Revista de Sistemas de Informação da FSMA
n. 12 (2013) pp. 41-48

SISTEMAS DE INFORMAÇÃO
http://www.fsma.edu.br/si/sistemas.html

TRILHA ESTUDANTIL

Classification of Parking Spots Using Multilayer Perceptron Networks

Heráclito Santos Falcão, Agnaldo Volpe Lovato, Alex Ferreira dos Santos, Lucas Santos de Oliveira, Robson Hebraico Cipriano Maniçoba, Marcelo Alves Guimarães e Murilo Silva Santana

Departamento de Química e Exata (DQE). Universidade Estadual do Sudoeste da Bahia (UESB), Jequié, Bahia, Brasil.

***Abstract*— This project intends to develop a prototype for the identification of free spots in open air parking area where there is a good aerial view without obstacles, allowing for the identification of occupied and free spots. We used image processing techniques and pattern recognition using Artificial Neural Networks (ANN). In order to help implement the prototype, we used Matlab. In order to simulate the parking area, we created a model so that we could acquire the images using a webcam, process them, train the neural network, classify the spots and finally, show the results. The results show that it is viable to apply pattern recognition through image capture to classify parking spots.**

***Keywords*— Artificial Intelligence, Neural Networks, Pattern Recognition.**

I. Introduction

GIven the increased income of all Bazilian population and the need for faster displacement in cities, there is a remarkable increase in the number of cars in Brazilian cities. The increased number of cars is causing medium and large town to become increasingly congested [1]. According to Denatran (National Department of Transit), the Brazilian vehicle fleet grew 119% in ten years, and at the end of 2010 there were 64.817 million registered vehicles (Figure 1).

Given the changes Brazilian society underwent and the increased number of cars in the cities, there is a characteristic that became very relevant: the need to find a parking spot.

Nowadays there is no efficient solution to solve this kind of problem, especially in urban centers wherein is the larger number of vehicles. For drivers that daily need to travel to a congested downtown, finding a parking spot is usually a stressful situation [2], and not only in the larger urban centers. Even in leisure moments, finding a parking spot has become a mission that is akin to torture and highly dispiriting. Near the waterfront, it is hard to find parking spots and drivers usually resort to illegal car guardians or illegal parking [3]. Some entrepreneurs created some solutions, such as underground parking, but this may be too expensive for some regions, even thought this solution is in the plan to end the lack of parking spots in the central area of Brasilia and close to the Ministries [4].

Figura 1: Evolução do total da frota de veículos na década. Fonte: Denatran

Nowadays there are softwares that use ultrasound sensors to detect parking spots. This technique consists in installing presence sensors in each spot in a parking place that ultimately gives us the number of available sports in each wing or floor or the parking area. Nevertheless, its high costs makes it not viable for installation in urban centers, especially given that Brazil is still a developing country.

Recent researches indicate that the biggest causes for traffic jams in cities are drivers looking for parking spots. The search for parking spots causes slow traffic and infractions, such as double parking and parking on walkways. [5].

Nowadays, given the popularization of low cost equipment capable of recording, transmitting and storing video it is possible to obtain dynamic or static information (video or photo) in an efficient way. Hence, the possibility of processing those images and classifying a parking spot is already a reality.

Artificial intelligence (IA) is in the Field of task systematization and automation. Hence, it is potentially relevant for any human intellectual task [6]. Inside this field there are the Artificial Neural Networks (ANN), which, according to Braga [7], is a non algorithmic computation form that is characterized by structures that resemble the structure of the human brain.

ANNs are built with neurons, a simple processing unit, whose goal is to compute certain mathematical functions that are usually non linear. These units are disposed in one or more layers and interconnected by a large number of connections that are usually unidirectional. In most models these connections are associated with weights that store the knowledge represented in the model and serve to weight the input received by every neuron in the network [14].

Neural networks are one of the most successful branches of artificial intelligence and have a wide range of applications. According to Braga [7], the solution using ANNs is very attractive.

This project intends to present how to use images captured through a camera in order to help identify free and occupied parking spots in a parking area. For that we will develop a neural network that may help in the classification of the parking spot state, given its ability to solve problems and for generalization.

I. THEORETICAL FOUNDATIONS

A. *Image Processing*

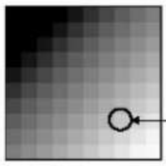
According to Marques Filho [8], digital image processing includes procedures that are usually expressed in an algorithmic form. This way, most of these image processing functions are performed using computers.

In [9] the author understands digital image processing as the manipulation of a computer image in a way that both the input and output of the process are images. The goal of this technique is to improve certain imperfections (such as noise or inadequate contrast) in order to improve the image acquired for the next steps.

In [8] it is pointed out that image processing makes viable a large number of applications in two different categories: (1) the improvement of pictorial information for human interpretation; and (2) the automatic analysis by a computer of information extracted from a scene.

The digital image is simply a matrix in which lines and columns identify a point in the image. A digital image is, hence, a function *f(x,y)* discretized both in amplitude and in space [8].

The words *picture element* (pixel) refer to the smallest unit of a digital image. In order to create a complex image we need a set of several (or even millions) pixels. In Figure 2, it is possible to see the representation of an mage with 256 shades of grey.



	01	02	03	04	05	06	07	08	09	10
01	0	9	26	48	72	92	110	127	142	154
02	10	25	45	68	91	109	127	142	154	165
03	27	45	68	91	112	127	141	155	167	178
04	48	67	91	111	128	140	153	165	178	190
05	72	91	112	129	144	155	166	178	190	202
06	91	109	127	141	154	166	179	190	202	213
07	110	126	141	153	165	178	190	202	214	226
08	128	142	156	165	178	191	203	214	226	238
09	142	154	166	177	190	201	214	226	241	249
10	154	167	179	190	203	214	228	239	247	254

Figure 2:Representation of an enlarged image of 10X10 pixels with 256 shades of grey [10].

After the image acquisition, we need to use image processing techniques to improve it, increasing the chances of success in the next steps [11]. Image processing consists in applying the necessary transformations to make it easier to extract the information contained in the image for a specific application. The techniques used are classified into two main categories, the first of which refers to the spatial domain and the second one to the direct manipulation of pixels in an image.

In this last technique, the image properties follow three different factors: first, the location of the point under analysis, as defined by its row and column; second, the radiometric resolution of the sensor system; and last, the spatial resolution, as defined by the pixel size.

B. *Artificial Neural Networks*

Artificial Neural Networks are a branch of Artificial Intelligence that Braga [12] states to be a non algorithmic form characterized by systems that resemble the structure of the human brain. Besides, given that neural computation is not based on rules, it becomes an alternative to the traditional algorithmic computation.

ANNs have the ability to calculate, store and distribute information in the interconnected network. They are distributed systems made of simple processing units (artificial neurons) that calculate certain mathematical functions, which are usually non linear. These units are organized into one or more layers and interconnections, usually unidirectional [7]. Braga [12] points out that in most models these connections are associated with weights whose function is to store the knowledge acquired by the model and are used to weight the input received by every neuron in the network.

Haykin [13] states that a neural network is a massively parallel processing unit made of several simple processing units, with a natural propension to store experimental knowledge and making it available for posterior use.

The steps an ANN takes to solve a set of problems begins with learning, where a set of examples is presented to the network, which extracts the characteristics of the problem. Based on those characteristics, it is possible to reuse them in the next steps in order to find possible answers to specific problems. One can say that neural networks can extract information presented non explicitly through examples [13].

The ability to learn through examples and to generalize the learned information is, no doubt, the main attraction of solving problems using ANNs.

Individual neurons have limited ability, but when there is a set of artificial networks connected to make a network, then this ability is largely increased, being able to solve high complexity problems.

Santiago Ramon y Cajal (1911) defined the neuron as the basic structure in the brain, being responsible for sending and receiving information and acting through the production and transmission of electric signals called neural impulses. It is basically made of dendrites and axons that respectively receive and transmit electric signals to other neurons and the cellular body (soma), responsible for the processing of those signals (Figure 3).

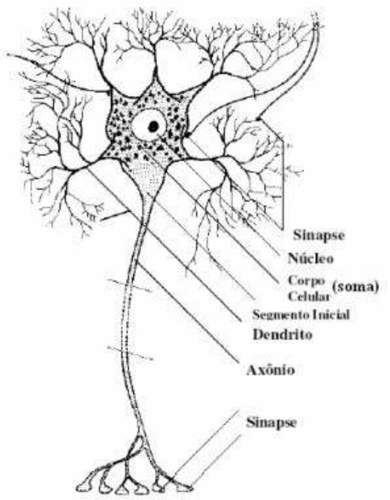


Figure 3: Schematics of a biological neuron [14].

In 1943, McCulloch and Pitts proposed a simplified mathematical model of the workings of the biological neuron (Figure 4). In this model the neuron has i inputs (corresponding to the dendrites) x_1, x_2, …,x_i and a single output (equivalent to the axon) y. In order to simulate the synapses, each input is associated to a weight w_1, w_2,…, w_i whose values can be either positive (excitatory) or negative (inhibitory). The goal of the weights is to store the knowledge and determine the intensity by which each input will contribute to the result of the neuron. The cellular body is simulated simply adding the values of the products of the inputs by their respective weights ($x_i.w_i$) and, if the sum is equal to or exceeds a specific threshold, the output is activated, with value 1.

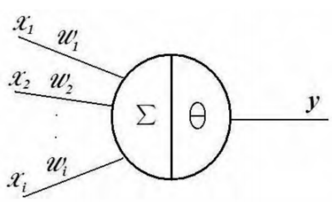


Figure 4: Mathematical model of a biological neuron [7].

In the following years, other researchers proposed changes to the McCulloch-Pitts model and the most important were performed in the activation function, with the inclusion of a bias.

An artificial neuron is composed by a set of inputs x_1, x_2, x_3...x_n which are the signals coming from the other neurons and an output y_k, (Σ) adders which is responsible for the effective calculation of the input, the weights that represent the synaptic weight of the connections between the neurons of the previous layer and of the current one, a *bias* (*Bk*) whose effect is to increase or decrease the net result of the activation function (the bias acts as an extra weight in the connections of the unit and its value is always 1 and an activation function $\varphi(.)$ (Figure 5).

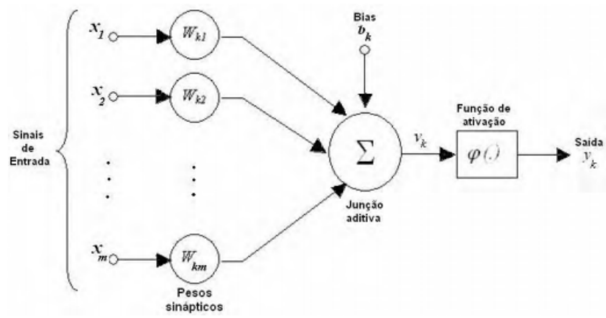


Figure 5: Base model for a neuron for ANN projects [13].

In an ANN structure for a specific problem, it is necessary to understand the following aspects: [12]:

- Problem complexity;
- Dimension of the input space;
- Dynamic and static characteristics;
- A priori knowledge on the problem;
- Data representativity;

These factors will influence directly the architecture or topology of the ANN.

As to the number of layers:

- Single layer networks – is the simplest form of a layered networks, which has a single input layer that projects the result to the output;
- Multiple layer networks – is different from the single layer network due to the present of one or more hidden layers.

Types of connections:

- *Feedforward* or acyclic – the output of the layer i does not input to layers whose index is smaller or equal to i;
- *Feedback* or cyclic – the output of a neuron from layer i is input to neurons in layers whose index is smaller or equal to i.

As to the process of learning in an ANN, Carlos [15] points out that the knowledge is store in the weights and in the topology. The weights represent the network knowledge and determine the importance of a specific input.

In the process of learning for an ANN the are two basic types: supervised and non-supervised. The type of learning is determined by the way the parameter modification occurs [13].

Supervised learning implies in the existence of an external supervisor or teacher, who is responsible to analyze and select the network inputs patterns and watch the output found, matching it to the desired output.

Braga [12] states that the supervised learning is applied to problems where we want a specific mapping from input and output patterns. The best known example is the Delta rule and its generalizations from multiple layer networks, such as the back-propagation algorithm.

In the non-supervised learning there is not external teacher to analyze – only the input patterns are available. This model is used to problems that intend to find relevant characteristics in the input data, such as the discovery of clusters or classes [14].

The ANNs evolved in the 1950s and in 1958 Frank Rosenblatt proposed the concept of learning to ANNs. Hence came the perceptron, composed by the McCulloch-Pitts structure, becoming then the simples ANN for pattern classification that could work for linearly separable problems, that is patterns that reside in opposed sides of a hyperplane [13].

Braga [12] explains that the original topology consisted of an input layer (retina), by an intermediate level formed by the association units and by an output level formed by the response unit. In spite of this topology having three layers, it is known an single layer perceptron, due to the fact that only the output layer (the response unit) presents adaptative properties. The retina consists of sensor units and the association unit, in spite of being formed by McCulloch-Pitts (MCP), has fixed weights defined in the training period.

In Figure 6 there are five inputs represented as x_1,...x_5, whose weights (synapses) are w_1,...,w_5, w_0 (bias) in the neuron input, d is the intermediate output and O is the network activation output.

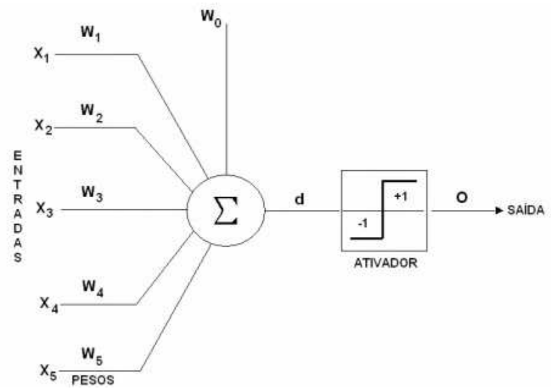


Figure 6: Perceptron model [16]

Medeiros [16] and Haykin [13] discuss the limitation of the Perceptron model for some problems that are not linearly separable. A possible solution would be increasing the training set to further refine the set of weights found, but there may be no well defined class separation, what may cause classification erros. Besides, there may also be specifics characteristics of the problem under study which may not allow a clear dividision between two or more classes through well defined planes. This means that single layer networks have limitations to solve only problems with linear characteristics.

Nevertheless, the multiple layer Perception is able to solve the non linear problems. Since most real problems are non-linear, then it is more adequate to problem resolution. The multilayer perceptron was created to solve complex problems that would not be solvable with the basic neuron model.

This neural network made of neurons with sigmoid functions in the intermediate layers are classed Multi-Layer Perceptrons (MLPs) [12]. Basically, a MLP network is made of an input layer, one or more hidden layer and an output layer and to train it we use the backpropagation algorithm [15].

In Figure 7 we can see the architecture of a MLP network with an input layer that has three neurons, two hidden layers, each one of which has four neurons and an output layer with two neurons. We can see that in the model every neuron in each layer is connected to all the neurons in the previous layers. The number of hidden layers and the number of neurons in each one of them is defined empirically.

Multilayer perceptrons have been applied successfully to solve many difficult problems, through their supervised training with error backpropagation [13]. Application examples include character recognition, stock price forecasting, signature verification, credit card transactions security, medical diagnosis and others [7].

In order to train the Multilayer Perceptron we use the backpropagation algorithm that has two phases: the first is the propagation of the signal (forward) and the second is the propagation (backward) of the error signals through the network.

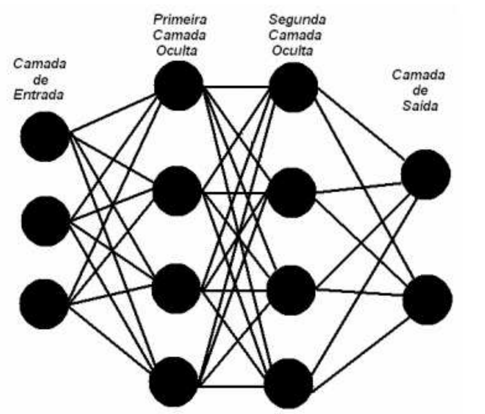


Figure 7: Architecture of a MLP ANN with two hidden layers.

In the propagation phase occurs the dat ainput through the retina (input layer), its propagation through the network and the generation of an output. Hence, this output is compared to the desired output and the error value is calculated. After that, we start the backpropagation phase and this error is propagated back to the previous layers of the neural network and used to adjust the weights, with the intention to reduce as much as possible the errors committed in each iteration so that the result becomes closer to the desired output (Figure 8).

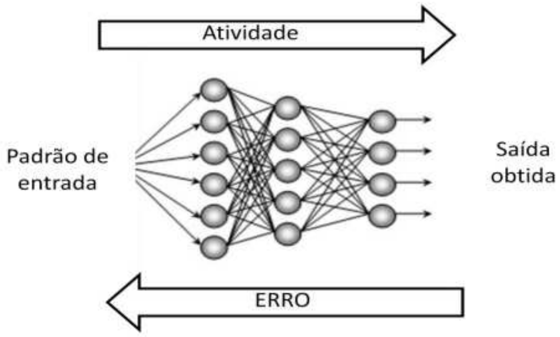


Figure 8: Schematics of the workings of the Backpropagation algorithm

The way the neurons are organized is strictly connected to the type of problem being solved. Hence, the neural networks are divided in three groups: feedforward networks with a single layer, feedforward networks with multiple layers and recurrent networks.

II. Project Execution

For the development of this project we used a webcam, a computer, a wood model, miniature cars and a simulation tool. Figure 9 shows the model and its components.

Figure 9 – Parking space model

Initially we researched existing Matlab functions to help capture and process images and then we analyzed which would be the best neural network model. We selected the MLP (*Multi-Layer Perceptron*) due to its generalization power and ability to solve problems with which the Perceptron is not able to deal.

The next step was to define the network architecture, modeling the number of neurons in each layer and the number of hidden layers. Afterwards, we acquired the data, trained the network and validated the training results, repeating this process each time the network did not present the expected results (Figure 10).

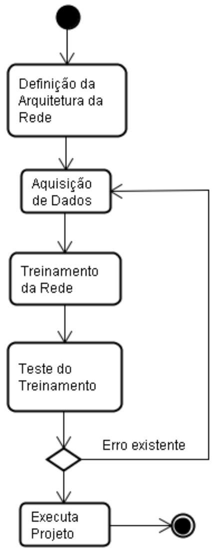


Figure 10 – Training process

All the development was performed using the Matlab tool and its library available for modeling and executing neural networks. The next sections will show these steps in further detail.

A. *Definition of Network Architecture*

In order to define the network architecture, it is important to initially analyze which will be the input and output data. The input data will be part of the image captured by the webcam which represent a single parking spot. In Figure 11 we can see the green rectangle that represents a parking spot captured for training. Each sport has a size of 39 x 24 pixels, which will be converted into a one-dimensional matrix with 936 rows which will be used as input to the network. The output will be represented by a single network that will return 0 if the parking spot is free and 1 if it is occupied.

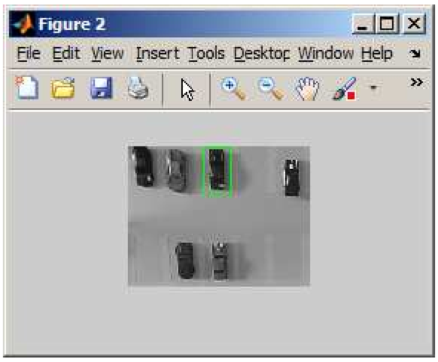


Figure 11 – Image of a parking place with the cut on a single spot (green rectangle).

The definition of the number of hidden layers and the number of neurons in each layer is done empirically. Studies performed showed that a single hidden layer in the MLP can solve classification problems, the type of problem we are trying to deal with. Hence, we decided to use a single hidden layer with five neurons, which were enough to learn the mapping function (Figure 12).

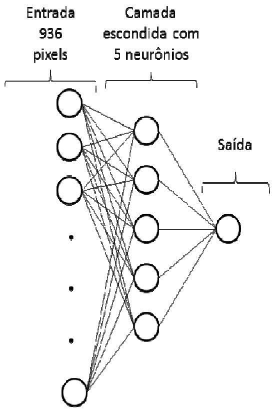


Figure 12 – Network architecture

B. Data acquisition

In order to acquire adata, we thought about analyzing a single spot (Figure 11), given that the neural network ability to generalize would allow it to deal with the other spots that have similar characteristics.

To capture the information we implemented the algorithm *AquisicaoDados.m*. In this algorithm the data are captured by the webcam (Figure 13) and the user informs the system whether the spot is free (zero) or occupied (one). Next, the algorithm asks if a new image must be captured.

During training, the image acquired was submitted to different lighting effects and ways to park the vehicle. Only 3 vehicles were used in this process, in order to test afterwards the neural network generalization ability.

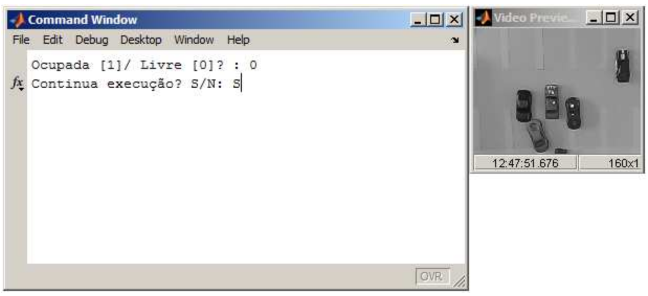


Figure 13 – Data Acquisition

C. Network training

The data acquired in the previous step were submitted to the network for training and for that we developed the algorithm *TreinamentoRede.m*. A total of 5 epochs were necessary to finalize the network training (Figura 14). We submitted 63 different images to the network, including free and occupied spots.

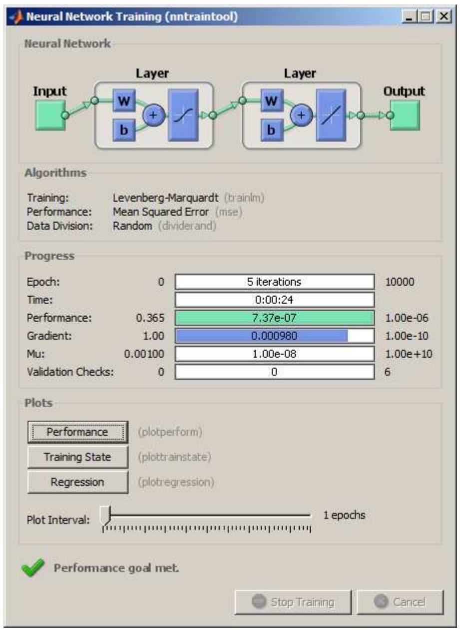


Figura 14 – Network training

At this stage, the network achieved a minimum for the error between expected and calculated output, given its weights updated through the backpropagation algorithm.

D. Training test

To validate the network training, we develop the algorithm *TesteTreinamento.m*. In this algorithm the captured images were used to identify the spot situation (free or occupied). In this phase, the captured images are different from those used in the training, in order to analyze the network ability for generalization.

Similar to what was done in training, different situations were submitted to the algorithm, using different vehicles from those used during training and different parking positions. It is important to state that initially a smaller number of images were submitted, achieving unsatisfactory results for the network. Following the process described in Figure 10, new images were acquired and presented for a new training.

Figure 15 and Figure 16 represent the network output when the parking spot is occupied or free.

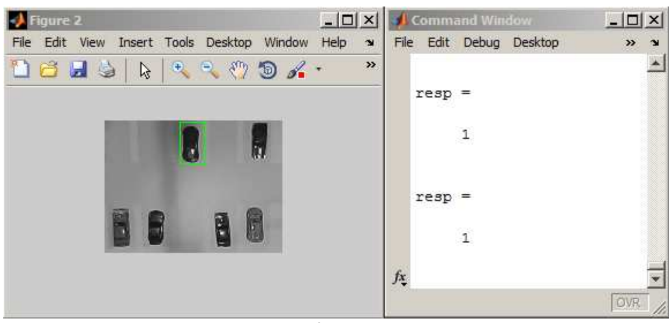


Figure 15 – Test for an occupied spot.

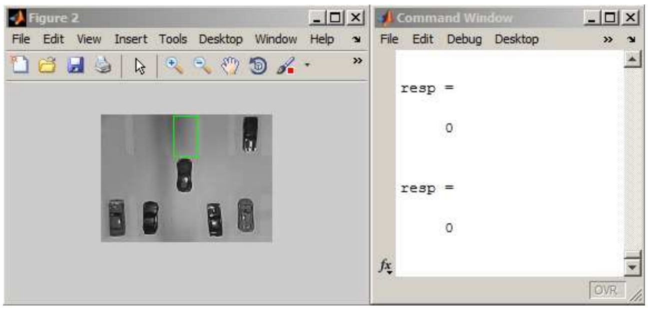


Figure 16 – Test for a free spot.

III. RESULTS FOUND

In order to present the results found, we developed the algorithms *ProjetoFinal.m* and *ProjetoFinalRapido.m*. It is important to stress that the second algorithm presents the same results in a faster way. While *ProjetoFinal.m* presents the results as images (Figura 17), *ProjetoFinalRapido.m* presents the results in matrix format (Figure 18).

The free spots are represented in green color and the occupied ones in red color.

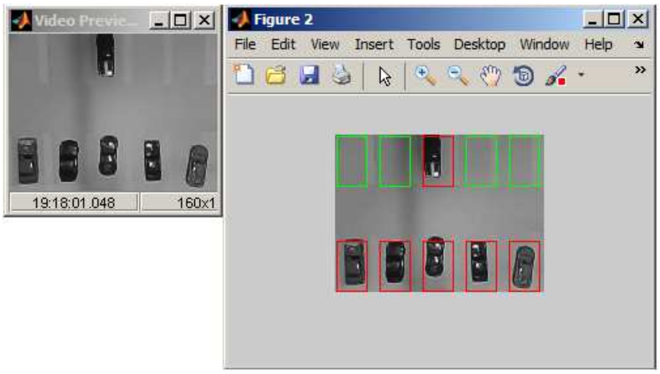


Figura 17 - ProjetoFinal – Result in image format

In Figure 18, zeroes represent empty spots and 1, the occupied spots.

In these algorithms, the network used for a single spot was replicated for the other spots, based on its ability for generalization.

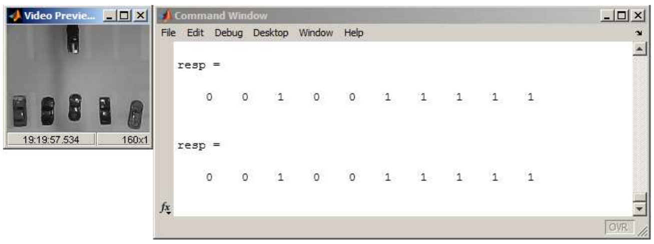


Figure 18 - ProjetoFinal – Text results

Figure 19 presents the individual spot test, where each spot was checked using a single vehicle.

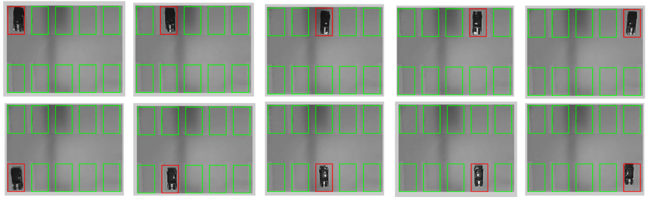
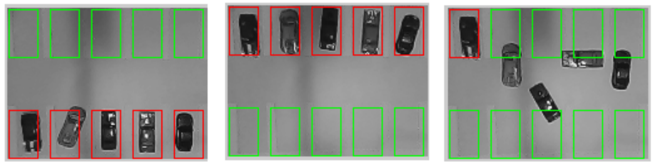
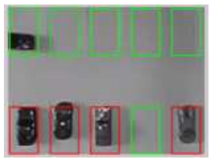


Figure 19 – Individual spot test

Figure 20 presents the collective spot test and in this test we inserted some vehicles partially in some spots (Figure 20c), not representing the occupation of those spots, but the movement of vehicles or their parking. Figure 20d shows that a vehicle must be correctly parked for the spot to be considered occupied.



(a) (b) (c)



(d)

Figure 20 – Collective spot test

IV. CONCLUSION

The project showed itself as remarkably viable for the identification of free spots in parking garages since the tests performed showed significative results. It was possible to attest the efficiency of the project using image processing techniques and pattern recognition using ANNs with the help of Matlab.

In order to execute this job, we considered the current need to facilitate the process of finding free parking spots, reducing the time needed to find them. For that, we observed the current discussions on this topic as well as the need for more affordable and reliable solutions.

The most relevant facts that can be taken into consideration is the idea of automatizing the system, in which robust and reliable methods are used in open parking garages where there is a good aerial view.

The results found show possibilities of perfecting the project for its implementation, given that good neural network training may make it viable to adapt the prototypes for environments where it is possible to capture the images. Given that the project uses the image capture in real time, these data can in the future be transmitted for the driver through Internet, Bluetooth, Wi-Fi, or other communication mechanisms.

REFERENCES

[1] BUARQUE, Ecomensagem, Asfixia_das_Cidades. Disponível em: http://www.ecolnews.com.br/asfixia_das_cidades.htm. Acessado em: 25 de jul. 2011

[2] MOURA, Elizângela. Zona Azul deve começar a funcionar em setembro, no centro de Mossoró. Correio Mossoró. Rio Grande do Norte, 28 jul. 2011. Disponível em: <http://www.correiodatarde.com.br/editorias/correio_mossoro-64228>. Acesso em: 28 jul. 2011.

[3] AZEVEDO, Gabriel. Falta de estacionamento vira problema na praia. Gazeta do povo. Paraná, 20/01/2011. Disponível em: http://www.gazetadopovo.com.br/verao/conteudo.phtml?id=1088408. Acesso em: 28 jul. 2011.

[4] MADER Helena; TEMÓTEO Antonio. Vagas subterrânea devem ser solução na área central de Brasília. Correio Braziliense. Brasília 15/07/2011. Disponível em: http://www.correiobraziliense.com.br/app/noticia/cidades/2011/07/15/interna_cidadesdf,261323/vagas-subterraneas-devem-ser-solucao-na-area-central-de-brasilia.shtml. Acessado em: 28 jul. 2011.

[5] JACOBE, Lupa, Zona Azul. Diário de Ciclista. Pará 14/05/2010. Disponível em: http://diariodeciclista.blogspot.com/2010/05/zona-azul.html . Acessado em: 25 de jul. 2011

[6] NORVING, Stuart Russell Peter, Inteligência Artificial. 2 Edição. Editora Campus: Rio de Janeiro, 2004.

[7] BRAGA, Antônio de Pádua; LUDERMIR, Teresa Bernarda; CARVALHO, André Ponce de Leon F. de Carvalho. Redes Neurais Artificiais: teoria e aplicações. Editora LTC: Rio de Janeiro, 2000.

[8] MARQUES FILHO, Ogê; VIEIRA NETO, Hugo. Processamento Digital de Imagens, Rio de Janeiro: Brasport, 1999. ISBN 8574520098.

[9] MARIA, L. G. F. Processamento Digital de Imagens. INPE, Junho de 2000.

[10] SILVA, Renato Ramos da. Reconhecimento de imagens digitais utilizando redes neurais artificiais. 2005. 51 f. Monografia (Graduação em Ciência da Computação). Universidade Federal de Lavras, Lavras – MG.

[11] GONZALEZ, R. C. WOODS, R. E. Processamento de Imagens Digitais. EDITORA EDGARD BLÜCHER LTDA: São Paulo, 2000.

[12] BRAGA, Antônio de Pádua; LUDERMIR, Teresa Bernarda; CARVALHO, André C. P. de Leon Ferreira. Redes Neurais Artificiais: teoria e aplicações. 2 edição. Editora LTC: Rio de Janeiro, 2007.

[13] HAYKIN, S. Redes Neurais: Princípios e prática. McMaster University, Hamilton, Ontário, Canadá, 2001.

[14] BITTENCOURT, J. R. Sistemas Inteligentes baseados em Redes Neurais Artificiais aplicados ao Processamento de Imagens, In: I Workshop de inteligência artificial, 2000, apostila-seminário, Santa Cruz do Sul, UNISC – Universidade de Santa Cruz do Sul - Departamento de Informática, 2000.

[15] CARLOS, J. F. P. Aplicação de Redes Neuronais no Processamento Digital de Imagens. Universidade Federal de Minas Gerais, 1994.

[16] MEDEIROS, L. F. de. Redes neurais em delphi. Florianópolis: Visual Books Editora, 2003, 115p.