User-Defined Object Classifier based in a Neural Network with Optimal Selection of Visual Features using a Genetic Algorithm

Manuel J. Reyes G.

School of Engineering and Applied Science
Columbia University, New York, NY USA

INTRODUCTION

The object recognition problem consist in: given some knowledge of how a certain objects may appear, plus an image of a scene possibly containing those objects, report which objects are present in the scene and where [1].

The first step for the solution of this problem is to segment the image into regions that would presumably contain the desired objects.

Some previous works on object recognition use the hypothesis-verification paradigm. The generation of hypothesis is trivialized in the top-down case. If an object class has a stable property, they assign the object class as an interpretation for any image region that has that stable property; the main drawback of this approach is that the assignment has utility only in very restrictive scene domains, which subtract flexibility to the systems. [1][2]

In this project we implement a Neural Network as the object classifier-recognizer, the input of the neural network is a set of "useful" region features that would enable the neural network to classify a given region as a "object-container" or as a "non-object-container" of the desire object. The set of "useful" region features is selected from a standard feature vector using a genetic algorithm.

The genetic algorithm would eliminate the redundant and irrelevant features increasing the performance of the neural network both in speed and predictive accuracy.

We present in this paper the results in the detection of faces in an image. The set of images chosen is highly unrestricted, in order to show that this approach is very flexible. As we would see the proper selection of the features by the genetic algorithm increases the neural network accuracy in almost 20%.

FACTORS TO CONSIDER IN THE SOLUTION'S DESIGN AND RELATED WORK.

As we mentioned before, object recognition consist in classifying an image region as "container" or as "non-container" of a given object. The first stage of an object recognition system is the image segmentation module, which divides the image in homogeneous regions, which should correspond to the different objects in the image.

The actual pixel data of the region cannot be use as the input of the neural network because the size of the segmented regions is not fixed, it varies in a large range and the running time of the neural network would be significant because the number of input units could be big. So we need to represent the information contained in the region in a more compacted and controlled way, we resolve this problem calculating a feature vector using the region pixel data to represent the characteristics of the given region. An optimal subset of this feature vector would be used as the input of the neural network.

A previous work has used a neural network and a set of features from the image regions to recognize objects[3]. In this work they use a system of three layers of neural networks for the understanding of images. The classification of the objects is done using a set of classification rules (IF-THEN rules) that are encoded in the weights of a neural network using simple perceptron connections; this neural network corresponds to one of the mentioned layers, the applications of this system are also very constrained.

Like in the previous mentioned work, in most of the object recognizer systems, the system designer using a set of classification rules predefines a model of the object.

In our approach the user defines its own classifier training the neural network with a set of regions that the user has already classified as: “object-container or as “non-object-container”, this “user definition” approach has been used in the work[4], but there, an algorithm using a variation of the K-Nearest-Neighbor Classification was used as the classifier, and the selection of the features used is not given by their “goodness” in the classification process.

Feature selection plays a central role in the data analysis process since irrelevant features often degrade the performance of algorithms devoted to data characterization, rule extraction and construction of predictive models, both in speed and in predictive accuracy. Irrelevant and redundant features interfere with useful ones, so that most supervised learning algorithms fail to properly identify those features that are necessary to describe the target concept.[5]

A genetic algorithm does the selection of the optimal subset; the number of features in this optimal subset would be also the input units number of the neural networks, so the feature selection has also a repercussion in the neural network architecture, in order to assure the best architecture for a specific subset of features, the genetic algorithm also encodes the number of hidden units to be used. So the genetic algorithm determines the optimal subset and the architecture of the neural network.

The system could be trained to classify more than one object at the time, if this is the case the user should proportioned regions instances for each one of the objects to classify, this would change the number of outputs of the neural network

USE OF A NEURAL NETWORK AS AN OBJECT CLASSIFIER.

As we mentioned before the image is segmented into regions with salient homogeneous properties, such as color and texture [6]. The parameters in the segmentation process should be calculated in such a way that the objects to be classified would be contained within a single region.

Then a feature vector of all the regions present in the image is calculated. For this project 22 features were calculated: The first three correspond to the values of the region’s average color in the LUV space, the next six features are related to the shape of the region, such as: roundness, aspect ratio, etc; the following four features are related to the size of the region: area, perimeter, maximum line and minimum line, the next six correspond to texture features of the region such as coarseness and contrast, finally the last three are related with the spatial location of the region with respect to the whole image.

The user should determine which are the regions that contain the desired object, so all the regions in the image are classified as “object-container” or “no-object-container”, this information is used as the target concept for the neural network to be implemented as the classifier.

The information provided by the feature vector is used as the input of the neural network, but, as we mentioned before, because irrelevant and redundant features often interfere with useful ones, then the optimal subset is calculated and used as the input of the neural network.

The complete feature vector contains 22 features, the first 19 are related to the intrinsic characteristics of the region, the last three give information about the location of the region within the image, so for classification purposes only the first 19 features would be used, the remaining 3 features are use only to report where a recognized object is located.

USE OF A GENETIC ALGORITHM FOR OPTIMAL FEATURE SELECTION IN OBJECT CLASSIFIERS.

The selection of the optimal subset of features is done using a genetic algorithm, the subset is encoded in a bit string, where a (1) in a certain position of the string denotes the selection of the feature in the given position and a 0 denotes the absence of the correspondent feature.

The number of features selected would also determine the number of inputs of the neural network and then also would affect its architecture, so we also encoded in the bit string the number of hidden units of the neural network in the last three bits of the string. So the range of hidden units to be tested is from 1 (0000) to 16 (1111).

An example of the bit strings used is the following:

1001100101000100000 0101

The first 19 bits determine which features are selected and the last three bits determine the number of hidden units of the neural network, for this particular case the number of hidden units encoded is six.

The genetic operators used are the standard single-point crossover and point mutation [7].

The selected set of features and the correspondent neural network architecture encoded in the bit string are used to train the neural network over a training set of segmented images with their correspondent regions previously classified by the user as container or non-container of the desired object.

The fitness function used is the predictive accuracy of a given neural network over a test set of segmented images. The overall system can be seen in figure #1.

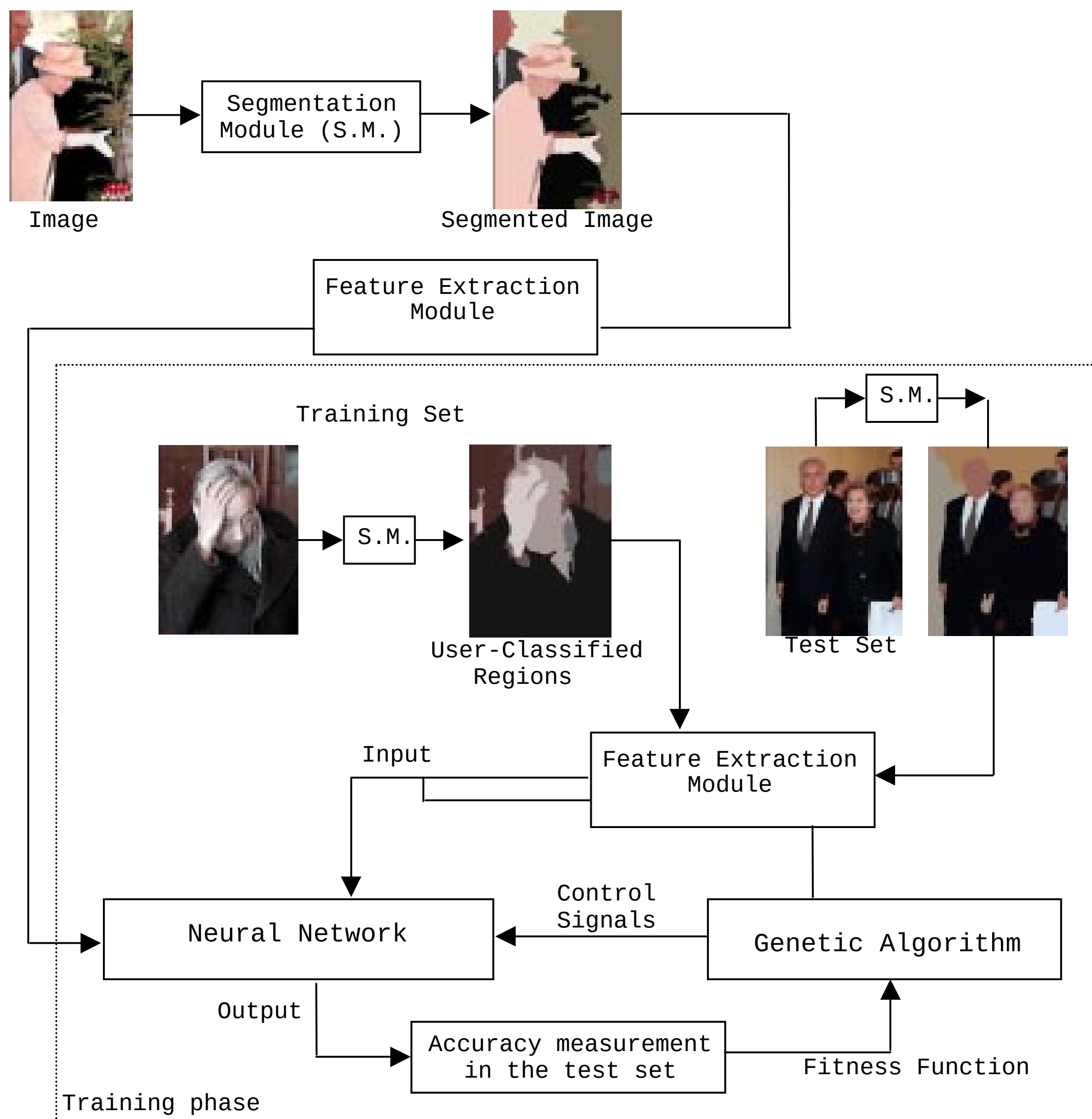

Figure # 1

RESULTS

The system was used to construct a face detector. We use a training set of 200 images containing faces in different orientations of persons of different races and ages; and a test set containing 400 images with the same diversity as in the training set. This is a very unconstrained set, because normally in face detectors the image of the face is straight or is constrained in terms of the race or age.

The accuracy obtained using the entire feature vector using 16 hidden units was 70.45%.

The value of the parameters used in the neural network and in the genetic algorithm are the following:

Neural Network Parameters	
Parameter	Value
Number of epochs	100
Learning Rate	0.3
Momentum	0.3
Genetic Algorithm Parameters	
Parameter	Value
Population	50
Number of generations	30
Crossover Probability	0.92
Mutation Probability	0.02
Fitness	0.90

The selection method used was the *fitness proportionate selection method.*[7]

In the following table we present the best individuals found by the genetic algorithm.

	Bit String	Accuracy
1	0001101100010100000 0111	84.18%%
2	0111010101010110111 1110	83.24%
3	0001100101000000011 0011	82.74%
4	1000100110001100110 0001	82.74%
5	1000010100100110000 0010	82.65%

As we can observe there is an improvement in the accuracy due the use the optimal subset of:

$$\Delta\text{Accuracy} = (84.18-70.45)/70.45$$

$$= 19.48\%$$

The only feature that appears in all the bit strings is the one in the position number eight, which corresponds to the shape measurement of compactness. So for this case the optimal subset of features is not exactly unique.

The accuracy in this example is not as high as in some previous works, but as we mentioned before the dataset used was very unconstrained, because we wanted to show the flexibility of the system.

CONCLUSIONS

The system proposed is very flexible because it could work with unconstrained datasets, even though the use of constrained datasets would increase its accuracy, it also permits the user to make their own object classifier. The use of the genetic algorithm really improves the performance of the system in speed and speed accuracy.

BIBLIOGRAPHY

[1] Pope Arthur R. Model-Based Object Recognition. A survey of recent research. Technical Report 94-04 January 1994.

[2] Binford Thomas O. Survey of Model-Based Image Analysis Systems The International Journal of Robotics Research Vol.1 No.1 1982

[4] Jaimes Alejandro, Chang S.F Model-Based Classification of visual Information for Content-Based Retrieval. Storage and Retrieval for Image and Video Databases VII, IS&T/SPIE99, San Jose, CA. January 1999.

[5] Kira K., Rendell L The feature selection problem: Traditional methods and a new algorithm. Proc. Of the Tenth Conference on Artificial Intelligence. MIT Press 1992.

[3] El-Kwae Essam A., Kabuka Mansur R. An Autonomous Boolean Neural Network Approach for Image Understanding.

[6] D. Zhong, S.F. Chang. Video Object Model and Segmentation for Content-Based Video Indexing ISCAS’97 June 11, 1997.

[7] Mitchell, Tom M. Machine Learning. McGraw Hill (1997)