Building Recognition Using Gabor Filter for Extraction of Texture, Wavelet Transformed in the Reduction of Dimensionality and Multilayer Perceptron Network for the Classification

Leandro Aureliano Da Silva[1], Gilberto Arantes Carrijo[2], Rodrigo Pinto Lemos[3], Cleiton Silvano Goulart[4], Antonio Carlos Lemos Júnior[5]

[1, 4, 5]*Department of Electrical Engineering, Faculty of Talentos Humanos, Uberaba, MG, Brazil*

[2]*Department of Electrical Engineering, Federal University of Uberlândia, Uberlândia, MG, Brazil*

[3]*Department of Electrical Engineering, Federal University of Goiás, Goiânia, GO, Brazil*

*Corresponding Author: *Leandro Aureliano Da Silva, Department of Electrical Engineering, Faculty of Talentos Humanos, Uberaba, MG, Brazil*

Abstract: *Building recognition has gained more attention in recent years due to the increasing popularity of georeferencing tools. Many works have accomplished this task using high complexity techniques and several image features, complicating their application on smartphones. Complexity has been reduced using only orientation, color and intensity features. Although texture comprises variations both in color and intensity, no work used texture for building recognition. So, we propose a system using only the texture feature to perform robust recognition of buildings under different points of view, lighting conditions or partial obstruction. We used Gabor filtering for feature extraction in multiple orientations and Daubechies wavelet transforms to reduce the dimensionality. Recognition was accomplished using a multilayer perceptron neural network. To assess the behavior of the network, the amount of orientations of the Gabor filter was varied during feature extraction. The proposed system has shown to perform quite satisfactorily when compared to other building recognition methods using the building database of the University of Sheffield.*

Keywords: *Building recognition, Gabor Filter, Feature extraction, Multilayer Perceptron network.*

1. INTRODUCTION

The popularization of georeferencing tools explains the growing interest on large scale building recognition, which became one of the big challenges in Pattern Recognition. However, building recognition can be also used for several other applications like automatic target detection and tracking, surveillance, architectural design, navigation for mobile devices and robot localization (Li and Allinson, 2013) (Li and Allisnon, 2009).

The biggest challenge of building recognition comes from the huge diversity of image acquisition conditions of the building to be recognized, like images acquired from different angles of view, under diverse lighting conditions, or even partially obstructed by trees, moving vehicles and other buildings (Li and Allinson, 2013). How to deal with this challenge is a recent issue that has been addressed by only a few building recognition systems in the last years, (Li and Allinson, 2013), (Hutchings and Mayol-Cuevas, 2005), (Li and Allinson, 2009), (Li and Shapiro, 2002), (Zhang and Košecká, 2006), (Duan and Allinson, 2010), (Chung, Han and He, 2009), (Groeneweg, et al., 2006), (Trinh, Kim and Jo, 2008).

Research in building recognition have been conducted under two main approaches: (i) research on effectiveness that use representation-based algorithms and image correspondence approaches; and (ii) research on efficiency that use dimensionality reduction methods and clustering algorithms (Li and Allinson, 2013).

Among the works on effectiveness, Hutchings and Mayol (2005) extracted the local features of images using the Harris corner detector the Scale Invariant Feature Transform (SIFT) descriptor. They re-

duced the computational effort by limiting the search space according to the geodesic coordinates of the query image. However this system failed in case of major changes in relation to various points of view.

Considering research on efficiency, Zhang and Košecká (2006) introduced Hierarchical Building Recognition (HBR) that is based on vanishing point detection using Expectation-Maximization (EM) and localized color histograms. Recognition was performed on ZuBud database (Shao, Svoboda and Van Gool, 2003) also using SIFT. Due to indexing using localized color histograms, this system showed to be relatively effective.

On the other hand, according to Li et al. (2014), the system of Zhang and Košecká is unsuitable for real time navigation because it requires the existence of a single building over a simple background on each image to be recognized and spends a long runtime extracting many characteristics of color images.

In order to reduce runtime in mobile phones, Groeneweg et al. (2006) deployed the Fast Offline Building Recognition method that uses local intensity detection and principal component analysis (PCA) to make clusters. Then the building images are recognized using Weighted Majority Algorithm (WMA).

Li and Shapiro (2002) used color, orientation an spatial information to represent each line segment of an image. Those segments were integrated and grouped on a kind of middle level feature to yield consistent line clusters. So, intracluster and intercluster relations were used to recognize different buildings.

Other interesting approach used the detection of both vanishing points and façade line segments to characterize and recognize the buildings, like Trinh et al. (2008), that also employed a SIFT descriptor for each building. By their turn, Duan and Allinson (2010) developed a method for precise automatic detection of vanishing points and grouping of lines associated to façades reconstructed in 3D. Tests on the ZuBud image bank (Shao, Svoboda and Van Gool, 2003) showed that the method was efficient for many kinds of building structures and conditions in which images are filtered using the vanishing points.

Similarly, Chung, Han and He (2009) used the sketch of the main structural components and the Maximally stable extremal regions (MSER) detector to find the windows and doors of a building. Then they normalized the output of MSER detector through oriented gradient histogram and used k-means for clustering the several structural and spectral components.

Following a different path, Li and Allinson (2009) proposed a building recognition system that sets up a saliency-based model by subdividing a multiple-feature map for each image. In order to reduce dimensionality, they compared algorithms for Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Semi-supervised Discriminant Analysis (SDA) and Locality Preserving Projections (LPP). Finally, they used "k-nearest neighbors" for classification.

Li and Allinson (2013) recently used oriented filters to represent the features of a building and developed a recognition model that can be divides in four parts: (i) feature representation; (ii) feature selection; (iii) dimensionality reduction; and (iv) classification. Pooling was employed for selection and Linear Discriminant Analysis (LDA) for dimensionality reduction. At last, they used support vector machines (SVM) for recognition.

All those systems required extracting and processing a large amount of features, like shape, texture, intensity and movement. The more features a system takes into account, the more accurate the classification but also the larger the required computational effort. In order to reduce it, Li and Allinson (2009) considered only three visual characteristics: orientation, intensity and color. Additionally, they combined color and intensity on a single feature for classification, given their joint importance for building recognition.

Although most of the works on building recognition neglected texture, the analysis of this feature is admittedly important for both classification and segmentation of images in general, since it comprises local spatial variation in both intensity and color (Ojala and Pietikäinen, 2015) (Zhou, 1999).

Then, this work proposes a building recognition system that uses only texture with the aim to reduce the computational effort and favor the use of building recognition applications in smartphones. However, using only texture requires integrating multiple techniques to complementarily exploit this single feature in order to achieve a recognition rate compatible to those found in the literature.

This system uses Gabor filtering for extracting texture under different angles and frequencies; wavelet transforms and multiresolution analysis for dimensionality reduction; and a Multilayer Perceptron (MLP) neural network for classification.

Gabor filtering must allow us to circumvent problems related to geometrical and photometrical transformations of images.

Wavelet transforms and multiresolution analysis are used for decomposing the Gabor filter response vector into approximation and detail coefficients through several scales. Approximation coefficients resulting from images of the same building allow us to not only reduce dimensionality but also to preserve common features of those images for classification.

Since Multilayer perceptron (MLP) neural networks show good generalization ability and high accuracy in pattern recognition (da Silva, et al. 2006), we used them to perform the classification task. To the best of our knowledge, this is the first time an MLP is employed in building recognition.

The outcomes of this work were particularly compared to those obtained by Li and Allinson (2013) (2009), since they used the same image database adopted here.

However, the way in which the features were extracted and the integration of the techniques mentioned above, when associated with the MLP, form a denser structure of learning, close to that which characterizes deep learning. (Guo, Yanming, et al., 2016), (Nousi, Pa-raskevi, and Anastasios Tefas, 2017), (Siniscalchi, Sabato Marco, et al. 2013).

This article is structured as follows: Section II describes feature extraction using Gabor filter. Wavelet transform and Multilayer Perceptron network are discussed in sections III and IV. Section V introduces the proposed method. The results are shown in section VI and section VII brings the conclusions.

2. FEATURE EXTRACTION USING GABOR FILTERING

The Gabor filter for two dimensions is given by (Piparsaniyan, Sharma and Mahapatra, 2014):

$$\left.\begin{array}{l} \psi_\Pi\left(f,\theta,\gamma,\eta\right)=\dfrac{f^2}{\pi\gamma\eta}e^{-\left(\frac{f^2}{\gamma^2}x_t^{\,2}+\frac{f^2}{\eta^2}y_t^{\,2}\right)}e^{j2\pi f x_t} \\[2mm] x_t=x\cdot\cos(\theta)+y\cdot\sin(\theta) \\[1mm] y_t=-x\cdot\sin(\theta)+y\cdot\cos(\theta) \end{array}\right\} \tag{1}$$

where $j=\sqrt{-1}$, f is the sinusoidal frequency, θ is the wavelet orientation, γ is the sinusoidal wavelength, η is the wave-length in wavelet space (orthogonal to the sine wave), and (x, y) are the pixel coordinates.

The frequency (f) and orientation (θ) can be determined through (Piparsaniyan, Sharma and Mahapatra, 2014):

$$\left.\begin{array}{l} f=\dfrac{f_{\max}}{\left(\sqrt{2}\right)^g} \\[3mm] \theta=\dfrac{h}{8}\pi \\[4mm] \psi_{g,h}(x,y)=\psi_\Pi\left(f,\theta,\gamma,\eta\right) \end{array}\right\} \tag{2}$$

We find the filter bank of Gabor wavelet representation by setting $\gamma = \eta = \sqrt{2}$, $f_{max} = 0,25$ and varying g from 0 to 4 and h from 0 to 7.

The image filtered with the representation in two dimensions of the Gabor wavelet is found by convolving the original image with the filter bank. This operation is described by (Piparsaniyan, Sharma, & Mahapatra, 2014):

$$O_{g,h}(x, y) = I(x, y) * \psi_{g,h}(x, y) \tag{3}$$

The convolution (3) originates sub-images with 5 frequency scales and eight orientations (Piparsaniyan, Sharma and Mahapatra, 2014).

The feature vector (F) is found by sub-sampling the rows and columns of the filtered image module by a factor of 10 (Piparsaniyan, Sharma and Mahapatra, 2014):

$$F = \left(O^T_{0,0} \; O^T_{1,1} \cdots O^T_{4,7} \right) \tag{4}$$

Then, the vector of features is normalized to zero mean and variance one. For an image $m \times n$, F has dimensions given by (Piparsaniyan, Sharma and Mahapatra, 2014):

$$d = \frac{m \cdot n \cdot g \cdot h}{10 \times 10} \tag{5}$$

where m and n are respectively the number of rows and columns of the original image; g is the variation of frequency scale, and h is the variation of orientations.

The functions used to represent Gabor wavelet are two-dimensional complex sinusoids that can be modeled by a two-dimensional Gaussian function as shown in (1). The purpose of these functions is to extract textures found in the image and described by the frequency (f) and the orientation (θ).

3. WAVELET TRANSFORM

The application of feature extraction to the image yields a feature vector of large dimensions. Thus, in order to provide dimensionality reduction, we used the wavelet transform (WT) to reduce the size of this vector.

Using the WT decomposes the feature vector into a tree of approximation and detail coefficients. Approximation coefficients (A_m) represent low frequency information, while detail coefficients (D_m) capture the high frequency behavior. Fig. 1 illustrates the decomposition tree used to reduce dimensionality by sub-sampling (Misiti, et al. 1996).
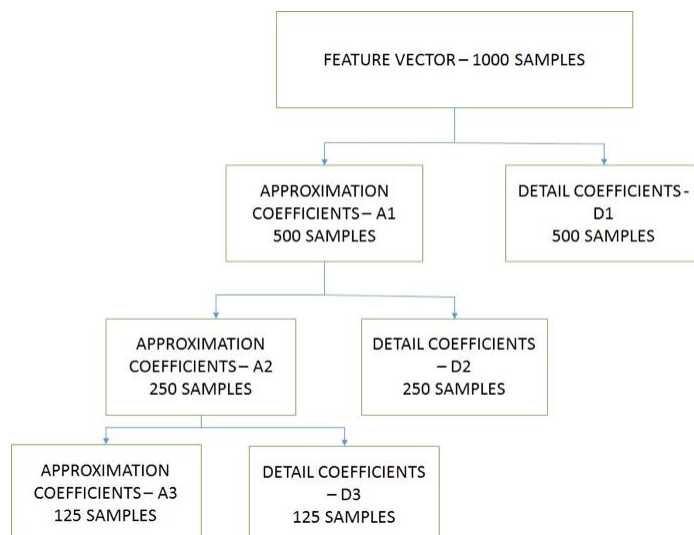
Figure1. *Example of decomposition tree used in the dimensionality reduction*

4. MULTILAYER PERCEPTRON NEURAL NETWORK

Multilayer Perceptron networks consist of a set of perceptrons disposed in an input layer, one or more hidden layers and an output layer. Neurons belonging to the input layer are called input units and simply propagate the the entries to subsequent layers. Intermediate or hidden layers process the in-

formation through the connections (synapsis) between input and output layers. At each synapsis the information is modified by synaptical weights. Finally, the output layer is responsible for providing the response of the network (Haykin, 2001).

The output of each neuron passes through an activation function that can be a linear, logistic or hyperbolic tangent function. This study used the hyperbolic tangent.

MLP uses supervised training that indicates the desired response to an input pattern such that the synaptical weights and neuron thresholds are ajusted to minimize the error between those responses during the learning phase. The most used training algorithm is the *backpropagation* with *forward* e *backward* phases. In the *forward* phase, entries are propagated throughout the network layer-by-layer. During this phase, the weights are fixed and the output of the network is compared to the desired response. In the *backward* phase, the output error is backpropagated from the output to the input layer and the synaptical weights are updated according to an error correction rule.

Given that the training of the MLP is supervised, it is necessary that the output layer is composed of binary or bipolar vectors. These vectors form a matrix, whose each column corresponds to a particular input. The binary and bipolar vectors are such that:

$$\vec{V}_{ij} = \begin{cases} 1\ para\ i = j \\ 0\ para\ i \neq j \end{cases} \tag{6}$$

$$\vec{V}_{ij} = \begin{cases} 1\ para\ i = j \\ -1\ para\ i \neq j \end{cases} \tag{7}$$

Recent studies (Manzan, et al. 2012) have shown that using orthogonal bipolar vectors (OBV) increases the Euclidean distance between the network output points. This increase benefits the mapping of the MLP network, resulting in higher success rates and faster training.

Fausset (1994) proposed an algorithm for generating the OBV. This algorithm uses a seed vector that generates a sequence of vectors until the final determination of OBV. The number of components of each OBV depends on the number of desired vectors and the number of components chosen for the seed vector. The number of components of each vector is obtained by:

$$n = 2^k m \tag{8}$$

where 2^k is the number of orthogonal vectors for $k > 0$, and m is the number of components in the seed vector. It is worth noting that the number of vectors is always a power of 2.

According to Fausset, a set OBV is constructed by making:

Step 1 - Depending on the application, the values of m and k should be determined by the user. The value of m starts at 1 and as it increases, the amount of OBV grows.

Step 2 - The seed vector is obtained by (9). It is worth emphasizing that this vector can have a single component.

$$V_m^0 = \overbrace{(1,1,...,1)}^{m}{}^T \tag{9}$$

Step 3 - The number of OBV components is obtained by (8).

Step 4 – From the seeds vector found in step 3, the vectors are built as $V_{2m}^1 = fcc\left(V_m^0, V_m^0\right)$ and $V_{2m}^2 = fcc\left(V_m^0, -V_m^0\right)$.

where $fcc\left(V_1, V_2\right)$ is the concatenation function of vectors V_1 and V_2. Note that the seed vector is used twice. In the first time V1 is concatenated with V2. In the second time, the signal of V2 changes.

Step 5 – From vectors V_1 and V_2 in step 4 an using the concatenation function, four new vectors are determined:

$$V_{4m}^1 = fcc\left(V_{2m}^1, V_{2m}^1\right), \; V_{4m}^2 = fcc\left(V_{2m}^1, -V_{2m}^1\right), \; V_{4m}^3 = fcc\left(V_{2m}^2, V_{2m}^2\right) \; e \; V_{4m}^4 = fcc\left(V_{2m}^2, -V_{2m}^2\right).$$

The vectors determined at step 4 are used twice in the concatenation function. In the first, the sign of the second argument is kept and in the second time it is changed.

Step 6 – Concatenations are made until we have 2^k orthogonal vectors with n components $(V_n^1, ..., V_n^{2^k})$.

5. PROPOSED METHOD

The proposed method consists of four steps. The first one preprocess the color image. In the second step, Gabor wavelet representation is applied in order to extract the preprocessed image features. The result of the filtered image using this representation is transformed into a feature vector. In the third step wavelet transform is used to reduce the dimensionality of this vector. Finally, in the fourth step training and testing of the MLP is carried out. Fig. 2 shows the block diagram of this method. In the following each of these steps is described.
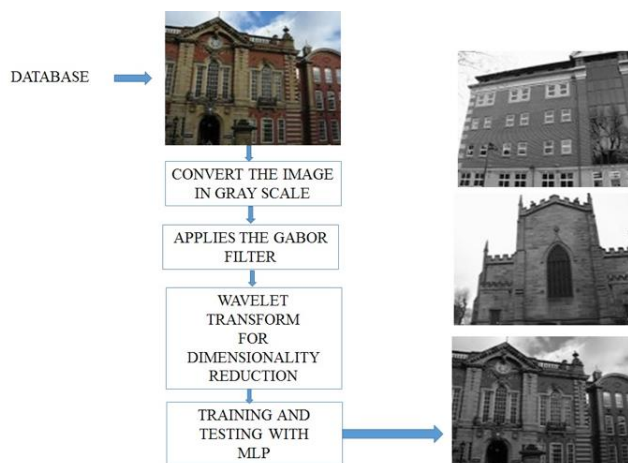
Figure2. *Block diagram of the proposed method*

5.1. Preprocessing

In this step, the image is read from the building database. Then, the image is converted to 256-level grayscale. Fig. 3 shows the original image and the result of this step.

The convertion of the color image to grayscale aims reducing the runtime of feature extraction.

5.2. Feature Extraction

In the second step, feature extraction is performed through the convolution between the grayscale image and the filter bank originated from the Gabor wavelet representation. In this case, the bank has five different scales built up to eight orientations. According to (1), the pixel coordinate (x, y) is determined (39, 39) (Haghighat, Zonouz, & Abdel-Mottaleb, 2013). Fig. 4 illustrates the result of this convolution module. In this figure is shown an example of 40 sub-images obtained using these scales and 8 orientations.

The first row in Fig. 4 show 8 sub-images obtained using the frequency 0.25 and eight directions (0°, 22.5°, 45°, 67.5°, 90°, 112.5°, 135 and 157.5°). Each one of the other rows is obtained using a frequency determined according to (2) and the set of orientations used in the first row.

The feature vector is obtained by subsampling by a factor of 4 the rows and columns of of each picture in Fig. 4, and then rearranging them into a column vector. In order to be processed by the MLP, this feature vector is normalized to zero mean and unit variance.

The dimension of the feature vector is found using (5). For images with 160 x 120 pixels, 5 frequency scales and 8 orientations, the size of the feature vector is 7680 samples. Then, we must reduce the dimensionality of this vector for faster network convergence

5.3. Dimensionality Reduction

In order to reduce the dimensionality of the feature vector, we decomposed it with Daubechies wavelet transform of order 5 until the level 3. As a result of this operation we built a new feature vector using the level 3 approximation coefficients. This was the vector used for training and testing the MLP.

(a)   (b)

Figure3. Grayscale conversion: (a) original image; (b) grayscale image.
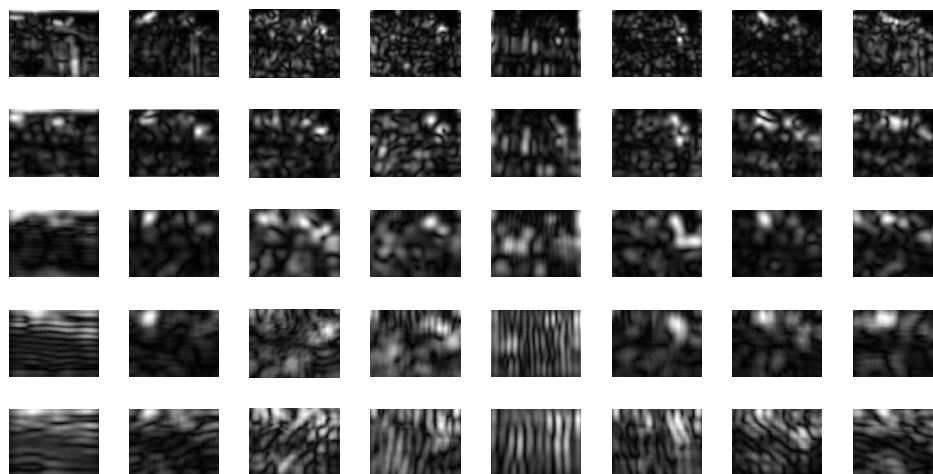
Figure4. *Modulus of the result of convolution between the grayscale image and the Gabor filters*

5.4. Training and Testing the MLP

The building recognition task was performed through an MLP whose input layer had the same dimension of the feature vector, the single hidden layer had 200 neurons and the output layer has 64 neurons to feed the OBV target vectors of length 64.

In order to choose 200 neurons in the single intermediate layer, it started from the minimum limit set by Fletcher-Gloss (da Silva, et al. 2006). It is noteworthy that the 200 neurons in the middle layer was the best result in the tests.

For training and testing the MLP we used the first 80 images of 21 categories of the building image bank, totaling 1680 samples such that 1344 among them (64 images in each category) were used for training and other 336 (16 images in each category) for testing.

As stopping criterion, we considered the mean square error. After some tests, we found a tolerable error rate of 0.001 and a learning rate of 0.01. To speed up the convergence of the network, we used the momentum term and an adaptive learning rate.

6. RESULTS

In order to validate the results of building image classification based only in texture, we considered the University of Sheffield SBID ([online], 2014) image database, the same used by Li and Allinson (2013) (2009). This image database contains 40 categories of buildings, totaling 3192 images in JPEG format with dimensions 160 x 120 pixels. SBID combines different variations of each building image concerning rotation, scaling, lighting conditions, occlusions and vibration.

All algorithms were implemented in MATLAB® R2013b and, including the artificial neural networks toolbox for the MLP. Execution was carried out on a Core I5 machine with 4 GByte RAM.

In order to reduce runtime, we used only 21 of the 40 categories of SBID. As said before, we considered only the first 80 images of the 21 categories, such that 64 (80%) were used for the training and 16 (20%) for testing the MLP. The network used steepest descent with momentum term equal to 0.8 and adaptive learning rate, 200 neurons in the hidden layer and 64 in the output layer, and tolerable error equal to 0.001.

We verified the behavior of the MLP by varying the number of orientations of the Gabor filter while keeping the 5 frequency scales. In order to reduce dimensionality, we adopted order 5 Daubechies wavelet transform.

Table I shows the recognition rates of the *MLP*. The largest recognition rate achieved was 96% using 4 orientations in feature extraction. For other orientations those rates varied from 91 to 95%. Table II compares our results to those found in literature concerning the recognition rate (Li, Huang, Shao, & Allinson, 2014).

Table1. *Recognition Rates Of Mlp Interms Of Orientation Ө Of Gabor Filter Using Order 5 Daubechies Wavelet Transform To Reduce Dimensionality*

N° of orientations θ	Recognition rate %	N° of epochs
1	93%	802
2	93%	1043
3	91%	207
4	**96%**	**379**
5	93%	160
6	94%	230
7	94%	161
8	95%	227

From Table II, the proposed method achieved the highest recognition rate. Considering only the work of Li and Allinson (2013) (2009) that used the same database, the proposed method was far more robust because it reached a higher regonition rate despite using a single feature (texture).

Table2. *Comparision of the Recognition Rates of the Proposed Method to Those Found In Literature*

Correlated works	Database	Recognition rate %
Proposed method	**SBID**	**96%**
Li and Allinson (2013)	SBID	94,6%
Li and Allinson (2009)	SBID	85,25%
Groeneweg. et al (2006)	ZuBud	91%
Li and Shapiro (2002)	977 Color images	94,2%
Zhang and Kosecká (2006)	ZuBud	95%
Chung, Han and He (2009)	ZuBud	81%

7. CONCLUSIONS

This paper presented a new building recognition system based only in texture feature. The system used Gabor filtering for extracting texture, Daubechies wavelet multiresolution analysis for dimensionality reduction and a *Multilayer Perceptron* (MLP) neural network for classification.

By varying the number of orientations of the Gabor filter from 0 to 7, the MLP achieved 96% of recognition rate using 4 orientations. Despite it used a single feature, it was able to overcome the re-

sults of the works of Li and Allinson (2013) (2009) and showed the highest recognition rate when compared to previous studies in the literature.

REFERENCES

Chung, Y. C., T. X. Han, and Z. He. "Building recognition using sketch-based representations and spectral graph matching." *IEEE 12th International Conference onComputer Vision, 2009.* IEEE, 2009. 2014-2020.

da Silva, Ivan Nunes, Danilo Hernane Spatti, Rogerio Andrade Flauzino, Luisa Helena Bartocci Liboni, and Silas Franco dos Reis Alves. *Artificial Neural Networks: A Practical Course.* Springer, 2006.

Duan, W., N. M. Allinson. "Vanishing points detection and line grouping for complex building facade identification." *WSCG*, 2010: 213-220.

Fausett, L. V. *Fundamentals of neural networks.* Prentice-Hall, 1994.

Groeneweg, Nikolaj JC, Bastiaan de Groot, Arvid HR Halma, Bernardo R Quiroga, Maarten Tromp, e Frans CA Groen. "A fast offline building recognition application on a mobile telephone." *International Conference on Advanced Concepts for Intelligent Vision Systems.* Springer, 2006. 1122-1132.

Guo, Yanming, et al. "Deep learning for visual understanding: A review." *Neurocomputing* 187 (2016): 27-48.

Haghighat, Mohammad, Saman Zonouz, and Mohamed Abdel-Mottaleb. "Identification using encrypted biometrics." *International Conference on Computer Analysis of Images and Patterns.* Berlin: Springer, 2013. 440-448.

Haykin, S. *Redes neurais.* Bookman, 2001.

Hutchings, R., W. Mayol-Cuevas. "Building recognition for mobile devices: incorporating positional information with visual features." *CSTR-06-017, Computer Science*, 2005.

Li, J., N. M. Allinson. "Building recognition using local oriented features." *IEEE Transactions on Industrial Informatics*, Aug. de 2013: 1697-1704.

—. "Subspace learning-based dimensionality reduction in building recognition." *Neurocomputing*, 2009: 324-330.

Li, J., W. Huang, L. Shao, N. M. Allinson. "Building recognition in urban environments: A survey of state-of-the-art and future challenges." *Information Sciences* (Elsevier) 277 (2014): 406-420.

Li, Y., L. G. Shapiro. "Consistent line clusters for building recognition in CBIR." *Proceedings. 16th International Conference on Pattern Recognition, 2002.* . IEEE, 2002. 952-956.

Manzan, José Ricardo Gonçalves, Shigueo Nomura, Keiji Yamanaka, Milena Bueno Pereira Carneiro, and Antônio C Paschoarelli Veiga. "Improving iris recognition through new target vectors in MLP artificial neural networks." *IAPR Workshop on Artificial Neural Networks in Pattern Recognition.* Berlin: Springer, 2012. 115-126.

Misiti, M., Y. Misiti, G. Oppenheim, J. M. Poggi. *Wavelet Toolbox User's Guide.* Natick: The Mathworks, 1996.

Nousi, Paraskevi, and Anastasios Tefas. "Deep Learning Algorithms for Discriminant Autoencoding." *Neurocomputing* (2017).

Ojala, T., M. Pietikäinen. *Texture Classification [online].* 2015. http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/OJALA1/texclas.htm.

Piparsaniyan, Y., V. K. Sharma, K. K. Mahapatra. "Robust facial expression recognition using Gabor feature and Bayesian discriminating classifier." *International Conference on Communications and Signal Processing (ICCSP), 2014.* IEEE, 2014. 538-541.

Shao, Hao, Tomáš Svoboda, and Luc Van Gool. "Zubud-zurich buildings database for image based recognition." *Computer Vision Lab* (Swiss Federal Institute of Technology) 260 (2003): 20.

Siniscalchi, Sabato Marco, et al. "Exploiting deep neural networks for detection-based speech recognition." *Neurocomputing* 106 (2013): 148-157.

Trinh, H. H., D. N. Kim, and K. H. Jo. "Facet-based multiple building analysis for robot intelligence." *Applied Mathematics and Computation*, 2008: 537-549.

Zhang, W., J. Košecká. "Localization based on building recognition." *CVPR Workshops. IEEE Computer Society Conference onComputer Vision and Pattern Recognition-Workshops, 2005.* IEEE, 2006. 21-28.

Zhou, D. "Texture analysis and synthesis using a generic Markov-Gibbs image model." *Ph.D. Thesis.* New Zealand: Department of Computer Science, University of Auckland, 1999.

[online]. 2014. http://www.sheffield.ac.uk/eee/research/iel/datasets.

DATA AVAILABILITY

The database for the execution of this work can be found in the electronic address:

https://www.sheffield.ac.uk/eee/research/iel/research