Concrete and Asphalt Runway Detection in High Resolution Images Using LBP Cascade Classifier

Juliano E. C. Cruz
Applied Computing Master Program
Instituto Nacional de Pesquisas Espaciais - INPE
São José dos Campos, Brazil
juliano.cruz@lac.inpe.br

Elcio H. Shiguemori
and Lamartine N. F. Guimarães
Instituto de Estudos Avançados - IEAv
São José dos Campos, Brazil
{elcio,guimarae}@ieav.cta.br

***Abstract*—Automatic object recognition in digital satellite images is not a simple task due to several variations present in the capture process and object appearance and pose, consequently, different general purpose techniques have been proposed. In this paper, an approach with LBP boosted cascade classifier for automatic runway detection in high resolution satellite imagery is analyzed. Promising results are obtained with the methodology presented in this work, considering objects with variations of scale, rotation and images obtained by different sensors.**

***Keywords*—runway detection, satellite imagery, boosting, LBP, boosted cascade**

I. INTRODUCTION

In satellite imagery with high spatial resolution, most objects can be recognized by experts. Some objects are easy to be visually recognized, others only with a high level of expertise. However, today, several applications require automatic detection or identification of objects present in those images.

One example of that system use is in UAV (Unmanned Aerial Vehicle) platforms that are currently having a boom, principally due to the growing use by military, police and civilian applications [1]. Runway recognition is an important task particularly for UAVs that can use it for landing, combat or even in self-localization procedures [2]. Generally, big and medium cities and air force bases have its runway made of concrete or asphalt. This approach can be useful to both autonomous and hybrid systems. The use of hybrid systems is also relevant because human-based recognition system is highly susceptible to errors due to the fact that recognition processes generally require a huge amount of data processing and it is a tedious task. Furthermore, in some cases the operator must be previously trained.

This paper analyzes the applicability of LBP (Linear Binary Patterns) cascade classifier [3] to concrete and asphalt runway detection. Despite this method has been used more frequently in other application areas, as seen in Section II, the choice of using LBP cascade classifier was made because its high "benefit-cost" ratio, that is, good accuracy and low computational cost. These characteristics are desirable particularly by autonomous systems that need to take real-time decisions [4][5].

II. RELATED WORKS

Since its creation, the LBP cascade classifier, has shown to be a robust classifier. It is considered to be a general purpose although most uses are for human feature recognition. One of the first works to employ the LBP with the cascade classifier concept for object recognition was in [3], where it proposed a face recognition approach.

As presented before, most applications of LBP have been for human feature recognition. Some works perform face recognition, as [6], [7], [8], [9] that also implements gender recognition and [10] that also proposes a LBP multi-block approach. Some other works perform pedestrian detection, as [11] that uses optical and thermal imagery, [12] that also performs tracking, and [13] that also performs face and head detection. As a biometric identification solution, [14] proposes a palmprint identification. There are also works that seek general object recognition, as [15].

There are papers that propose runway recognition, but none of them employs the use of LBP cascade classifier. Due to the fact that a runway is an exact straight line, some works employ Canny operator with Hough transform for such task [16][5][17].

III. LBP CASCADE CLASSIFIER

Texture is defined as a function of spatial variations in the pixel intensity of an image, and it has being used in a wide variety of applications [18].

The Local Binary Pattern operator, also known as LBP, was first introduced in [19] through the adaptation of the work [20] and has shown to be a powerful texture descriptor. The idea behind this operator is that common features such as edges, lines, point, among others, can be represented by a value in a particular numerical scale. Therefore, using a set of extracted values *a priori* it is possible to recognize objects in an image.

The original LBP labels the pixels by thresholding the 3x3 neighborhood in relation to the central pixel value, as seen in Figure 1, but there are recent studies that extend this version [21][15][10]. The "bit" obtained for each neighboring pixel is used through a pre-defined order to form a final value, using 8 neighboring pixels that have value between 0 and 255, as can be seen in Equation 1.

$$LBP(x_p, y_p) = \sum_{n=0}^{7} s(i_n - i_p)2^n \qquad (1)$$

where (x_p, y_p) is the pixel of an image, n represents the neighboring pixel, i_n and i_p the respective gray level of

978-1-4799-3194-1/13 $31.00 © 2013 IEEE
DOI 10.1109/BRICS-CCI-CBIC.2013.83

neighboring and central pixel, and $s(x)$ can be described by the Equation 2.

$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \tag{2}$$

An extension to LBP was proposed by [10] and is called multi-block LBP, or simply MB-LBP. Instead of using single pixels, MB-LBP applies the LBP operator to pixel blocks. All blocks must have the same size and must respect the 3x3 formation, as seen in Fig. 1. Then, what LBP operator uses for its computation is the pixel block mean value. The advantage of MB-LBP is to have a faster and more precise classifier [10].

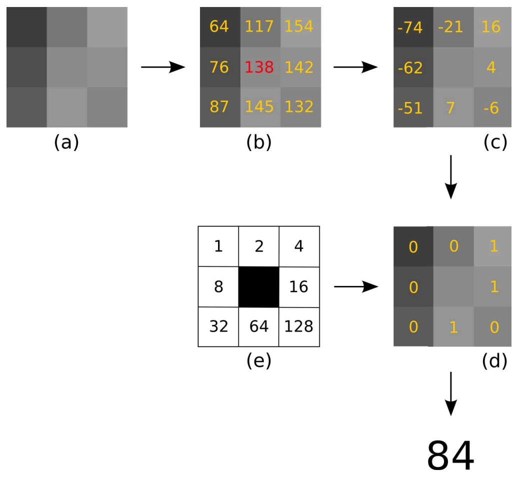

Fig. 1. LBP extraction process, where (a) is the image fragment to be processed, (b) shows gray levels, (c) shows subtraction results among periphery pixels and the central one, in (d) it is assigned 0 to subtraction results less than 0 and 1 to subtraction results greater or equal 0, in (e) are shown the binary matrix values and in the final step, it is summed the correspondent cell values in binary matrix that in (d) are 1.

MB-LBP binary patterns can be used to detect diverse image structures such as edges, lines, spots, flat areas and corners at different scale and location. Since MB-LBP features are non-metric values, therefore it is not possible to use a threshold-based function as weak classifier. Thus, a decision tree is used as weak classifier. These weak classifiers are classifiers that are slightly better than a random guess, but when set in cascade, they become a strong classifier (Fig. 3) with a high discrimination power, capable of detecting structures despite of illumination, color or scale variation [22]. It combines successively weak classifiers, starting with simple ones and ending with the most complex ones. In spite of the fact that it seems to be an exhaustive search, the classifier building characteristic allows an early rejection with a minimum evaluation and consequently, it has a low computational cost. It lies in the fact that the majority of detection windows are negative and there are 'few' windows that go through all stages. Therefore, the computational power will be focused in the windows that have higher probability to be positive, once they already passed through the initial stages [23][4].

In any window inside a image, a huge amount of MB-LBP features can be found. It is necessary then, during the training stage, to focus on a small set of critical features, discarding most of non-critical ones, in order to increase significantly classification speed without affecting accuracy. Boosting [24], an effective learning algorithm and has strong bounds on generalization performance, is then, responsible to solve that problem [4].

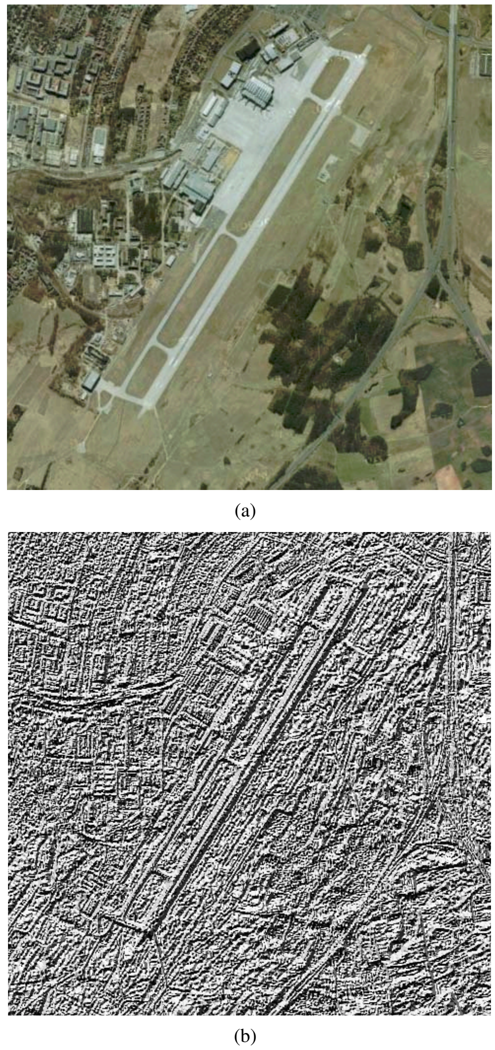

Fig. 2. (a) Original image and (b) resultant image after applying LBP operator.

Objects can appear in different regions of the image and in different scales. In order to solve this problem, the sliding window method is used [25]. It consists of a detection window that slides over an image extracting regions and classifying them. A Gaussian pyramid [26] is also applied to the image during the classification stage in order to perform a scale search. It is important to highlight that before the training stage is performed, the training samples must be resized to the detection window size.

Due to the fact that detection windows overlap, the same object can be count as various instead of just one. Then, it is employed what is called as non-maximum suppression, that is, windows with a local maximum of the classifier response suppress nearby windows [25].

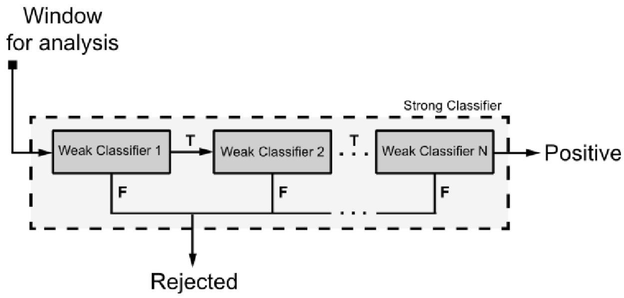

Fig. 3. Cascade classifier concept.

IV. Methodology

Due to the fact that there is not a common use runway dataset available and the previous works [16][5][17] proposing runway detection did not make their own dataset available , it was necessary then, to create our own.

The training dataset utilized is a high spatial resolution image set composed of 2040 positive (Fig. 4) and 1600 negative samples. All images were obtained from Google Maps application [27] where the positive samples are runway images from different parts of the world. It is important to highlight that Google Maps images have watermarks with Google logo and year of capture scattered throughout the image. In order to give certain rotation invariance, the positive set were built by rotating the original images forty times, 9 degrees each time. Due to memory limitations, the detection window size and consequently the size of positive set images during the training procedure should be, at maximum, 50x50 pixels. Two sizes have been used, the maximum one (50x50) and a 32x32 one, but negative samples have not been resized, because during the training stage, fragments with the same size as the positive samples are systematically extracted from the negative training set images.

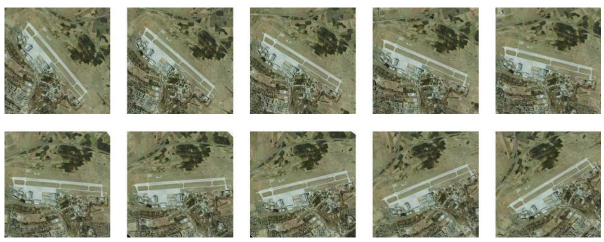

Fig. 4. Positive data set samples from the high resolution image set [27].

The LBP cascade classifier configuration for training was set with 5 different stage sets (20, 25, 30, 35 and 40 stages), 0.999 for minimum hit rate per stage and Gentle Adboost as boost algorithm [28].

V. Results

Objects detected by this approach are possible to be categorized in three classes: True Positives(TP), Fig. 5, that are runways correctly recognized, False Positives(FP), Fig. 6 and Fig. 7, that are regions of the image that were erroneously classified as runway and False Negatives(FN) that are non-recognized runways. In this work, True Negatives, that are non-runway regions correctly classified, were not used due to the fact that they appear in a huge number and do not add any substantial information besides the ones already gotten with TP, FP and FN.

Two metrics [29] were utilized in order to measure classification accuracy performance of an image set obtained from different sensors. The metrics are: True Positive Rate (Eq. 3), also known as Hit Rate and Precision (Eq. 4).

$$\text{TPR} = \frac{\text{TP}}{\text{Total Positives}} \times 100 \tag{3}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \times 100 \tag{4}$$

where Total Positives are the total amount of runways contained in the test dataset, that is, $TP + FN$.

In order to measure processing speed performance there were utilized two metrics: Average Time per Image (ATI), Eq. 5, and Average Time per Pixel (ATpx), Eq. 6.

$$\text{ATI} = \frac{\text{Total Processing Time}}{\text{Total Image Amount}} \tag{5}$$

$$\text{ATpx} = \frac{\text{Total Processing Time}}{\text{Total Pixel Amount}} \tag{6}$$

In the classification test, 50 images, that were not present in the training dataset, were processed. They are not synthetic images, therefore they were processed just the way they appear in Google Maps. Each image contains a runway, where the image size average are 1200x600 pixels and the runway length varies from 200 pixels to 400 pixels. The classification process was not parallelized and it was executed in a 2.40Ghz Intel i7-3630QM processor with 8GB RAM. The classification results, the performance indexes and the average time measures are shown in Table I and the II.

Stages	TP	TPR	FP	FN	Prec.	ATI	ATpx
20	39	78%	11	11	78%	367.66 ms	$5{\cdot}10^{-4}$ ms
25	39	78%	11	11	78%	209.65 ms	$2.85{\cdot}10^{-4}$ ms
30	39	78%	11	11	78%	124.57 ms	$1.69{\cdot}10^{-4}$ ms
35	40	80%	11	10	78.4%	81.75 ms	$1.11{\cdot}10^{-4}$ ms
40	38	76%	24	10	61.3%	57.15 ms	$0.78{\cdot}10^{-4}$ ms

TABLE I. Classification results with a 32px detection window.

Stages	TP	TPR	FP	FN	Prec.	ATI	ATpx
20	39	78%	11	11	78%	3002 ms	$40.8{\cdot}10^{-4}$ ms
25	39	78%	11	11	78%	1945.7 ms	$26.4{\cdot}10^{-4}$ ms
30	40	80%	10	10	80%	1124.4 ms	$15.3{\cdot}10^{-4}$ ms
35	40	80%	10	10	80%	649.24 ms	$8.82{\cdot}10^{-4}$ ms
40	40	80%	10	10	80%	393.55 ms	$5.35{\cdot}10^{-4}$ ms

TABLE II. Classification results with a 50px detection window.

Fig. 5. Cropped region showing a true positive example.

Fig. 6. Cropped region showing a false positive example.

Fig. 7. Cropped region showing a false positive example.

VI. Conclusion

In this paper, it is proposed an approach for concrete and asphalt runway recognition with LBP cascade classifier in high resolution satellite images. It has shown a high performance with object recognition in the previous works and this work confirms its applicability for runway detection in satellite imagery and also, its applicability in real-time decision making systems.

The classifier parameters have been set in order to check the variations in the classification accuracy and speed performance. All of the classifiers had somehow a similar classification accuracy performance, the outlier is the classifier with 40 stages and 32 pixel size (TPR=76% and Precision=61.3%). The possible explanation is classifier overfitting due to the parameters combination. It was not possible to identify whether false positives follow a pattern or not, but what is possible to say is that most of false positives occur in the same region in almost all classifiers and that some false positives are composed by strong lines however they are not necessarily straight.

Regarding the processing speed performance, it can be clearly seen that the more stages a classifier has, more faster it is, and also, the smaller its descriptor is, the faster it gets.

Analyzing only the classification accuracy, TPR and Precision of 80%, it is possible to affirm that the results got a good performance but for some applications they still need improvements. But when one looks the relation between classification accuracy and processing speed, it turns to be a promising result. The best classifier, considering this ratio, is the classifier with 35 stages and 32 pixel descriptor size(TPR=80%, Precision=78.4% and ATI=81.75ms).

Another interesting point to note is that the watermarks present in the high resolution training images had none or little impact on the classifier performance, due mainly to the classifier internal architecture.

The three previous works [16][5][17] about runway detection did not make their own image dataset available, consequently it was not possible to make comparisons about the classification accuracy. Most of them also, did not report the processing speed performance of their approach either. The only one [5] that made it, processed a 256x256 pixel image in 27ms, that is, it has ATpx=$4.12{\cdot}10^{-4}$ ms. The approach used in this work showed a much better performance. Using a 32px detection window and a 25, 30, 35 or 40 stage cascade, it showed to be at least 1.45 time faster and at most 5.28 times faster.

REFERENCES

[1] K. Nonami, F. Kendoul, S. Suzuki, W. Wang, and D. Nakazawa, *Autonomous Flying Robots: Unmanned Aerial Vehicles and Micro Aerial Vehicles*. Springer, 2010.

[2] R. Rodrigues, H. Shiguemori, C. Forster, and S. Pellegrino, "Color and Texture Features for Landmarks Recognition on UAV Navigation," *Anais do XIV Simpsio Brasileiro de Sensoriamento Remoto*, 2009.

[3] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face recognition with local binary patterns," in *Computer Vision - ECCV*, ser. Lecture Notes in Computer Science, T. Pajdla and J. Matas, Eds. Springer Berlin Heidelberg, 2004, vol. 3021, pp. 469–481. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-24670-1_36

[4] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1. Los Alamitos, CA, USA: IEEE Comput. Soc, Apr. 2001, pp. 511–518.

[5] N. Di, M. Zhu, and Y. Wang, "Real time method for airport runway detection in aerial images," *International Conference on Audio, Language and Image Processing, 2008. ICALIP 2008.*, pp. 563 – 567, Jul. 2008.

[6] C. Ma, T. Tan, and Q. Yang, "Cascade boosting lbp feature based classifiers for face recognition," in *Intelligent System and Knowledge Engineering, 2008. ISKE 2008. 3rd International Conference on*, vol. 1, 2008, pp. 1100–1104.

[7] T. Gritti, C. Shan, V. Jeanne, and R. Braspenning, "Local Features based Facial Expression Recognition with Face Registration Errors," *IEEE International Conference on Automatic Face and Gesture Recognition*, 2008.

[8] W. Wang, W. Chen, and D. Xu, "Pyramid-based multi-scale lbp features for face recognition," in *Multimedia and Signal Processing (CMSP), 2011 International Conference on*, vol. 1, 2011, pp. 151–155.

[9] Y. Fang and Z. Wang, "Improving lbp features for gender classification," in *Wavelet Analysis and Pattern Recognition. ICWAPR '08. International Conference on*, vol. 1, 2008, pp. 373–377.

[10] L. Zhang, R. Chu, S. Xiang, S. Liao, and S. Li, "Face detection based on multi-block lbp representation," in *Advances in Biometrics*, ser. Lecture Notes in Computer Science, S.-W. Lee and S. Li, Eds. Springer Berlin Heidelberg, 2007, vol. 4642, pp. 11–18. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-74549-5_2

[11] D. Xia, H. Sun, and Z. Shen, "Real-time infrared pedestrian detection based on multi-block lbp," in *Computer Application and System Modeling (ICCASM), 2010 International Conference on*, vol. 12, 2010, pp. 139 – 142.

[12] Y. Ma, X. Chen, and G. Chen, "Pedestrian detection and tracking using hog and oriented-lbp features," in *Network and Parallel Computing*, ser. Lecture Notes in Computer Science, E. Altman and W. Shi, Eds. Springer Berlin Heidelberg, 2011, vol. 6985, pp. 176–184. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-24403-2_15

[13] E. Corvee and F. Bremond, "Haar like and LBP based features for face, head and people detection in video sequences," in *International Workshop on Behaviour Analysis and Video Understanding (ICVS 2011)*, Sophia Antipolis, France, Sep. 2011, p. 10. [Online]. Available: http://hal.inria.fr/inria-00624360

[14] X. Wang, H. Gong, H. Zhang, B. Li, and Z. Zhuang, "Palmprint identification using boosting local binary pattern," in *Pattern Recognition, ICPR. 18th International Conference on*, vol. 3, 2006, pp. 503–506.

[15] J. Trefnỳ and J. Matas, "Extended set of local binary patterns for rapid object detection," *Computer Vision Winter Workshop*, vol. 10, pp. 37–43, 2010.

[16] X. Wang, B. Li, and Q. Geng, "Runway Detection and Tracking for Unmanned Aerial Vehicle Based on an Improved Canny Edge Detection Algorithm," *4th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2012*, pp. 149 – 152, Aug. 2012.

[17] D. Liu, L. He, and L. Carin, "Airport detection in large aerial optical imagery ," *IEEE International Conference onAcoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP '04).*, pp. 761 – 764, Jul. 2004.

[18] C. Chen, *Handbook of Pattern Recognition and Computer Vision*. World Scientific Publishing Company, Incorporated, 2009. [Online]. Available: http://books.google.com.br/books?id=TaevS0ibLqwC

[19] T. Ojala, M. Pietikainen, and D. Harwood, "Performance evaluation of texture measures with classification based on kullback discrimination of distributions," in *Pattern Recognition, 1994. Vol. 1 - Conference A: Computer Vision amp; Image Processing., Proceedings of the 12th IAPR International Conference on*, vol. 1, oct 1994, pp. 582 –585 vol.1.

[20] L. Wang and D. He, "Texture classification using texture spectrum," *Pattern Recognition*, vol. 23, no. 8, pp. 905–910, 1990.

[21] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 7, pp. 971 – 987, jul 2002.

[22] P. Viola and M. Jones, "Robust Real-Time Face Detection," *International Journal of Computer Vision*, 2004.

[23] Q. Chen, N. D. Georganas, and E. M. Petriu, "Real-time Vision-based Hand Gesture Recognition Using Haar-like Features," in *Instrumentation and Measurement Technology Conference Proceedings, 2007. IMTC 2007. IEEE*, May 2007, pp. 1–6.

[24] Y. Freund and R. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," in *Computational Learning Theory*, ser. Lecture Notes in Computer Science, P. Vitányi, Ed. Springer Berlin / Heidelberg, 1995, vol. 904, pp. 23–37.

[25] D. Forsyth and J. Ponce, *Computer Vision: A Modern Approach*. Prentice Hall, 2011. [Online]. Available: http://books.google.com.br/books?id=gM63QQAACAAJ

[26] A. Witkin, "Scale-space filtering: A new approach to multi-scale description," *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '84.*, vol. 9, pp. 150 – 153, mar 1984.

[27] Google Maps, "Google," http://maps.google.com, 2013.

[28] R. Lienhart, A. Kuranov, and V. Pisarevsky, "Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection," *Pattern Recognition*, pp. 297–304, 2003.

[29] T. Fawcett, "An introduction to ROC analysis," in *ROC Analysis in Pattern Recognition*. New York, NY, USA: Elsevier Science Inc., Jun. 2006, vol. 27, no. 8, pp. 861–874.