A Hybrid Feature Extraction Method for Offline Handwritten Math Symbol Recognition

Carlos Ramírez-Piña[1], Josep Salvador Sánchez[2(✉)],
Rosa M. Valdovinos-Rosas[1], and José A. Hernández-Servín[1(✉)]

[1] School of Engineering, Universidad Autónoma del Estado de México,
Toluca, Mexico
xoseahernandez@uaemex.mx

[2] Institute of New Imaging Technologies,
Department Computer Languages and Systems,
Universitat Jaume I, Castelló de la Plana, Spain
sanchez@uji.es

Abstract. This paper introduces a feature extraction scheme for offline handwritten math symbol recognition. It is a hybrid model that involves the basic ideas of the wavelet and zoning techniques so as to define the feature vectors with both statistical and geometrical properties of the symbols, with the aim of overcoming some limitations of the individual algorithms used. Experiments over a medium-sized database of isolated math symbols investigate the performance of the new hybrid technique in comparison to other algorithms. The results show that the new model performs significantly better than the rest of algorithms tested, independently of the symbol category.

Keywords: Feature extraction · Handwritten math symbol recognition

1 Introduction

Recognition of handwritten mathematical expressions allows to transform formulas in scientific documents into an electronic representation. Although it might appear that the mathematical expression recognition is equivalent to the recognition of plain text, there exist differences that make it unrealistic to apply standard solutions of handwritten character recognition to mathematical notation. First, a line of text is one-dimensional and discrete, whereas symbols in mathematical expressions are spatially arranged into complex two-dimensional structures. Second, symbol recognition is a nontrivial problem because the vocabulary is very large (digits, Latin and Greek letters, operator symbols, relation symbols, etc.) with a variety of typefaces (regular, bold, italic, calligraphic) and several font sizes (subscripts, superscripts, limit expressions) [2]. Third, mathematical

R. Vera-Rodriguez et al. (Eds.): CIARP 2018, LNCS 11401, pp. 893–901, 2019.
https://doi.org/10.1007/978-3-030-13469-3_103

handwriting may involve large operators such as matrix brackets, fraction bars or square roots.

Recognition of a mathematical expression comprises two main steps [5]: symbol recognition and structural analysis. The recognition stage translates the input image into a set of mathematical symbols present in the expression, being as task of most relevance. In general, symbol recognition comprises a set of processes that are applied to the input image: pre-processing, segmentation to isolate symbols, feature extraction, and classification. On the other hand, the objective of structural analysis is to determine the relations among the symbols recognized in the previous stage in order to build a complete structure that represents the mathematical expression.

The scope of this paper focuses on the field of isolated math symbol recognition, which is deemed to be a hard problem [8]. From the different operations included in this stage, feature extraction is one of the most critical elements of a mathematical recognition system because it provides the set of features used to describe each symbol precisely.

This work presents a feature extraction algorithm for offline handwritten math symbol recognition. It combines the wavelet and zoning techniques to obtain a feature vector with both statistical and geometrical properties of the symbols, thus overcoming some limitations of those individual feature extraction algorithms. The performance of the new hybrid method is compared against that of four feature extraction techniques when applied to a medium-sized database of isolated math symbols.

2 Some Feature Extraction Techniques

Feature extraction methods can be classified into two major groups: statistical and structural [2]. In the statistical approach, a character image is represented by a set of d features that are derived from the statistical distributions of pixels and can be considered as a point in d-dimensional feature space. In the structural category, various local and global properties of the character can be represented by geometrical and topological characteristics. Note that structural and statistical features are deemed to be complementary in the sense that they emphasize different properties of the characters.

2.1 The FKI Algorithm

Given a binary image I of size $M \times N$, the FKI algorithm [9] computes a set of nine geometrical values for each image column y, obtaining 9-dimensional vectors $v(y) = [v_1(y), \ldots, v_9(y)]$. The algorithm uses a sliding window of size 1, moving from the very left of the image to the very right, to calculate a set of geometrical values.

2.2 The Wavelet Method

Wavelet transform is a multi-resolution signal decomposition tool that provides a representation of an image at different levels of resolution. This work utilizes 3-level Daubechies discrete wavelet transform (DWT) [7], which recursively decomposes an input image I of size $M \times N$ into one low-frequency component (a thumbnail of the input image) and three high-frequency components for each level of decomposition j. The contour of the image is in the low-frequency sub-band and contains the approximation or scale coefficients (A_j), whereas the high-frequency sub-band includes the so-called detail coefficients H_j (horizontal), V_j (vertical) and D_j (diagonal).

The input image is fed into two filters h and g, which produce the approximation coefficient A_j and the three detail coefficients H_j, V_j and D_j, which are all down-sampled by a factor of 2. Since images are two-dimensional structures, filtering and sub-sampling are first applied along the rows of the image and then along the columns of the transformed image. The result of these operations is a transformed image with four distinct bands: the upper left band corresponds to a down-sampled version of the original image that has been, the bottom left band tends to preserve localized horizontal features, the upper right band tends to preserve localized vertical features, and the bottom right band tends to isolate localized high-frequency point features in the image. Additional levels of decomposition can be applied only to the upper left band of the transformed image at the previous level in order to extract lower frequency features in the image.

Frequency domain analysis is the background of representation of the feature vector (with a size of $\frac{M}{2}\frac{N}{2}$), but a total of 54 textural and statistical values are also computed to enhance the feature vector [10]. In particular, entropy, mean and standard deviation are computed on the gray-scale, binary and twelve sub-band images. Analogously, the Shannon entropy, the 'log energy' entropy, the threshold entropy, the sure entropy and the norm entropy are also calculated on the approximation coefficient sub-band.

2.3 The Zoning Technique

The zone-based feature extraction algorithm [3] used in this paper follows the foundations of the procedure proposed by Ashoka et al. [1]. It splits a binary image I of size $M \times N$ into a number of squared, non-overlapping zones or patches of a predefined size $m \times n$. For each zone Z_i, two values are calculated to build up the feature vector: one is the density of black pixels and the second corresponds to the normalized coordinate distance of black pixels.

Firstly, a grid L of size $M \times N$ is superimposed on the image, where the (x, y)-th element of L will be assigned to 1 if the pixel $I(x, y)$ is black and 0 otherwise. Then the density of black pixels in a zone Z_i can be computed as

$$v_1(Z_i) = \frac{1}{mn} \sum_{l(x,y) \in L} l(x, y) \tag{1}$$

where mn is the total number of pixels in Z_i and $l(x, y)$ denotes the value of the (x, y)-th element of L.

For the second value, consider the bottom left corner of each grid as the absolute origin $(0, 0)$ and compute the coordinate distance $\delta_j(Z_i)$ of the j-th pixel in zone Z_i at location (x, y). Then the normalized coordinate distance of black pixels can be obtained by dividing the sum of coordinate distances of black pixels (i.e., elements of the grid L whose value is equal to 1) by the sum of coordinate distances of all pixels in zone Z_i:

$$v_2(Z_i) = \frac{\displaystyle\sum_{j \in Black(Z_i)} \delta_j(Z_i)}{\displaystyle\sum_{j=1}^{mn} \delta_j(Z_i)} \tag{2}$$

where $Black(Z_i)$ denotes the set of black pixels in zone Z_i.

2.4 The Binarization Algorithm

The binarization technique for feature extraction aims to minimize the useless information that can be present in an image [6]. Accordingly, it is assumed that a binary image I has black pixels, which correspond to the characters, and white pixels for the background. Thus, it is possible to represent the image by a matrix $\mathbf{W} = [\mathbf{w_{xy}}]_{\mathbf{M \times N}}$ where the (x, y)-th component of $\mathbf{W}$ will be assigned to 1 if the pixel $I(x, y)$ is black, and to 0 for a white pixel. This matrix $\mathbf{W}$ can be then reshaped in a row first manner to a column vector of size $M \times N \times 1$.

3 Methodology

This section presents the proposed methodology, which follows the phases of a standard image recognition system. Apart from describing the specific tasks performed at each stage, we will also introduce a new algorithm for feature extraction, which is the result of hybridizing the foundations of the DWT and zoning methods presented in the previous section.

3.1 Image Acquisition, Binarization and Segmentation

A total of 185 gray-scale images of size 4160 × 1200 were obtained by scanning notes and documents handwritten by a pool of writers. The documents consisted not only of mathematical expressions, but also of plain text, diagrams and graphics. With the purpose of discarding useless information and handling an image formed only by a set of mathematical symbols, the region of interest with the handwritten mathematical expression or formula was selected manually.

Then the gray-scale images were converted into binary using the Otsu's thresholding method [11], which assumes that the distribution of the pixel intensities is bi-modal: dark pixel intensities (corresponding to the object or character) can be separated from light pixel intensities (the background) in the gray-level histogram. The central idea of this method is to find the threshold that maximizes the between-class variance.

For the segmentation of the binary images, we chose a technique based on labeling connected regions (which correspond to symbols). The algorithm starts from the first foreground pixel found and then, it propagates to any of the pixel's 4-neighbors. Each already visited pixel cannot be explored again; after the entire connected region has been labeled, a region number is assigned to its pixels. Afterwards, each connected region, which has been labeled with a region number, is enclosed by a bounding box.

The coordinates of these bounding boxes allow to describe the relationships between the input symbols and distinguish single symbols from those symbols that are composed of two or more strokes. To check whether or not two or more bounding boxes correspond to the same symbol, we analyzed some characteristics of the boxes: length, height, distance between boxes, and size. Boxes complying with these characteristics were re-labeled, indicating that they belong to the same symbol. Finally, each symbol image was resized to a fixed size of 120×120.

3.2 Combining Wavelets and Zoning for Feature Extraction

We introduce a new method, hereafter called c-WZ, to extract discriminant features from the binary image and build up a feature vector by combining the bases of the DWT and zoning techniques with the aim of using both statistical and geometrical characteristics of the image.

Firstly, the 3-level Daubechies DWT decomposes the binary image I of size $M \times N$ $(M = N = 120)$ in order to obtain the coefficients of the third block, which correspond to those with the most representative geometrical characteristics of the image. The approximation coefficient A_2 represents a thumbnail of I, whereas the detail coefficients H_2, V_2 and D_2 contain characteristics related to the contour of the symbol. Each of these coefficients is of size $\frac{M}{m_w}\frac{N}{n_w}$ with $m_w = n_w = 8$, leading to a total of 900 features. Next, the mean, the standard deviation and the entropy for the coefficients A_2, H_2, V_2 and D_2 are also calculated. In addition, the Shannon entropy, the 'log energy' entropy, the threshold entropy, the sure entropy and the norm entropy are calculated for the approximation coefficient A_2. Thus the wavelet-based stage of the c-WZ algorithm produces a feature vector with 917 textural and statistical values as a result of the frequency domain analysis.

Then, the zoning-based stage of c-WZ divides the image I into squared zones of size $m_z \times n_z$, and two values are calculated for each zone Z_i: the total number of black pixels (instead of the density of black pixels as done in the standard zoning technique) and the normalized coordinate distance of black pixels. This produces a feature vector of size $2(\frac{M}{m_z}\frac{N}{n_z})$ with $m_z = n_z = 15$, which gives a total of 128 values. Finally, the feature vectors that result from wavelet and zoning stages are concatenated to build up the feature vector of the c-WZ algorithm.

4 Experiments

The aim of the experiments is to compare the c-WZ method against FKI, wavelet, zoning and binarization. Six standard classifiers were applied to the sets of samples generated by the feature extraction algorithms using 10-fold cross-validation: the nearest neighbor (1-NN) rule with the Euclidean distance, the naive Bayes classifier (NBC), a Bayesian network (BN), a multi-layer perceptron (MLP) with one hidden layer, a support vector machine (SVM) with a linear kerne ($C = 1.0$), and the C4.5 decision tree with pruning by the subtree raising approach.

The empirical analysis was performed over the English database generated by Campos et al. [4], which includes digits (10 classes) with 527 samples, the uppercase Latin letters (upLatin) with 26 classes and 1402 samples, and the lowercase Latin letters (lowLatin) with 26 classes and 1321 samples. In addition, by means of the methodology described in Sect. 3.1, we also incorporated the uppercase and lowercase Greek letters (upGreek and lowGreek, respectively) with 24 classes each, and a miscellany of mathematical symbols (24 classes), all of them with 1320 samples. Putting these sets (types) of characters all together leads to a database with a total of 7210 samples of isolated math symbols that belong to 134 different classes.

Although images were resized to a fixed size of 120×120 pixels, the dimension of the feature vectors depends on each feature extraction algorithm (see Table 1).

Table 1. Dimension of the feature vectors

	Dimensionality	Size
FKI	$9N$	1080
DWT	$(\frac{M}{2}\frac{N}{2}) + 54$	3654
Zoning	$2(\frac{M}{m}\frac{N}{n})$	1152
Binar.	$M \times N$	14400
c-WZ	$4(\frac{M}{m_w}\frac{N}{n_w}) + 17 + 2(\frac{M}{m_z}\frac{N}{n_z})$	1045

5 Results

Table 2 reports the accuracy rates when using the feature extraction methods with each classifier over the whole data set (7210 samples). The values for the best performing algorithm with each classifier are highlighted in bold face. As can be seen, the proposed c-WZ method achieved the highest rates when using SVM, MLP and C4.5, whereas its accuracy were not too far from that of the best technique for the rest of classifiers. To assess the statistical significance of these results, the Friedman's average rank for each algorithm was also calculated (note that the one with the lowest average rank corresponds to the best strategy), showing that the recognition rates using the c-WZ technique were better than those obtained with any other feature extraction method.

Table 2. Accuracy rate and Friedman's rank over the whole data set

	BN	NBC	SVM	MLP	1-NN	C4.5	Rank
FKI	91.82	87.22	93.71	92.04	90.20	82.26	4.00
DWT	**92.65**	86.85	95.97	94.17	85.43	81.58	3.16
Zoning	91.50	88.05	95.76	94.14	**94.38**	83.35	2.66
Binarization	90.58	**89.99**	95.60	93.87	93.34	81.18	3.50
c-WZ	92.51	89.20	**96.54**	**94.90**	93.15	**83.83**	**1.66**

Table 3. Accuracy rate and Friedman's rank for each set of characters

		BN	NBC	SVM	MLP	1-NN	C4.5	Rank
Digits	FKI	85.85	78.02	91.06	89.16	84.05	72.80	4.83
	DWT	**89.31**	79.46	95.47	93.87	83.01	73.94	3.66
	Zoning	88.85	84.32	96.31	93.95	93.11	**82.18**	2.33
	Binarization	88.43	**84.35**	96.33	94.87	**94.24**	78.73	**2.00**
	c-WZ	89.06	83.21	**96.56**	**94.91**	91.89	77.39	2.16
upLatin	FKI	91.44	85.37	92.32	89.22	85.40	80.36	3.83
	DWT	91.75	83.37	95.68	94.01	75.90	79.18	3.50
	Zoning	90.50	83.47	94.89	93.80	94.27	**81.70**	3.00
	Binarization	88.07	**87.47**	94.93	93.37	**94.48**	80.33	3.00
	c-WZ	**91.89**	85.77	**95.77**	**94.40**	90.39	81.21	**1.66**
lowLatin	FKI	79.24	74.10	83.10	82.29	80.74	62.06	3.50
	DWT	**79.96**	69.36	87.64	83.97	71.81	54.82	3.33
	Zoning	76.50	73.98	86.99	83.42	**86.07**	**62.25**	3.00
	Binarization	76.89	**77.83**	87.02	83.04	84.77	60.67	2.83
	c-WZ	78.66	75.19	**89.58**	**84.63**	82.55	60.47	**2.33**
upGreek	FKI	98.87	97.61	99.08	97.95	98.14	95.92	3.33
	DWT	98.57	98.15	**99.29**	98.52	95.37	96.80	2.66
	Zoning	98.54	**99.09**	99.25	98.48	**98.56**	93.92	2.66
	Binarization	97.36	97.46	98.81	97.50	96.00	91.63	4.83
	c-WZ	**99.05**	98.46	99.26	**98.60**	98.25	**97.36**	**1.50**
lowGreek	FKI	**99.72**	88.77	96.76	93.64	93.10	82.68	3.66
	DWT	96.39	90.89	97.79	94.97	86.70	84.82	2.83
	Zoning	94.86	87.61	97.20	95.38	94.28	80.46	3.33
	Binarization	94.17	**93.53**	96.58	94.85	90.79	76.13	4.00
	c-WZ	96.45	92.68	**98.11**	**97.05**	**95.79**	**86.57**	**1.66**
Math	FKI	99.81	99.47	**99.97**	**100**	99.81	99.79	2.83
	DWT	**99.95**	99.92	**99.97**	99.72	99.81	**99.97**	2.50
	Zoning	99.75	99.83	99.95	99.85	**100**	99.61	3.33
	Binarization	98.57	99.33	99.95	99.62	99.81	99.61	4.66
	c-WZ	**99.95**	**99.93**	**99.97**	99.85	**100**	**99.97**	**1.66**

After evaluating the performance when all the characters were put into a unique data set, one might wonder whether the feature extraction algorithms would show the same behavior irrespective of the set of characters being analyzed or on the contrary, they would perform differently with each set. To outline an answer to these question, Table 3 provides the accuracy rates and the Friedman's average ranks when the feature extraction algorithms were applied to each set of characters. Bold-faced values highlight the best feature extraction algorithm for each classifier and each data set, whereas underlined values indicate the best performing classifier for each feature extraction method and each set of symbols.

The only case in which the c-WZ algorithm did not received the best Friedman's rank corresponds to the set of digits, although it was very close to the lowest ranking assigned to binarization. For the remaining data sets, c-WZ showed the best overall behavior (the lowest Friedman's average rank). In general, FKI and binarization were the techniques with the poorest performance: FKI took the highest average rank when applied to digits, uppercase Latin letters and lowercase Latin letters, and binarization overuppercase Greek letters, lowercase Greek letters and math symbols. On the other hand, the results in Table 3 also reflect that SVM was the model with the highest recognition rate independently of the feature extraction method and the set of characters.

6 Conclusions

A hybrid feature extraction method for offline handwritten math symbol recognition has been introduced. The bases of this model relies on statistical and geometrical characteristics of the symbol images, which have been obtained from the combined application of an extended version of DWT and a zoning technique.

Experiments have revealed that the hybrid method performs better than other standard feature extraction algorithms, both over the whole database with 7210 samples from 134 different classes and over almost each set of characters. Besides, we have observed that SVM and MLP can be deemed as the most appropriate classifiers to be used with the new technique. Another point of interest refers to the fact that the c-WZ method has led to feature vectors of size smaller than those given by any of the remaining feature extraction algorithms, which supposes some computational advantages for the subsequent recognition tasks.

Acknowledgement. Partially supported by the Mexican CONACYT through the Student Support Program [CVU 702528], the Generalitat Valenciana [PROMETEOII/2014/062], and the Universitat Jaume I [P1-1B2015-74].

References

1. Ashoka, H., Manjaiah, D., Bera, R.: Feature extraction technique for neural network based pattern recognition. Int. J. Comput. Sci. Eng. **4**(3), 331–339 (2012)
2. Blostein, D., Zanibbi, R.: Processing mathematical notation. In: Doermann, D., Tombre, K. (eds.) Handbook of Document Image Processing and Recognition, pp. 679–702. Springer, London (2014). https://doi.org/10.1007/978-0-85729-859-1_21
3. Bokser, M.: Omnidocument technologies. Proc. IEEE **80**(7), 1066–1078 (1992)
4. Campos, T.E., Babu, B.R., Varma, M.: Character recognition in natural images. In: Proceedings of the International Conference on Computer Vision Theory and Applications, Lisbon, Portugal, pp. 273–280 (2009)
5. Chan, K.F., Yeung, D.Y.: Mathematical expression recognition: a survey. Int. J. Doc. Anal. Recog. **1**, 3–15 (2000)
6. Choudhary, A., Rishi, R., Ahlawat, S.: Off-line handwritten character recognition using features extracted from binarization technique. AASRI Procedia **4**, 306–312 (2013)
7. Daubechies, I.: The wavelet transform, time-frequency localization and signal analysis. IEEE Trans. Inform. Theory **36**(5), 961–1005 (1990)
8. Koerich, A.L., Sabourin, R., Suen, C.Y.: Large vocabulary off-line handwriting recognition: a survey. Pattern Anal. Appl. **6**(2), 97–121 (2003)
9. Marti, U.V., Bunke, H.: Using a statistical language model to improve the performance of an HMM-based cursive handwriting recognition system. Int. J. Pattern Recogn. **15**(1), 65–90 (2001)
10. Obaidullah, S.M., Halder, C., Das, N., Roy, K.: Numeral script identification from handwritten document images. Procedia Comput. Sci. **54**, 585–594 (2015)
11. Otsu, N.: A threshold selection method from gray-level histograms. IEEE Ttans. Syst. Man Cyb. **9**(1), 62–66 (1979)