Route planning for active classification with UAVs

Kelen C. T. Vivaldini[1], Vitor Guizilini[3], Matheus D. C. Oliveira[2], Thiago H. Martinelli[2], Denis F. Wolf[2] and Fabio Ramos[3]

***Abstract*— The mapping of agricultural crops by capturing images obtained with UAVs enables fast environmental monitoring and diagnosis in large areas. Airborne monitoring in agriculture can a substantially impacts on the identification of diseases and produce accurate information on affected areas. The problem can be formulated as a classification task on aerial images with significant opportunities to impact other fields. This paper presents an active learning method through route planning for improvements in the knowledge on visited areas and minimization uncertainties about the classification of diseases in crops. Binary Logistic Regression and Gaussian Process were used for the detection of pathologies and map interpolation, respectively. A Bayesian optimization strategy is also proposed for the planning of an informative trajectory, which resulted in a maximized search for affected areas in an initially unknown environment.**

I. INTRODUCTION

Unmanned aerial vehicles (UAVs) provide significant opportunities that revolutionize industrial agriculture through automation of the mapping and disease monitoring process. By autonomously flying over large areas, UAVs can build a precise picture of the farm and how crops evolve over time, which impact on all levels of decisions, from sowing and harvest times to effective prevention of pests.

Previous research in Geographic Information System (GIS) and remote sensing has used satellite images and videography for the evaluation and monitoring of plantations. The detection and characterization of the extent of soil contamination, drainage, vegetation, and degree of change caused by the impact and subsequent monitoring of the affected areas are of vital importance for an environmental impact assessment [1]. A continuous monitoring can avoid substantial losses caused by the spread of diseases in planted areas. In Brazil, eucalyptus is one of the main sources of raw material. An estimated $400 million per year is lost due to diseases spread over in 5,1 million hectares of eucalyptus forests [2][3].

Remote sensing technology with UAVs (Unmanned Aerial Vehicles) has been adopted to assist in monitoring applications. UAVs provide low-cost data acquisition platforms, high spatial and temporal resolutions and fast scanning of large areas in comparison to manned aircraft or satellites [4]. Several techniques for the extraction and classification of aerial images were compared in [5]. UAVs can be controlled remotely or fly autonomously on a predefined route. The route selected has a significant impact on the monitoring performance. According to Karakaya [6], the flight route planning should enable the monitoring of all or a maximum number of given targets, however, in real scenarios, the route might be insufficient to cover all targets. The authors proposed a modified Max-Min Ant System (MMAS) algorithm to cover more targets with shorter distances.

Marchant and Ramos [7][8] proposed a method for environmental monitoring that uses mobile robots to calculate continuous paths for spatial-temporal prediction of a dynamic phenomenon. The method adopts Bayesian Optimization (BO) [9] and builds a Gaussian Process (GP) to model an environmental phenomenon and choose an appropriate set of parameters for the BO acquisition function. Such function deals with the exploration-exploitation trade-off automatically and takes into account the reduction in travel distances (Distance-based Upper Confidence Bound - DUCB and Upper Confidence Bound Continuous Sampling - UCBC). Tests were conducted to monitor the luminosity of an environment, so as to improve predictions. Experiments show a 40% reduction in the distance traveled and a notable improvement in the monitoring of high luminosity areas. Souza et al. [10] developed a method based on Bayesian Optimization to reduce the vibration during robot navigation in different terrains. As a result, the robot navigated in areas of lower vibration, improved safety and reduced the energy consumption and operational costs.

Lavalle [11] developed a path planning algorithm called Rapid-Exploring Random Trees (RRT), which constructs a path between two given points to avoid obstacles. Based on this method combined with GP occupancy mapping, Yang et. al [12] proposed a path planner method to explore unknown and cluttered areas and maximize information about the environment. A collision-free path is generated from an initial state to goal state. The GP occupancy map is then used with RRT planning since it works as a collision detection module that implicitly represents free space. The RRT expands the search as part of the exploration mission to obtain information on the unknown cluttered environment.

This paper addresses the development of a method based on Bayesian optimization that unifies planning under uncertainty by combining perception, environment representation, and route planning into a common framework. As opposed

[1]Kelen C. T. Vivaldini is with the Department of Computer Science, Federal University of São Carlos (UFSCar), SP, Brazil, vivaldini@dc.ufscar.br

[2] Matheus D. C. Oliveira, Thiago H. Martinelli and Denis F. Wolf are with the Mobile Robotics Lab, Institute of Mathematics and Computer Science (ICMC), University of São Paulo (USP), SP, Brazil, denis@icmc.usp.br.

[3]Vitor Guizilini and Fabio Ramos are with the School of Information Technologies, University of Sydney, Australia, vitor.guizilini@gmail.com; fabio.ramos@sydney.edu.au.

to previous strategies that use way-point greedy solutions for acquiring new observations, we reason in the space of continuous sampling paths taking into account predictions propagated over time. We obtain a probabilistic model that represents complex objective functions the robot attempts to maximize as part of its mission. The predictive mean and variance of the model were used for the exploration-exploitation trade-off in a principled manner following a Bayesian optimization procedure. The methodology was applied for the monitoring and identification of pathogens in plantations and tested in the monitoring of *Ceratocystis fimbriata* disease in real eucalyptus plantations.

II. METHODOLOGY

The framework developed consists of 4 modules (Fig. 1), namely Coordinates Systems and Transformations, Classification, Map interpolation and Route Planning. The Coordinates Systems and Transformations module localizes the UAV's images in relation to the world; the Logistic Regression model separates disease affected trees from healthy trees and other structures; the Gaussian Process receives the coordinates and values of classified points for the interpolation and creation of a continuous map of the area; finally, the route planning algorithm is executed to minimize the distance traveled and ensure a good coverage.

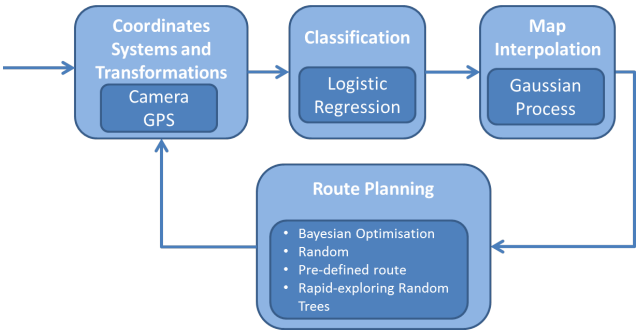


Fig. 1: Proposed methodology

The method used in each module can be freely changed, therefore, the same framework can be employed under different circumstances. We tested different methods for route planning to validate the efficiency of the proposed framework in maximizing classification performance. Tests were carried out in the Morse Simulator representing a realistic scenario of eucalyptus plantations, prior to the testing of the algorithms in real UAVs. The proposed framework can be applied to both scenarios (simulated and real environments), and requires only the adjustment of camera parameters, coordinate systems and battery.

A. Coordinates Systems and Transformations

Images are captured by a UAV with known GPS coordinates and orientation. From the transformation relationships among the adopted coordinate systems in the UAV, camera, GPS and the world, each image is spatially transformed into an equivalent normal view through the creation of a new image and mapping of each pixel's location in relation to the GPS coordinates. The information of the GPS is based on the geodetic coordinate system and the rotation matrix of the UAV follows Euler's XYZ convention.

B. Logistic Regression

1) Overview: The Logistic Regression model [13] is a linear classifier that inserts the linear regression model in a logistic function and produces probability values from $0 \leq p(X) \leq 1$. It is formulated as

$$p(Y|X) = \frac{1}{1 + \exp\left(-Y(w^T X + b)\right)}, \tag{1}$$

where X is the data input vector and Y is the class label. $w = (w_1, w_2, ..., w_m)$ and b is the weight vector and bias constant of the separator hyperplane, respectively.

2) Contextual Block Classification: We used the Contextual Block Classification methodology [5] to detect and classify eucalyptus crops affected by Ceratocystis wilt disease. A squared sliding window (block) of 4x4 pixels runs on the image and extracts the following features: mean and variance of each individual color channel, gray-scale image, and Local Binary Patterns (LBP) and entropy from the image converted to CIELab color space. Each block is surrounded by a larger contextual block of 16x16 pixels (Fig 2). Visual features are extracted from each block and its contextual block and concatenated to form the feature vectors. A manually classified map (ground-truth) assigns: 0 for diseased trees e 1 for healthy trees/other structures. Finally, the Logistic Regression classification method is trained to create a model. This configuration achieved the best results in terms of F-measure in the test set.

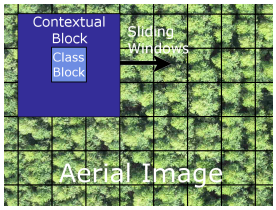


Fig. 2: Contextual block methodology, adapted from [5]

C. Gaussian Process Regression

A Gaussian Process (GP) [14] is a non-parametric Bayesian regression technique, as it does not rely on an explicit model to perform inference. Instead, it stores a set of samples of the underlying function $f(.)$ to be modeled and uses this information to estimate its value in unobserved areas of the input space. Here, this set of samples is given by a training dataset $\mathcal{D} = (X, \mathbf{y}) = (\mathbf{x}_i, y_i)_{i=1}^n$, composed of n inputs $\mathbf{x}_i$ containing latitude-longitude coordinates and their corresponding probability of disease y_i, given by the Logistic Regression algorithm. The relationship between the underlying function and those probabilities is given by

$$y(\mathbf{x}_i) = f(\mathbf{x}_i) + \epsilon_i, \tag{2}$$

where ϵ is an independent noise component usually assumed to be a zero-mean Gaussian distribution with constant

variance σ_n^2, i.e. $\epsilon \sim \mathcal{N}(0, \sigma_n^2)$. In addition to a noise function, a mean and covariance functions are also used to encode our prior knowledge of the underlying function. Mean function $m(\mathbf{x}; \boldsymbol{\theta}_m)$ represents the average distribution value at each point of the input space, while covariance function $k(\mathbf{x}_i, \mathbf{x}_j; \boldsymbol{\theta}_k)$ quantifies the correlation between any two points of the input space. The literature reports a multitude of well-established covariance functions [15] that use different metrics to establish this correlation and are better suited for different modeling scenarios. Parameters $\boldsymbol{\theta} = (\boldsymbol{\theta}_m, \boldsymbol{\theta}_k, \sigma_n)$ for the functions are commonly referred to as *hyperparameters* in Bayesian statistics and obtained through the maximization of the log-marginal likelihood function

$$\log p(\mathbf{y}|X) = \log \mathcal{N}(\mathbf{y}|m(X; \boldsymbol{\theta}_m), K_{nn} + \sigma_n^2 I) \quad (3)$$

This equation balances between data fit and model complexity encoding the Occam's Razor [16] principle to avoid over-fitting. Once the optimal hyperparameters have been determined, the predictive distribution at a test point $\mathbf{x}_*$ is a Gaussian distribution with mean μ_* and variance $\mathcal{V}_*$ given by

$$\mu_* = K_{*n} \left(K_{nn} + \sigma_n^2 I\right)^{-1} (\mathbf{y} - m(\mathbf{x}_*)) \quad (4)$$

$$\mathcal{V}_* = K_{**} \ - K_{*n} \left(K_{nn} + \sigma_n^2 I\right)^{-1} K_{n*}, \quad (5)$$

where K_{nn} is the $n \times n$ covariance matrix with entries $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j, \boldsymbol{\theta}_k)$, K_{*n} is the covariance matrix between test and training points and K_{**} is a diagonal covariance vector between test points.

D. Bayesian Optimization

1) Overview: Bayesian Optimization (BO) is an iterative procedure that finds the maximum of a *noisy* function $f(.)$ that is either unknown, or too costly to be evaluated [17]. The method uses the Bayes theorem to combine prior knowledge with observations for the production of a new estimation of f while attempting to find its maximum. In each iteration, new samples are selected by an incomplete model composed of previously acquired data. The model is then incrementally updated by the addition of such samples.

Here, the prior is a GP model with a mean component that captures both the estimated value of $f(.)$ at each position of the input space, and the respective variance (uncertainty of its corresponding estimate). Each observation is a noisy sample collected from $f(.)$. The method uses an *acquisition function* for effectivelt sampling $f(.)$. The function enables this iterative process to converge to a local maximum of the underlying function. The choice of the acquisition function is crucial for the BO performance [7] [9], since it determines the most relevant samples to be acquired in each iteration.

2) BO for Path Planning: Our focus is on the determination of an UAV path that minimizes the distance traveled and ensures a proper coverage of the surveyed area. The GP model contains classification results from the Logistic Regression algorithm as mean values, which range from 0 (diseased trees) to 1 (healthy trees / other structures).

Algorithm 1 Continuous Path Bayesian Optimization

Require: $f, h, \mathcal{C}$
  for $i = \{1, 2, 3, \cdots,$ iterations$\}$ **do**
    Find $\beta^* = \arg\max_\beta r(\mathcal{C}(u, \beta)|h)$
    $\{\mathbf{x}, y\}_\mathcal{C} \leftarrow \mathcal{C}(u, \beta^*)|_{u=0}^{1}$ %
    $\mathcal{GP} \leftarrow \{\mathbf{x}, y\}_\mathcal{C}$
  end for

Intermediary values indicate ambiguous areas, with 0.5 representing no knowledge about the classification at a particular point. The fundamental property of the acquisition function, therefore, should be the minimization of this ambiguity (variance) for the construction of a map of confidently classified regions. Such behavior can be codified by the following acquisition function

$$h(\mathbf{x}) = -\sigma_v^2 * \exp\left(-\frac{1}{2}\left(\frac{\mu(\mathbf{x}) - 0.5}{\sigma_l}\right)^2\right). \quad (6)$$

The above equation places a Gaussian distribution centered in 0.5, with amplitude and standard deviation determined by σ_v and σ_l respectively. Such parameters are selected empirically, according to the type of behavior desired for that particular application; i.e, lower values of σ_l tend to prioritize areas closer to 0.5, while σ_v is an overall regularizer (in all experiments, we used $\sigma_v^2 = 100$ and σ_l^2 = 0.02). The negative sign flips the Gaussian distribution, since this framework performs minimization during the optimization process.

Furthermore, whereas the standard BO derivation is discrete (i.e. concerned only with the end goal of each iteration), here we are interested in the extension to a continuous domain [8], taht also takes into account the trajectory between the goal. This is particularly useful for the application at hand, since the vehicle can collect images during navigation with minimal effort. Score s for each trajectory $\mathcal{C}$ is obtained through the integration of the acquisition function over its length

$$s(\mathcal{C}(u, \boldsymbol{\beta}|h)) = \int_{\mathcal{C}(u,\boldsymbol{\beta})} h(u)du, \quad (7)$$

where $\boldsymbol{\beta}$ are the parameters that define the trajectory and $u = [0, 1]$. A sampling strategy or a rectangle-rule quadrature approximation [18] can be used if Eq. 7 provides no analytic solution. Once the optimal trajectory $\boldsymbol{\beta}^*$ has been determined, samples $\{\mathbf{x}, y\}_\mathcal{C}$ are collected along the way (i.e. at fixed-length intervals) and incorporated into the GP model. The process is then repeated and a new optimal path based on this updated dataset (as depicted in Algorithm 1) is generated for the maximization of a generic reward $r(\mathcal{C}(u, \beta)|h)$.

E. Rapidly-exploring Random Trees for Path Planning

This section concisely describes Rapidly-exploring Random Tree (RRT) used for comparisons in the experiments. RRT is a data structure and algorithm designed for the searching of nonconvex high-dimensional spaces through the random construction of a space-filling tree. RRTs are constructed incrementally from a random search space. The

Algorithm 2 Rapidly-exploring Random Trees

BUILD_RRT $(q_{init}, K, \Delta q)$;
for $i = \{1, 2, 3, \cdots, \mathrm{K}\}$ **do**
  $q_{rand} \leftarrow RAND\_CONF()$;
  $q_{near} \leftarrow NEAREST\_VERTEX()$;
  $q_{new} \leftarrow NEW\_CONF(q_{qnew}, \Delta q)$;
  $G.add\_vertex(q_{new})$;
  $G.add\_edge(q_{near}, q_{new})$;
  $Return\, G$;
end for

tree is inherently biased to grow towards large unsearched areas of the problem. RRTs are indicated for path planning problems with obstacles and nonholonomic or kinodynamic differential constraints. This technique is suitable for the generation of open-loop trajectories for nonlinear systems with state constraints. An RRT can be intuitively considered a Monte Carlo way of biasing search into the largest Voronoi regions. Some variations can be considered stochastic fractals [11]. An RRT whose root is in configuration q_{init} and has K vertices is constructed by Algorithm 2.

III. Simulations

MORSE (Modular Open Robots Simulation Engine) has been widely adopted for the testing and evaluation of robot software in complex missions [19]. Using Blender to simulate photo-realistic 3D worlds and the associated physics engine, it brings enough realism for the evaluation of complete sets of software components within a wide range of application contexts [20]. For environment configuration, we used real images of eucalyptus plantations captured by UAVs. Each image has an area of 25715 m^2 and represents different scenarios (Fig.3).

(a) Scenario 1 (b) Scenario 2 (c) Scenario 3

Fig. 3: Examples of scenarios used in this study

The simulated UAV can travel in the area at any altitude and obtain the necessary information from the sensors and camera, as in a real environment (Fig.4). Each image captured within the larger scenario has an area of 1681 m^2.

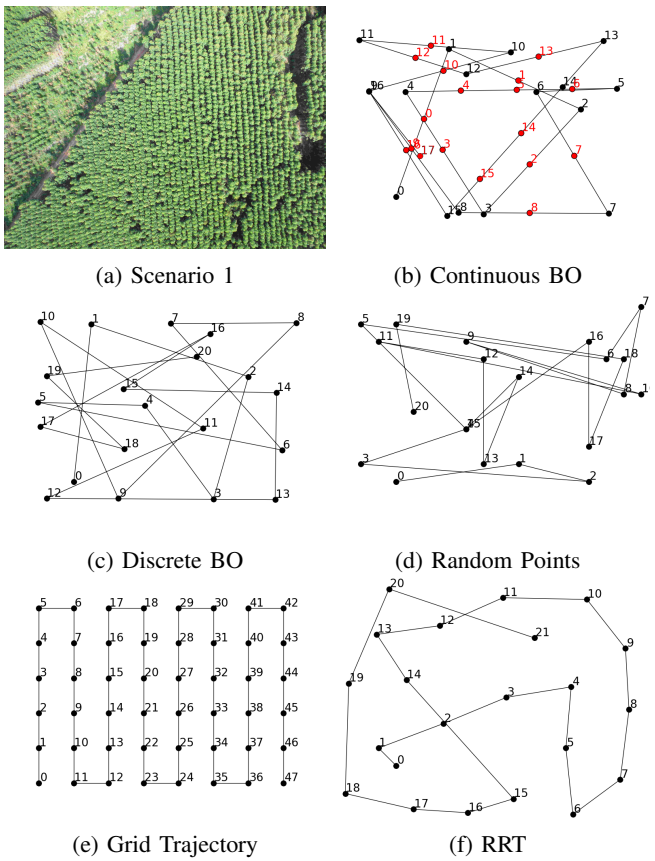
IV. Experimental Results

We used 10 different scenarios in the simulated environment to evaluate the results and efficiency of the Continuous BO. In each scenario, the following route planning methods were implemented (with and without a heuristic function): Continuous BO (Fig 5b), Discrete BO (Fig 5c), Random Points (Fig 5d), Grid Trajectory (Fig 5e) and RRT (Fig 5f). All methods search for destination points starting from the origin of the UAV up to a distance of 650 meters. Each node represents an image captured by the UAV.

(a) Simulated quadrotor (b) Camera view

Fig. 4: The UAV captures high-resolution aerial photos in the simulated environment.

The Continuous BO considers the uncertainty of intermediate nodes (i.e red nodes in Fig 5b) from one point to another for a better selection of the destination node. The Discrete BO considers only the uncertainty of the destination node. The Random Points method raffles random destination points without a heuristic function. The Grid Trajectory algorithm has a predefined route to cover the entire area. Finally, the RRT follows a method similar to the Continuous BO, and initianlley defines a destination point and then a route, considering the uncertainty along the path.



(a) Scenario 1 (b) Continuous BO

(c) Discrete BO (d) Random Points

(e) Grid Trajectory (f) RRT

Fig. 5: Path created by different algorithms for Scenario 1

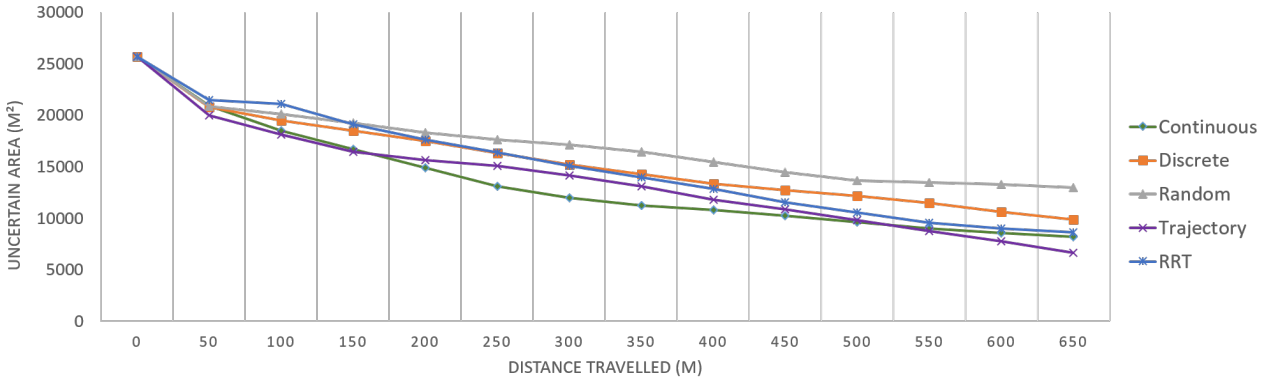


Fig. 6: Total amount of areas with values ranging from $0.3 \leq p(X) \leq 0.7$ (uncertain), regarding distance traveled

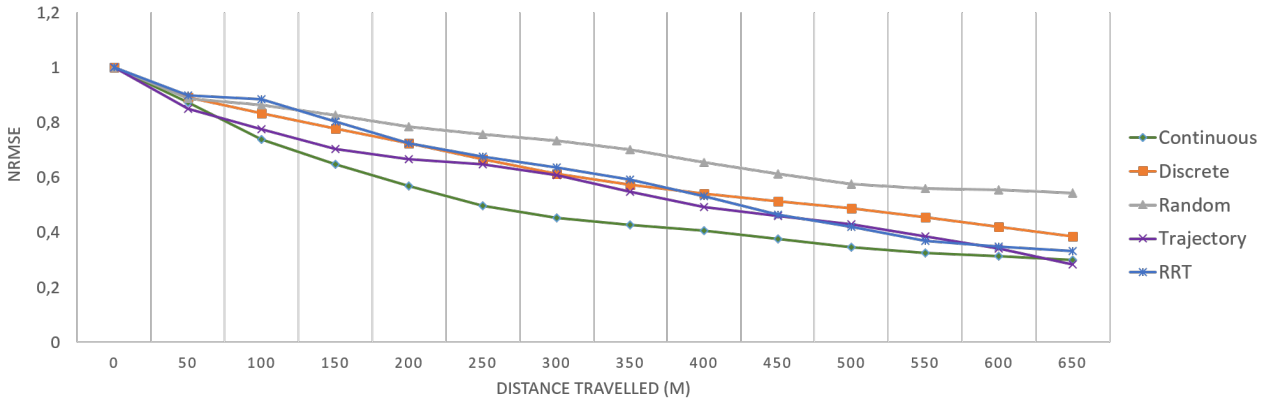
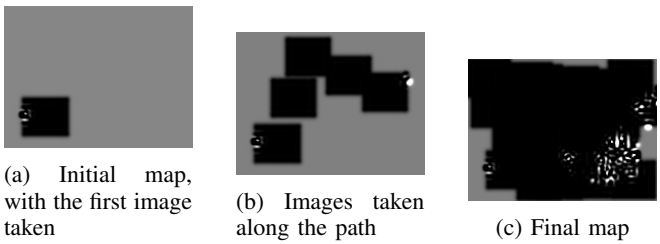


Fig. 7: NRMSE values regarding distance traveled. The predicted map approximates the reference map

A. *Uncertainty Decrease Over Distance*

For each scenario, the GP map was initialized with values of 0.5, which indicate a totally unknown environment (Fig 3a). As the UAV travels through the map, the values are updated by the active classification results. We considered the classification values of 0 to 0.3 as diseased trees (white), $0.3 \leq p(X) \leq 0.7$ uncertain areas (grey) and 0.7 to 1 healthy trees/other structures (black) (Fig 8). For a statistical analyses of the results, we averaged the values obtained in all 10 scenarios for each route planning method, regarding distance and uncertainty over the map area (generated by Logistic Regression model and GP interpolation).



(a) Initial map, with the first image taken

(b) Images taken along the path

(c) Final map

Fig. 8: Classification results. Images captured by the UAV are represented by white pixels (diseased trees) and black pixels (healthy trees, other structures)

Fig. 6 shows how the area considered uncertain decreases over distance. The Continuous BO started to decrease faster than other methods, at 200 meters, until it reached 450 meters. The Grid Trajectory showed the smallest number of uncertain areas, because its path ensures a total coverage of the scenario. The Continuous BO was in second position with fewer uncertain areas, followed by RRT. The results have validated the similar nature of the algorithms, as they both define destination points and a route considering uncertainty along the path. Discrete BO and Random Points performed the worst. Although the Discrete BO indicates the best points for the route, many images are overlapped and require a longer distance for traveling the entire map. The Random Points did not have a formal metric to choose the path, therefore it provided the worst result.

B. *NRMSE Decrease Over Distance*

The Normalized Root-Mean-Square Error (NRMSE) was used in the comparison between the predicted probabilities from the GP map and the manually classified image (or reference image). The metric was chosen because it quantifies how the predicted image approximates the reference image, ranging from 1 (completely different) to 0 (the same). Therefore, the probabilistic aspect of the estimates provided by LR and interpolated with GP was not discarded. In Fig. 7 the NRMSE values from Continuous BO, from 50 meters to 450 meters, decrease faster than other methods,

since its routes are more informative. The Grid Trajectory algorithm showed the lowest final error, as it provides non overlapping pictures of the entire environment. The NRMSE from Discrete BO was still higher than RRT, because it contained areas without classification at the end of the path. Random showed the highest error (0.54), as expected.

C. Comparison of NRMSE from LR and GP with Continuous BO

This test analyzed the average NRMSE values for the LR model in various resolutions and the corresponding values for the test with Continuous BO at 650 meters (Figure 9). For full (4608x3456 pixels) and half (2304x1728 pixels) resolutions, LR NRMSE was nearly zero and showed a good LR classification performance in comparison to the manually classified image. The error rate increases as the images become smaller, and provides the worst value of 0.94 at 144x108 pixels. For an image of same size on the GP map (185x139 pixels), the LR NRMSE is 0.88. Once the GP model is added to interpolate the points predicted by LR, even in such low resolutions, the error rates decrease significantly, from 0.88 to 0.33, which validate to the quality of the interpolation method used.

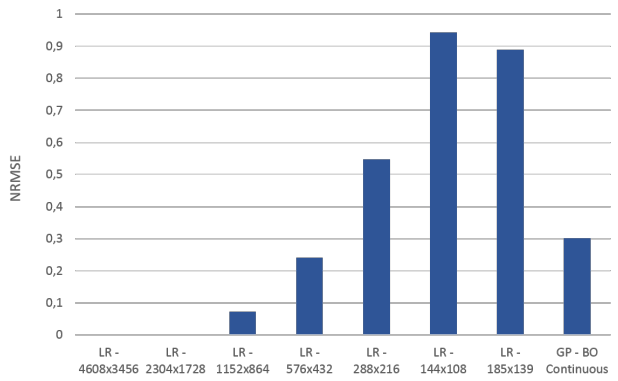


Fig. 9: NRMSE values from Logistic Regression classification of several map sizes without a GP model (columns 1 to 7) and after GP interpolation (column 8)

V. CONCLUSION

A methodology of route planning with active classification for UAVs has been proposed, to enhance the knowledge of visited areas and minimize the uncertainties on the classification of diseased trees. Five different route planning algorithms, namely Continuous BO, Discrete BO, Random Points, Grid Trajectory and RRT were evaluated. In all methods, the LR model provides the classified images and a Gaussian Process interpolates the information on the map. Unlike traditional methods of route planning, Continuous BO with active classification does not have all targets in the route pre-established, but instead generates new targets along the path. The main advantage of Continuous BO is the combination of route planning (building a route between source and goal points) and active classification (that enables the choice of goals based on how they affect environment uncertainty). It is suitable for scenarios where a UAV looks for a specific pattern, performing a search on an unknown environment (i. e. forest fires, national boundaries, illegal deforestation, etc).

ACKNOWLEDGMENT

The authors acknowledge CNPq and CAPES for the financial support to this research.

REFERENCES

[1] G. A. Longhitano, J. A. Quintanilha, "Rapid Acquisition of Environmental Information after Accidents with Hazardous Cargo through Remote Sensing by UAV", ISPRS Hannover Workshop 2013, Hannover. Int. Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. XL-1/W1, pp. 201-205.

[2] D. R. Negrao and T. A. F. S. Junior and J. R. S. Passos and C. A. Sansgolo and M. T. A. Minhoni and E. L. Furtado, Biodegradation of eucalyptus urograndis wood by fungi, Int. Biodeterioration and Biodegradation, 2014, vol. 89, pp. 95102.

[3] ABRAF, Associacao Brasileira de produtores de Florestas plantadas. Anuario Estatistico da ABRAF. Tech. Rep., 2013.

[4] J. R. JENSEN, "Remote sensing of the environment: an earth resource perspective". 2. ed. Pearson Prentice Hall, ISBN-10: 0131889508, 2007.

[5] J.R. Souza, C. C. T. Mendes, V. Guizilini, K.C.T. Vivaldini, A. Colturato, F. Ramos, D. F. Wolf, "Automatic detection of Ceratocystis wilt in Eucalyptus crops from aerial images," in Conf. 2015 - IEEE Int. Conf. on Robotics and Automation (ICRA), pp.3443-3448.

[6] Karakaya, M., "UAV route planning for maximum target coverage. Computer Science and Engineering: Int. J. (CSEIJ), Vol. 4, No. 1, February 2014

[7] R. Marchant , F. Ramos, "Bayesian Optimisation for Intelligent Environmental Monitoring," in Conf 2012 - IEEE Intelligent Robots and Systems (IROS), 2012.

[8] R. Marchant , F. Ramos, "Bayesian Optimisation for Informative Continuous Path Planning," in Conf IEEE/RSJ International Conf. Intelligent Robots and Systems (IROS), 2012 2014.

[9] E. Brochu, V. Cora, N. Freitas, "A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning, " 2010.

[10] J.R. Souza, R. Marchant, L. Ott, D. F. Wolf, F. Ramos, "Bayesian optimisation for active perception and smooth navigation," in Conf. 2014 - IEEE Int. Conf. on Robotics and Automation (ICRA), pp.4081-4087, 2014.

[11] M. Lavalle, S J. J. Kuffner, "Rapidly-exploring random trees: Progress and prospects," in Proc. Workshop on the Algorithmic Foundations of Robotics, 2000.

[12] K. Yang, S. K. Gan, A. Sukkarieh, "Gaussian process-based RRT planner for the exploration of an unknown and cluttered environment with an UAV," Advanced Robotics, vol. 27, no.6, 2013.

[13] E. Hyttinen,D. Kragic, R. Detry, "Learning the tactile signatures of prototypical object parts for robust part-based grasping of novel objects," in Conf. 2015 - IEEE Int. Conf. on Robotics and Automation (ICRA), pp.4927-4932.

[14] C. E. Rasmussen, "Evaluation of Gaussian processes and other methods for non-linear regression," Diss. University of Toronto, 1996.

[15] C. E. Rasmussen, K. I. Williams, "Gaussian Processes for Machine Learning", The MIT Press, 2006.

[16] C. E. Rasmussen, Z. Ghahramani, "Infinite mixtures of Gaussian process experts," Advances in neural information processing systems, vol. 2 (2002), pp- 881-888.

[17] S. Jasper, H. Larochelle, R. P. Adams, "Practical Bayesian optimization of machine learning algorithms," Advances in neural information processing systems, 2012.

[18] W. Gauschi, "Numerical Analysis," Springer, New York. 2012.

[19] G. Echeverria, N. Lassabe, A. Degroote, and S. Lemaignan. Modular openrobots simulation engine: MORSE. In Proceedings of the IEEE ICRA, 2011.

[20] G. Echeverria, S.Lemaignan, A. Degroote, S. Lacroix, M. Karg, "Simulating complex robotic scenarios with MORSE". 3rd Int. Conf. on Simulation, Modeling, and Programming for Autonomous Robots, Nov 2012, Tsukuba, Japan. pp.197-208, 2012.