

A Pyramidal Neural Network Based on Nonclassical Receptive Field Inhibition

Bruno J. T. Fernandes and George D. C. Cavalcanti
Federal University of Pernambuco (UFPE)
Informatics Center (CIn)
Recife, PE, Brazil
{bjtf,gdcc}@cin.ufpe.br

Abstract

This paper presents a new artificial neural network, called I-PyraNet. This new architecture is based on the combination between concepts of the recently described PyraNet and the nonclassical receptive fields inhibition, integrating the feature extraction and the classification stages into the same structure which is formed by 2-D and 1-D layers. The main difference between the PyraNet and the I-PyraNet is that while in the first a 2-D neuron always provide the same output, in the I-PyraNet the signal of the output of a 2-D neuron will invert when it appears inside a inhibitory field. Furthermore, the I-PyraNet is applied over a face detection task where different configurations of the network are tested.

1. Introduction

In recent years there has been a growing interest in the application of the artificial neural networks (ANNs) to solve the more different kinds of tasks, like machine vision, robust pattern detection, signal filtering, data segmentation, data compression and more. The motivation of the ANNs is based on the design and functioning of human brains and components thereof, distinguishing in this way them from other mathematical techniques. Therefore, their main advantage is the ability to learn like their biological counterparts, being very useful in tasks that are difficult to find an analytic solution. Also, the ANNs have been successfully applied to solve visual recognition problems, like face recognition and face detection [1].

Phung and Bouzerdoun [2] proposed a new ANN, the PyraNet, applying it to solve a face recognition problem. The PyraNet is based on the concepts of receptive fields and weight sharing over a 2-D topology. It is based on the concepts of receptive fields and weight sharing, motivated by the successfully results obtained by the image pyramids. However, the PyraNet does not contemplate the inhibitory

behavior of the neurons that surround any given receptive field. This kind of inhibitory stimulus is present in 80% of the orientation-selective cells and has been successfully applied to solve a contour detection problem [3]. Therefore, in this paper, is proposed a new neural network, called I-PyraNet. It is a hybrid implementation between the PyraNet and the inhibitory fields. The paper is organized as follows. In Section 2 is presented the entire architecture of the I-PyraNet. In Section 3, are presented the experimental results of the I-PyraNet over a face detection task. Finally, in Section 4 it is drawn some concluding remarks.

2. I-PyraNet

I-PyraNet is a hybrid implementation of the PyraNet [2] involving also the concept of inhibitory receptive field, where a neuron might send excitatory or inhibitory signals to other neurons in posteriors layers. The main difference between the two structures is that in the I-PyraNet the neuron in a 2-D layer has the possibility of change the signal of its output when it appears in a inhibitory field. A more detailed description of the I-PyraNet is here presented.

2.1. I-PyraNet Architecture

The I-PyraNet is a feedforward neural network with two kinds of processing layers:

- 2-D layers, arranged into a 2-D array, responsible for the feature extraction and is situated in the bottom of the network. Each neuron in this kind of layer is connected to a receptive field in the previous layer surrounded by a inhibitory field. The input for the first 2-D layer is the input image.
- 1-D layers responsible for the pattern classification, located at the top of the network working as a multilayer perceptron. The input for the first 1-D layer is the output of the last 2-D layer. The output of the last 1-D layer is the network output.

The output of a 2-D neuron y in the layer l at the (u, v) position is given by

$$y_{u,v} = f \left(\sum_{i,j \in R_{u,v}} w_{i,j} * y_{i,j} - \sum_{i,j \in I_{u,v}} w_{i,j} * y_{i,j} + b_{u,v} \right), \quad (1)$$

where f is a nonlinear activation function, $w_{i,j}$ denotes the weight associate with the input position (i, j) to the 2-D layer l , $b_{u,v}$ is the bias of the neuron y , $y_{i,j}$ the output of the neuron (i, j) in the previous layer $l - 1$ and $R_{u,v}$ and $I_{u,v}$ are respectively the receptive and the inhibitory field of the neuron (u, v) .

The output of 1-D neuron y in position n at layer l is given by

$$y_n^l = f \left(\sum_{m=1}^{N_{l-1}} w_{m,n} * y_m^{l-1} + b_n^l \right), \quad (2)$$

where N_{l-1} represents the amount of neurons in the previous 1-D layer $l - 1$, $w_{m,n}$ is the synaptic weight from the neuron m in the layer $l - 1$ to the neuron n at layer l , y_m^{l-1} is the output of the neuron m at layer $l - 1$ and b_n^l is the bias of the neuron n at layer l . The I-PyraNet architecture is presented in the Figure 1.

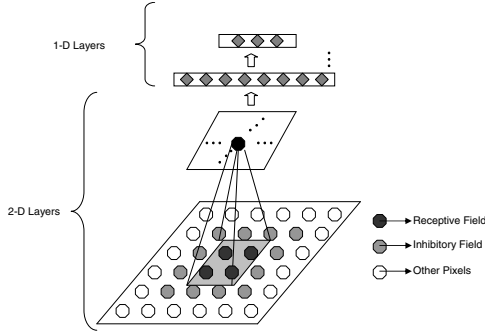


Figure 1. I-PyraNet architecture

2.2. I-PyraNet Training

The I-PyraNet is trained based on the difference between the output desired and the output obtained. The cross-entropy function (CE) [4] is used to calculate the network outputs estimating for the image input the a posteriori probability for each known class. Being y_n^L the output of the neuron n in last network layer L for an input image k , the estimated a posteriori probability p_n is given by

$$p_n^k = \exp(y_n^{L,k}) / \sum_{i=1}^{N_L} \exp(y_i^{L,k}), \quad (3)$$

where N_L is the amount of neurons in the layer L . The error obtained in the network output is backpropagated to the entire network. The error sensitivity for the neurons will provide the gradient error for recalculate the network weights.

The error sensitivity δ for each neuron n at the 1-D output network layer L_{1D} , for an input image k is given by

$$\delta_n^{L_{1D},k} = e_n^k f' (s_n^{L_{1D},k}), \quad (4)$$

where e_n^k is the output y_n^k produced by the neuron n at the last 1-D layer L_{1D} minus the desired output d_n^k , then $e_n^k = y_n^k - d_n^k$. And $s_n^{L_{1D},k}$ is the weighted sum input to neuron n at layer L_{1D} and f' is the differential of the activation function f . Then, for the neurons in the others 1-D layers $l_{1D} < L_{1D}$ the error sensitivity is given by

$$\delta_n^{l_{1D},k} = f' (s_n^{l_{1D},k}) * \sum_{m=1}^{N_{l_{1D}+1}} \delta_m^{l_{1D}+1,k} * w_{n,m}, \quad (5)$$

where $N_{l_{1D}+1}$ represents the amount of neurons in the next layer $l_{1D}+1$, $w_{n,m}$ is the synaptic weight from the neuron n in the layer l_{1D} to the neuron m at layer $l_{1D}+1$ and $\delta_m^{l_{1D}+1,k}$ is the error sensitivity of the neuron m at layer $l_{1D}+1$.

The error sensitivities for the last 2-D layer are calculated using the previous equation but rearranged into a 2-D grid. In the others 2-D layers l_{2D} , the error sensitivity for each neuron at the position (u, v) is given by

$$\delta_{u,v}^{l_{2D},k} = f' (s_{u,v}^{l_{2D},k}) * w_{u,v} * \sum_{i=i_l^{max}}^{i_h^{max}} \sum_{j=j_l^{max}}^{j_h^{max}} \gamma_{i,j}^{l_{2D}+1,k}, \quad (6)$$

where $s_{u,v}^{l_{2D},k}$ is the weighted sum input for the neuron (u, v) , $w_{u,v}$ is the weight associate to the neuron (u, v) at layer l_{2D} and $\gamma_{i,j}^{l_{2D}+1,k}$ is given by

$$\gamma_{i,j}^{l_{2D}+1,k} = \begin{cases} \delta_{i,j}^{l_{2D}+1,k} & i_l \leq i \leq i_h, j_l \leq j \leq j_h \\ -\delta_{i,j}^{l_{2D}+1,k} & \text{otherwise} \end{cases}, \quad (7)$$

being $\delta_{i,j}^{l_{2D}+1,k}$ the error sensitivity for the neuron (i, j) at the next layer, and $i_l^{max}, i_h^{max}, j_l^{max}, j_h^{max}, i_l, i_h, j_l$ and j_h are calculated by

$$i_l^{max} = \left\lceil \frac{u - (r_{l+1} + h_{l+1})}{g_{l+1} - h_{l+1}} \right\rceil + 1, i_l = \left\lfloor \frac{u - r_{l+1}}{g_{l+1}} \right\rfloor + 1 \quad (8)$$

$$i_h^{max} = \left\lfloor \frac{u - 1}{g_{l+1} - h_{l+1}} \right\rfloor + 1, i_h = \left\lfloor \frac{u - 1}{g_{l+1}} \right\rfloor + 1 \quad (9)$$

$$j_l^{max} = \left\lceil \frac{v - (r_{l+1} + h_{l+1})}{g_{l+1} - h_{l+1}} \right\rceil + 1, j_l = \left\lfloor \frac{v - r_{l+1}}{g_{l+1}} \right\rfloor + 1 \quad (10)$$

$$j_h^{max} = \left\lfloor \frac{v - 1}{g_{l+1} - h_{l+1}} \right\rfloor + 1, j_h = \left\lfloor \frac{v - 1}{g_{l+1}} \right\rfloor + 1 \quad (11)$$

Therefore, the error gradient to the weights and the biases can be obtained through the next equations.

- **1-D Weights:** the error gradients for the 1-D synaptic weight $w_{m,n}$ from the neuron m at layer $l_{1D} - 1$ to the neuron n at layer l_{1D} for all the images input K , are given by

$$\frac{\partial E}{\partial w_{m,n}} = \sum_{k=1}^K \delta_n^k y_m^{l_{1D}-1,k}. \quad (12)$$

- **2-D Weights:** the 2-D synaptic weight $w_{u,v}$ of neuron (u, v) at layer l_{2D} to layer $l_{2D} + 1$ is calculated by

$$\frac{\partial E}{\partial w_{u,v}} = \sum_{k=1}^K \left\{ y_{u,v}^{l_{2D},k} \times \sum_{i=i_1^{max}}^{i_h^{max}} \sum_{j=j_1^{max}}^{j_h^{max}} \gamma_{u,v}^{l_{2D}+1,k} \right\}, \quad (13)$$

- **Biases:** the error gradients for the bias of neuron n , b_n , at 1-D layer l_{1D} and of neuron u, v , $b_{u,v}$, at 2-D layer l_{2D} are respectively given by

$$\frac{\partial E}{\partial b_n} = \sum_{k=1}^K \delta_n^k, \quad \frac{\partial E}{\partial b_{u,v}} = \sum_{k=1}^K \delta_{u,v}^k \quad (14)$$

The weights in this work are then recalculated through the use of the Gradient Descent [5] and this completes the training phase of the I-PyraNet.

3. Experimental Results

Face recognition is a very analyzed task in the machine vision area. In [2] the PyraNet classifier is tested over a face-nonface classification task, what means that the neural network will try to define if a given image is or is not a face pattern. In another work, Osuna et al. [6] used a Support Vector Machine (SVM) to detect faces in an image. Therefore, in this work the face-nonface recognition problem is applied over many different configurations of the I-PyraNet and the results obtained through it are compared to the ones using the regular PyraNet and the SVM. The method used to realize all the tests is the cross-validation 10-fold repeated 10 times.

3.1. Data Preparation

The data used to form the image base of this work was taken from different places. The ones that compose the faces patterns were extracted from the Essex Database¹ and are composed of faces from different men and women. The others, belonging to the nonfaces patterns, were taken from

¹<http://cswww.essex.ac.uk/mv/allfaces/index.html>

free Web pages images in the network being randomly extracted from scenery photos. All the images used in this work were resized to 20×20 pixels. An amount of 271 patterns were used for each class totalizing 542 images. Figure 2 shows image samples used in this work.

3.2. Analysis of Face-Nonface Classifiers

First-of-all is important to say that a regular PyraNet is nothing more than an I-PyraNet with all the values of the inhibitory fields equals to 0. The I-PyraNet tested has two pyramidal layers and one output layer with two neurons. In order to detect the best network parameters, many layers configurations were tested. First, the overlaps and inhibitory fields sizes of both pyramidal layers were set to 0 and only the size of the receptive fields varied, the error rates for such configurations are presented in Table 1 where the first column shows the sizes for the receptive fields of the first and the second pyramidal layer, respectively.

Table 1. Error Rates in % Varying the Receptive Fields

Receptive Fields	1,1	2,1	4,2	5,1	5,2	5,4
Error Rate	42.0	13.0	9.1	8.0	7.7	7.9

The best configuration for the sizes of the receptive fields were 5×5 and 2×2 for the first and the second pyramidal layer, respectively. Therefore, in order to find the best overlaps parameters for such receptive fields configuration, new tests were realized varying only the overlaps sizes, this results are presented in Table 2 where the first column shows the sizes for the overlaps of the first and the second pyramidal layer, respectively.

Table 2. Error Rates in % Varying the Overlaps Sizes

Overlaps Sizes	0,0	1,0	2,0	2,1	2,2
Error Rate	7.7	10.6	7.2	10.5	10.7

The best configuration for the sizes of the overlaps were 2 and 0 for the first and the second pyramidal layer, respectively, what means that the neurons in the second pyramidal layer might have no overlap among their inputs. The last parameter configuration is the inhibitory fields sizes. Table 3 presents the error rates for a I-PyraNet with receptive fields sizes of 5×5 and 2×2 for the first and the second pyramidal layer, respectively, being the overlap factor in the first pyramidal layer equals to 2. The second pyramidal layer does not overlap its inputs. The first column in Table 3 presents

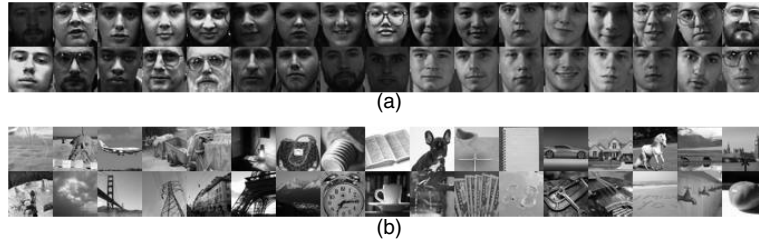


Figure 2. Images example: (a) faces patterns, (b) nonfaces patterns

the sizes of the inhibitory fields for the first and the second layers, respectively; 0,0 means that no inhibitory field is used working as a regular PyraNet.

Table 3. Error Rates in % Varying the Inhibitory Fields

Inhibitory Fields	0,0	1,0	1,1	2,1
Error Rate	7.2	5.9	9.1	8.9

The application of the inhibitory fields concepts in the first pyramidal layer achieved the lowest error rate among all the results. Finally, in Table 4 are presented the best results obtained with the I-PyraNet against the ones obtained with the PyraNet and the SVM. The SVM tested has a Multilayer Perceptron as its kernel function and uses the Quadratic Programming (QP) method to find the separating hyperplane. The I-PyraNet presented the best results among all the classifiers, reaching an error rate of only 5.9%. Therefore, the I-PyraNet classification error rate was smaller than the PyraNet classification error rate and both of them were smaller than the ones obtained with the SVM. However, it is important to note that the bad results of the SVM classification might be explained by the fact that no method for feature extraction is used, since this step is realized inside the I-PyraNet and the PyraNet classifiers.

Table 4. Results Comparison

Classifier	I-PyraNet	PyraNet	SVM
Error Rate	5.9	7.2	21.93

4. Conclusions

In this work was proposed a hybrid implementation of the PyraNet based on the idea of inhibitory fields, called I-PyraNet. We have applied many different configurations of the classifier I-PyraNet over the face-nonface recognition problem and it achieved the best results in comparison with

the regular PyraNet and a Support Vector Machine. Therefore, the I-PyraNet was presented as a very good approach with its architecture based on the biological schema of the human brain. Although the neural network presented in this work was used only to realize a facial detection task, its use is not restrict only to this task, the I-PyraNet can be applied in many different visual recognition problems.

Acknowledgments

This paper is supported by the Provider Tecnologia de Sistemas LTDA.

References

- [1] H. A. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, pp. 23–38, 1998.
- [2] S. L. Phung and A. Bouzerdoum, "A pyramidal neural network for visual pattern recognition," *IEEE Transactions on Neural Networks*, vol. 18, no. 2, pp. 329–343, 2007.
- [3] C. Grigorescu, N. Petkov, and M. A. Westenberg, "Contour detection based on nonclassical receptive field inhibition," *IEEE Transactions on Image Processing*, vol. 12, no. 7, pp. 729–739, 2003.
- [4] C. M. Bishop, *Neural Networks for Pattern Recognition*. Oxford, U.K.: Clarendon, 2007.
- [5] G. E. H. D. E. Rumelhart and R. J. Williams, "Learning internal representations by error propagation," *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, vol. 1, pp. 318–362, 1986.
- [6] E. Osuna, R. Freund, and E. Girodit, "Training support vector machines: an application to face detection," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 130–136, 1997.