Lateral Inhibition Pyramidal Neural Network for Image Classification

Bruno José Torres Fernandes, *Member, IEEE*, George D. C. Cavalcanti, *Member, IEEE*, and
Tsang Ing Ren, *Member, IEEE*

Abstract—The human visual system is one of the most fascinating and complex mechanisms of the central nervous system that enables our capacity to see. It is through the visual system that we are able to accomplish from the most simple task such as object recognition to the most complex visual interpretation, understanding and perception. Inspired by this sophisticated system, two models based on the properties of the human visual system are proposed. These models are designed based on the concepts of receptive and inhibitory fields. The first model is a pyramidal neural network with lateral inhibition, called lateral inhibition pyramidal neural network. The second proposed model is a supervised image segmentation system, called segmentation and classification based on receptive fields. This work shows that the combination of these two models is beneficial, and the results obtained are better than that of other state-of-the-art methods.

Index Terms—Image processing, neural network, pattern recognition, receptive fields.

I. INTRODUCTION

AT the beginning of the 1960s, an important region of neurons in the human brain was discovered. This region was called the receptive field, and it was identified in many parts of the human brain, such as the auditory, somatosensory, and visual systems [1]. The concept of receptive field was defined by Levine and Shefner [2] as an area in which the presence of an appropriate stimulus could lead to the response of a sensitive neuron. Rizzolatti and Camarda [3] demonstrated that another stimulus, applied simultaneously with the receptive field stimulus, can also affect the neuron. This stimulus originated from a region called the nonclassical receptive field or extraclassical receptive field (ECRF) [4], and most of the time, this stimulus presents a lateral inhibitory effect. Therefore, we refer to it as an inhibitory field.

The applications of the concepts of receptive and inhibitory fields in the area of pattern recognition range from contour detection [5] to texture analysis, such as the Gabor filter [6], [7]. These concepts have also been applied in the development of neural models. Sun *et al.* [8] proposed a neurocomputational model for object detection in the spatial and temporal domains. Park *et al.* [9] proposed a neural network architecture based on the radial basis function having in its topology a collection of receptive fields. Ghosh and Pal [10] proposed a model for the ECRFs using the receptive and inhibitory field concepts for object detection. These results demonstrate that the receptive fields of the primary visual cortex can be applied for image classification tasks.

Fukushima *et al.* [11], [12], inspired by the concept of receptive fields, proposed the neocognitron: a multilayered neural network for visual pattern recognition. He also proposed several modifications in the neocognitron, such as the inhibitory surround [13]. The presence of the inhibitory surround increased the neocognitron recognition rate and simplified the network architecture. Motivated by the neocognitron model, Perez *et al.* [14] combined a feature extraction neural network and a neural classifier to also perform visual pattern recognition. The receptive and inhibitory field concepts were embedded in the proposed neural network. However, their model was not able to generate the receptive fields by itself since it used a genetic algorithm in order to find the most suitable receptive field configuration: dimensions, angle of orientation, and bias. LeCun *et al.* [15], [16] proposed the convolutional neural network (CNN) using the same type of cells as the neocognitron and taking into consideration the fact that a specific neural network architecture based on *a priori* knowledge is able to improve the generalization ability of the model. Tivive and Bouzerdoum [17], [18] improved the CNN using an inhibitory stimulus. They obtained good classification rates in a face detection task using an architecture that has a less complex structure and fewer trainable parameters.

Another important method based on the concept of receptive field was presented by Phung and Bouzerdoum [19]. They proposed an artificial neural network called PyraNet to perform visual pattern recognition. The PyraNet receives as input an image, and the network output is the image classification. In this model, the sizes of the receptive fields do not change during the training, but the orientation of the angle and the bias are adjusted by tuning the neuron weights during the error backpropagation of the neural network.

The concept of lateral inhibition [20] has already been applied in other neural models. The analysis of the recurrent neural network with lateral inhibition was presented by Mao and Massaquoi [21], based on the work developed by

B. J. T. Fernandes is with the Polytechnic School, University of Pernambuco, 50750-470 Recife-PE, Brazil, and also with the Center for Informatics, Federal University of Pernambuco, 50740-560 Recife-PE, Brazil (e-mail: bjtf@ecomp.poli.br; http://cin.ufpe.br/~viisar).

G. D. C. Cavalcanti and T. I. Ren are with the Center for Informatics, Federal University of Pernambuco, 50740-560 Recife-PE, Brazil (e-mail: gdcc@cin.ufpe.br; tir@cin.ufpe.br; http://cin.ufpe.br/~viisar).

Coultrip *et al.* [22]. They showed that lateral suppression by neighboring neurons in the same layer makes the network more stable and efficient. Other interesting research works that explore the ideas of lateral inhibition were developed by Chen *et al.* [23] who proposed an unsupervised neural model based on Hebbian rule and lateral inhibition, called neuronal cluster, and by Fang *et al.* [24] who presented a study about dynamical neural networks with lateral inhibition. Arkachar and Wagh [25] presented a neural model to study the lateral inhibition influence in edge enhancement and demonstrated that, when the ratio between the inhibitory and excitatory weights approaches a critical rate, the edge enhancement increases, and the neural network becomes unstable when the ratio is higher than the critical rate.

Wilson *et al.* [26] presented two different inhibitory stimulus mechanisms in the brain circuits. The first one comes from the somatostatin-expressing neurons performing the subtraction of responses in their targets. The second stimulus comes from the parvalbumin-expressing neurons implementing divisive normalization. The inhibition in the somatostatin-expressing cells is more uniform than that in the parvalbumin-expressing cells, leading to a broader range of targets and sharpened orientation selectivity. Such inhibition has been described in neurons in the V1 area of the visual cortex.

In this paper, we propose two models for pattern classification based on the concepts of receptive and inhibitory fields: a classifier and a supervised image segmentation method. The models are combined to perform image classification. The proposed classifier is a pyramidal neural network with lateral inhibition, the lateral inhibition pyramidal network (LIPNet), which is based on the PyraNet architecture. However, while the PyraNet only considers the excitatory stimulus of the receptive fields, we propose a neural network that incorporates the concept of lateral inhibition implementing a subtractive inhibitory stimulus. Such approach is justified for two reasons: It gives better stability and efficiency to the neural network [21], and it improves the texture analysis by ignoring inner-texture contours [5]. The image segmentation method was developed based on the concepts of receptive fields, called the segmentation and classification based on receptive fields (SCRF). This model divides a given image into overlapped subimages and classifies each of them with a supervised classifier. The output of the classifier is the probability that a given subimage belongs to a known class. Thus, based on such classification, the SCRF model is able to classify each pixel of the image.

This paper is organized as follows. Section II describes the LIPNet. Section III presents the SCRF model. In Section IV, experiments using satellite image for segmentation and face detection are shown. Finally, in Section V, we present the concluding remarks.

II. LIPNet

LIPNet is an artificial neural network developed to perform image classification. LIPNet is inspired by the PyraNet [19], which was motivated by the CNNs [15] and by the concepts of receptive fields. However, the PyraNet considers only the excitatory effects of the neurons inside a receptive field.
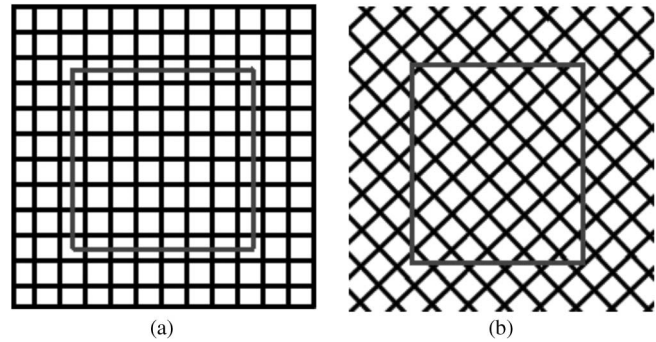

Fig. 1. Visual effect caused by the lateral inhibition. It is easier to distinguish the dark gray square in the black grid shown in (b) than it is in (a).

In the LIPNet, the lateral inhibition signal sent by the neurons in the same layer is also considered. Blakemore and Tobin [27] measured the response of a neuron to a bar inside a receptive field surrounded by other bars in different orientations. They observed an inhibitory stimulus caused by the bars surrounding the receptive field. The strength of the inhibition is improved when the bars had the same orientation as the one inside the receptive field. Fig. 1 illustrates the visual effects caused by the lateral inhibition. A given contour is difficult to observe when it is in the presence of other contours in the same orientation. The sides of the dark gray square in Fig. 1(a) are not perceptible at first; a more careful observation is necessary in order to determine the sides. On the other hand, in Fig. 1(b), the black bars have a different orientation from the bars of the gray square, and the lateral inhibition does not have an effect on the square detection.

In the LIPNet architecture, the output of the neuron consists of the excitatory stimulus in its receptive field and the inhibitory stimulus caused by the neurons in its neighborhood.

A. LIPNet Architecture

The LIPNet architecture is composed of a multilayer network with two kinds of layers:

1) Two-dimensional layers: perform feature extraction and data reduction. They are located at the base of the network. In a 2-D layer, the neurons are arranged in a matrix.
2) One-dimensional layers: perform image classification. They are located at the top of the network.

Each 2-D layer in the LIPNet has a parameter that sets the size of its receptive field, defined as r, and another parameter that sets the size of the overlap between adjacent receptive fields, defined as o. The relation between the receptive field and the overlap region is defined as $g = r - o$. In addition, the LIPNet has two other parameters: the size of the inhibitory neighborhood, denoted by h, and the inhibition weight, denote by δ. The neurons in the inhibitory area send a negative response when inside an inhibitory field. Fig. 2 shows the LIPNet architecture. The entire network is connected in cascade (i.e., the output of one layer is considered as the input to the next one). The input of the first 2-D layer is the image to be classified, and the input of the first 1-D layer is the output of the last 2-D layer rearranged in a vector. Each neuron in a 2-D
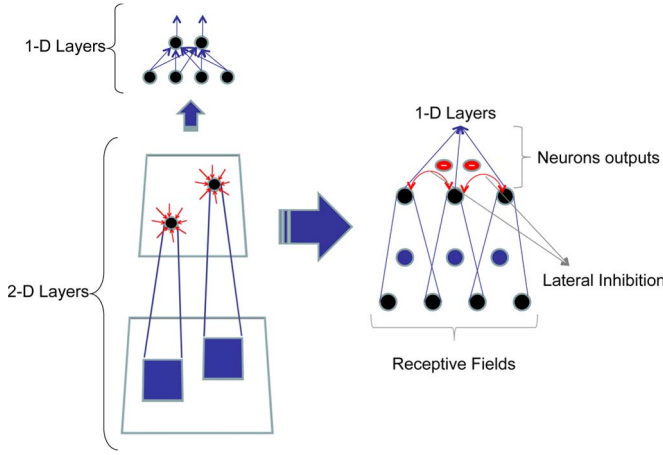
Fig. 2. LIPNet architecture. The left side shows the general architecture, while the right side shows the relation between the receptive field and the inhibitory field of a neuron in the last 2-D layer.

layer is connected to a receptive field in the previous layer, and it is surrounded by an inhibitory field in the same layer. Neurons in the same layer might also share neurons inside their receptive field.

In the 2-D layer, the weights are associated with the neuron itself, instead of being associated with the connection between neurons (i.e., neurons in the same layer share not only the output of an overlapped neuron but also the weight associated with it). Each neuron is also connected to an inhibitory field in the same layer. Thus, the lateral inhibition strength of a given layer l is given by

$$\psi^l = \frac{\delta^l}{\left(h^l \times 2 + 1\right)^2 - 1} \tag{1}$$

where δ^l is the lateral inhibition weight and h^l is the size of the inhibitory field of the 2-D layer l. If δ^l is too high, the output of the neurons is completely inhibited. On the other hand, if the inhibitory field is too big, the inhibition may not take effect.

The output of the 2-D neuron consists in applying a nonlinear activation function to the weighted summation of the neurons inside its receptive field. The result is then subtracted by the weighted summation of the neurons surrounding it in the inhibitory field. We define (u, v) as the position of a neuron in a 2-D layer l, (i, j) as the position of the neuron in the previous layer $(l - 1)$, and $b_{u,v}$ as the bias of the neuron at (u, v). For each layer, the output y of the neuron is calculated in three steps.

1) Receptive field: For each neuron in a layer l, the excitatory stimulus is calculated using the following equation:

$$\chi_{u,v}^l = \underbrace{\sum_{i,j \in R_{u,v}^l} w_{i,j}^l \times y_{i,j}^{l-1} + b_{u,v}^l}_{Receptive \; Field} \tag{2}$$

where $w_{i,j}^l$ represents the weight associated with the input position (i, j) to the layer l and $R_{u,v}^l$ and $b_{u,v}^l$ are the receptive field and the bias of the neuron at position (u, v) in layer l, respectively.

2) Inhibitory field: For each neuron in the layer l, the lateral inhibition is calculated using the following equation:

$$\iota_{u,v}^l = \psi^l \times \underbrace{\left(\left(\sum_{i=u-h}^{u+h} \sum_{j=v-h}^{v+h} \chi_{i,j}^l\right) - \chi_{u,v}^l\right)}_{Inhibitory \; Field}. \tag{3}$$

3) Activation function: consists in applying a nonlinear activation function to the weighted summation of the neurons inside its receptive field subtracted by the weighted summation of the neurons inside its inhibitory field. The mathematical formulation is given by

$$y_{u,v}^l = f\left(\chi_{u,v}^l - \iota_{u,v}^l\right). \tag{4}$$

The output of a neuron in the 1-D layer is also defined as a nonlinear activation function applied to the weighted summation of the neurons in the previous layer. In the 1-D layer, the weights are associated with the connection between the neurons. Then, the output y of a neuron at the position n of the 1-D layer l is given by

$$y_n^l = f\left(s_n^{l_{1D},k}\right) \tag{5}$$

having

$$s_n^{l_{1D},k} = \sum_{m=1}^{N_{l-1}} w_{m,n} \times y_m^{l-1} + b_n^l \tag{6}$$

where N_{l-1} is the number of neurons in the previous layer $l - 1$, $w_{m,n}$ is the synaptic weight of the neuron m in the layer $l - 1$ to the neuron n in the layer l, and b_n^l is the bias associated with neuron n in the 1-D layer l. The output of the last 1-D layer is the network output.

The activation functions used in this work are the logistic sigmoid for the 2-D neurons and the hyperbolic tangent for the 1-D neurons.

B. LIPNet Training

LIPNet is first trained to be able to perform visual pattern recognition. As a supervised neural network, its objective is to reduce the error that is defined as the difference between the desired and the obtained output. This is achieved by adjusting the weights of the neuron connections in the LIPNet. The optimization function used to perform this task was the cross-entropy (CE) function [28], in which the network output estimates the *a posteriori* probability for each known class. The advantages of using the CE function for training the neural network were presented by Kline and Berardi [29].

The output of the neuron n in the last network layer L for an input image k is defined as y_n^L. Thus, the estimated *a posteriori* probability for a class associated with the neuron n is given by

$$p_n^k = \exp\left(y_n^{L,k}\right) / \sum_{i=1}^{N_L} \exp\left(y_i^{L,k}\right) \tag{7}$$

where N_L is the number of neurons in the layer L. Therefore, in order to adjust the weights of the LIPNet, the error gradient of the weights is calculated using the error sensitivity δ of each neuron n at the 1-D output layer L_{1D}. For an input image k, the error sensitivity δ is defined as

$$\delta_n^{L_{1D},k} = e_n^k f'\left(s_n^{L_{1D},k}\right) \tag{8}$$

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, i.e., $e_n^k = y_n^k - d_n^k$, and f' is the differential of the activation function f. Thus, for the neurons in the other 1-D layers $l_{1D} < L_{1D}$, the error sensitivity is given by

$$\delta_n^{l_{1D},k} = f'\left(s_n^{l_{1D},k}\right) \times \sum_{m=1}^{N_{l_{1D}+1}} \delta_m^{l_{1D}+1,k} \times w_{n,m} \tag{9}$$

where $N_{l_{1D}+1}$ represents the number of neurons in the next layer $l_{1D} + 1$, $w_{n,m}$ is the synaptic weight from neuron n in layer l_{1D} to the neuron m in layer $l_{1D} + 1$, and $\delta_m^{l_{1D}+1,k}$ is the error sensitivity of the neuron m in 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 other 2-D layers l_{2D}, the error sensitivity for each neuron at the position (u, v) is calculated using three steps for each layer.

1) Sensitivity of the next layer: calculated using the summation of the neurons in $l_{2D} + 1$ that contains the neuron in the $(u, v)^l$ position in the receptive fields χ

$$\gamma_{u,v}^{l,\chi,k} = \sum_{i=i_l}^{i_h} \sum_{j=j_l}^{j_h} \delta_{i,j}^{l_{2D}+1,k} \tag{10}$$

where $\delta_{i,j}^{l_{2D}+1,k}$ is defined as the error sensitivity of the neuron (i, j) in the next layer, and i_l, i_h, j_l, and j_h are defined as

$$i_l = \left\lceil \frac{u - r_{l+1}}{g_{l+1}} \right\rceil + 1 \tag{11}$$

$$i_h = \left\lfloor \frac{u - 1}{g_{l+1}} \right\rfloor + 1 \tag{12}$$

$$j_l = \left\lceil \frac{v - r_{l+1}}{g_{l+1}} \right\rceil + 1 \tag{13}$$

$$j_h = \left\lfloor \frac{v - 1}{g_{l+1}} \right\rfloor + 1. \tag{14}$$

2) Sensitivity of the same layer: calculated using the summation of the neurons in l_{2D} that contains the neuron in the $(u, v)^l$ position in the inhibitory fields ι

$$\gamma_{u,v}^{l,\iota,k} = \psi^l \times \left(\left(\sum_{i=u-h^l}^{u+h^l} \sum_{j=v-h^l}^{v+h^l} \delta_{i,j}^{l_{2D},k} \right) - \delta_{u,v}^{l_{2D},k} \right). \tag{15}$$

3) Sensitivity of the neuron

$$\delta_{u,v}^{l_{2D},k} = f'\left(s_{u,v}^{l_{2D},k}\right) \times w_{u,v}^l \times \left(\gamma_{u,v}^{l,\chi,k} - \gamma_{u,v}^{l,\iota,k}\right). \tag{16}$$

The error gradient of the weights and the biases can then be derived using the following equations.

1) One-dimensional weights: The error gradient for the 1-D synaptic weight $w_{m,n}$ of the neuron m in layer $l_{1D} - 1$ to the neuron n in layer l_{1D} for all the input images K is given by

$$\frac{\partial E}{\partial w_{m,n}} = \sum_{k=1}^{K} \delta_n^k y_m^{l_{1D}-1,k}. \tag{17}$$

2) Two-dimensional weights: The 2-D synaptic weight $w_{u,v}$ of neuron (u, v) in layer l_{2D} to layer $l_{2D} + 1$ is calculated by

$$\frac{\partial E}{\partial w_{u,v}} = \sum_{k=1}^{K} \left\{ y_{u,v}^{l_{2D},k} \times \sum_{i=i_l}^{i_h} \sum_{j=j_l}^{j_h} \delta_{i,j}^{l_{2D}+1,k} \right\}. \tag{18}$$

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

$$\frac{\partial E}{\partial b_n} = \sum_{k=1}^{K} \delta_n^k \tag{19}$$

$$\frac{\partial E}{\partial b_{u,v}} = \sum_{k=1}^{K} \delta_{u,v}^k. \tag{20}$$

Finally, in order to recalculate the weights of the neural network, a training method, such as the gradient descent [30] or the resilient propagation [31], is applied.

C. LIPNet Remarks

PyraNet can be considered a special case of the LIPNet, in which the lateral inhibition effect does not exist. Moreover, a neuron inside a 2-D layer in the PyraNet produces the same input as the neurons in the next layer. In the LIPNet, however, the output of a neuron can also be used to inhibit another neuron in the same layer. As pointed out by Grigorescu *et al.* [5], the use of lateral inhibition is useful to suppress contours inside a region that has the same texture. Therefore, the application of the inhibitory field leads the neural network to consider not only how useful a given information might be but also how this information impairs the neural network performance. In the LIPNet, the sizes of the receptive and inhibitory fields are given as input to the neural network, while the bias and angle of orientation of the receptive and inhibitory fields are simultaneously adjusted during the training.

III. SCRF MODEL

The SCRF model is used to perform supervised image segmentation. This model combined with a supervised classifier defines the probability that a given subimage extracted from the original image belongs to each one of the known classes. The LIPNet trained with the CE function is an appropriated method to perform the classification steps of the SCRF since it receives as input a 2-D image extracted using the SCRF model and gives
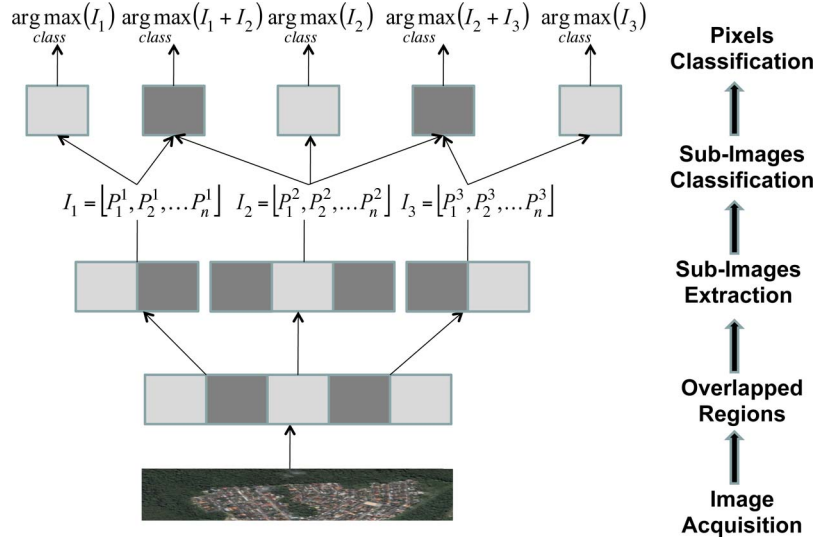
Fig. 3. SCRF model, where n is the number of known classes and P_n^i is the probability of the subimage i belonging to the class n.

as output the probability of the image to belong to each one of the training classes.

The aim of the SCRF model is to divide an image into subimages. Therefore, the classification of each subimages is used to classify each pixel in the image. This is done based on the concepts of the receptive field applied on an image. The basic idea is to generate subimages that share some pixels. This leads to the following advantage: The classification of a pixel does not depend only on itself, but it also depends on the classification of the subimages that contain the pixel (i.e., the pixels in its neighborhood also affect the classification). Fig. 3 presents the proposed model.

The SCRF model is described as follows.

1) First, a 2-D image is acquired (Image Acquisition).
2) The input image is divided into subimages (Subimage Extraction). The subimages have a predefined size of $r_S \times r_S$ and share a region of overlapped pixels among adjacent subimages, which is defined as o_S. The size of the extracted subimages (h, w) is given by

$$h = \left\lfloor \frac{H - o_S}{g_S} \right\rfloor \tag{21}$$

$$w = \left\lfloor \frac{W - o_S}{g_S} \right\rfloor \tag{22}$$

where g_S is the gap defined as $g_S = r_S - o_S$ and H and W are the height and width of the original image, respectively. A subimage SI is composed of a group of pixels according to its spatial position

$$SI_{u,v} = \{x_{i,j} | u \times g_S \le i < u \times (g_S + r_S), v \times g_S$$
$$\le j < v \times (g_S + r_S)\}. \tag{23}$$

3) Next, the probability of each subimage belonging to a known class must be calculated (Subimage Classification). This is accomplished using a supervised classifier.
4) Finally, in order to classify each pixel in the input image, the model defines the classification of a pixel as the class

that presents the highest probability among all the subimages that contain the pixel (Pixel Classification). However, if the pixel is not in an overlapped area, meaning that only one subimage contains the pixel, therefore, only one probability per class is generated, and the classification of the pixel is straightforward.

The following equation defines how the classification is obtained for a single pixel:

$$C_{x_{i,j}} = \underset{class\ c}{\arg\max} \left(\sum_{SI|x_{i,j} \in SI} P(c, SI) \right) \tag{24}$$

where $x_{i,j}$ is a single pixel in the (i, j) image position, $C_{x_{i,j}}$ is the pixel classification, c denotes one of the possible classes, SI represents a subimage, and $P(c, SI)$ is the a posteriori probability of a given subimage SI belonging to a given class c. Although the summation of the probabilities is used to define the pixel class, any other metric that uses the obtained probabilities can be applied. The LIPNet model is used as the supervised classifier of the SCRF model, and it calculates the class of a given pixel applying the following equation:

$$C_{x_{i,j}} = \underset{class\ c}{\arg\max} \left(\sum_{u=i_l}^{i_h} \sum_{v=j_l}^{j_h} p_c^{SI_{u,v}} \right) \tag{25}$$

where $p_c^{SI_{u,v}}$ is defined by (7) and i_l, i_h, j_l, j_h are given by

$$i_l = \left\lceil \frac{u - r_S}{g_S} \right\rceil + 1 \tag{26}$$

$$i_h = \left\lfloor \frac{u - 1}{g_S} \right\rfloor + 1 \tag{27}$$

$$j_l = \left\lceil \frac{v - r_S}{g_S} \right\rceil + 1 \tag{28}$$

$$j_h = \left\lfloor \frac{v - 1}{g_S} \right\rfloor + 1. \tag{29}$$

One of the advantages of the SCRF is that it can be easily combined with other image processing techniques. The method fits well with the LIPNet model since it requires the classification of the generated 2-D images and such classification is obtained using the proposed model.

Moreover, the overlap between the pixels of different subimages leads to a better level of acuity in the image. The reason for that is because the overlapped regions can have different classifications from the regions to which they belong, increasing the level of detail of the segmented image.

Finally, the use of the concepts of receptive field makes the model more fault tolerant. If noise exists in a subimage, it can be easily ignored based on its neighborhood.

IV. EXPERIMENTAL RESULTS

The proposed approaches were tested on three different classification tasks: forest detection using real and synthetic satellite images and face detection. In all the experiments, the LIPNet architecture was composed of two 2-D layers and one 1-D layer with two neurons (each neuron estimates the *a posteriori* probability of a given image to belong or not to a given class).

However, for the forest detection task, only the SCRF model was applied, which generated subimages of 18×18 pixels with an overlap of 6 pixels. Since all the satellite images had a size of 900×450 pixels, a total of 1250 subimages for each tested image was created. Also, different image processing algorithms were tested in comparison with SCRF and LIPNet.

A. Data Preparation

Three different image databases were used in the experiments. The first data set was developed for the segmentation of real satellite images, and they were composed of images from regions of different Brazilian cities collected from Google Maps. All the images have approximately the same scale and light conditions.

The real satellite images database is composed of two training images of 900×450 pixels representing a forested and a nonforested area [32], [33]. The database also contains nine test images of 900×450 pixels and nine manually segmented images for each test image, which were used as an evaluation set. The test images received the following nomenclatures: Jundiai-1, Jundiai-2, Jundiai-3, Manaus-1, Manaus-2, Manaus-3, Manaus-4, Recife-1, and Recife-2.

Fig. 4 shows the images Jundiai-3 and Manaus-1 and the manually segmented representations.

The second data set was developed for the synthetic satellite image segmentation task. The database was generated using the functions described in [34], which were applied in the evaluation images of the real satellite image database. Table I presents the used functions that are associated with urban and forested areas, according to their gray level homogeneity. The parameters used to generate such images for the functions K_a and G_a^0 are as follows:

1) $K_a(\alpha, \lambda, n)$ where $\alpha = 2$, $\lambda = 0.00023$, and $n = 3$;
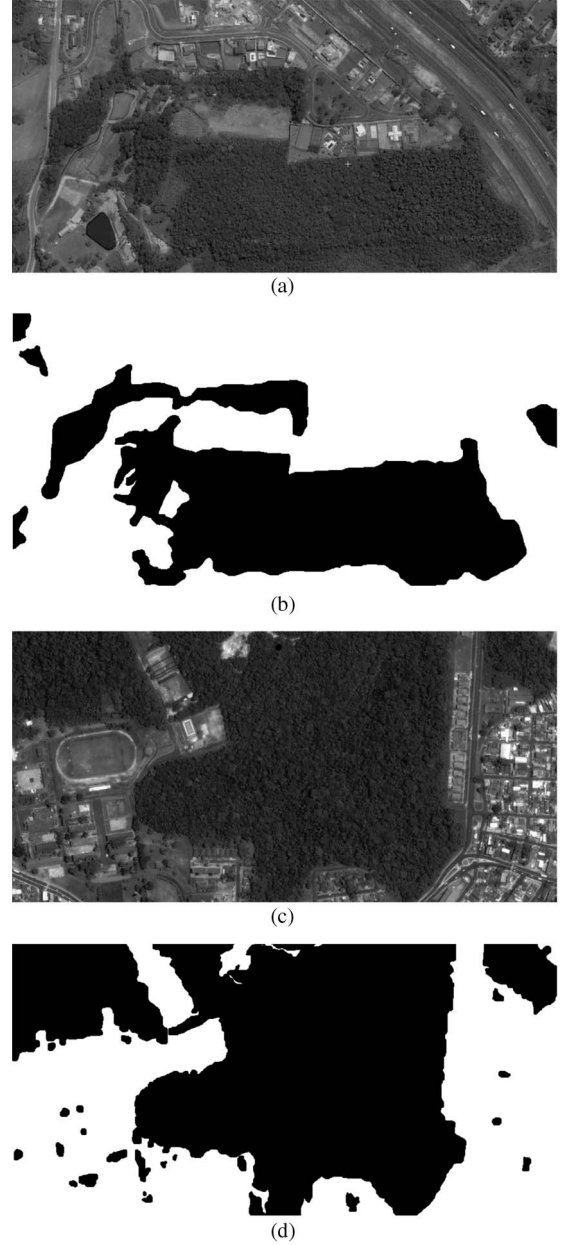2) $G_a^0(\alpha, \gamma, n)$ where $\alpha = -5$, $\gamma = 203{,}987$, and $n = 3$.

Fig. 4. Examples of real satellite images. (a) Jundiai-3 and (b) evaluation image of Jundiai-3. (c) Manaus-1 and (d) evaluation image of Manaus-1.

TABLE I
MULTIPLICATIVE DISTRIBUTIONS FOR DIFFERENT
SATELLITE IMAGE REGIONS

Region	Distribution
Forested	$K_a(\alpha, \lambda, n) = \Gamma^{1/2}(\alpha, \lambda) \times \Gamma^{1/2}(n, n)$
Urban	$G_a^0(\alpha, \gamma, n) = \dfrac{\Gamma^{1/2}(n,n)}{\Gamma^{1/2}(\alpha,\gamma)}$

Fig. 5 shows the images generated using the evaluation images Jundiai-3 and Manaus-1. The advantage of such database is that the evaluation images are perfectly segmented from the test images. Thus, the error rate represents an exact number of misclassified pixels, in opposition to the experiments performed with the real satellite image database where the evaluation images may contain some inaccuracy. The satellite images
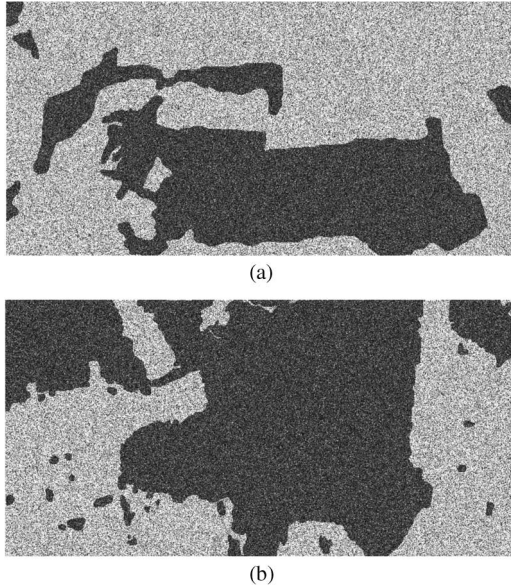
Fig. 5. Examples of synthetic image. (a) Simulation of Jundiai-3. (b) simulation of Manaus-1.

Fig. 6. Example of images from the MIT CBCL Face Database. (a) Training face image. (b) Testing face image. (c) Training nonface images. (d) Testing nonface images.

that compose the databases are available at http://cin.ufpe.br/~viisar/databases/SCRF.

For the face detection experiments, images from the MIT CBCL Face Database #1 [35] were used. The database contains 2429 face images and 4548 nonface images for training and 472 face images and 23 573 nonface images for testing. Fig. 6 shows examples of images taken from this database.

B. Experiments Using Real Satellite Images

This experiment shows the capability of the SCRF model and the LIPNet classifier used as a supervised image segmentation method. Recent works in satellite image recognition used different image spectra [36], [37] to accomplish the desired results. However, the task applied here detect forested areas in satellite images displayed in gray level. For comparison purposes, k-NN, MLP [38], and the Bayesian classifier using the histogram technique [39] were also tested.

The classification procedure for each method was calculated using a pixel-by-pixel comparison between the generated image and the manually segmented version of the same image. The error rate was then obtained by dividing the number of pixels

TABLE II
ERROR RATE IN PERCENT FOR THE FOREST DETECTION USING
LIPNet WITH DIFFERENT RECEPTIVE FIELD (R)
AND OVERLAP (O) CONFIGURATIONS

LIPNet Structure	Error Rate
$(r_1 = 3, o_1 = 1), (r_2 = 3, o_2 = 1)$	7.33
$(r_1 = 3, o_1 = 2), (r_2 = 3, o_2 = 1)$	**6.92**
$(r_1 = 4, o_1 = 2), (r_2 = 3, o_2 = 2)$	7.43
$(r_1 = 4, o_1 = 2), (r_2 = 4, o_2 = 2)$	7.36
$(r_1 = 5, o_1 = 2), (r_2 = 3, o_2 = 2)$	9.03
$(r_1 = 5, o_1 = 2), (r_2 = 4, o_2 = 1)$	8.96

TABLE III
ERROR RATE IN PERCENT FOR FOREST DETECTION USING THE
LIPNet WITH DIFFERENT LATERAL INHIBITION DIMENSION (H)
AND WEIGHT (δ) CONFIGURATIONS

LIPNet Structure	Error Rate
$(h_1 = 1, \delta_1 = 1.15), (h_2 = 0)$	**6.13**
$(h_1 = 2, \delta_1 = 1.15), (h_2 = 0)$	6.57
$(h_1 = 3, \delta_1 = 1.15), (h_2 = 0)$	6.88

misclassified by the total of pixels in the image. Each image was segmented ten times in each method. The results are reported using the mean error rate and standard deviation.

The neural network employed in this experiment was trained using the gradient descent method. Several configurations for the 2-D layers of the LIPNet without lateral inhibition combined with the SCRF model were tested. Table II presents the mean error rate of the tested images using different receptive field sizes and overlap regions. The best configuration presents an error rate of 6.92%, and it was obtained by the second configuration.

Different configurations for the lateral inhibition of the LIPNet model were also tested. Table III shows the error rate for different sizes of inhibitory fields using the same configuration of the best configuration without lateral inhibition. In this case, the first configuration showed the lowest error rate of 6.13%. Thus, the results of the following experiments were obtained using this configuration for receptive fields, inhibitory fields, and overlap factors.

Table IV presents the classification error rates for all the tested methods with the best results shown in bold. The deterministic methods k-NN and the Bayesian classifier using the histogram technique do not present standard deviations. Each segmentation method follows a specific nomenclature, as described hereinafter:

1) *SCRF-LIPN*: the SCRF model combined with the LIPNet classifier;
2) *SCRF-PN*: the SCRF model combined with the PyraNet classifier;
3) *SCRF-NN*: the SCRF model combined with the k-NN classifier;
4) k-NN: the k-NN classifier applied pixel by pixel with $k = 100$;
5) MLP: the MLP classifier applied pixel by pixel;
6) BHT: Bayesian classifier with the histogram technique applied pixel by pixel.

The k-NN applied pixel by pixel had the highest error rate. The MLP classifier obtained the second highest error rate, and the BHT showed a slight improvement, reaching an error

TABLE IV
ERROR RATE IN PERCENT FOR FOREST DETECTION USING THE SUPERVISED CLASSIFIERS ($\bar{X}(s)$)

Image	SCRF-LIPN	SCRF-PN	SCRF-NN	100-NN	MLP	BHT
Jundiai-1	$5.94(1.25)^-$	**4.13(0.28)**	28.17	37.81	14.89(4.54)	20.65
Jundiai-2	$\mathbf{7.89(0.27)}^+$	10.35(1.11)	19.80	35.77	18.22(0.86)	18.31
Jundiai-3	$\mathbf{8.43(0.22)}^\sim$	8.75(0.47)	13.61	35.77	15.45(0.83)	14.66
Manaus-1	$6.07(0.53)^+$	7.77(0.60)	**5.67**	55.73	17.75(3.56)	12.82
Manaus-2	$6.08(0.39)^+$	7.29(0.58)	**5.49**	53.63	15.13(1.55)	12.94
Manaus-3	$8.47(0.66)^+$	10.79(0.88)	**6.48**	16.03	27.42(3.64)	22.47
Manaus-4	$\mathbf{6.51(0.37)}^\sim$	6.67(0.52)	14.79	31.75	26.33(3.59)	21.46
Recife-1	$2.61(0.27)^+$	3.38(0.45)	3.65	48.26	2.38(0.37)	**1.85**
Recife-2	$\mathbf{3.09(0.38)}^\sim$	3.20(0.28)	4.91	47.18	3.05(0.73)	3.75
X	**6.13**	6.93	11.40	40.21	15.62	14.31

rate of 14.31%. It is also important to note that the lowest error rates for almost all the images were obtained using the SCRF model. The SCRF-NN model showed the lowest error rate in three images, and BHT presented the best result for one image, while SCRF-LIPN had the best results in the other images and achieved the lowest mean error rate of 6.13%. The hypothesis tests performed with Student's t-test with 5% of significance level demonstrate that the classifiers SCRF-LIPN and SCRF-PN perform differently in six images where "+" and "−" mean that the null hypothesis was rejected and the SCRF-LIPN presented a better or worse error rate than the SCRF-PN, respectively, and "∼" means that the results are not significantly different. An important observation is that the SCRF-LIPN takes less than a second to perform the segmentation task, while the SCRF-NN takes more than 77 s.

The standard deviations also points to the fact that the LIPNet classifier is more stable. Taking into consideration that the LIPNet and the PyraNet were trained with the same number of epochs, we can conclude that the presence of lateral inhibition increases the convergence speed of the neural network.

C. Experiments Using Synthetic Satellite Images

Despite the low error rates, the experiments performed with real satellite images might not be exact since the manual segmentation of the image can contain some inaccuracy. Thus, this experiment used synthetic images that were automatically generated from the images in the previous experiment. The evaluation images correspond 100% to each segmented test images since the test image was generated using its evaluation image. Thus, the obtained error rate for each classifier is the exact number of misclassified pixels. The neural networks were trained using the resilient propagation method.

Since the data of this experiment were generated by statistical functions that represent a region with the pixel intensity following a given distribution, the pixel-to-pixel classifiers (k-NN, MLP, and Bayesian classifier with the histogram technique) were not tested. The results shown here were also derived from a pixel-to-pixel comparison between the image generated by the segmentation algorithm and its original segmentation.

As in the previous experiment, it was assumed that the LIPNet and PyraNet have the same receptive fields and overlap region. However, different configurations for the lateral inhibition of the LIPNet were also tested. Table V presents the error rate for different sizes of inhibitory fields. The last tested configuration achieves the best classification rate.

TABLE V
ERROR RATE IN PERCENT OF THE FOREST DETECTION IN SYNTHETIC IMAGES USING THE LIPNet WITH DIFFERENT LATERAL INHIBITION CONFIGURATIONS

LIPNet Structure	Error Rate
$(h_1 = 1, \delta_1 = 0.9), (h_2 = 0)$	7.19
$(h_1 = 2, \delta_1 = 0.7), (h_2 = 0)$	6.83
$(h_1 = 2, \delta_1 = 0.7), (h_2 = 2, \delta_2 = 0.8)$	6.71
$(h_1 = 2, \delta_1 = 0.7), (h_2 = 3, \delta_2 = 0.8)$	**6.68**

TABLE VI
ERROR RATE IN PERCENT FOR FOREST DETECTION IN THE SYNTHETIC IMAGES ($\bar{X}(s)$)

Image	SCRF-LIPN	SCRF-PN	SCRF-NN
Jundiai-1	$\mathbf{8.84(0.35)}^\sim$	9.39(0.76)	10.79
Jundiai-2	$\mathbf{8.08(0.17)}^+$	8.57(0.58)	11.05
Jundiai-3	$\mathbf{6.97(0.25)}^+$	7.54(0.87)	9.77
Manaus-1	$\mathbf{6.55(0.24)}^+$	7.16(0.78)	9.27
Manaus-2	$\mathbf{4.84(0.10)}^+$	5.30(0.53)	6.35
Manaus-3	$\mathbf{5.55(0.24)}^+$	6.07(0.64)	7.46
Manaus-4	$\mathbf{13.17(0.15)}^+$	14.17(0.92)	15.16
Recife-1	$\mathbf{3.25(0.09)}^+$	3.92(0.64)	4.99
Recife-2	$\mathbf{2.85(0.14)}^+$	3.37(0.57)	4.11
X	**6.68**	7.28	8.77

Table VI presents the error rates and the respective standard deviations obtained by each classifier for all the synthetic images. The combination between the SCRF model and the LIPNet classifier obtained the lowest error rates for all the tested images, and the Student's t-test performed demonstrates that the proposed model performs statistically better than the SCRF-PN in most of the cases. The same nomenclature and significance level of the previous experiment are also used here.

D. Face Detection Experiment

The face detection problem can be summarized as the dichotomy task of determining if a given pattern is a face or not. Makinen and Raisamo [40] presented an evaluation of different methods for face alignment in order to improve the face detection rates. They demonstrated that manual alignment was the only kind of alignment that improved the detection rate. In all of the evaluated cases, support vector machine (SVM) [41] showed the best classification rates. Osuna *et al.* [42] also showed the advantages of using an SVM approach to perform face detection, and Waring and Liu [43] applied a method composed of spectral histograms and SVMs to achieve better results. Thus, the results obtained using LIPNet, PyraNet,

TABLE VII
AREA UNDER THE ROC CURVE FOR THE FACE
DETECTION USING PYRANET

LIPNet Structure	AUC
$(r_1 = 3, o_1 = 1), (r_2 = 3, o_2 = 1)$	0.833
$(r_1 = 4, o_1 = 2), (r_2 = 3, o_2 = 2)$	0.844
$(r_1 = 4, o_1 = 1), (r_2 = 4, o_2 = 0)$	**0.860**
$(r_1 = 4, o_1 = 2), (r_2 = 4, o_2 = 2)$	0.844
$(r_1 = 5, o_1 = 2), (r_2 = 3, o_2 = 2)$	0.851
$(r_1 = 5, o_1 = 2), (r_2 = 4, o_2 = 1)$	0.849

TABLE VIII
AREA UNDER THE ROC CURVE FOR THE FACE
DETECTION USING LIPNET

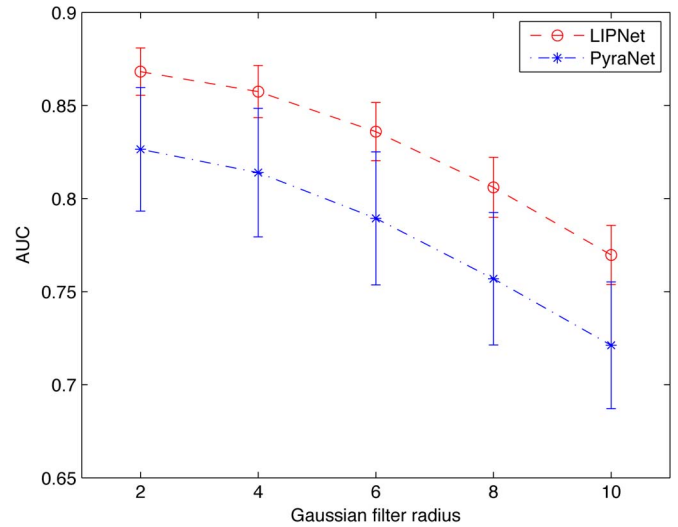| Lateral Inhibition Structure	AUC
$(h_1 = 1, \delta_1 = 3.0), (h_2 = 0)$	0.872
$(h_1 = 2, \delta_1 = 1.8), (h_2 = 0)$	0.875
$(h_1 = 2, \delta_1 = 1.6), (h_2 = 0)$	**0.878**
$(h_1 = 2, \delta_1 = 1.4), (h_2 = 0)$	0.873
$(h_1 = 3, \delta_1 = 2.8), (h_2 = 0)$	0.870
$(h_1 = 4, \delta_1 = 2.4), (h_2 = 0)$	0.872
$(h_1 = 5, \delta_1 = 2.2), (h_2 = 0)$	0.871

Fig. 7. Comparison among the area under the ROC curves of LIPNets and PyraNets with the test images blurred with a Gaussian filter with different radius sizes (2, 4, 6, 8, and 10).
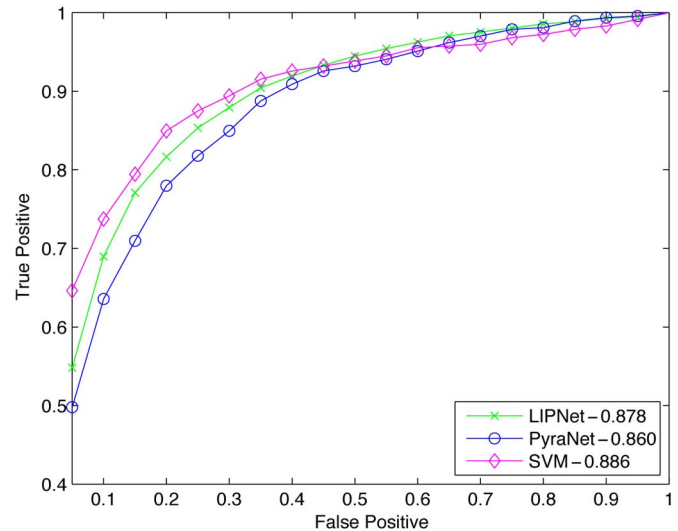

Fig. 8. Comparison among LIPNet, PyraNet, and SVM classifiers.

TABLE IX
COMPUTATIONAL COST OF THE LIPNET, PYRANET,
AND SVM CLASSIFIERS

Classifier	CPU Time (ms)	Stored parameters
LIPNet	0.04	447
PyraNet	0.04	443
SVM	7.00	89,066

and SVM for the face detection problem are presented. In this experiment, all the images had their histograms equalized.

The experimental results are presented using ROC curves [44] that present the true positive classification rates for different false positive error rates. The true positive rate is defined as the number of faces correctly classified divided by the total of faces in the database, while the false positive rates are defined as the number of nonface images wrongly classified divided by the total of nonface images in the database. The result of the ROC curve is determined by the area under the curve (AUC) [45].

Many different configurations for the LIPNet were tested. Table VII presents the results obtained for different LIPNet configurations without lateral inhibition. The third configuration presents the highest AUC. Table VIII shows the results obtained for different lateral inhibition configurations in the LIPNet with the same structure. The highest AUC obtained was 0.878.

Fig. 7 shows error bar graphs from the comparison between the AUCs of the LIPNet and the PyraNet with the test images blurred with a Gaussian filter with different radius sizes (2, 4, 6, 8, and 10). The LIPNet presents statistically better results than the PyraNet with a Student's t-test with 5% of significance level, and the highest difference between the LIPNet and PyraNet rates is obtained with the maximum radius in the Gaussian filter tested. These results are consistent with the results demonstrated by Fukushima [13] that state that the presence of lateral inhibition improves the model in order to recognize the patterns even after a blurring operation. The LIPNet also presents a lower confidence interval than the PyraNet.

Fig. 8 shows a comparison among LIPNet, PyraNet, and SVM with the best configurations. The LIPNet model showed better results compared with the PyraNet, but SVM presented the biggest AUC. However, the LIPNet obtained the best classification rate when the false positive rates were greater than 40%. Also, the use of the LIPNet might be justified because of the time needed to perform the classification. While the LIPNet takes only 0.04 ms to classify a pattern and it has 447 stored parameters, the SVM takes 7 ms, and it has, on average, 361 support vectors or 89 066 stored parameters. This means that

LIPNet is 175 times faster than SVM and consumes only 0.5% of the memory in comparison with SVM. Table IX presents the computational cost of the LIPNet, PyraNet, and SVM in the face detection task. The experiments were performed over a Pentium Dual Core with a 1.73-GHz CPU and 2-GB RAM.

Taking into consideration that LIPNet performs much faster than SVM, a committee [46] of 20 LIPNets was tested having as output the average value of all neural networks. The committee of LIPNets obtained an AUC of 0.892 against the 0.886 obtained by using SVM, and it is still more than eight times faster than the SVM model. The comparison is presented in Fig. 9.
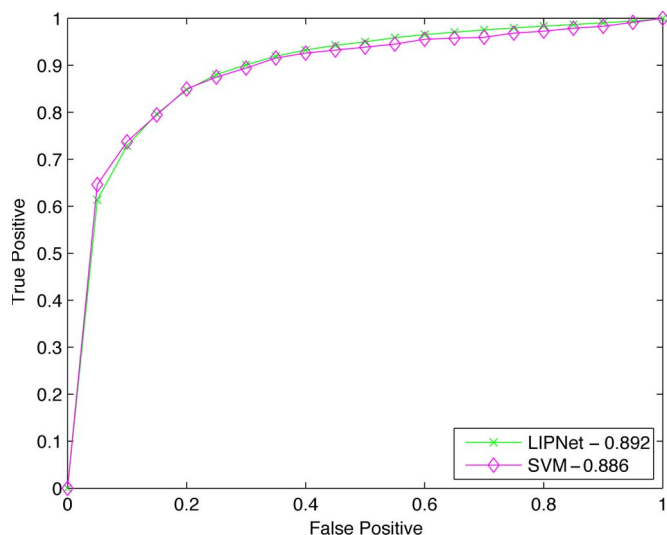
Fig. 9. Comparison among the committee of LIPNets and the SVM classifier.

V. CONCLUSION

This work presented two new models inspired by biological behaviors of the brain. The first model was developed to classify 2-D patterns (images), based on a combination of the neural network PyraNet and the concepts of inhibitory fields, called LIPNet, and the second model was developed as a supervised segmentation method based on the concepts of receptive fields, called SCRF. Both models were applied in experiments involving satellite image segmentation, and LIPNet was applied in a face detection problem. The proposed models achieved good results, and they have the following advantages: low error rate, fast performance, and low memory consumption. The use of the inhibitory concepts creates a new neural network (LIPNet) which is more stable and efficient.

Moreover, the fact that the LIPNet is much faster and consumes less memory in comparison to other methods such as SVM motivates its use in embedded systems, where processing restrictions are important. Other training algorithms can be used to improve the LIPNet results, such as neuroevolutionary methods.

REFERENCES

[1] D. H. Hubel, "The visual cortex of the brain," *Sci. Amer.*, vol. 209, no. 5, pp. 54–62, Nov. 1963.

[2] M. Levine and J. Shefner, *Fundamentals of Sensation and Perception*. London, U.K.: Oxford Univ. Press, 2000.

[3] G. Rizzolatti and R. Camarda, "Inhibition of visual responses of single units in the cat visual area of the lateral suprasylvian gyrus (Clare-Bishop area) by the introduction of a second visual stimulus," *Brain Res.*, vol. 88, no. 2, pp. 357–361, May 1975.

[4] C. Sun, X. Chen, L. Huang, and T. Shou, "Orientation bias of the extraclassical receptive field of the relay cells in the cat's dorsal lateral geniculate nucleus," *Neuroscience*, vol. 125, no. 2, pp. 495–505, Feb. 2004.

[5] C. Grigorescu, N. Petkov, and M. A. Westenberg, "Contour detection based on nonclassical receptive field inhibition," *IEEE Trans. Image Process.*, vol. 12, no. 7, pp. 729–739, Jul. 2003.

[6] J. Daugman, "Two-dimensional analysis of cortical receptive field profiles," *Vis. Res.*, vol. 20, no. 10, pp. 847–856, 1980.

[7] J. Jones and P. Palmer, "An evaluation of the two-dimensional Gabor filter model of simple receptive fields in cat striate cortex," *J. Neurophysiol.*, vol. 58, no. 6, pp. 1233–1258, Dec. 1987.

[8] H. Sun, L. Liu, and A. Guo, "A neurocomputational model of figure-ground discrimination and target tracking," *IEEE Trans. Neural Netw.*, vol. 10, no. 4, pp. 860–884, Jul. 1999.

[9] H.-S. Park, W. Pedrycz, and S.-K. Oh, "Granular neural networks and their development through context-based clustering and adjustable dimensionality of receptive fields," *IEEE Trans. Neural Netw.*, vol. 20, no. 10, pp. 1604–1616, Oct. 2009.

[10] K. Ghosh and S. Pal, "Some insights into brightness perception of images in the light of a new computational model of figure-ground segregation," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 40, no. 4, pp. 758–766, Jul. 2010.

[11] K. Fukushima, S. Miyake, and T. Ito, "Neocognitron: A neural network model for a mechanism of visual pattern recognition," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-13, no. 5, pp. 826–834, Sep./Oct. 1983.

[12] K. Fukushima, "Neocognitron: A hierarchical neural network capable of visual pattern recognition," *Neural Netw.*, vol. 1, no. 2, pp. 119–130, Jun. 1988.

[13] K. Fukushima, "Neocognitron for handwritten digit recognition," *Neurocomputing*, vol. 51, pp. 161–180, Apr. 2003.

[14] C. A. Perez, C. A. Salinas, P. A. Estévez, and P. M. Valenzuela, "Genetic design of biologically inspired receptive fields for neural pattern recognition," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 33, no. 2, pp. 258–270, Apr. 2003.

[15] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, Winter 1989.

[16] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[17] F. H. C. Tivive and A. Bouzerdoum, "A new class of convolutional neural networks (SICoNNets) and their application of face detection," in *Proc. Int. Joint Conf. Neural Netw.*, 2003, vol. 3, pp. 2157–2162.

[18] F. H. C. Tivive and A. Bouzerdoum, "Efficient training algorithms for a class of shunting inhibitory convolutional neural networks," *IEEE Trans. Neural Netw.*, vol. 16, no. 3, pp. 541–556, May 2005.

[19] S. L. Phung and A. Bouzerdoum, "A pyramidal neural network for visual pattern recognition," *IEEE Trans. Neural Netw.*, vol. 18, no. 2, pp. 329–343, Mar. 2007.

[20] B. Nabet and R. Pinter, *Sensory Neural Networks: Lateral Inhibition*. Boston, MA: CRC Press, 1991.

[21] Z.-H. Mao and S. G. Massaquoi, "Dynamics of winner-take-all competition in recurrent neural networks with lateral inhibition," *IEEE Trans. Neural Netw.*, vol. 18, no. 1, pp. 55–69, Jan. 2007.

[22] R. Coultrip, R. Granger, and G. Lynch, "A cortical model of winner-take-all competition via lateral inhibition," *Neural Netw.*, vol. 5, no. 1, pp. 47–54, Jan. 1992.

[23] D. Chen, L. Zhang, and J. Weng, "Spatio-temporal adaptation in the unsupervised development of networked visual neurons," *IEEE Trans. Neural Netw.*, vol. 20, no. 6, pp. 992–1008, Jun. 2009.

[24] Y. Fang, M. Cohen, and T. Kincaid, "Dynamic analysis of a general class of winner-take-all competitive neural networks," *IEEE Trans. Neural Netw.*, vol. 21, no. 5, pp. 771–783, May 2010.

[25] P. Arkachar and M. D. Wagh, "Criticality of lateral inhibition for edge enhancement in neural systems," *Neurocomputing*, vol. 70, no. 4–6, pp. 991–999, Jan. 2007.

[26] N. R. Wilson, C. A. Runyan, F. L. Wang, and M. Sur, "Division and subtraction by distinct cortical inhibitory networks in vivo," *Nature*, vol. 488, no. 7411, pp. 343–348, Aug. 2012.

[27] C. Blakemore and E. Tobin, "Lateral inhibition between orientation detectors in the cat's visual cortex," *Exp. Brain Res.*, vol. 15, no. 4, pp. 439–440, Sep. 1972.

[28] C. M. Bishop, *Neural Networks for Pattern Recognition*. Oxford, U.K.: Clarendon, 2007.

[29] D. M. Kline and V. L. Berardi, "Revisiting squared-error and cross-entropy functions for training neural network classifiers," *Neural Comput. Appl.*, vol. 14, no. 4, pp. 310–318, Dec. 2005.

[30] D. Rumelhart, G. Hinton, and R. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, Oct. 1986.

[31] M. Riedmiller and H. Braun, "A direct adaptive method for faster back-propagation learning: The RPROP algorithm," in *Proc. IEEE Int. Conf. Neural Netw.*, 1993, pp. 586–591.

[32] B. J. T. Fernandes, G. D. C. Cavalcanti, and T. I. Ren, "Classification and segmentation of visual patterns based on receptive and inhibitory fields," in *Proc. 8th Int. Conf. Hybrid Intell. Syst.*, 2008, pp. 126–131.

[33] B. J. T. Fernandes, G. D. C. Cavalcanti, and T. I. Ren, "Nonclassical receptive field inhibition applied to image segmentation," *Neural Netw. World*, vol. 19, no. 1, pp. 21–36, Feb. 2009.

[34] A. C. Frery, H. Muller, C. Yanasse, and S. Sant'Anna, "A model for extremely heterogeneous clutter," *IEEE Trans. Geosci. Remote Sens.*, vol. 35, no. 3, pp. 648–659, May 1997.

[35] B. Heisele, T. Poggio, and M. Pontil, "Face Detection in Still Gray Images," Center for Biol. and Comput. Learn., MIT, Cambridge, MA, USA, Tech. Rep. 1687, 2000.

[36] Y. Venkatesh and S. K. Raja, "On the classification of multispectral satellite images using the multilayer perceptron," *Pattern Recognit.*, vol. 36, no. 9, pp. 2161–2175, Sep. 2002.

[37] K. Venkatalakshmi, S. Sridhary, and S. MercyShalinie, "Neuro-statistical classification of multispectral images based on decision fusion," *Neural Netw. World*, vol. 16, no. 2, pp. 97–107, 2006.

[38] S. L. Phung, A. Bouzerdoum, and D. Chai, "Skin segmentation using color pixel classification: Analysis and comparison," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 1, pp. 148–154, Jan. 2005.

[39] M. Jones and J. Rehg, "Statistical color models with application to skin detection," *Int. J. Comput. Vis.*, vol. 46, no. 1, pp. 81–96, Jan. 2002.

[40] E. Makinen and R. Raisamo, "Evaluation of gender classification methods with automatically detected and aligned faces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 3, pp. 541–547, Mar. 2008.

[41] V. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer-Verlag, 1995.

[42] E. Osuna, R. Freund, and E. Girosit, "Training support vector machines: An application to face detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 1997, pp. 130–136.

[43] C. Waring and X. Liu, "Face detection using spectral histograms and SVMs," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 35, no. 3, pp. 467–476, Jun. 2005.

[44] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874, Jun. 2006.

[45] A. Bradley, "The use of the area under the ROC curve in the evaluation of machine learning algorithms," *Pattern Recognit.*, vol. 30, no. 7, pp. 1145–1159, Jul. 1997.

[46] A. Verikas, A. Lipnickas, K. Malmqvist, M. Bacauskiene, and A. Gelzinis, "Soft combination of neural classifiers: A comparative study," *Pattern Recognit. Lett.*, vol. 20, no. 4, pp. 429–444, Apr. 1999.

Bruno José Torres Fernandes (M'11) received the B.Sc. and M.Sc. degrees in computer science from the Federal University of Pernambuco, Recife, Brazil, in 2007 and 2009, respectively, where he is currently working toward the D.Sc. degree.

He is an Assistant Professor with the University of Pernambuco, Recife. His research interests include machine learning, computer vision, image processing, and neural networks.

Mr. Fernandes was the recipient of several awards, including the 2008 Google Academic Prize as the Top M.Sc. Student in the Federal University of Pernambuco.

George D. C. Cavalcanti (M'09) received the D.Sc. degree in computer science from the Center for Informatics, Federal University of Pernambuco, Recife, Brazil.

He is currently an Associate Professor with the Center for Informatics, Federal University of Pernambuco. His research interests include machine learning, pattern recognition, computer vision, and biometrics.

Tsang Ing Ren (M'11) received the B.Sc. degree in electronic engineering from the Federal University of Pernambuco, Recife, Brazil, and the Ph.D. degree in physics from the University of Antwerp, Antwerp, Belgium.

He is currently an Associate Professor with the Center for Informatics, Federal University of Pernambuco. His research interests include machine learning, computer vision, image processing, and biometrics.