

TERRAIN CLASSIFICATION USING STATIC AND DYNAMIC TEXTURE FEATURES BY UAV DOWNWASH EFFECT

Submitted: 8th October 2018; accepted: 7th February 2019

João Pedro Carvalho, José Manuel Fonseca, André Damas Mora

DOI: 10.14313/JAMRIS_1-2019/11

Abstract:

Knowing how to identify terrain types is especially important in the autonomous navigation, mapping, decision making and emergency landings areas. For example, an unmanned aerial vehicle (UAV) can use it to find a suitable landing position or to cooperate with other robots to navigate across an unknown region. Previous works on terrain classification from RGB images taken onboard of UAVs shown that only static pixel-based features were tested with a considerable classification error. This paper presents a computer vision algorithm capable of identifying the terrain from RGB images with improved accuracy. The algorithm complement the static image features and dynamic texture patterns produced by UAVs rotors downwash effect (visible at lower altitudes) and machine learning methods to classify the underlying terrain. The system is validated using videos acquired onboard of a UAV with a RGB camera.

Keywords: Image processing, Texture, Machine Learning, Terrain Classification, Neural Networks, UAV

1. Introduction

Nowadays, due to UAVs' higher availability and capabilities, there is a research trend to explore innovative applications of UAVs useful to the society. They are having a major impact on search and rescue missions, in logistics, in precision agriculture, among other applications. Key issues are to provide a safe and reliable operation and to perceptionate the surrounding area. This latter, within this paper, will be to identify the underlying terrain. Terrain classification is a crucial functionality for a wide range of autonomous vehicles [12]: either for ground vehicles to avoid water bodies, aerial vehicles to determine suitable landing areas, or surface vehicles to detect safe passageways. As further explained in section 2, several approaches have been used for terrain classification. However, there is still margin for improving accuracy by extracting more complex image features. When at lower altitudes, UAV's rotors downwash effect create singular image texture patterns depending on the type of terrain, which can be used to differentiate them.

The main goal of this paper is to propose a computer vision algorithm that using RGB images captured by a camera onboard of a UAV is capable of classifying a terrain by analysing static image features (colour and texture) and rotors downwash effect on the underlying surface. There are several issues that must be addressed in order to achieve this goal, namely:

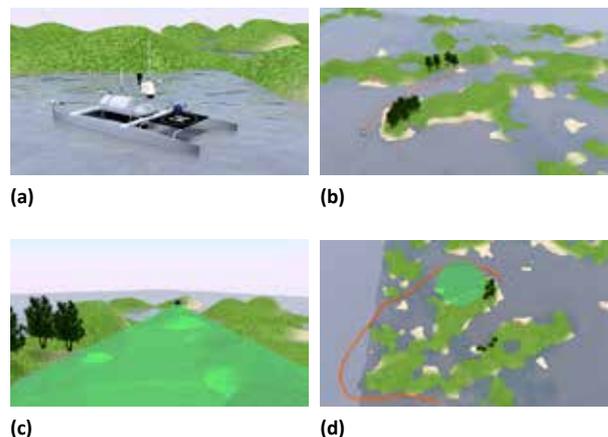


Fig. 1. a) Cooperation between Unmanned Aerial Vehicle (UAV) and Unmanned Surface Vehicle (USV) to improve the autonomous navigation. b) shows a plan made by only USV. c) and d) show a cooperative UAV to improve the plan made by USV [12]

- Which terrains can be more accurately classified using the downwash effect?
- Which are the texture and motion patterns of each terrain (water movement for example)?
- Which static and dynamic image features can be extracted to classify the terrain?

To address these challenges, new optimization procedures and techniques will be proposed in this paper, aiming the best possible performance.

This paper is structured with six sections starting with an introductory section and followed by a presentation of related works. In the experimental setup section the system background, namely the hardware and the terrain types, are described. On the Terrain Classification Method the system architecture, the static and dynamic texture features and the machine learning classifier will be presented. The article finishes with the experimental results and drawn conclusions.

2. Related Work

UAVs (Unmanned Aerial Vehicle) play an important role on the new generation of information technology and is predicted to have a major impact in the human life in the near future [1]. One of the areas is in computer vision, where it is possible to acquire, process, analyse and understand aerial images. Many researchers have proposed terrain classification systems based on features derived from colour informa-

tion [2], texture patterns [4], [8], [11], [9] and from additional sensors, as is the case of laser scan systems [15], [14], [6]. Although many of these algorithms are for terrestrial unmanned ground vehicles, currently there is a shift towards UAVs, where the visual features have wider importance.

One of the most recent works of terrain detection and classification is presented in [13]. The authors use the concept of optical flow to detect the water texture direction in images acquired by an RGB camera onboard of a UAV. From the directions of the textural features, the algorithm determines if the terrain, where the UAV is flying over, is water or non-water. One of the problems identified is that the UAV must be stable over the target while identifying the type of terrain, which, in the best case, takes four seconds to execute. Another reason that requires the UAV to be stand still during calculations is that the computer vision algorithm does not compensate the UAV movement. Thus, when the directions of the features are calculated, the results do not represent the reality.

A classification method using colour features was proposed in [2]. The proposed method converts a RGB image into an image entitled "normal RGB", where each pixel is divided by the square root of the three colour channels. Thus, each terrain will emphasize the colour that represents it (for example, green for vegetation). The proposed method was limited due to the fact that it varies significantly with illumination.

Laser scanners have proven to be important to distinguishing between land and water as presented in [15], [14] and [6]. However, in low water depths the laser sensor produces incorrect results, due to the fact that it captures reflections from the seabed and misclassifies it as non-water terrain. Therefore, this laser scan approach, by itself, reveals to be insufficient and requires additional equipment.

3. Experimental Setup

3.1. UAV Platform Design

The aerial vehicle used in the experiments and named as Vigil R6 (Figure 2), is a six-rotor solution that endows the system with graceful degradation, as it is able to land with one motor off, although without yaw control in that condition.



Fig. 2. Aerial robot Vigil R6

Some of the Vigil R6 specifications are:

- A VRBRAIN hardware is used to control the low level operation. This hardware contains the IMU and GPS to know the UAV's position and orientation. Also, the

VRBRAIN connects to the Odroid-U2 embedded system via MAVlink protocol and the UAV's battery. Lastly, to control the UAV's motors it is connected to a UHF receiver;

- An Odroid-U2 is used to control the high level operation. It receives data from the distance sensor (LIDAR), the RGB camera and communicates with external devices via Wifi link;
- A RGB camera with a gimbal stabilizer, is installed capturing onboard images at a resolution of 640x480 pixels;
- The camera and lens specifications are known, allowing the field of view (FOV) and the pixel size in meters to be computed.

For a better understanding of the communications between various system layers, in Figure 3 it is represented the system architecture.

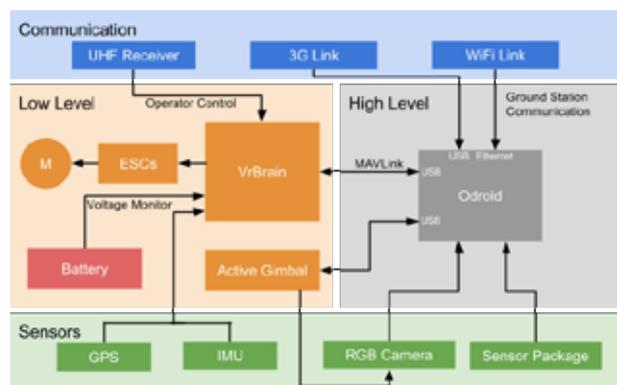


Fig. 3. Layer communication

3.2. Terrain Types

The dynamic of different terrains when exposed to wind provoke singular texture patterns that can be used in their identification. In this paper we study the importance of static image features, such as colour and texture, when compared with the dynamic features exhibited by the downwash effect, for terrain classification.

In this work three different terrain types (water, vegetation and sand), which can benefit from the downwash effect for their identification (Figure 4) were identified. It can be seen that the downwash effect produces: on water a circular dynamic texture; on vegetation a linear spread from inside outwards; and on sand it is almost stable or it moves outwards.

4. Terrain Classification Method

If different types of terrain behave differently when exposed to UAV rotors downwash effect, then it should be possible to obtain unique information for their identification. Based in this research hypothesis, it is possible to obtain some conclusions. When exposed to the downwash effect, water particles' movement is always greater than in vegetation and sand terrains. Also, regarding static texture, usually vegetation has a more rough texture than sand or water terrains; water only presents roughness when exposed to

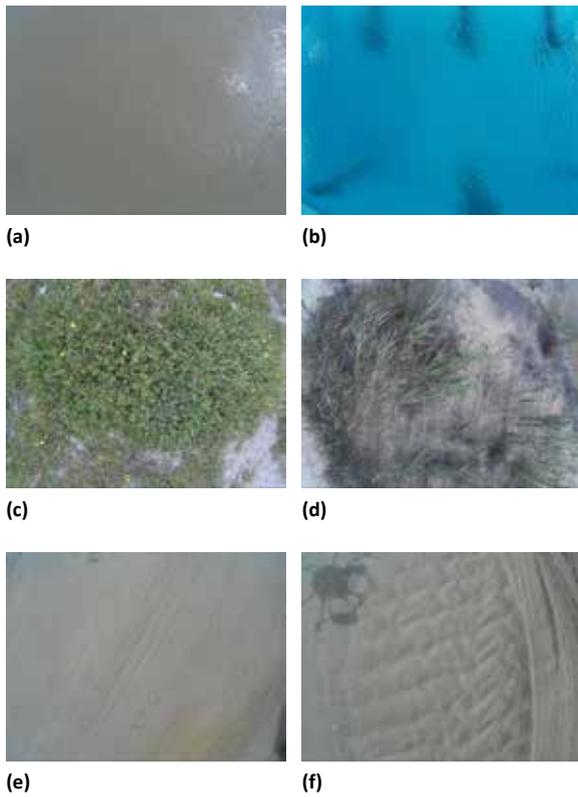


Fig. 4. Examples of terrain types: water (a)(b); vegetation (c)(d); and sand (e)(f)

wind and downwash effect; and sand (fine grains) has a lower roughness. It can also be seen that sand depends on the patterns already in the terrain, showing usually a more irregular texture (figures 4.e and 4.f) when compared with water that shows a unique signature and regular texture when exposed to wind (figures 4.a and 4.b).

4.1. System Architecture

The proposed system architecture to classify the terrain using texture information is shown in Figure 5. As previously identified in sections 1 and 4, two texture features are proposed to classify the terrain, namely, static and dynamic textures. At this stage it were also assessed the features that can be computed in parallel, in order to speedup the system execution time.

Five main processes were identified in the architecture (figure 5), namely:

- **Rectified Image:** Performs lens geometrical corrections;
- **Texture Filter:** Extracts terrain's static textural information using Gabor filters;
- **Threshold:** A thresholding is applied to the static texture image to highlight the terrain roughness;
- **Projections:** Vertical and horizontal projections were applied to the thresholded image, extracting unique features that help differentiate the different types of terrains;
- **Motion Analysis:** Extracts information from dynamic textures. Optical flow and thresholding techni-

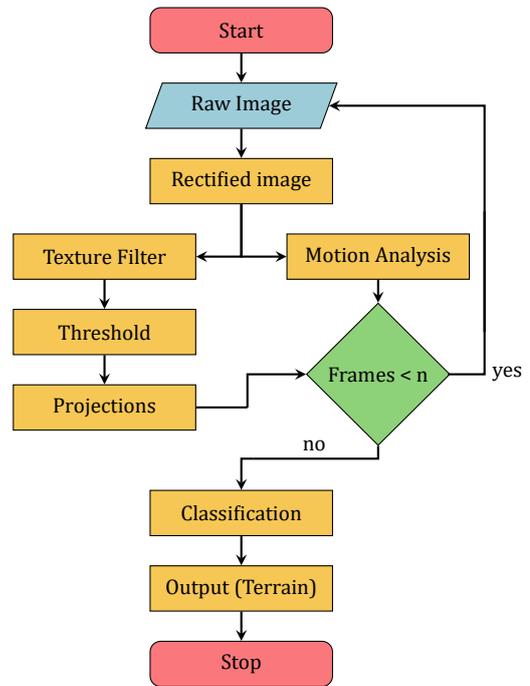


Fig. 5. Proposed system architecture

ques are used to identify the moving parts;

- **Classification:** The extracted features are used as inputs of an automatic classifier to identify the type of terrain. Machine learning techniques already proved to be efficient for terrain classification [10], [7], [5].

4.2. Static Textures

This section presents the proposed method for extracting terrain's static textures, based on the Gabor filter to be able to choose multiple texture directions. This filter is the impulse response formed by a multiplication of a sinusoidal signal with a Gaussian envelope function and can be computed using the following complex equation:

$$G(x, y, \lambda, \theta, \psi, \sigma, \gamma) = e^{\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right)} e^{i\left(2\pi\frac{x'}{\lambda} + \psi\right)} \quad (1)$$

Its real and an imaginary components can be obtained by equations 2 and 3, respectively:

$$G(x, y, \lambda, \theta, \psi, \sigma, \gamma) = e^{\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right)} \cos\left(2\pi\frac{x'}{\lambda} + \psi\right) \quad (2)$$

$$G(x, y, \lambda, \theta, \psi, \sigma, \gamma) = e^{\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right)} \sin\left(2\pi\frac{x'}{\lambda} + \psi\right) \quad (3)$$

where:

$$x' = x \cos(\theta) + y \sin(\theta) \quad (4)$$

$$y' = -x \sin(\theta) + y \cos(\theta) \quad (5)$$

These equations (1, 2 and 3) require as input parameters:

- **x and y**: Filter coordinates, where x represents the columns and y the rows;
- **Lambda (λ)**: Represents the sinusoid's wavelength;
- **Theta (θ)**: Defines the Gaussian envelope orientation;
- **Psi (ψ)**: Symbolizes the phase offset;
- **Sigma (σ)**: Describes the Gaussian envelope size;
- **Gamma (γ)**: Reflects the shape of the ellipse in the gabor filter space.

In this work we used only the real component of the Gabor function (equation 2). After obtaining the multiplication of a Gaussian with a sinusoidal function, i.e. the kernel of the filter, it will be convolved with the original image (equation 6). The result of the Gabor filter applied over a water surface is presented in Figure 6.

$$f[x, y] * g[x, y] = \sum_{-n1}^{n1} \sum_{-n2}^{n2} f[n1, n2] \cdot g[x - n1, y - n2] \quad (6)$$

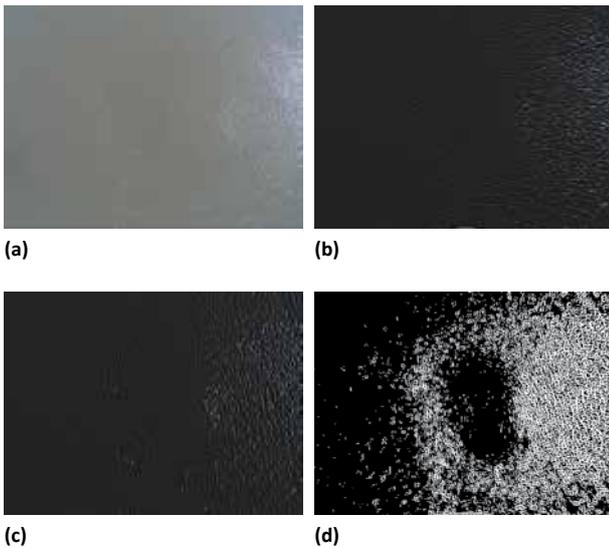


Fig. 6. Example of a static texture extraction: a) Raw image; b) c) Convolution with the Gabor filter $\theta=0$ degrees (b) and $\theta=90$ degrees (c); d) Sum of images b) and c) after thresholding

As can be seen in figure 6, it is possible to obtain the texture of a water-type terrain when it is affected by the downwash effect of the UAV. From the binarized image, a width projection was made to see the singular features of this terrain type (Figure 7).

From the observed vertical projection of water type terrain (figure 7) it can be seen that it produces an undulatory effect with a local minimum in the centre of the downwash. This effect in water type terrains is due to the lower roughness in the centre of the downwash. However, due to the water movement, around the centre a higher roughness is observed

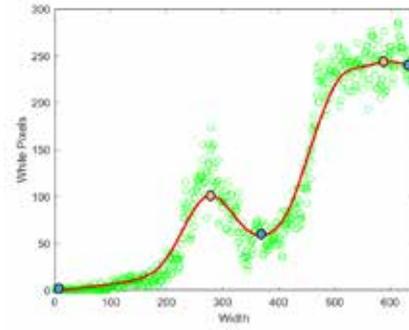


Fig. 7. Width projection of the example in Figure 6.d

(white pixels in the binarized image in figure 6.d). The next step was to translate this observed feature into a computational model.

By calculating the local maxima and minima of the vertical projection in figure 6.d, it is possible to calculate a line (red line in Figure 7) that most closely approximates these points. A polynomial regression was used.

The Lowess's theory calculates a line closest to a given set of points, through weights assigned to each point in a neighbouring window. In this work, to reduce the number of points, only the projection's local maxima and minima were used. The number of points that this window can contain is defined by the user. Once defined, it will traverse all points of the vector. In each set of n points, the best values for the slope (m) and the intercept (b) that minimize the sum squared residuals are calculated using equation 7:

$$S(m, b) = \sum_{i=1}^n w_i ((m \cdot x_i + b) - y_i)^2 \quad (7)$$

where, the weight function w is:

$$w(d) = \begin{cases} (1 - |d|^3)^3 & \text{for } |d| < 1 \\ 0 & \text{for } |d| \geq 1 \end{cases} \quad (8)$$

In equation 8, d is the distance between each window point and its neighbours. The final step is to obtain for each window the smoothed projection values from the obtained line equation (m and b), i.e., the red line in figure 7.

After obtaining this smoothed projection, new local minima and maxima are calculated and used to obtain two features:

- 1) Area measured between the local minimum and its respective two local maxima;
- 2) Integral between the local minimum and its two respective local maxima.

The first has the advantage of being relative to minima and maxima values, while the integral gives an absolute value and will vary for lower and higher roughness.

4.3. Dynamic Textures

This section presents the proposed method for extracting dynamic terrain textures.

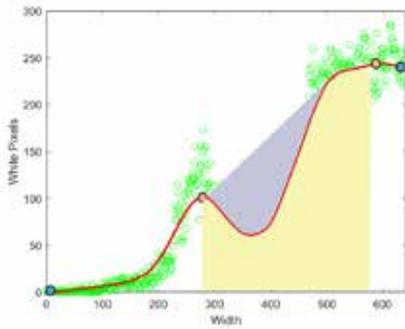


Fig. 8. Width projection showing the area (gray color) in relation with integral (yellow)

As mentioned in section 4, water-type terrain only exhibit dynamic texture when exposed to the down-wash effect. However, in spite having a dynamic texture, when analysing the optical flow it is never stronger than the dynamic observed for sand and vegetation. As referred in section 3, the optical flow method can calculate the distance travelled by block matching features in a given frame sequence. In this paper, the Farneback algorithm [3] was used to detect the movement of these features. One of the advantages to using the Farneback algorithm is the direct flow, F_d , return of features between two frames. The flow displacement F_d between features in frame n and $n - 1$ can be obtained from equation 9:

$$F_d = S_n - S_{n-1} \quad (9)$$

where S_n and S_{n-1} are the sample pixels between two frames.

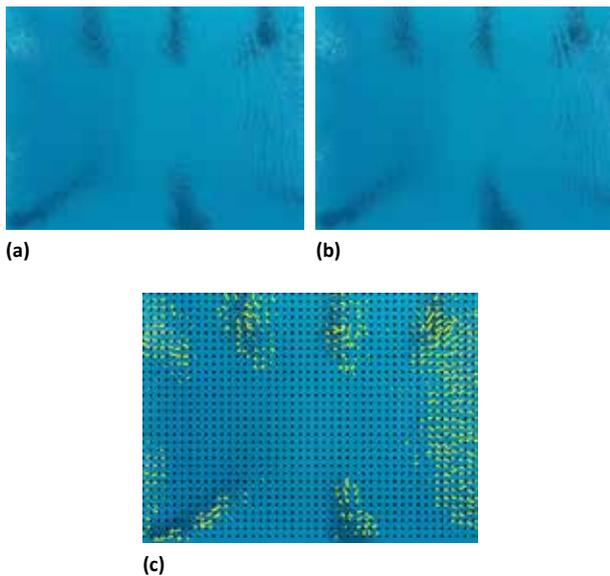


Fig. 9. Example of a Optical Flow concept using Farneback algorithm: a) Current frame; b) Next frame; c) Optical Flow result

With the obtained flow, as shown in Figure 9 is then used to calculate the distance travelled (trajec-

tory) by each feature in a sequence of frames:

$$Travel_{distance} = \sum_{i=2}^n \sqrt{A_x(i) + B_y(i)} \quad (10)$$

where:

$$A_x(i) = [x_1 - x_{i-1} + F_{dx}]^2 \quad (11)$$

$$B_y(i) = [y_1 - y_{i-1} + F_{dy}]^2 \quad (12)$$

and x_i and y_i are the positions in x and y in the most recent frame (n), x_1 and y_1 are the initial positions ($n = 1$) and F_{dx} and F_{dy} are the flow displacements between frames n and $n - 1$. We used normalized x and y coordinates for the calculations.

To eliminate features that did not move or were almost static in a sequence of frames, it was imposed that:

$$\begin{cases} \text{Red feature} & \text{for Trajectory} \geq \text{Threshold} \\ \text{Nothing} & \text{for Trajectory} \leq \text{Threshold} \end{cases} \quad (13)$$

From the equation 13 and knowing the maximum number of features (it is pre-defined the number of features in rows and cols in the image), we calculate the percentage of dynamic features that appear in the image (equation 14):

$$Dynamic_{feature} = \frac{\text{filtered features}}{\text{Total features}} \cdot 100\% \quad (14)$$

From equation 14, an example showing this theory is shown in Figure 10.

4.4. Classification

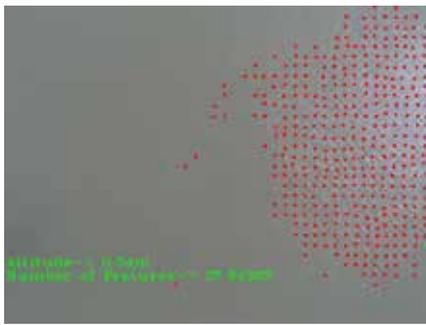
The data from dynamic textures and static textures are quite alike. It's not a trivial task for a human being to identify a terrain from the outputs of the two analyses. In order to merge the data while increasing certainty and automating the classification of the type of terrain, a machine learning technique was used, namely a feed-forward neural network (NN).

The architecture of the designed neural network, was composed by two layers, a hidden layer with 10 neurons and an output layer with 3 neurons (water, vegetation and sand).

The NN received the integral and area between the two local maxima as mention in sub-section 4.2, and the number of dynamic features detected as shown in sub-section 4.3. A sigmoidal function (equation 6) was used as activation function and the final classification was derived from the output neuron with highest activation value.

$$S_{\theta}(x) = \frac{1}{1 + e^{-\theta T x}} \quad (15)$$

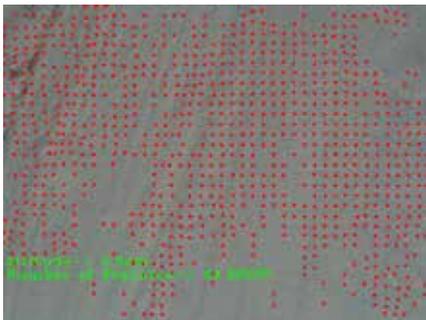
The training dataset was composed by 251 samples, from which 70% were for training, 15% for testing and 15% for validation. Even though we were



(a)



(b)



(c)

Fig. 10. Dynamic textures detection by Farneback algorithm and distance travelled calculation. a) water; c) vegetation and d) sand

using a quite simple NN, after training the NN, it was obtained 92.9% accuracy on the training set and 93.8% on the test dataset.

5. Experimental Results

To validate the proposed static and dynamic texture features for terrain classification, a total 251 frames from several types of terrains were used to validate the proposed system. From these 90 frames were for water, 88 frames for vegetation and 73 frames for sand.

Regarding the static texture feature the area and integral were calculated and displayed in Figure 11. It is possible to observe a clear separation between water, vegetation and sand, even with some outliers. In water type terrain, the three clusters that can be noticed for the integral feature, were mainly due to different water environments (lake and pool).

To validate the dynamic texture feature it was calculated in a three frame period ($n = 3$) and plotted

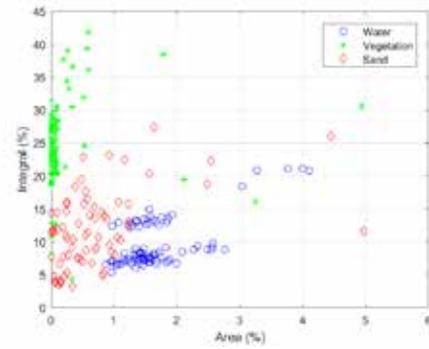


Fig. 11. Static Texture - Relation in area with respect to the integral of minimum and maxima locals

against the area feature from the static texture. This feature shows the same discriminant level to separate the different terrains. From Figure 12, it can be seen that water type terrain obtained a lower dynamic texture value ($< 45\%$), which can be due to a higher concentration of these dynamic features in the downwash centre and outside hasn't exceeded the threshold. Sand and vegetation shown a more uniform pattern, obtaining a higher number of features. On average, sand presents a percentage between 55% to 90%. Finally, vegetation with a percentage of features between 90% and 100%, is the terrain with highest dynamic texture, i.e., moving features.

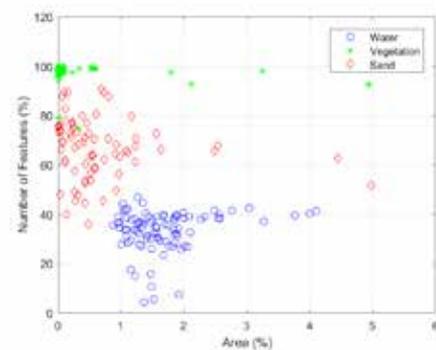


Fig. 12. Dynamic Texture - Relation in number of features with respect to the integral of minimum and maxima locals

Finally, these features were extracted from the figures 4.a-f and shown to the neural network classifier, which outputted the automated terrain classification. The extracted features and the classification result is shown in Table 1. As expected the proposed features and classification method, shown good results by classifying correctly all the six examples, reinforcing the idea that a combination of static and dynamic texture can be used to automatically extract terrain type from RGB images.

6. Conclusion

The main objective of this paper was to design a computer vision system capable of extracting static and dynamic image features, such as optical flow and

Tab. 1. Experimental Results

Figure	Static Texture		Dynamic Texture	Classification
	Area (%)	Integral (%)	Number of Features (%)	
1	1.44	8.32	32.88	water
2	1.55	7.59	36.70	water
3	0.01	27.42	98.32	vegetation
4	0.05	23.71	98.69	vegetation
5	0.24	14.67	63.74	sand
6	0.07	4.07	59.30	sand

texture features, to identify the type of terrain with improved accuracy by taking advantage of the the UAV's rotors downwash pattern effect. For this, it was necessary to conduct a research into detection methods already implemented and of interest to this work.

Texture features, such as Gabor filtering (static textures) and optical flow (dynamic textures), were studied to improve terrain classification aiming the best possible performance.

We emphasize that by implementing the static textures filter, vegetation-like terrains were found to have a higher texture than sand and water type terrains. On the other hand, water-type terrain, also presents a singular characteristic due to the downwash effect provoked by the UAV, which can be decisive to different it from other terrain types.

Although the system takes only 90 ms to classify the type of terrain where the UAV is flying over, the algorithm is not taking into account the drone movement which will bring some noise in the execution of the dynamic texture for the terrains classification. The main goal will be, with the data from IMU, GPS, GLONASS, barometer and accelerometer, merged in a kalman filter, give more reliable position of the UAV and thus it will be possible to remove the movement of the drone and consequently increase the accuracy of the system for terrain classification.

One of the major problems encountered throughout this work was the irregularity that the sand presents. As this terrain presents irregular textures, the static part of the system may take on forms of all kinds (but never presents a high texture - almost white image), including the wave form of the water-type terrain. It is for this reason that it is necessary to join more parallel logic to this algorithm in order to avoid this type of problems.

Classification is an interesting and engaging topic on nowadays research community. On the future, we expect to explore different techniques in order to achieve higher accuracy meanwhile having the most generic model. As a start, we need to acquire a larger data set with unbiased and varied frames. From that we can obtain different static features using state of the art Deep Neural Networks. As input of the deep net, we can plug the each frame and the information from the Dynamic textures analyses. From that, we expect the network to find it's own static features that represent a terrain, and increase the classification precision from the motion data.

Another approach we intend to explore is the in-

clusion of more drone flight data into the Machine Learning algorithm in use. For example, Altitude data gives a good clue to wish information can be extracted from each pixel in the image, and the drone drift can influence the image stabilization. All this precious information can be merged to increase precision and the resilient to unexpected behaviours and scenarios.

ACKNOWLEDGEMENTS

This work was partially funded by FCT Strategic Program UID/EEA/00066/203 of the Center of Technologies and System (CTS) of UNINOVA – Institute for the Development of new Technologies. In last, this work was not possible without the support and commitment of several fellow colleagues and friends, namely: Ricardo Mendonça, Francisco Marques, André Lourenço, Eduardo Pinto and José Barata.

AUTHORS

João Pedro Carvalho – Computational Intelligence Group of CTS/UNINOVA, FCT, University NOVA of Lisbon, e-mail: jp.carvalho@uninova.pt.

José Manuel Fonseca – Computational Intelligence Group of CTS/UNINOVA, FCT, University NOVA of Lisbon, e-mail: jmf@uninova.pt.

André Damas Mora – Computational Intelligence Group of CTS/UNINOVA, FCT, University NOVA of Lisbon, e-mail: atm@uninova.pt.

REFERENCES

- [1] Y. Bestaoui Sebbane, *Intelligent Autonomy of UAVs: advanced missions and future use*, number no. 3 in Chapman & Hall/CRC artificial intelligence and robotics, Chapman and Hall/CRC: Boca Raton, FL, 2018.
- [2] F. Ebadi and M. Norouzi, "Road Terrain detection and Classification algorithm based on the Color Feature extraction". In: *2017 Artificial Intelligence and Robotics (IRANOPEN)*, 2017, 139–146, 10.1109/RIOS.2017.7956457.
- [3] G. Farnebäck, "Two-Frame Motion Estimation Based on Polynomial Expansion". In: J. Bigun and T. Gustavsson, eds., *Image Analysis*, 2003, 363–370.
- [4] Q. Feng, J. Liu, and J. Gong, "UAV Remote Sensing for Urban Vegetation Mapping Using Random Fo-

- rest and Texture Analysis”, *Remote Sensing*, vol. 7, no. 1, 2015, 1074–1094, 10.3390/rs70101074.
- [5] A. Giusti, J. Guzzi, D. C. Cireşan, F. He, J. P. Rodríguez, F. Fontana, M. Faessler, C. Forster, J. Schmidhuber, G. D. Caro, D. Scaramuzza, and L. M. Gambardella, “A Machine Learning Approach to Visual Perception of Forest Trails for Mobile Robots”, *IEEE Robotics and Automation Letters*, vol. 1, no. 2, 2016, 661–667, 10.1109/LRA.2015.2509024.
- [6] W. Gruszczynski, W. Matwij, and P. Cwiakala, “Comparison of low-altitude UAV photogrammetry with terrestrial laser scanning as data-source methods for terrain covered in low vegetation”, *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 126, 2017, 168–179, 10.1016/j.isprsjprs.2017.02.015.
- [7] B. Heung, H. C. Ho, J. Zhang, A. Knudby, C. E. Bulmer, and M. G. Schmidt, “An overview and comparison of machine-learning techniques for classification purposes in digital soil mapping”, *Geoderma*, vol. 265, 2016, 62–77, 10.1016/j.geoderma.2015.11.014.
- [8] Y. N. Khan, P. Komma, K. Bohlmann, and A. Zell, “Grid-based visual terrain classification for outdoor robots using local features”. In: *2011 IEEE Symposium on Computational Intelligence in Vehicles and Transportation Systems (CIVTS) Proceedings*, 2011, 16–22, 10.1109/CIVTS.2011.5949534.
- [9] J. P. Matos-Carvalho, J. M. Fonseca, and A. D. Mora, “UAV downwash dynamic texture features for terrain classification on autonomous navigation”. In: *Annals of Computer Science and Information Systems*, vol. 15, 2018, 1079–1083.
- [10] A. Mora, T. M. A. Santos, S. Łukasik, J. M. N. Silva, A. J. Falcão, J. M. Fonseca, and R. A. Ribeiro, “Land Cover Classification from Multispectral Data Using Computational Intelligence Tools: A Comparative Study”, *Information*, vol. 8, no. 4, 2017, 147, 10.3390/info8040147.
- [11] M. Pietikäinen, A. Hadid, G. Zhao, and T. Ahonen, *Computer Vision Using Local Binary Patterns*, number Volume 40 in Computational Imaging and Vision, Springer: London, 2011.
- [12] E. Pinto, F. Marques, R. Mendonça, A. Lourenço, P. Santana, and J. Barata, “An autonomous surface-aerial marsupial robotic team for riverine environmental monitoring: Benefiting from coordinated aerial, underwater, and surface level perception”. In: *2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014)*, 2014, 443–450, 10.1109/ROBIO.2014.7090371.
- [13] R. Pombeiro, R. Mendonça, P. Rodrigues, F. Marques, A. Lourenço, E. Pinto, P. Santana, and J. Barata, “Water detection from downwash-induced optical flow for a multirotor UAV”. In: *OCEANS 2015 - MTS/IEEE Washington*, 2015, 1–6, 10.23919/OCEANS.2015.7404458.
- [14] L. Wallace, A. Lucieer, Z. Malenovský, D. Turner, and P. Vopěnka, “Assessment of Forest Structure Using Two UAV Techniques: A Comparison of Airborne Laser Scanning and Structure from Motion (SfM) Point Clouds”, *Forests*, vol. 7, no. 3, 2016, 62, 10.3390/f7030062.
- [15] W. Y. Yan, A. Shaker, and N. El-Ashmawy, “Urban land cover classification using airborne LiDAR data: A review”, *Remote Sensing of Environment*, vol. 158, 2015, 295–310, 10.1016/j.rse.2014.11.001.

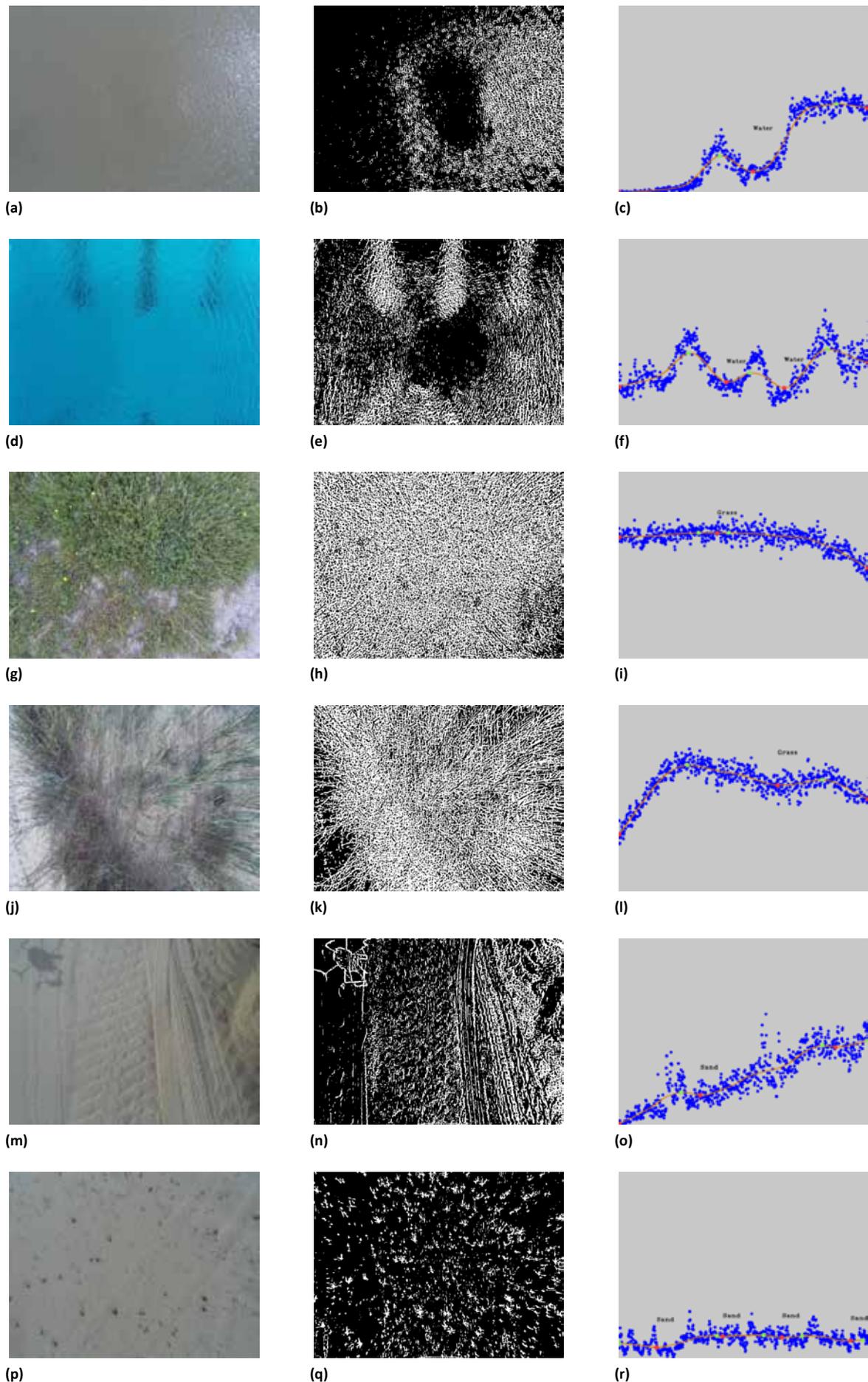


Fig. 13. Examples of terrain tested for their classification using static texture algorithm. With the input image (first column) it is possible to use the gabor filter concept (second column) and therefore the width projection calculation (third column)

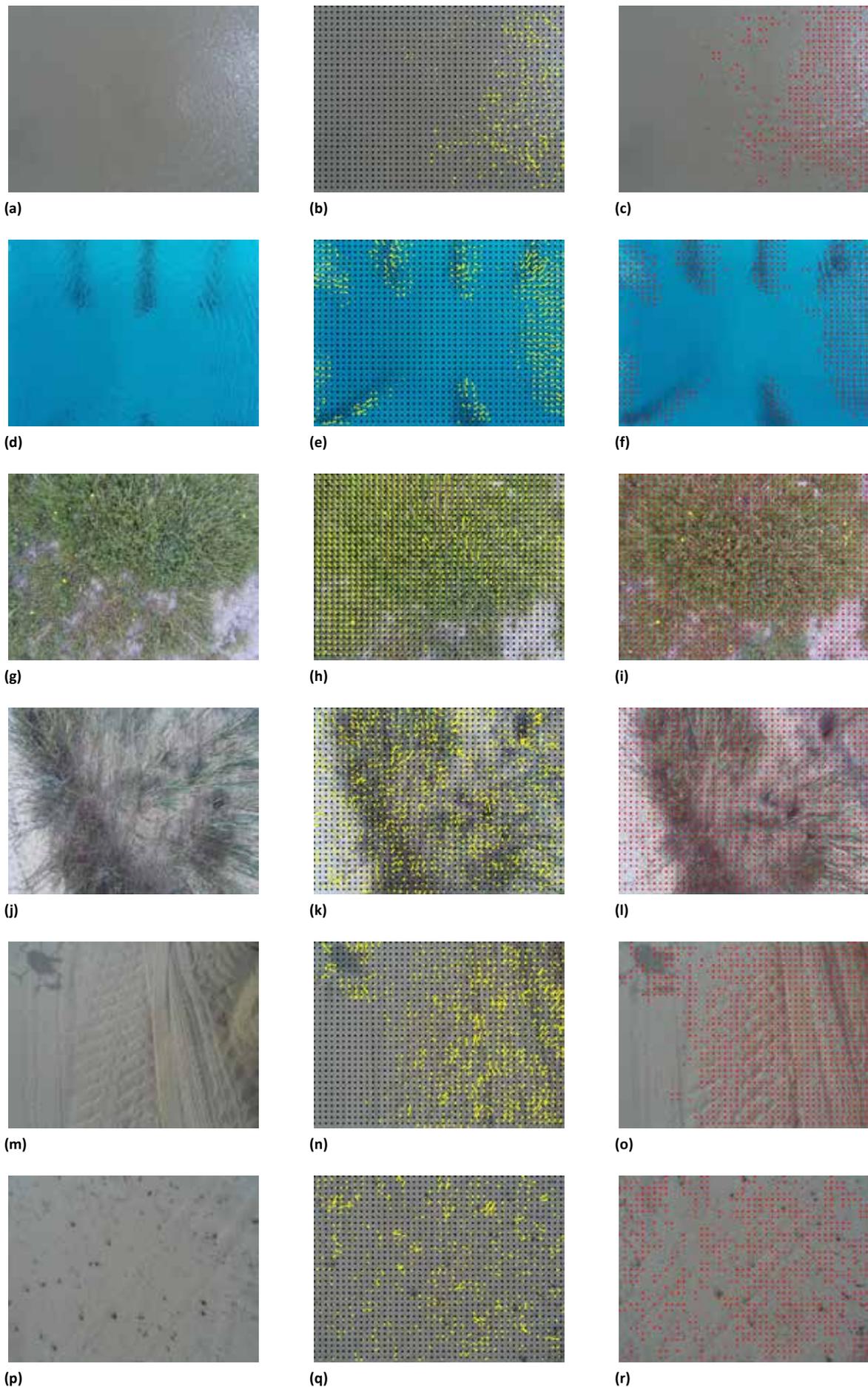


Fig. 14. Examples of terrain tested for their classification using dynamic texture algorithm. With the input image (first column) it is possible to use the optical flow concept (second column) and therefore the travel distance calculation (third column)

