges, "The MNIST Database of handwritten digits," 2002.
- [12] Y. LeCun, "LeNet-5, convolutional neural networks," *URL http://yann.lecun.com/exdb/lenet*, vol. 20, 2015.
- [13] E. Stomatias, D. Neil, F. Galluppi, M. Pfeiffer, S.-C. Liu, and S. Furber, "Live demonstration: handwritten digit recognition using spiking deep belief networks on SpiNNaker," in *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2015, p. 1901.
- [14] S. Pan, Y. Wang, C. Liu, and X. Ding, "A discriminative cascade CNN model for offline handwritten digit recognition," in *2015 14th IAPR International Conference on Machine Vision Applications (MVA)*, 2015, pp. 501–504.
- [15] H. Cecotti, "Active graph based semi-supervised learning using image matching: application to handwritten digit recognition," *Pattern Recognit. Lett.*, vol. 73, pp. 76–82, 2016.
- [16] J. Qiao, G. Wang, W. Li, and M. Chen, "An adaptive deep Q-learning strategy for handwritten digit recognition," *Neural Networks*, vol. 107, pp. 61–71, 2018.
- [17] R. V. K. Reddy and U. R. Babu, "Efficient Handwritten Digit Classification using User-defined Classification Algorithm," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 8, no. 3, pp. 970–979, 2018.
- [18] J. V. S. Srinivas, P. V. Kumar, and P. Premchand, "Hand Written Digit Recognition Using Elman Neural Network," *Int. J. Eng. Innov. Technol.*, vol. 2, no. 5, pp. 9–13, 2012.
- [19] A. Goltsev and V. Gritsenko, "Investigation of efficient features for image recognition by neural networks," *Neural Networks*, vol. 28, pp. 15–23, 2012.
- [20] F. Lauer, C. Y. Suen, and G. Bloch, "A trainable feature extractor for handwritten digit recognition," *Pattern Recognit.*, vol. 40, no. 6, pp. 1816–1824, 2007.
- [21] D. J. Romero, L. Seijas, and A. Ruedín, "Directional continuous wavelet transform applied to handwritten numerals recognition using neural networks," *J. Comput. Sci. Technol.*, vol. 7, 2007.
- [22] C.-L. Liu, K. Nakashima, H. Sako, and H. Fujisawa, "Handwritten digit recognition: investigation of normalization and feature extraction techniques," *Pattern Recognit.*, vol. 37, no. 2, pp. 265–279, 2004.
- [23] F. Bortolozzi, A. de Souza Britto Jr, L. S. Oliveira, and M. Morita, "Recent advances in handwriting recognition," *Doc. Anal.*, pp. 1–31, 2005.
- [24] Ø. D. Trier, A. K. Jain, and T. Taxt, "Feature extraction methods for character recognition-a survey," *Pattern Recognit.*, vol. 29, no. 4, pp. 641–662, 1996.
- [25] M. Shi, Y. Fujisawa, T. Wakabayashi, and F. Kimura, "Handwritten numeral recognition using gradient and curvature of gray scale image," *Pattern Recognit.*, vol. 35, no. 10, pp. 2051–2059, 2002.
- [26] L.-N. Teow and K.-F. Loe, "Robust vision-based features and classification schemes for off-line handwritten digit recognition," *Pattern Recognit.*, vol. 35, no. 11, pp. 2355–2364, 2002.
- [27] P. Y. Simard, D. Steinkraus, and J. C. Platt, "Best practices for convolutional neural networks applied to visual document analysis," in *Icdar*, 2003, vol. 3, no. 2003.
- [28] G. Y. Chen, T. D. Bui, and A. Krzyzak, "Contour-based handwritten numeral recognition using multiwavelets and neural networks," *Pattern Recognit.*, vol. 36, no. 7, pp. 1597–1604, 2003.
- [29] P. Zhang, T. D. Bui, and C. Y. Suen, "A novel cascade ensemble classifier system with a high recognition performance on handwritten digits," *Pattern Recognit.*, vol. 40, no. 12, pp. 3415–3429, 2007.
- [30] Y. LeCun *et al.*, "Learning algorithms for classification: A comparison on handwritten digit recognition," *Neural networks Stat. Mech. Perspect.*, vol. 261, p. 276, 1995.

- [31] J. Cai and Z.-Q. Liu, "Integration of structural and statistical information for unconstrained handwritten numeral recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 3, pp. 263–270, 1999.
- [32] M. Hanmandlu, M. V. Krishna, and P. C. Krishna, "Some Approaches to the recognition of handwritten numerals," in *National Conference on Document Analysis and Recognition*, 2001.
- [33] R. M. O. Cruz, G. D. C. Cavalcanti, and T. I. Ren, "Handwritten digit recognition using multiple feature extraction techniques and classifier ensemble," in *17th International Conference on Systems, Signals and Image Processing*, 2010, pp. 215–218.
- [34] S.-B. Cho, "Neural-network classifiers for recognizing totally unconstrained handwritten numerals," *IEEE Trans. Neural Networks*, vol. 8, no. 1, pp. 43–53, 1997.
- [35] Z. Dan and C. Xu, "The recognition of handwritten digits based on bp neural network and the implementation on android," in *2013 Third International Conference on Intelligent System Design and Engineering Applications*, 2013, pp. 1498–1501.
- [36] M. Wu and Z. Zhang, "Handwritten digit classification using the mnist data set," *Course Proj. CSE802 Pattern Classif. Anal.*, 2010.
- [37] Y.-S. Hwang and S.-Y. Bang, "An efficient method to construct a radial basis function neural network classifier," *Neural networks*, vol. 10, no. 8, pp. 1495–1503, 1997.
- [38] Y. Li, M. Dong, and J. Hua, "Simultaneous localized feature selection and model detection for Gaussian mixtures," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 5, pp. 953–960, 2009.
- [39] J. Dong, A. Krzyzak, and C. Y. Suen, "Fast SVM training algorithm with decomposition on very large data sets," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 4, pp. 603–618, 2005.
- [40] F. Colas and P. Brazdil, "Comparison of SVM and some older classification algorithms in text classification tasks," in *IFIP International Conference on Artificial Intelligence in Theory and Practice*, 2006, pp. 169–178.