Reducing the Amount of Input Data in Traffic Sign Classification

Ana Granados, Agapito Ledezma, Germán Gutiérrez, and Araceli Sanchis

Universidad Carlos III de Madrid, Computer Science Department, Avenida de la Universidad nº 30, CP 28911, Leganés, Madrid, Spain
{agranado, ledezma, ggutierr, masm }@inf.uc3m.es

Abstract. Several complex problems have to be solved in order to build Intelligent Transport Systems. Among them, it is worth mentioning the detection and classification of traffic signs which could appear at any position within a captured image. This paper analyzes the influence of the number of attributes in the field of classification of traffic signs when automatic learning techniques are used. In order to face this task, four different approaches have been considered, three of them symbolic and one sub-symbolic. These techniques have been applied using two different input pattern dimensions and their performances have been compared.

1 Introduction

Detection and classification of traffic signs within a captured image are problems that have not yet been definitely solved in the Intelligent Transport Systems research area. Researchers in this area have mainly focused their efforts in other problems, such as border road detection [1, 2] and obstacle detection (i.e. pedestrians [3, 4], other vehicles [3, 5]). Some of these works have been used to develop automatic driver systems. However, designing new techniques and proposing new solutions to detect and classify traffic signs could provide new interesting strategies that could be applied to the implementation of automatic drivers or driver assistants with better performances.

Several strategies, like image processing algorithms or automatic learning models, have been applied to solve the traffic sign classification problem (see next section for more details and references). Most of them, use complete images as input patterns. When these images are too big, the amount of information to process increases considerably, and the computational cost becomes very high as well. Thus, reducing the amount of information could be interesting in order to simplify the problem and reduce both the training and testing times. This paper presents the results of applying four symbolic and sub-symbolic techniques. These results reflect the way in which the reduction of the number of input attributes affects the recognition rate.

A comparison between the recognition rates obtained for each technique was made. Each method was applied twice using two sets of patterns (one using all

the pixels in each image to generate the subsequent pattern, and the other using only a subset of these pixels) to study the influence of changing the amount of input information for each technique. In line with this, our main goal was to compare how the amount of information per pattern has an influence on the classification rate when each method is applied.

The paper is organized as follows. In section 2 several related works with image classification problems are briefly described. In section 3 all the developed tasks are described in low detail. In section 4 the chosen methods are explained, evaluated and compared in depth. Finally, in section 5 the conclusions and future work are summarized.

2 Related Work

This section reviews previous works in the traffic sign classification research area. Both symbolic and sub-symbolic methods are discussed.

Artificial Neural Networks (ANNs) are an instance of sub-symbolic methods. When working with ANNs to solve the image classification problem, usually the input to the network is a normalized image [6–9]. In other words, if we use 20x20 pixel images, the input layer of the network would have 400 neurons, one neuron for each pixel. Depending on the color system used, each input neuron would be fed with the numeric value of a pixel [6–9] or a component of the HSV color System [10].

Sometimes it is possible to use the ANN with different types of information, e.g. using a set of extracted features based on Fast Fourier Transform (FFT). These acquired data can be used to feed the network [11, 12]. Actually even ANNs themselves have been used to extract features of the image [12, 13]. Several kinds of algorithms, such as BackPropagation [6, 7, 11, 12, 14], RPROP [9], or ART2 [12], have been used to train the network.

In this paper we propose an ANN as an instance of a sub-symbolic method to solve the traffic sign classification problem. The difference between previous approaches in this field and ours is that the amount of information per image required to classify it is much lower when our method is applied, and therefore the computational cost is also reduced.

Decision Trees (DT) are symbolic techniques. DT can be used to tackle the image classification problem as well. Marée et al. [15] suggest a solution based on DT ensembles and local sub-windows for image classification. Geurts [16] shows that extremely randomized trees can be applied on pixel values to solve the image classification problem.

Classical methods of image recognition have been used in general fields [17]. Among them, the following ones are worth mentioning: color analysis, border detection, color regions detection, color thresholding, image normalization, filters application correlation, etc.

Classical approaches have two main important problems when applied to our domain. Firstly, they depend on the environmental conditions, and they do not work well when these conditions change. For example, if the image is too dark, or if it is out of focus. Secondly, their computational cost is too high, so it becomes impossible to apply them to build a real time classification system.

Our approach attempts to overcome these problems in order to lay the foundations for a future real time system, which should be change-tolerant, by reducing the amount of input information to all the used techniques.

3 Our approach

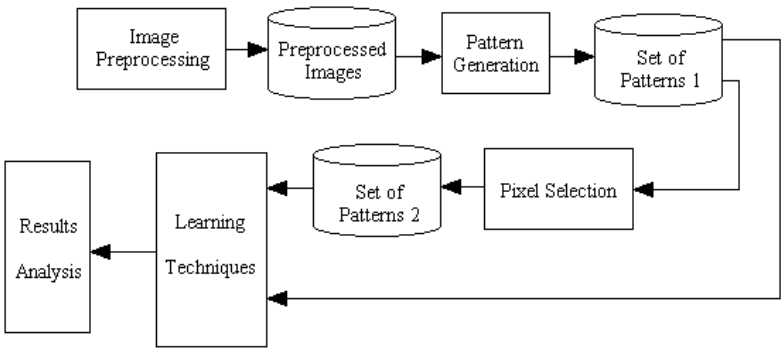


Fig. 1. General view of our approach.

Before applying symbolic and sub-symbolic learning techniques, an image preprocessing phase was required. This phase consisted of an initial color thresholding, a border detection, and a subsequent normalization. In this paper, this specific issue is not discussed, but detailed information about it can be found in [17].

Once the preprocessed and normalized gray intensity level images had been created, two patterns per image were generated:

a. A pattern with an attribute for each of the pixels in the image (see *Set of Patterns 1* in Figure 1).

b. Another pattern with an attribute for each of those pixels selected by the attribute selection algorithm (see *Set of Patterns 2* in Figure 1).

After this, four different learning techniques were used to classify traffic signs. All of them were executed twice, using the patterns corresponding to the complete image first, and using the patterns generated selecting only a few pixels in the image later. Finally, all the obtained results were contrasted to determine the influence of the amount of information used as input. Figure 1 shows a schematic view of the whole process.

4 Experimental Setup and Results

A set of real urban outdoor images was used to generate preprocessed images by applying image preprocessing methods based on color and shape analysis over the collected images. We used 254 preprocessed and normalized gray intensity level images of 32x32 pixels. In order to carry out the cross validation process, subsets of 229 and 25 were considered. In this first approach, the learning task was to classify traffic signs in two classes, *prohibition* and *danger*. We used, on the one hand, three symbolic methods (a DT generator algorithm: C4.5[18], a rule generator algorithm: Part[19] and an ensemble of classifiers generator: AdaBoost with C4.5 as boost learning algorithm[20]). On the other hand, we used an ANN as a representative of sub-symbolic methods to classify the traffic signs. In order to select the most representative attributes we applied an attribute selection approach: the Best First algorithm[21]. SNNS tool [22] was used to work with ANNs, while WEKA[23] tool was applied to work with Best First, C4.5, AdaBoost and Part.

In the first experimental phase, the ANN was a feed forward network [24] with 32x32 input nodes, two hidden layers (15 neurones in first hidden layer and 5 neurones in second hidden layer), and an output layer with 3 output neurones. Each output neuron represented a *prohibition* sign, a *danger* sign, and *no sign* respectively. The ANN architecture was chosen using the results and conclusions obtained from [7]. The BackPropagation algorithm [24] was used to train the network. A recognition rate of 98,8% was obtained in this experimental phase.

The C4.5 algorithm was used to build a DT using the same patterns as in the training phase of the ANN. It can be observed that the built DT only used a few pixels to classify the image (see Figure 2). This result shows that only a small amount of pixels in the image are required to classify the traffic signs, obtaining a good recognition rate (88,19%). Later on, an attribute selection algorithm would be applied to reduce the amount of input information of each pattern.

In addition, a rule generator algorithm (Part) and an ensemble of classifiers (AdaBoost) were applied to solve the traffic sign classification problem and to

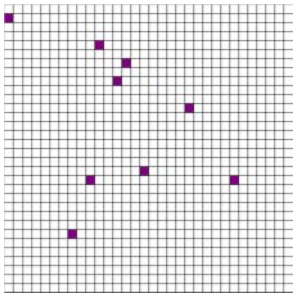


Fig. 2. Pixels used by C4.5 to classify 32x32 images.

study the number of pixels used by each method to classify the images. The resulting recognition rates were 94,49% and 91,34% respectively.

Before the second phase of our approach, we used the attribute selection algorithm Best First to select the most significant attributes. We obtained 36 attributes when this algorithm was applied. As a result, the amount of information to process was significatively reduced. Figure 3 shows the selected pixels.

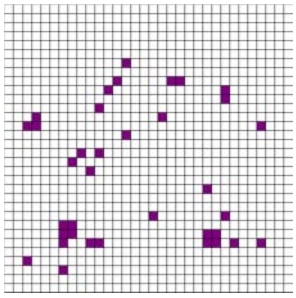
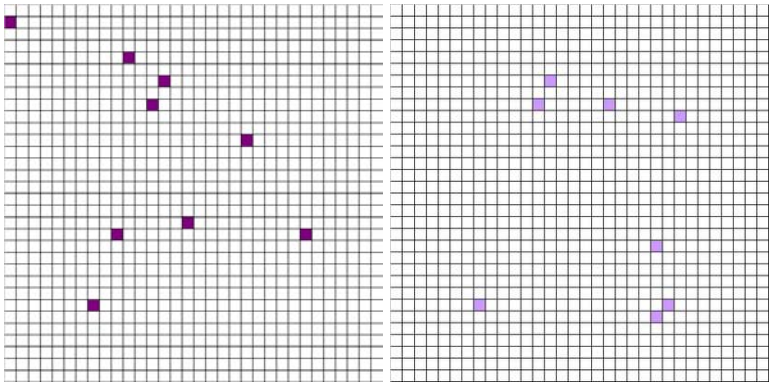


Fig. 3. Pixels selected using the Best First algorithm in a 32x32 image.

Once the pixels had been selected, a new set of patterns was generated using only 36 pixels of each image to create the subsequent pattern. We applied the same four techniques mentioned above to this new reduced patterns to solve the classification problem. We wanted to check whether or not getting good results using only a few pixels was feasible.

The ANN used for the first experiment was designed according to this architecture: 36/15/5/3. The number of input nodes was reduced because of the fact that now each input pattern consisted only of 36 attributes. The training algorithm used was BackPropagation again. A good recognition rate was achieved

(94%), when pixel selection was performed, but we observed that it might be possible to face the same classification task using a smaller ANN. Therefore, a second kind of experiments were carried out applying a second and smaller ANN, with an architecture of 36/15/3. The input to this network consisted of the same 36 pixels chosen by the Best First algorithm, but the second hidden layer had been removed. Our hypothesis was confirmed because the obtained recognition rate was the same when both ANN architectures (36/15/5/3 and 36/15/3) were applied. Furthermore, we could conclude that when the number of input attributes is reduced, and we only classify traffic signs in two classes, the recognition rate is still good, although it is a little lower than the one obtained when all the pixels are used as input attributes.

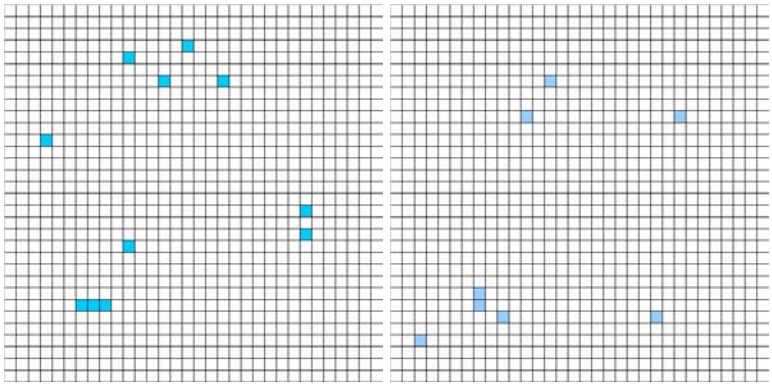


(a) (b)

Fig. 4. Pixels used by c4.5 algorithm to build the DT when 1024 attributes are used as input (a) or when only 36 attributes are used (b)

The C4.5 and Part algorithms were then applied to solve the traffic sign classification problem, using the second set of patterns as input. The recognition rate was higher in both cases than the one previously obtained. The recognition rates were 91,34% and 93,31 when C4.5 and Part were applied respectively. Figure 4 shows the selected pixels in both tests phases, using both sets of patterns to build the DT by executing the C4.5 algorithm. Figure 5 shows the pixels used by Part method in both cases as well.

The 36 selected pixels were also used as input to the AdaBoost method. The recognition rate remained the same as in the 1024 attribute patterns (94,49%). Figure 6 shows the pixels used by the AdaBoost method in both test phases. It can be observed that the number of pixels used by the algorithm increased from 9 to 28 when only 36 attributes were used as input.



(a) (b)

Fig. 5. Pixels used by Part algorithm to extract the rules when 1024 attributes are used as input (a) or when only 36 attributes are used (b)

To sum up, several experiments were carried out to empirically test the behavior of the classification methods under comparison. The experiments showed the performances and the comparison, when several techniques, C4.5, AdaBoost, Part and ANNs, were used to implement the approach.

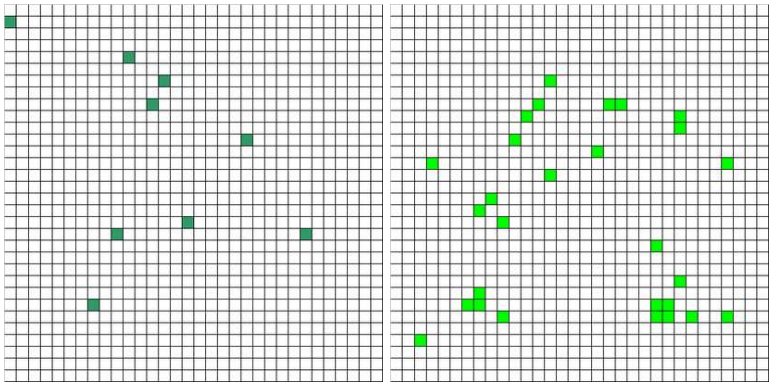
Table 1 shows the results of applying the four different algorithms to solve the classification problem. There are two results for each method. One corresponds to the experiments carried out using 1024 attributes (the normalized gray intensity levels of each pixel in the 32x32 image), and the other to the tests made using only 36 attributes (those selected by the Best First algorithm).

Table 1. Recognition rate results for all the methods.

Input attributes	ANN	C4.5	AdaBoost	Part
All the pixels in the image	98,8 %	88,19 %	94,49 %	91,34 %
Selection of 36 pixels	94 %	91,34 %	94,49 %	93,31 %

The recognition rate was very high (98,8%) when a complete gray level image was used to feed the ANN. However, when only 36 pixels per image were taken into account for classification purposes, the recognition rate was not so high (94%). In spite of this, the amount of information and the computational cost were much lower. This is a highly desirable aspect that should be stressed if a future real time classifier were to be developed.

Nine pixels were used as nodes of the DT when the first set of patterns was used as input, and 8 when the second set was used (see Figure 4). The time



(a) (b)

Fig. 6. Pixels used by Adaboost algorithm when 1024 attributes are used as input (a) or when only 36 attributes are used (b)

taken to build the model was 2,94 seconds and 0,25 seconds respectively.

Four rules were generated by the Part algorithm in both experiments. However, the time taken to build the model was 4,81 seconds when the first set of patterns was used and 0,1 seconds when the second set was used. The model used the value of 11 and 8 pixels respectively to classify the images using the rules generated by the Part algorithm (see Figure 5).

The recognition rate remained constant when the AdaBoost method was applied, independently of the set of input patterns used. When the first set of patterns (1024 attributes per pattern) was used, the number of pixels used by AdaBoost was 9. When the second set of patterns (36 attributes per pattern) was used, AdaBoost used 28 pixels (see Figure 6). The time taken to build the model decreased from 3,18 seconds to 0,93 seconds.

5 Conclusions

Four techniques aimed at classifying traffic signs based on ANNs, C4.5, AdaBoost and Part have been tested and compared, using two kind of input patterns. The initial objective was to study how the number of attributes for each pattern has an influence on the performances.

Very high recognition rates have been obtained when an attribute selection algorithm was applied to the pixels of the image in order to select the most significant ones and generate the set of reduced patterns. It has been observed that the recognition rate decreases when the number of input attributes is reduced and a sub-symbolic technique is applied. As opposed to this, the rate increases

when symbolic techniques are used in two cases (C4.5 and Part), and remains constant in the other one (AdaBoost).

This paper lays the foundations to implement a new traffic sign classification method that will be applied to classify traffic signs in urban scenes. In the future we will apply its results with the purpose of developing a traffic sign classifier which should be able to classify traffic sings into a larger number of classes.

Acknowledgements

The research reported here was carried out as a part of the research project CICYT TRA2004-07441-C03-01.

References

1. E.D. Dickmanns and A. Zapp. A curvature-based scheme for improving road vehicle guidance by computer vision. In *Proc. SPIE conference on Mobile Robots.*
2. D. Pommerleau and T. Jochem. Rapidly adapting machine vision for automated vehicle sterring. In *IEEE Expert.*
3. U. Franke, D. Gavrila, S. Gorzig, F. Lindner, F. Paetzold, and C. Wohler. Autonomous driving approaches downtown. In *IEEE Intelligent systems.*
4. U. Handmann, T. Kalinke, C. Tzomakas, and M. Werner. A image processing system for driver assistance. In *Image and Vision Computing.*
5. M. Bertozzi and A. Broggi. Gold: a parallel real-time stereo vision system for generic obstacle and lane detection. In *IEEE Transactions on Image Processing.*
6. Y. Aoyagi and T. Asakura. A study on traffic sign recognition in scene image using genetic algorithms and neural networks. In *IEEE IECON, 22nd Int. Conf. on Industrial Electronics, Control, and Instrumentation.*
7. A. de la Escalera, L. Moreno, M.A. Salichs, and J. M^{a}. Armingol. Road traffic sign detection and classification. In *IEEE Transaction on Industrial Electronics.*
8. Dan Ghica, Si Wei Lu, and Xiaobu Yuan. Recognition of traffic signs by artificial neural network. In *IEEE International Conference on Neural Networks.*
9. Lutz Priese, Raimund Lakmann, and Volker Rehrmann. Ideogram identification in a realtime traffic sign recognition system. In *Intelligent Vehicles Symposium.*
10. S. Vitabile and F. Sorbello. Pictogram road signs detection and understanding in outdoor scenes. In *SPIE Conference on Enhanced and Synthetic Vision.*
11. Phil Douville. Real-time classification of traffic signs. In *Real-time Imaging.*
12. N. Kehtarnavaz and A. Ahmad. Traffic sign recognition in noisy outdoor scenes. In *Intelligent Vehicles Symposium.*
13. Chiung-Yao Fang, Sei-Wang Chen, and Chiou-Shann Fuh. Road-sign detection and tracking. In *IEEE Transactions on Vehicular Technology.*
14. D.S. Kang, N.C. Griswold, and N. Kehtarnavaz. An invariant traffic sign recognition system based on sequential color processing and geometrical transformation. In *Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation.*

15. Raphaël Marée, Pierre Geurts, Justus Piater, and Louis Wehenkel. A generic approach for image classification based on decision tree ensembles and local subwindows. In K.-S. Hong and Z. Zhang, editors, *Proceedings of the 6th Asian Conference on Computer Vision*, volume 2, pages 860–865. Asian Federation of Computer Vision Societies (AFCV), 2004.
16. P. Geurts. *Contributions to decision tree induction: bias/variance tradeoff and time series classification.* PhD thesis, University of Liège, Department of Electrical Engineering and Computer Science, May 2002.
17. *Digital Image Processing.* Addison-Wesley Publishing Company, 1993.
18. J.R. Quinlan. *C4.5: Programs for Machine Learning.* Morgan Kaufmann, San Mateo, CA, 1993.
19. E. Frank and I. Witten. Generating accurate rule sets without global optimization. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 144–151. Morgan Kaufmann, 1998.
20. Y. Freund and R. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In Springer-Verlag, editor, *Proceedings of the Second European Conference on Computational Learning Theory*, pages 23–37, 1995.
21. J. Pearl and R. Dechter. Generalized best-first search strategies and the optimality of a*. In *Journal of the ACM.*
22. A. Zell et al. *Stuttgart Neural Network Simulator. User Manual, Version 4.1.* Institute for Parallel and Distributed High Performance Systems (IPVR), University of Stuttgart.
23. *Data mining: practical machine learning tools and techniques with Java implementations.* 2000.
24. Simon Haykin. *Neural Networks. A Comprehensive Foundation.* Prentice Hall, Upper Saddle River, New Jersey 07458, second edition edition, 1999.