COMPARISON OF CASCADE CLASSIFIERS FOR AUTOMATIC LANDING PAD DETECTION IN DIGITAL IMAGES

C.S. de Oliveira*, A.P. Anvar†, A. Anvar†, M.C. Silva Jr.*, A. Alves Neto*, L.A. Mozelli ‡*

*UFSJ - Universidade Federal de São João Del-Rei
CELTA - Center for Studies in Electronics Engineering and Automation
Rod. MG 443 km 7 – 36420-000 – Ouro Branco, MG, Brazil

†The University of Adelaide – School of Mechanical Engineering
Adelaide, SA 5005, Australia

Emails: caterine@live.de, {amir.p.anvar, amir.anvar}@adelaide.edu.au,
{mariocupertino, aaneto, mozelli}@ufsj.edu.br

Abstract— In this paper, two methods for local feature detection in digital images are compared for the task of automatic detection of landing pads. Most landing pads are marked by traditional characters "H" and "X" inside of circles. In this case, by using a data set of positive and negative images, a cascade classifier is trained to achieve high accuracy for different lighting and background conditions, as well as distinct attitude and altitude. The strong classifier is obtained by advancing weak classifiers based either on Haar-like features or Local Binary Patterns.

Keywords— Aerial Robotics, Quadrotors, Classifiers Performance, Machine Vision

Resumo— Neste artigo, dois métodos para detecção de padrões em imagens são comparados para o problema de detecção automática de locais de pouso. Grande parte das pistas de pouso são demarcados pelas tradicionais letras "H" e "X" dentro de círculos. Neste caso, usando uma base de imagens com e sem marcadores, um classificador em cascata é treinado visando alta precisão para diferentes condições de iluminação, fundo, orientação e altura. O classificador forte é obtido aprimorando-se classificadores fracos em série, baseados em estruturas tipo Haar ou padrões binários locais.

Palavras-chave— Robótica Aérea, Quadricópteros, Desempenho de classificadores, Visão Computacional

1 Introduction

Within the last few years, great efforts have been made towards the development of fully autonomous Unmanned Aerial Vehicles (UAVs) (Tomic et al., 2012). Vertical Taking-off and Landing (VTOL) vehicles, (see Figure 1) such as helicopters and quadrotors (Mozelli et al., 2015) deserved attention, since they are effective in undertaking several tasks including: forest monitoring and measurement; crowds management; crop pulverization; urban surveillance and cargo deployment (Munro et al., 2012).

They possess high maneuverability and versatility because can fly at both low and high speeds, perform lateral and longitudinal trajectories and hover, near the ground, which justifies their capability to conduct several tasks.

An important task for UAVs is the capability to perform automatic landing with safety. Towards fully automation, the landing-pad detection via digital images is a critical mission. Methodologies can be categorized according to the characteristics of the landing site. Roughly, it is possible to distinguish methods that detect a known target from methods that identify suitable sites in unknown environment. Another division is between stationary and moving sites. The first case is done by searching known patterns, which can

Figure 1: UAV during the landing procedure.

be artificial (like landing pads) or natural markers. In the second case, relevant characteristics of potential landing sites, such as flatness or terrain roughness, are measured for unstructured environments. In artificial markers category, Saripalli et al. (2002) used on-board cameras to detect simple markers ("H" characters marked on the ground) by resorting to previous known characteristics such as perimeter, shape and area, and combining the information provided by a GPS receiver. In Merz et al. (2006), complex markers are used (dismissing GPS), allowing the estimation of distance and orientation. For natural markers, Frew et al. (2004) uses pavemented roads as references for landing. For unprepared environments, Scherer et al. (2012)

‡Corresponding author.

uses Lidar sensors to estimate terrain flatness and roughness. However, some advanced sensors are very expensive and require vehicles with greater payload bearing capability. Another approach relies on the optical flow mimicking vision of flying insects (Herisse et al., 2008).

This paper focuses on automatic landing-pad detection, based on digital images. The goal is to detect traditional characters "H" or "X", inside circles. This task is part of an effort to develop UAV's control system for oceanic rescue missions. Although the vehicle is equipped with GPS, the inherent error for altitude measurement can be compensated by the visual detection system. The system is going to support the final approach, meaning that detection must occur within the vicinity of 5 m and bellow in any given orientation and angle. To achieve this goal two methods are presented, which are based on detecting local relevant features in images and using boosted cascade classifiers to perform supervised learning on a comprehensive data set.

2 Methodology

In recent years, machine learning has successfully been used for recognition of license plates (Wang et al., 2009; Zhao et al., 2010) and handwriting (Yilmaz et al., 2011). Supervised training methodologies seem robust for detection of known features. Due to the flexibility within the landing environment, which for every mission scenario could be subjected to changes, inspire the present research to limit and focus on the landing-pads with artificial markers that resemble characters "H" an "X" inscribed into a circle.

2.1 Multi-scale Block Local Binary Patterns

Local Binary Patterns (LBPs) are generally used as local descriptors for microstructures of images (Ojala et al., 1996). In this approach, each pixel of an image is labeled by thresholding the 3×3-neighborhood according to the center value (Figure 2). As a result, a texture unit is assigned to each pixel and the overall possibilities are called texture spectrum. Each texture unit can represent a spot, an edge and so on. LBP features have high tolerance against illumination changes and their computational cost is simple.

According to Liao et al. (2007), it is a powerful local descriptor, however, the original LBP operator has a very small spatial support area, which results in the bit-wise comparison between two single pixel values being affected by noise. Moreover, features calculated in such local neighborhood cannot capture higher scale structure that may be dominant features of many objects of interest.

Aiming to overcome those limitations, Liao et al. (2007) proposed an innovative representation,

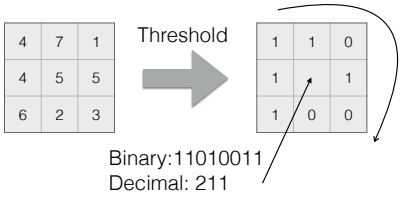


Figure 2: Original LBP operation. After applying the threshold to the neighbors, value 211 is assigned to the central pixel.

called Multi-scale Block LBP (MB-LBP), which has its calculation based on average values of block subregions, instead of individual pixels. This representation revels some advantages: higher robustness when compared to LBP; encoding of micro and macrostructures of image patterns; efficient computation using integral images. The original LBP can be regarded as a particular case of the MB-LBP.

2.2 Extended Set of Haar-Like Features

Haar-like features in two dimensions are composed of adjacent light and dark rectangles. Figure 3 shows some typical features used for recognition. They are usually used to detect and recognize objects, and were successfully used in Viola and Jones (2001) for real-time facial detection.

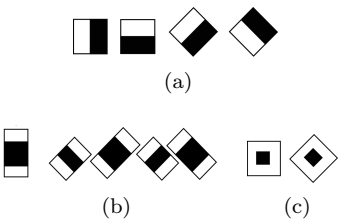


Figure 3: Typical Haar-like features: (a) edges, (b) lines and (c) center surrounded.

The presence of a Haar-like feature is determined by adjacent rectangular regions at a particular location within the detection window. The pixel intensities in these regions are summed and the difference between these regions is calculated. This is then used to place different subsections of an image into different categories. If the difference exceeds a threshold, then the feature is said to be present.

To compute the existence of several hundred Haar-like features at different scales and locations in a single image, Viola and Jones (2001) proposed the integral image technique. Each pixel is assigned as the sum of all pixels above it and to its left.

An extension to their approach is proposed in Messom and Barczak (2006), which analyses a set of rotated Haar-like features. This set enriches the simple features and can also be calculated efficiently. As a result, an average of 10 % lower

false alarm rate at a given hit rate is achieved. However, since some real time applications use low resolutions images, some numerical errors appear, impairing the use of rotated features.

2.3 Cascade Classifiers

Cascade classifiers are based on the idea of combining several weak and inaccurate classifiers, *i.e.*, better than random guessing, in series to achieve a highly accurate predictor. Each classifier is a decision stump, labeling a sample either as negative or positive. If the decision is negative, the process ends, otherwise it passes to the next classifier. Therefore each classifier must have a low false negative rate, since a mistake cannot be corrected in the next stages. However, in some circumstances, they may have a high false positive rate, since a mistake can be corrected in subsequent stages.

The method described in previous sections is the foundations of the weak classifiers. In each case, the local features, either MB-LBP or Haar-like, are searched within an image in the training set.

Boosting is the method responsible for raising the cascade classifier accuracy. A binary weak classifier must find a weak hypothesis, $h_t : \mathcal{X} \rightarrow \{-1, +1\}$, where the inputs for training belong to the domain set $\mathcal{X}$ and the labels $\{-1, +1\}$ are negative and positive, respectively. The final hypothesis is done by doing an weighted combination of the weak hypotheses:

$$H(x) = \sum_{t=1}^{T} \alpha_t h_t(x), \tag{1}$$

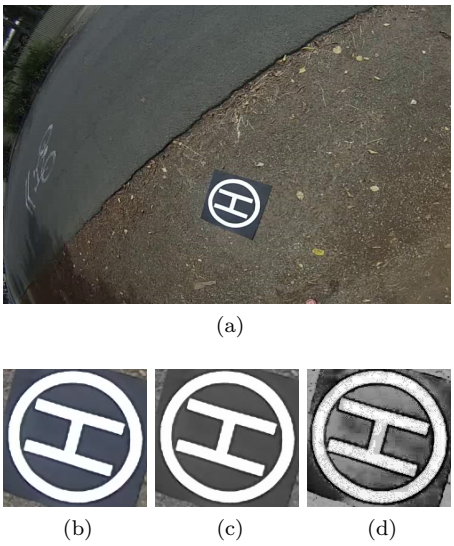
which is equivalent to saying that the final result is an election where the winner is chosen by the majority, but each elector t may have a different weight α_t. When there is a substantial majority, the prediction has a large confidence level.

In this paper, AdaBoost algorithm, proposed by Freund and Schapire (1997), is used, since it remais widely adopted and studied. Over the years, a great deal of effort there has been made in trying to explain the AdaBoost as a learning algorithm (Schapire, 2013). AdaBoost classifiers with more weight come first to eliminate negative regions more quickly.

2.4 Training Data Set

The training data set was acquired during an undergraduate internship of a co-author, sponsored by Brazil Scientific Mobility Program, in The Univesity of Adelaide /Australia.

The positive training set comprises 6,000 images of landing pads with two different features, "X" and "H". The database videos of the lading pads were captured with different cameras, having distinct resolution, noise and lens distortion. The shots were taken with diverse backgrounds (grass, asphalt and clay), under different lighting conditions and from different altitudes (0 to approximately 5 m), orientation and perspective angles. These videos were post-processed to extract one frame per second to *jpeg* sample images. The landing pad region of interest (ROI) was extracted in each image and, finally, saved as a positive sample. To complete the process, all images were converted to grayscale and had their histograms equalized. The process of creating a positive sample is summarized in Figure 4.



(a)

(b) (c) (d)

Figure 4: Generating a positive sample: (a) original frame sampled; (b) land pad extraction; (c) grayscale conversion; (d) equalized histogram.

The negative training set comprises 12,000 labeled images of objects, scenes, landscapes, animals and people, among others. These images were downloaded through a loaded random crawl of the Web, converted to grayscale and had their histograms equalized. The negative sample creation process is summarized in Figure 5.

(a) (b) (c)

Figure 5: Generating a negative sample: (a) original image downloaded; (b) grayscale conversion; (c) equalized histogram.

3 Experimental Results

During the courseof this research, for the classifiers training stage, an *Intel i3 Core* processor with 2.53 GHz and 4 GB RAM was used. All the training and testing process, including pre-processing of

Table 1: Parameters for different classifiers using Haar-like features or LBP.

Parameter	$\mathcal{C}^1$	$\mathcal{C}^2$	$\mathcal{C}^3$
Positive samples	6,000	4,875	1,125
Negative samples	12,000	9,750	2,250
Stage type	BOOST	BOOST	BOOST
Feature type (Haar)	Haar	Haar	Haar
Feature type (LBP)	LBP	LBP	LBP
Sample size (w x h)	24×24	24×24	24×24
Boost type	GAB	GAB	GAB

the samples, was performed using the *Open Computer Vision* (OpenCV), library version 2.4.11, on Linux platform.

For the experiments, a total of six classifiers were trained, three variations of each of the two methods discussed above: one using the total set of positive samples, which includes partial or entire images of "H" and "X" landing pads, named $\mathcal{C}^1$; another one using only pieces of "H" and "X" samples, called $\mathcal{C}^2$; and the last one using only entire images of "H" and "X" for training, named $\mathcal{C}^3$. Parameters common for both features are shown in Table 1. Unique parameters are identified as (Haar) or (LBP), respectively.

3.1 Classifiers Test

For the training stage, a test database has also been formed from the earlier extracted frames, excluding the images contained in the training dataset. This set is composed by 1,218 samples, 1,100 containing pieces or entire lading pads images and 118 non-containing any.

The classifiers trained with samples comprising only partial objects of interest were combined with the classifiers trained with samples containing only the entire object of interest in the same code. As a result, it was possible to compare the performances of a single classifier, trained to identify partial or complete landing pads, and a combination of each individual classifier, trained to recognize separately each of those two configurations.

3.2 Analysis

The receiver operating characteristic (ROC) curve is a graphical plot to display performance and quality of a binary classifiers (Fawett, 2004). To draw the ROC curve the positive rate is plotted against negative rate as some parameter that affects discrimination changes. Here, the parameter chosen was the minimum number of overlapping rectangles to mark and conduct detection within a ROI. This is based on the fact that every object of interest can be detected with several windows of consecutive sizes. So, if there is a detection only for a single window size, it can be assumed as false alarm.

For the creation of ROC curves, 11 different values were tested for window size detection; *i.e.* the neighborhood of detection, which varied from 0 pixels (infinite detections) to 1,000 pixels (≈ 0 detections). Each value was tested using the complete set of 1,218 test samples, totalizing 53,592 images processed. These images were then classified and separated manually as True Positive (TP), True Negative (TN), False Positive (FP) or False Negative (FN). Once classified each tested sample, it was possible to map the results between true positive rate (TPr) and false positive rate (FPr), given by:

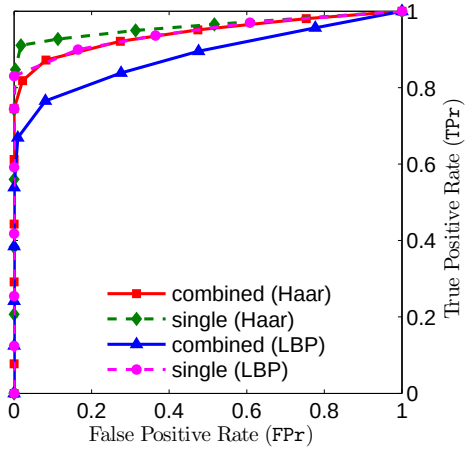


Figure 6: ROC curves for all evaluated classifiers.

$$\mathtt{TPr} = \frac{\mathtt{TP}}{\mathtt{TP} + \mathtt{FN}} \quad \text{and} \quad \mathtt{FPr} = \frac{\mathtt{FP}}{\mathtt{FP} + \mathtt{TN}},$$

Finally, the ROC curves were drawn. Furthermore, the accuracy of each classifier can be measured by the area under the curve (AUC) coefficient, which is obtained by the integral of the ROC curve from 0 to 1 (Ferri et al., 2011).

3.3 Discussion

Figure 6 presents the ROC curve analysis for the entire set of classifiers evaluated in this work. Generally speaking, it is possible to conclude, (for this specific application), that single classifiers present better results than combined ones, though the difference between combined Haar (red continuous line) and single LBP (magenta dashed line) is barely noticeable.

It is also possible to see the advantage of using the single Haar-like classifier instead of others, since it seems to be more precise. The AUC values of each detector, presented in Table 2, surpass 90 % which indicates that the classifiers have high hit rate with good accuracy, and low false alarming (zero for the best results achieved). The single Haar-like classifier gives the highest AUC (96.35 %), which is an indication that it is the best option, though the difference is not so significant.

Table 3 shows the TP, TN, FP and FN parameters for the best result achieved with the single

Table 2: different AUC values for different classifiers.

Method	Single	Combined
MB-LBP	0.963	0.912
Haar-like	**0.964**	0.958

Haar and LBP classifiers, as well as coefficients `TPr` and `FPr`.

Table 3: Performance of best result achieved for the classifiers based on Haar-like and LBP features.

Parameter	Haar	LBP
`TP`	1,016	1,001
`TN`	118	118
`FP`	0	0
`FN`	82	99
`TPr`	92.36 %	91.00 %
`FPr`	0 %	0 %

In this process, the average run-time was evaluated in order to identify the lower consumption time among the classifiers, which is important for real-time application, when a vehicle has limited on-board computational power. Once the training stage was performed, 29 random selected images were classified, and the time spent in the task was measured. With 97.5 %, the confidence interval for the single Haar-like method was [338.9, 377.6] ms., whereas for the single LBP was [216.2, 252.4] ms. Combined classifiers using Haar-like method produced [578.8, 657.6] ms whereas LBP method produced [394.7, 476.5] ms. Since the combined classifiers run 2 steps of detection in series, one trained for complete images followed by the other for partial images, they took almost twice much time.

Finally, Figure 7 provides some output data using the Haar-like single classifier, facing great variation on lens distortion, scale, attitude, texture and illumination conditions among the images.

4 Conclusion and Future Work

In this paper, we presented a comparison between Haar-like and LBP classifiers in the problem of detecting artificial landing marks for autonomous Unmanned Aerial Vehicles. We trained six classifiers providing different sets of input data and used the ROC curve analysis to compare then, in order to identify the best one. In conclusion, we have shown single classifiers $\mathcal{C}^1$ for both, Haar-like or LBP methods, provide better results than combine ones $\langle \mathcal{C}^2, \mathcal{C}^3 \rangle$. Particularly, the Haar-like was the more accurate among the four evaluated methods, though the difference was rather small. If time consumption is of more importance than accuracy, then LBP approaches are more desired, since they spent less time in the classification step.

Figure 7: Output of the classifiers on several test images.

As future work, several extensions are possible. For instance, other classifiers may be tested and compared with the two previous presented. These methods could also tested and confronted by looking for non-artificial landmarks on unstructured unknown areas.

Acknowledgments

The authors would like to thank the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) and the Fundação de Amparo à Pesquisa de Minas Gerais (FAPEMIG).

References

Fawett, T. (2004). ROC graphs: Notes and practical considerations for researchers, *Technical report*, HP Laboratories, Palo Alto, CA.

Ferri, C., Hernández-Orallo, J. and Flach, P. A. (2011). A coherent interpretation of AUC as a measure of aggregated classification performance, *Int. Conf. on Mach. Learning*, pp. 657–664.

Freund, Y. and Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting, *J. Comput. Syst. Sci.* **55**(1): 119 – 139.

Frew, E., McGee, T., Kim, Z., Xiao, X., Jackson, S., Morimoto, M., Rathinam, S., Padial, J. and Sengupta, R. (2004). Vision-based road-following using a small autonomous aircraft, *IEEE Aerosp. Conf.*, Vol. 5, pp. 3006–3015.

Herisse, B., Russotto, F.-X., Hamel, T. and Mahony, R. (2008). Hovering flight and vertical landing control of a VTOL unmanned aerial vehicle using optical flow, *IEEE Int. Conf. on Intel. Robots and Syst.*, pp. 801–806.

Liao, S., Zhu, X., Lei, Z., Zhang, L. and Li, S. (2007). Learning multi-scale block local binary patterns for face recognition, *IAPR/IEEE Int. Conf. on Biometrics*, Vol. NCS-4642, pp. 828–837.

Merz, T., Duranti, S. and Conte, G. (2006). Autonomous landing of an unmanned helicopter based on vision and inertial sensing, *in* J. Ang, Marcelo H. and O. Khatib (eds), *Experimental Robotics IX*, Vol. 21 of *Springer Tracts in Advanced Robotics*, Springer Berlin Heidelberg, pp. 343–352.

Messom, C. H. and Barczak, A. L. C. (2006). Fast and efficient rotated Haar-like features using rotated integral images, *2006 Australian Conf. on Robot. and Autom.*, pp. 1–6.

Mozelli, L. A., Alves-Neto, A. and Campos, M. F. M. (2015). Attitude of quadrotor-like vehicles: Fuzzy modeling and control with prescribed rate of convergence, *IEEE Int. Conf. on Robotics and Automation 2015 - ICRA '15*, IEEE, Seatle, USA, pp. 1710–1715.

Munro, B., Lim, D. and Anvar, A. (2012). A feasibility-study of a micro-communications sonobuoy deployable by UAV robots, *World Academy Sci., Eng. and Technology* **2012-10-28**: 584–588.

Ojala, T., Pietikainen, M. and Harwood, D. (1996). A comparative study of texture measures with classification based on feature distributions, *Pattern Recognition* **29**(1): 51–59.

Saripalli, S., Montgomery, J. and Sukhatme, G. (2002). Vision-based autonomous landing of an unmanned aerial vehicle, *IEEE Int. Conf. on Robot. Autom. - ICRA'02*, Vol. 3, pp. 2799–2804.

Schapire, R. E. (2013). Explaining adaboost, *in* B. Scholkopf, Z. Luo and V. Vovk (eds), *Empirical Inference*, Springer Berlin Heidelberg, pp. 37–52.

Scherer, S., Chamberlain, L. and Singh, S. (2012). Autonomous landing at unprepared sites by a full-scale helicopter, *Robot. Autonomous Syst.* **60**(12): 1545 – 1562.

Tomic, T., Schmid, K., Lutz, P., Domel, A., Kassecker, M., Mair, E., Grixa, I., Ruess, F., Suppa, M. and Burschka, D. (2012). Toward a fully autonomous UAV: Research platform for indoor and outdoor urban search and rescue, *IEEE Robot. Automat. Mag.* **19**(3): 46–56.

Viola, P. and Jones, M. (2001). Rapid object detection using a boosted cascade of simple features, *Proc. 2001 IEEE Conf. Comput. Vision and Pattern Recognition*, Vol. 1, pp. I–511–I–518 vol.1.

Wang, Y., Zhang, H., Fang, X. and Guo, J. (2009). Low-resolution chinese character recognition of vehicle license plate based on albp and gabor filters, *Int. Conf. Advances in Pattern Recognition*, pp. 302–305.

Yilmaz, M., Yanikoglu, B., Tirkaz, C. and Kholmatov, A. (2011). Offline signature verification using classifier combination of HOG and LBP features, *Inter. Joint Conf. on Biometrics*, pp. 1–7.

Zhao, Y., Gu, J., Liu, C., Han, S., Gao, Y. and Hu, Q. (2010). License plate location based on Haar-like cascade classifiers and edges, *2nd WRI Global Cong. on Intell. Syst. (GCIS)*, Vol. 3, pp. 102–105.