Integration of Shape Context and Neural Networks for Symbol Recognition

Frank Julca-Aguilar, Christian Viard-Gaudin, Harold Mouchère, Sofiane Medjkoune, Nina S. T. Hirata

HAL Id: hal-01150830

https://hal.science/hal-01150830

Submitted on 11 May 2015

Integration of Shape Context and Neural Networks for Symbol Recognition

Frank Julca–Aguilar* — **Christian Viard–Gaudin**** — **Harold Mouchère**** — **Sofiane Medjkoune**** — **Nina Hirata***

* *Department of Computer Science, Institute of Mathematics and Statistics, University of São Paulo, São Paulo, Brazil.*
** *LUNAM Université, Université de Nantes, IRCCyN UMR CNRS 6597 Polytechnique Nantes rue Christian Pauc – BP 50609 – 44306, Nantes, France.*

ABSTRACT. *Using shape matching within a k–nearest neighbor approach, shape context descriptor has been applied in several classification problems with outstanding results. However, the application of this framework on large datasets or online scenarios is challenging due to its computational cost. To overcome this limitations, we evaluate the use of shape context as input features for neural networks. We test the proposed method in a problem of recognition of handwritten mathematical symbols. For a total of 75 classes of symbols, we obtained a recognition rate of 89.8%, comparable with a k–nearest neighbor approach, but with reduced time complexity.*

RÉSUMÉ. *Les descripteurs de contexte de formes ont été utilisés comme caractéristiques dans les classifieurs k–plus proches voisins avec des résultats remarquables. Néanmoins, l'utilisation de cette approche sur de grosses bases de symboles ou dans des contextes applicatifs à la volée reste difficile à cause de sa complexité calculatoire. Pour dépasser ces limitations, nous proposons l'utilisation des descripteurs de contexte de formes avec des réseaux de neurones au lieu de l'approche k–ppv. Nous évaluons la méthode proposée dans un problème de reconnaissances de symboles mathématiques en–ligne. Pour un total de 75 classes de symboles, un taux de reconnaissance est de 89.8% est obtenu, ce qui est comparable aux résultats de l'approche initiale mais avec une complexité nettement diminuée.*

KEYWORDS: *shape context, artificial neural networks, symbol classification.*

MOTS-CLÉS : *descripteurs de contexte, réseaux de neurones, reconnaissance de symboles*

1. Introduction

Shape context descriptor has been applied in several Computer Vision classification problems with outstanding accuracy. In (Belongie *et al.*, 2002), the authors evaluated shape context on the recognition of handwritten digits – using the MNIST dataset (Lecun *et al.*, 1998) – and 3D objects – using the COIL-20 dataset (Murase and Nayar, 1995) – and obtained around 99.4% of accuracy on both problems. In (Prasanth *et al.*, 2007) shape context was used in a recognition problem that involved a large number of classes: using a handwritten Tamil scripts dataset with 156 classes of symbols, results of 79% top 1 and 92% top 2 accuracy were reported. More recently, the work in (Wang *et al.*, 2011) combined shape context and SIFT descriptors for leaf image classification (220 leaf species) and obtained 91% of accuracy.

Shape context is generally used within a k–nearest neighbor approach. However, the k–nearest neighbor approach does not scale well on large databases, due to the computational load to find the nearest neighbor(s) and the need to store the class models (Hastie *et al.*, 2009). For instance, results in (Prasanth *et al.*, 2007) reported a recognition time of 240 seconds per symbol. This also makes online applications a challenging scenario for this method, because of the need to immediate system responses.

Symbol classification is a key step in the problem of online recognition of handwritten mathematical expressions. This problem presents two issues that make the application of shape context difficult: (1) large quantity of (symbol) classes, which makes essential the use of large datasets, and (2) the need for efficient classification methods to allow immediate recognition results for the input expressions.

To overcome the computational cost of the k–nearest neighbor approach, we evaluate the use of shape context descriptors as input features for a neural network classifier. We test the proposed approach on the Competition on Recognition of On-line Mathematical Expressions (CROHME 2012) dataset (Mouchere *et al.*, 2012), that is composed of 75 symbol classes. The proposed method achieved 89.8% of correct recognition rate, being comparable to the 90.16% by the nearest neighbor approach. Further, while the nearest neighbor implementation has a recognition rate of 15 seconds per symbol, the corresponding neural network performance is around 0.001 seconds per symbol, being an efficient way to introduce shape context descriptor in online recognition scenarios.

The rest of the paper is organized as follows: we discuss shape context calculation and issues in k–nearest neighbor computation in Section 2. In Section 3, we describe the neural network structure and integration with shape context. The experimental setup is described in Section 4 and results are reported and discussed in Section 5. Finally, conclusions and further work are presented in Section 6.

2. Shape Context

Shape context was originally proposed as a shape descriptor in (Belongie *et al.*, 2002). This descriptor can be calculated from offline or online data. In the first case, the input is an image and an edge extraction process must be previously applied to determine a set of points that represent the object shape. In the second case, the object shape is directly captured by an input device and a software that samples the written strokes. In this process, time information can also be captured by the device. As sampling of points may not be homogeneous (quick writing may cause hardware to sample less points in some parts of the symbols), before shape context calculation, preprocessing techniques, such as smoothing or interpolation must be applied.

In this work, we focus on online data. In addition, without loss of generality, we will hereafter consider a symbol as a set of points.

2.1. *Shape context calculation*

Given a symbol $P = \{p_1, p_2, \ldots, p_n\}$, the shape context of a point p_i in P is a log polar histogram that expresses the distribution of the remaining points relative to p_i (Belongie *et al.*, 2002). Figure 1 illustrates shape context calculation for two points of a symbol. In that example, the symbol area is divided into 8 angular regions and 3 radial regions, for a total of 24 bins, where a bin contains the quantity of points spatially placed in that bin.

Given two symbols $P = \{p_1, \ldots, p_n\}$ and $Q = \{q_1, \ldots, q_n\}$, the similarity measure, or matching cost, between shape contexts of a point p_i in P and q_j in Q is given by the χ^2 test statistic (Belongie *et al.*, 2002):

$$C_{ij} = \frac{1}{2} \sum_{x=1}^{X} \frac{[hp_i(x) - hq_j(x)]^2}{hp_i(x) + hq_j(x)} \qquad [1]$$

where X is the number of bins, and $hp_i(x)$ and $hq_j(x)$ denote the histogram value of p_i and q_j at bin "x". If both $hp_i(x)$ and $hq_j(x)$ are empty, the term is not considered in the summation.

By using the above measure, the total cost of a one-to-one matching of the points in P with points in Q is given by the sum of the cost of all matching pairs. To calculate a similarity measure between P and Q, we look for a matching that minimizes the total matching cost. To find a matching with minimum cost, for each pair of points p_i and q_j, a χ^2 cost is calculated. The calculated costs are stored in a matrix of dimension n x n. Then, we use the Hungarian algorithm (Kuhn, 1955), with the matrix of costs as input, to determine a matching with minimum cost.

With such an approach, the time order of the points is not kept, and only the best offline matching guides the pairing process. This can be seen as a way to circumvent
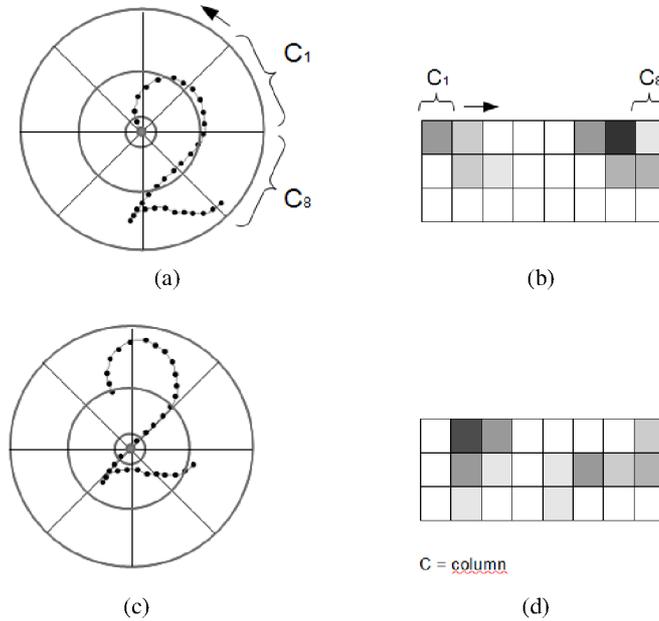
Figure 1. *Shape context of two points of a symbol "2": (a) and (c) show the sampled points and log polar histogram bins used to calculate shape context. (b) and (d) show the shape context histogram relative to (a) and (c) respectively. Dark cells mean higher values.*

the writing variability producing the same pattern with different starting and ending points, when writing a "0", for instance, or when changing the stroke order.

2.2. *k–nearest neighbor classification using shape contexts*

Most shape context applications use a k–nearest neighbor approach for classification. Two drawbacks of the this method are the computational load to find the nearest neighbor(s) and store the class prototypes (Hastie *et al.*, 2009).

In the first issue, given a query, classification implies a comparison of the query with all prototypes, to select the most "similar" prototype(s). By using the Hungarian algorithm, the cost to determine the similarity between an input symbol and a prototype takes $O(n^3)$(where n is the number of sampled points). So, the cost of classifying an input symbol using m prototypes is $m * O(n^3)$.

On the second issue, the computational space to store prototypes is directly related to the number of classes and variability of each class – for example, we can expect that number four presents more variability than zeros, so we could need more proto-

types for fours than those for zeros (Belongie *et al.*, 2002). Although several methods have been proposed to reduce the set of prototypes without reducing accuracy, the scalability of the method is still a weak point (Hastie *et al.*, 2009).

In (Mori *et al.*, 2005) the authors proposed a modified version of the shape context in order to more efficiently search for the best prototypes. The method is divided into two steps: fast pruning of candidates by means of a *light* version shape context comparison, followed by a selection of the best prototype (from the remaining prototypes) using a richer, but more expensive, shape context version. Although this method improves the search of best candidates, it does not cope with the problem of storing prototypes; the scalability of the method to problems with high number of classes is still challenging.

3. Shape contexts as input features for Neural Networks

By using shape contexts as input features for neural networks, we loose the explicit similarity measure on which shape context classification is based on. In some sense, it is up to the neural network to figure out which histogram is similar to which one. Roughly speaking, shape contexts features of a symbol can be seen as a set of histograms calculated at each sampled point of the symbol; but with the particularity that histograms are defined over a log polar space, which makes shape context of a point more sensitive to positions of nearby sample points (local features) than those of points farther away (global features) (Belongie *et al.*, 2002).

As input for neural networks, shape contexts must follow an order relative to the units of an input layer. Unfortunately, the order in which strokes are introduced or writing direction may change the order of shape contexts. This problem may be solved, at some extent, using enough training data to reflect different writing variations, so that it is possible for neural networks to learn these kinds of variations.

Considering reported applications (Belongie *et al.*, 2002; Prasanth *et al.*, 2007; Mori *et al.*, 2005), we can see that the number of sampled points and bins for shape context calculation vary around 30 and 40, respectively. Using such parameters, the number of input features for neural networks could reach values of 1200 features.

To reduce the number of input features, we maintain the rate of sample points (30 points per symbol), but extract shape contexts of only a subset of points. For instance, with the parameters mentioned above, we could calculate only the shape context of points at positions 1, 15 and 30, and get a total of 120 features (3 shape contexts, each containing 40 bins). This method is based on the fact that shape contexts of near points of a same symbol may be similar, thus, it allows us to reduce redundancy on features. Section 4 describes the evaluated reduced versions of shape contexts.

4. Experimental setup

As mentioned in Section 1, to evaluate the proposed methods, we used the CROHME 2012 dataset (Mouchere *et al.*, 2012). [1] The dataset consists of 1,824 mathematical expressions with a total of 23,375 symbols and 75 symbol classes. The expressions were handwritten by 347 people from France, India and Korea; using several kinds of input devices as digital pen technologies, white-boards and tablets with sensible screens. Given this variety of devices, symbols were sampled in different scales and resolutions.

It is important to note that symbols which are considered here come from entire mathematical expressions. However, we are considering only the sub–problem of symbol recognition, since we use the ground–truth segmentation to extract the symbols from their expressions. Hence, a symbol might be influenced by its context. For example, symbols written before and after a symbol X may influence the way people write X. This may have introduced more variability on symbol classes.

To evaluate the neural network classifier, we divided the dataset into two sets: one with 1,336 expressions for training and the other with 488 expressions for test (as defined on Part-III dataset in (Mouchere *et al.*, 2012)). From the training set, we used 70% to train neural networks with different shape contexts configurations and the rest for validation.

In our shape context implementation for neural networks, we used 3 radial regions and 8 angular regions, for a total of 24 bins. We extracted 30 points per symbol and evaluated the recognition performance considering from 3 to 10 shape contexts per symbol. For the nearest neighbor based approach, we extracted 30 points per symbol. Shape contexts configuration was fixed to 4 radial regions and 10 angular regions, for a total of 40 bins. These values were fixed to obtain best recognition rates using the validation dataset part.

In addition to shape context features, we used raw online data as features in a neural network classifier described in (Awal, 2010). We compare neural network performance using both feature sets separately and jointly.

We used a Multilayer Perceptron neural network. For configuration, the input layer size was determined by the number of input features and the output layer size by the number of symbol classes (75). We used one hidden layer with 100 nodes. This configuration was determined using the validation dataset.

---

1. The CROHME 2012 dataset is publicly available at: `http://www.iapr-tc11.org/mediawiki/index.php/Datasets_List`.

5. Results and discussion

Table 1 shows neural network performance using different number of shape context features. The reported recognition rate was calculated using the validation dataset, after training on the train part.

Table 1. *Impact on Neural network performance of varying the number of shape context features.*

# s. contexts (# features)	(%) Recognition rate
3 (72)	84.55
4 (96)	86.1
5 (120)	87.8
6 (144)	88.32
7 (168)	89.05
8 (192)	89.47
9 (216)	89.14
10 (240)	89.22
30 (720)	89.9

Results show that by using more than 8 shape contexts, the recognition rate does not improve considerably. Thus, for combination with the online features, we used 8 shape contexts per symbol and trained neural networks for the three cases: only online features, only shape context and with both feature sets. This evaluation was also done using the train and validation parts. For evaluation on the test set we include the 1–nearest approach. Table 2 compares the recognition performance of the classifiers in the test set.

Table 2. *Recognition rate for nearest neighbor approach and neural networks (NN).*

Classifier	(%) Recognition rate
1-nearest neighbor	90.16
NN with online features	88.6
NN with shape context	87.8
NN with online features and shape context	89.8

We can see that the neural network approach using shape context and online features performed almost as good as the 1–nearest neighbor approach. On the other hand, there is a considerable difference in recognition time: 15 seconds per symbol with the 1–nearest neighbor versus 0.001 seconds per symbol with the neural network approach.

As mentioned in Section 3, a concern on the application of shape context as input features to a neural network was the variability of handwritten symbols. However,

the variety of training samples per symbol class seems to help the neural network to "learn" this variability.

With regard to the nature of the recognition task, it is important to note that some symbols can not be recognized by considering only shape information. For example, symbols "x", "X" and $\times$ ("\times" latex code) may be handwritten with an identical shape. To solve those ambiguities, context from the mathematical expressions must be used.

6. Conclusions and Further work

To overcome the expensive computational cost of the shape context matching based approach, we used shape context as input features for neural networks. To reduce the descriptor dimension, we extracted shape contexts from only a subset of points of a symbol, instead of extracting one shape context per each point. Results showed that this reduced version of shape contexts alone or combined with online features, performs almost as good as a 1–nearest neighbor approach and makes a considerable improvement in terms of time recognition.

The neural network approach showed to be a practical method to introduce shape context descriptor in an online recognition problem. Further work includes evaluation of the classifier in the context of recognition of online handwritten mathematical expressions. In this problem, symbols are not already segmented, and during the segmentation process wrong symbol hypothesis may be generated. Identification of those wrong segmented symbols is a main concern in this problem. In this regard, rich features as shape context may help to improve performance recognition.

7. References

Awal A.-M., Reconnaissance de structures bidimensionnelles : Application aux expressions mathématiques manuscrites en-ligne, PhD thesis, Image and VideCommunication, research group, IRCCyN, 2010.

Belongie S., Malik J., Puzicha J., "Shape Matching and Object Recognition Using Shape Contexts", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, p. 509-522, April, 2002.

Hastie T., Tibshirani R., Friedman J., *The Elements of Statistical Learning*, Springer Series in Statistics, Springer New York Inc., New York, NY, USA, 2009.

Kuhn H. W., "The Hungarian method for the assignment problem", *Naval Res. Logist. Quart.*, vol. 2, p. 83-97, 1955.

Lecun Y., Bottou L., Bengio Y., Haffner P., "Gradient-based learning applied to document recognition", *Proceedings of the IEEE*, vol. 86, nº 11, p. 2278-2324, 1998.

Mori G., Belongie S., Malik J., "Efficient shape matching using shape contexts", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 27, nº 11, p. 1832-1837, 2005.

Mouchere H., Viard-Gaudin C., Kim D., Kim J., Garain U., "ICFHR 2012 Competition on Recognition of On-Line Mathematical Expressions (CROHME 2012)", *Frontiers in Handwriting Recognition (ICFHR), 2012 International Conference on*, p. 811-816, 2012.

Murase H., Nayar S. K., "Visual Learning and Recognition of 3-D Objects from Appearance", *Int. J. Comput. Vision*, vol. 14, nº 1, p. 5-24, January, 1995.

Prasanth L., Babu V., Sharma R., Rao G. V., M. D., "Elastic Matching of Online Handwritten Tamil and Telugu Scripts Using Local Features", *Proceedings of the Ninth International Conference on Document Analysis and Recognition - Volume 02*, p. 1028-1032, 2007.

Wang Z., Lu B., Chi Z., Feng D., "Leaf Image Classification with Shape Context and SIFT Descriptors", *Digital Image Computing Techniques and Applications (DICTA), 2011 International Conference on*, p. 650-654, 2011.