A Symbol-dependent Writer Identification Approach in Old Handwritten Music Scores

Alicia Fornés, Josep Lladós
Computer Vision Center, Dept. of Computer Science
Campus UAB, Ed. O, 08193, Bellatera (Barcelona), Spain
{*afornes,josep*}*@cvc.uab.es*

Abstract

Writer identification consists in determining the writer of a piece of handwriting from a set of writers. In this paper we introduce a symbol-dependent approach for identifying the writer of old music scores, which is based on two symbol recognition methods. The main idea is to use the Blurred Shape Model descriptor and a DTW-based method for detecting, recognizing and describing the music clefs and notes. The proposed approach has been evaluated in a database of old music scores, achieving very high writer identification rates.

1. Introduction

Writer identification is focused on the identification of the author of a piece of handwriting from a set of writers. Writer identification in handwritten text documents is a mature area of study [3], whereas very few research has been done in identifying the writer of graphical documents, such as music scores. To the best of our knowledge, only one work has been performed about writer identification in music scores [2]. The authors have developed a prototype that analyzes the whole music score and extracts some features about structural information of the music symbols. However, since the huge difficulties in the recognition of the whole score, the feature extraction was performed manually, and as far as we know, this work has not been continued.

Traditionally, writer identification approaches can be divided in text-dependent and text-independent, depending on whether or not the meaning of the text is known. When dealing with graphical information, we referred to the above concepts as *symbol-dependent* and *symbol-independent*. In [6] and [7] we presented two different *symbol-independent* writer identification approaches in music scores, which avoid the recognition of the elements in the score. In this paper, a *symbol-dependent* writer identification method is proposed, which is only focused on the detection and recognition of certain music symbols, and thus, avoiding the high difficulties in recognizing the whole music score. A similar idea has been proposed for writer identification in Hebrew documents [1]. It is based on the detection and extraction of features from three predefined Hebrew characters, and the rest of the characters are not taken into account.

After analyzing the different music elements and their characteristics, we can conclude that the most discriminant properties of the handwriting style are music clefs. Clefs can be seen as a characteristic individual signature of a writer, having a high discrimination power. An important advantage is that there are only three different clefs to consider (alto, bass or trebble clef). In addition, clefs are usually appearing in each music sheet, allowing the comparison between music scores. Concerning music notes, the higher is the number of writer to consider, the higher is the confusion between their writing styles of music notes, and consequently, the discrimination power becomes low. However, since there are not many clefs in the music score, the music notes can be used when the information about music clefs is not enough. Other music symbols (e.g. rests, accidentals, dynamics, time signature) could also be considered, but the probability to find the same symbol in different music sheets is very low. Lyrics will not be used for writer identification because not all the music scores contain text (e.g. music scores for instruments), and in addition, the writer of the lyrics and the writer of the music notation is not always the same.

Our proposed *symbol-dependent* method detects and recognizes the music clefs and notes, and then, it performs writer identification based on the symbol descriptors computed from each clef (or note). Two main tasks are addressed here: symbol detection (and recognition) and the classification in terms of the writer based on the segmented symbols. One can note the *chicken & egg* problem as the *segmentation-recognition* paradox, because we can not decide between segmenting for recognizing and recognizing for segmenting, being the ideal solution to perform both tasks at the same time. The aim of symbol detection is the localization and segmentation of the target symbol in the image, discarding the other symbols. The detection techniques can rely on different pattern recognition methods, such as geometric features [5], relational indexing of numeric primitive descriptors [12], or the structural symbol representation [13]. Symbol recognition [10] is focused on the description of the symbol for finding its corresponding true class given a set of classes. Due to the large different kinds of problems in symbol recognition applications, a symbol descriptor usually reaches good performance in some aspects, but fails in others. For our purpose, we require symbol recognition methods which are tolerant to noise, degradation and elastic deformations typically found in old handwritten documents.

In this paper we propose a writer identification approach using a two-class symbol segmentation. Firstly, the music clefs and notes are detected and segmented using a combination of the Blurred Shape Model (BSM) [4] and a DTW-based method [8]. They have shown to be able to cope with hand-drawn distortions and also with the inaccuracy on the symbol segmentation. Then, we compare the BSM descriptors of the segmented symbols for the identification of the writer, which is performed using a voting scheme.

The remainder of the paper is structured as follows. In the next section, the preprocessing step is presented. In Section 3 the two symbol recognition methods are presented. The detection technique is described in Section 4. The classification in terms of the writer is presented in Section 5. Experimental results are shown in Section 6. Finally, Section 7 concludes the paper.

2 Preprocessing

The preprocessing phase consists in binarizing the image, deskewing it and removing the staff lines. Firstly, the gray-level scanned image (at a resolution of 300 dpi) is binarized (using the Niblack's method [11]), and the Hough Transform is applied for detecting the staff lines, and for obtaining the rotation angle (in case it the deskewing is necessary). The third step consists in removing the staff lines in order to isolate music symbols. First, we obtain a coarse approximation of the staff lines using median filters, and then, a contour tracking process is used for following and removing every staff line, taking into account the coarse approximation when gaps are appearing. For further details, see [6].

3 Symbol Recognition Methods

The proposed *symbol-dependent* writer identification approach requires symbol recognition methods which can cope with degradation, noise and distortions. For this purpose, we combine the Blurred Shape Model descriptor (BSM) and a Dynamic Time Warping (DTW) based method. The idea is to first use a coarse descriptor (BSM) and then a fine approach (the DTW-based one). Both methods are briefly described next.

3.1 Blurred Shape Model

The second method proposed defines the Blurred Shape Model (BSM) descriptor. This descriptor encodes the spatial probability of appearance of the shape pixels and their context information in the following way: The image is divided in a grid of $n \times n$ equal-sized subregions, and each bin receives votes from the shape points in it and also from the shape points in the neighboring bins. Thus, each shape point contributes to a density measure of its bin and its neighboring ones. The output descriptor is a vector histogram where each position corresponds to the amount of shape points in the context of the sub-region. The resulting vector histogram, obtained by processing all feature points, is normalized in the range $[0..1]$ to obtain the probability density function (pdf) of $n \times n$ bins. In this way, the output descriptor represents a distribution of probabilities of the object shape considering spatial distortions. As a result, a robust technique in front of noise and elastic deformations is obtained. For further details, see [4].

3.2 Dynamic Time Warping-based Method

The first symbol recognition method is based on the Dynamic Time Warping algorithm (DTW) [9]. The DTW algorithm was proposed for comparing signals, distorting the time axis for finding the best matching between two sequences. For recognizing bi-dimensional handwritten symbols, we have proposed a variation of the DTW algorithm, which is not only robust to hand-drawn distortions, but it is also robust to rotation. The steps of the method are the following: first, for every column of the two symbols to be compared, we extract a set of features, consisting in the number of foreground pixels, the upper and lower profiles, and the sum of pixels of several column regions. The DTW distance between these two symbols is computed. This process is repeated for different orientations of the two symbols. Finally, the minimum DTW distance will decide the best matching (and also the rotation angle applied) for both symbols. For further details, see [8].

4 Clefs and Notes Detection

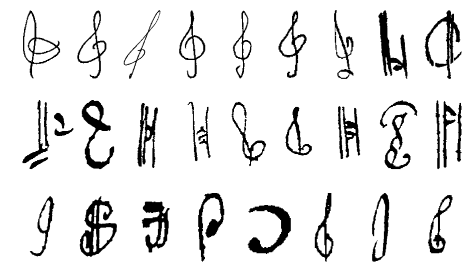
The method proposed for writer identification is composed of two tasks, namely, symbol detection and symbol description. The first step consists in detecting the music clefs and notes in the music score. For this purpose, we use a combination of the above described BSM descriptor and the DTW-based method.

A symbol detection method requires a good localization strategy and a robust symbol descriptor. Concerning the localization step, the aim is to localize the target symbol while discarding the most part of the image. In addition, one should avoid the analysis of the whole image with a sliding window for saving time. Referring the detection step, the descriptor should cope with deformation, distortions, noise and segmentation inaccuracies. It should be said that for obtaining characteristics of the music clefs and notes, it is not necessary to detect all the clefs and notes of the image. Contrary to Optical Music Recognition, badly segmented or incomplete music symbols could be left out, in order to avoid the introduction of noise to the writer identification step.

In order to design a symbol detection methodology, we need to define two stages. A first stage should learn to distinguish among the target symbol and the background (e.g. learning a binary classifier). A second stage should perform a search over the whole image using the trained classifier in order to locate those regions containing the target symbol.

4.1 Training Stage

For the first step, we propose to learn a hierarchical cascade of 2 classifiers with a set of positive and negative symbol instances (see Fig.1 for some examples), manually extracted from a set of music scores. Initially, the set of positive clefs and the set of positive notes samples consists in clefs and notes extracted from the music scores, whereas the negative examples are basically, examples of other music symbols (e.g. accidentals, rests). In the training stage, the suitable parameters of the BSM and the DTW-based descriptors are found, and the set of negative examples can be modified. First, different grid sizes and the rejection threshold for the BSM descriptor are tested until a minimum accuracy is achieved. Then, the set of negative examples is modified, adding the images of the false detections found. Secondly, different number of features (the number of regions) and the rejection threshold for the DTW-based symbol recognition method are tested. Finally, the set of false detections can be also increased by adding the images of the false detections found. This strategy is detailed in Algorithm 1.



(a)

(b)

Figure 1. Some examples of images used in the training step. (a) Positive images of clefs. (b) Positive images of music notes.

Algorithm 1 Symbol Detection Training algorithm for the cascade of two classifiers.

Require: A set of positive examples P and a set of negative examples N, a maximum false alarm rate f, a minimum accuracy a.

- $F_i \leftarrow 1, n_i \leftarrow 0$
- **while** $F_i > f$ **do**
  - $n_i \leftarrow n_i + 1$
  - Use P and N to train a classifier using the BSM descriptor with n_i as the grid size
  - $F_i \leftarrow$ Evaluate current classifier on validation set
  - Decrease threshold for the *i*th classifier until the current cascaded classifier satisfies a detection rate of a (this also affects F_i)
- **end while**
- $N \leftarrow 0$
- Evaluate the BSM-based detector on the set of non-symbol images and put any false detections into the set N.
- $F_i \leftarrow 1, n_i \leftarrow 0$
- **while** $F_i > f$ **do**
  - $n_i \leftarrow n_i + 1$
  - Use P and N to train a classifier using the DTW-based method with n_i regions
  - $F_i \leftarrow$ Evaluate current classifier on validation set
  - Decrease threshold for the *i*th classifier until the current cascaded classifier satisfies a detection rate of a (this also affects F_i)
- **end while**
- Evaluate the DTW-based detector on the set of non-symbol images and put any false detections into the set N.

Ensure: A cascade h of the BSM and DTW-based classifiers for symbol detection.

4.2 Detection Stage

Once both classifiers are trained, the different elements must be segmented from the input image. For this purpose, a merging approach is used, which consists in applying a morphological dilate using disks of different sizes as the structuring element. Then, the connected components whose size and area are not under certain restrictions (no clef is smaller than the half of the staff length and bigger than twice the staff length) are removed. This step is used for discarding the too small or too big symbols, which consequently, are not music clefs nor notes. Afterwards, the BSM descriptor is computed for each remaining connected component, and compared with the BSM descriptors of the set of positive and negative examples. The comparison is performed using the Euclidean distance and the k-NN classifier. If the BSM-based classifier accepts the candidate connected component as a clef or note, then, the DTW-based features are computed for this region, and compared with the DTW-based features of the set of positive and negative examples. If the DTW-based classifier also accepts the candidate connected component, then the candidate region is accepted as a music clef or note. The method is described in Algorithm 2.

Algorithm 2 Symbol detection using a cascade of two classifiers.

Require: An image I, a cascade of classifiers h, an initial structuring disk element of size D_I, a final disk size D_F, and a disk increment i.

$R \cup 0$
Compute the BSM and DTW features of all the set of positive examples P and the negative examples N.
for each structuring element D of size D_I, increasing by i, to D_F **do**
    $ImDilated =$ dilation of I using the disk D
    for each connected component r in $ImDilated$ of accepted size and area **do**
        test cascade h over region r

$$h(r) \quad = \quad \begin{cases} 1 & \text{if} \quad \text{target detection, save region } R = R \cup r \\ 0 & \text{if} \quad \text{background classification} \end{cases}$$

    end for
end for
Remove from R the repeated instances of a same symbol.

Ensure: Target symbol regions R

In this way, only those regions that arrive to the last stage of the cascade are classified as clefs or notes, and the rest of the regions are rejected. Each stage analyzes only the candidates accepted by the previous stages, and thus, the non-clefs/notes are analyzed only until they are rejected by a stage. The BSM descriptor is used for the first classifier, because it is very fast to compute. Notice that when dilating the image with different disk sizes, several instances of a same symbol may be accepted. In this cases, only one instance of each symbol is stored.

5 Writer Identification Based on Clefs

Once we have the symbols extracted from each music sheet of the database, the classification in terms of the writer is performed. As stated in the Introduction, since the music clefs have higher discrimination power than notes, the identification will be initially based on the shape of clefs. In case the number of clefs in the music score is not enough for a reliable writer identification, we will use notes instead of clefs.

The writer identification based on clefs can be seen as a multi-class clef classification, in which all the clefs detected from each page must be assigned to the same writer. We propose a non-supervised approach, avoiding the definition of the clef for each writer in the database. Thus, the idea is to compare the detected clefs of the test music page with the clefs of the training database. For this purpose, the BSM descriptors previously computed are used to compute the distance between each clef (using the Euclidean distance and the k-NN classifier). The BSM features have been chosen because the segmented symbols usually have important noise and gaps (the DTW-based features are more sensitive to discontinuities than the BSM features).

Then, the combination of the classification results of all the clefs (belonging the same music sheet) is performed so that each clef gives votes to the writer class of its nearest neighbor symbols of the training (see Fig.2). This process has the following steps. First, each test clef is compared to the clefs of the training set using the k-NN classifier. For each clef, a list of the k nearest neighbor clefs is obtained, and sorted so that the first candidate is the nearest neighbour of all. Then, the first ranked clef adds k votes to its corresponding class, the second nearest neighbor clef gives $k - 1$ votes to its class, and this process is repeated until the last candidate adds one vote to its corresponding class. After the voting performed for each clef belonging to the music page, the test music score will be classified as the class which has received the maximum number of votes.

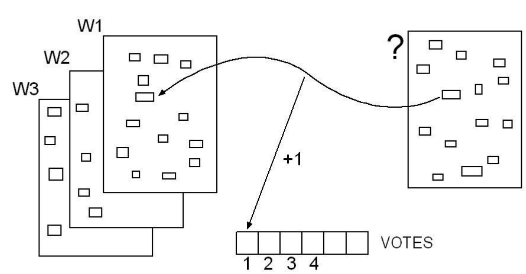


Figure 2. The identification of the input test music sheet is based on the votes from every clef detected when compared to the clefs in the training set.

It must be said that if an input clef has no nearest neighbors in the database (the distance to all the BSM descriptors is higher than the value set in the training step), then, it is discarded, and consequently, it can not vote. In this way, the symbols that could be wrongly accepted as clefs (false positives), could be detected, and consequently, rejected from the voting stage.

6 Results

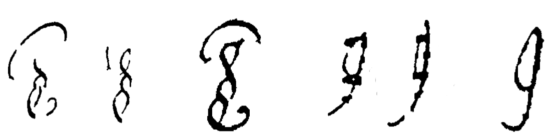
We have tested our method in a data set composed of 200 music sheets, consisting of 10 pages for each one of 20 different writers. They have been obtained from a collection of music scores of the 17th, 18th and 19th centuries, from the spanish archives of Seminar of Barcelona and Canet de Mar.

After the preprocessing of each music sheet, the symbol detection technique above described has been applied to extract the music clefs and notes. In the learning stage of the detection process, the parameters for the BSM and the DTW-based method have been trained. As a result, the grid size for the BSM descriptor has been set to 25, and 7 features are used (the upper and lower profile, and 5 zones) for the DTW-based method.

6.1 Detection Results

Concerning the evaluation of the symbol detection step, the database has a total of 733 music clefs, and 592 clefs has been correctly detected, 141 clefs have been missed (false negatives), and 697 regions have been wrongly detected (false positives). Thus, the detection rate is 80.8% $(592/733)$, the false positive rate is 54% $(697/1292)$ and the false negative rate is 19.2% $(141/733)$. We can affirm that although the detection rate is acceptable, there is an important rate of missed clefs and false positives. It must be noticed that in the classification step, the most part of these false positives will find no nearest neighbors, and consequently, they will not be allowed to participate in the voting.

After examining the missed clefs, we can see that most of them are the result of a bad segmentation, with important noise and gaps. These segmentation problems are due to the binarization and staff removal stages applied to very degraded documents. As an example, Figure 3(a) shows two badly segmented clefs (with gaps) and the corresponding manually segmented clef (Fig.3(b)); and Figure 3(c) shows two badly segmented clefs (with noise from the staff lines) and the corresponding manually segmented clef (Fig.3(d)).



(a) (b) (c) (d)

Figure 3. Examples of segmented clefs. (a),(c) Segmented clefs and their corresponding ideal segmented clef (b),(d).

6.2 Classification Results

For the writer identification experiments, we have used five independent test subsets, randomly chosen, containing one page per writer. For the BSM descriptor, the grid size with value 25 has been used. For each test subset or 20 images, the remaining 180 images are used for training. The classification has been performed using a k-Nearest Neighbor (with $k = 5$) classifier based on Euclidean distance and cross validation, with the voting step previously described. Since there are four writers whose music sheets contain no clefs (see Fig.4) or they only have one clef for each page (not being enough for a good classification), we have decided to use the information about notes for the identification of these four writers. For the rest of the 16 writers, only information about music clefs has been used for the identification.

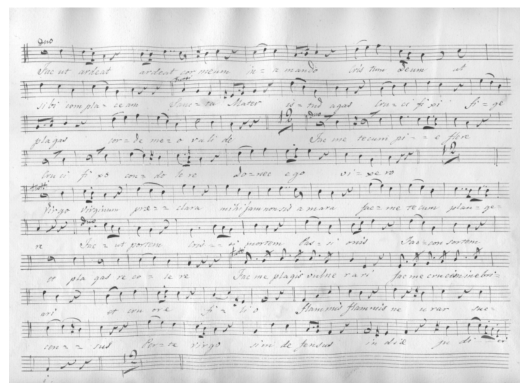


Figure 4. A music score without any clef.

We have compared the proposed method with two *symbol-independent* writer identification (W.I.) methods. The first one is based on the extraction of 100 typical features for handwritten line text recognition [6], whereas the second one is based on the extraction of textural features [7]. Figure 5 shows the W.I. rates of the three approaches for different database sizes.

The comparison of the results of the two *symbol-independent* approaches shows that both approaches reach similar performance (73% and 76% of writer identication rate for 20 writers). Our *symbol-dependent* W.I. approach significantly outperforms the others, with a W.I.rate of 93%, also showing a good scalability degree (from 96% with 5 writers to 93% with 20 writers). In this sense, the textural approach decreases significantly when adding more writers to the database (from 92% with 5 writers to 73% with 20 writers).

It must be noticed that *symbol-independent* approaches are very robust, avoiding the dependence on a good recognizer. Contrary, the performance of a *symbol-dependent* writer identification method is closely related to the performance of the detection and segmentation of symbols. Consequently, a more accurate symbol-detection technique, will obviously increase the final writer identification rate, and vice-versa.

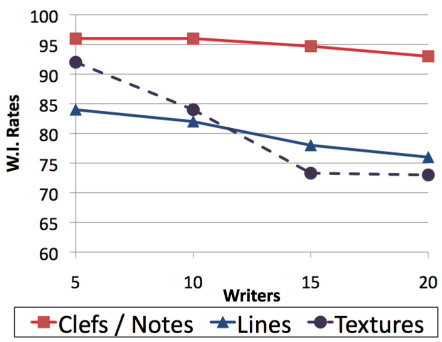


Figure 5. Writer identification rates of the three W.I. approaches for an increasing number of writers.

7 Conclusions

In this paper we have proposed a *symbol-dependent* writer identification method based on the shape or music clefs and notes. It has been performed using a cascade of two classifiers for saving computational cost time. The classifiers are based on the computation of the BSM descriptor and the DTW-based features. After detecting and segmenting the clefs and notes, the classification is performed using a non-supervised approach, in which the clefs (or notes) belonging to the test music pages are compared to the clefs (or notes) from the training music sheets.

Experimental results show that although there is an important amount of false positives in the segmentation, the retrieval of symbols is enough accurate for the writer identification method. Results show a very high writer identification rate (93%) in a database of 20 writers and 200 music pages. Although the method should be applied to a bigger database, the promising results show that this method has a very high discriminatory power.

Acknowledgements

The authors would like to thank Prof. Josep Maria Gregori Cifré from Art Department of UAB for his help in accessing to old resources of the archives of Seminar of Barcelona and Canet de Mar (in Catalonia, Spain). This work has been partially supported by the Spanish projects TIN2008-04998, TIN2009-14633-C03-03 and CONSOLIDER - INGENIO 2010 (CSD2007-00018).

References

[1] I. Bar-Yosef, I. Beckman, K. Kedem, and I. Dinstein. Binarization, character extraction, and writer identification of historical Hebrew calligraphy documents. *International Journal on Document Analysis and Recognition*, 9(2):89–99, 2007.

[2] I. Bruder, T. Ignatova, and L. Milewski. Integrating knowledge components for writer identification in a digital archive of historical music scores. In *Proceedings of the 4th ACM/IEEE-CS Joint Conference on Digital libraries (JCDL)*, pages 397–397, New York, NY, USA, 2004. ACM.

[3] M. Bulacu and L. Schomaker. Text-independent writer identification and verification using textural and allographic features. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(4):701–717, 2007.

[4] S. Escalera, A. Fornés, O. Pujol, P. Radeva, G. Sánchez, and J. Lladós. Blurred Shape Model for binary and grey-level symbol recognition. *Pattern Recognition Letters*, 30(15):1424–1433, 2009.

[5] M. Fonseca and J. Jorge. Towards content-based retrieval of technical drawings through high-dimensional indexing. *Computers & Graphics*, 27(1):61–69, 2003.

[6] A. Fornés, J. Lladós, G. Sánchez, and H. Bunke. Writer identification in old handwritten music scores. In *8th International Workshop on Document Analysis Systems*, pages 347–353, Nara, Japan, September 2008.

[7] A. Fornés, J. Lladós, G. Sánchez, and H. Bunke. On the use of textural features for writer identification in old handwritten music scores. In *Document Analysis and Recognition (ICDAR). Tenth International Conference on*, volume 2, pages 996–1000, July 2009.

[8] A. Fornés, J. Lladós, G. Sánchez, and D. Karatzas. Rotation invariant hand drawn symbol recognition based on a dynamic time warping model. *International Journal on Document Analysis and Recognition*, preprint(DOI: 10.1007/s10032-010-0114-8), 2010.

[9] J. B. Kruskal and M. Liberman. The symmetric time-warping problem: From continuous to discrete. In D. Sankoff and J. B. Kruskal, editors, *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*, pages 125–161, Reading, Massachusetts, September 1983. Addison-Wesley Publishing Co.

[10] J. Lladós, E. Valveny, G. Sánchez, and E. Martí. Symbol Recognition: Current Advances and Perspectives. In *Lecture Notes in Computer Science, vol. 2390*, pages 104–128. Springer, 2002.

[11] W. Niblack. *An Introduction to Digital Image Processing*. Prentice Hall, 1986.

[12] M. Rusiñol, A. Borràs, and J. Lladós. Relational indexing of vectorial primitives for symbol spotting in line-drawing images. *Pattern Recognition Letters*, 31(3):188–201, 2010.

[13] D. Zuwala and S. Tabbone. A Method for Symbol Spotting in Graphical Documents. *Lecture Notes in Computer Science*, 3872:518–528, 2006.