Visual Servo Control in Quadrotor for Power Line Tracking

Luciano do V. Ribeiro*[1], Caique de S. F. Duarte[1], Alexandre C. B. Ramos[1], and Félix Mora-Camino[2]

[1]Federal University of Itajubá, Itajubá, MG, Brazil
[2]Federal University Fluminense, Niterói, RJ, Brazil

Abstract

In this work, Visual Servo Control is employed for the guidance of an Unmanned Aerial Vehicle quadrotor over the transmission power lines during an inspection. With this technique, it is possible to fly the UAV in a GPS-denied environment. A camera is attached in the UAV and with the processing of the images, the lines forming the transmission lines are extracted. These lines are used to feed a PID controller to correct the UAV pose. The image processing is made in a Raspberry Pi 2 embedded in the UAV and the software was made using the ROS middleware.

1 Introduction

Electric energy distribution companies carry out regular inspections and preventive maintenance to ensure the operation of the network. With the inspections, it is possible to analyze the health of the components that are part of the transmission line like the conductors, insulators, tower and the vegetation near the power line corridor. Traditionally the surveys are made by a manned helicopter containing sensors to data acquisition, as high-resolution cameras. Such data will be analyzed later for identification of anomalies. This type of work is very dangerous, costly and time consuming.

Different approaches to conducting inspections in transmission lines have been studied [1]. In recent years, interest in the use of VTOL (Vertical Takeoff and Landing) Unmanned Aerial Vehicles (UAVs) to automate the acquisition of images for transmission line inspection has grown [2]. Inspection with UAV platforms can be made remotely controlled by an experienced pilot, or autonomously, with the pre-programmed route in the form of waypoints in the Ground Control Station (GCS).

UAVs typically rely on GPS signal fused with the IMU (Inertial Measurement Unit) data to calculate your position and speed over time. However, its accuracy is not good enough to allow a flight near structures as is the case of line tracking. Another method for determining its position should be investigated. Computational visual and image processing techniques can be used to obtaining the position and attitude of a UAV.

*Email address: lucianoribeiro@unifei.edu.br

In this work, a low cost VTOL UAV platform was developed for inspections on transmission lines. The main objective of the project is to develop an autonomous navigation system to make the path between two consecutive towers. Visual information will be used to correct the orientation and position of the UAV along the path. The UAV will run over the transmission line, and a downward-facing camera will capture images. An embedded computer will identify the transmission lines and errors for the orientation and distance perpendicular to them will be determined. Finally, a PID controller will be implemented to correct the route, using image processing as feedback.

The remainder of this paper is organised as follows: In section 2 are presented work related to inspection of transmission lines with UAVs. In Section 3, the UAV platform developed and the problem description will be performed. The architecture of the image processing software and simulations are presented in Section 4. And finally, the conclusion in section 5.

2 Background and Related Work

The researches for inspection of transmission lines are usually divided into two groups. The first is focused on developing techniques for identifying lines in the image and another on developing control techniques given the line that must be followed. For the identification and tracking of transmission lines, some considerations are usually made:

- Transmission lines are linear objects viewed from above.
- They have a uniform brightness due to the material they are made of.
- The power lines are parallel to each other.
- They occupy most of the image.

Given these characteristics, most algorithms for detecting transmission lines are based on the Hough Transform and some other technique for grouping the detected lines, to discriminate lines that are not transmission lines. The great difficulty is in discriminating the transmission lines of other linear objects from the background.

Zhengrong Li et al. [3, 4] had developed a PCNN (Pulse-Coupled Neural Network) filter with the objective of removing background noise in the detection stage of the lines. Then Hough Transform is applied for straight line detection and K-means is used to cluster straight lines that have the same slope. Their implementation was tested on real data captured by aircraft but was not used for guidance.

Yang et al. also used the Hough Transform to detect lines in the binarized image [5]. As a clustering method, fuzzy C-means was implemented, having as features to determine a cluster the size and slope of the lines.

Mills e Aouf [6] had developed an Imaged Based Visual Servo Control application for orientation correction and displacement of a fixed-wing aircraft, to follow the transmission line in the presence of wind. The results were obtained in simulation.

Zhang et al. made an important work in the detection and tracking of transmission lines to be used in real time [7]. In addition to Hough Transform and K-means for straight line clustering, the Kalman filter is applied to track the straight lines in the Hough space. The processing time of each frame was 40ms, ideal to be realized in real time, but the hardware used was not described, and if it has the capacity to be embedded in a UAV

Araar e Nabil [8] had developed two solutions, IBVS e PPBVS. Their proposal was tested in a simulated and real environment. Not much focus was given on transmission line detection and tracking. It was not considered a complex background for the power lines. However, the way the detection was handled was used as a reference in our solution.

3 Materials and Methods

3.1 UAV Plataform

A low-cost 450mm frame quadrotor was developed for this project, as can be seen in Figure 2. The quadrotor is equipped with four 1000KV brushless motors and 8 × 4.5" propellers. The open-hardware Pixhawk autopilot board was used [9]. It is responsible for the low-level control of the quadrotor. It is equipped with accelerometers, gyroscopes, magnetometers and with a GPS + compass mounted externally. The firmware running on Pixhawk is the ArduPilot stack [10]. It is responsible for reading the sensors, the radio controller signals and independently controlling each motor. It also implements MAVLink, a protocol developed especially for UAVs that allows the communication with other devices.

A Raspberry Pi 2 was embedded in the quadrotor to enable autonomous flight. It has Ubuntu-Mate 16.04 and the middleware Robot Operating System (ROS) Kinetic installed [11]. A downward-looking Raspberry Pi camera was attached to the quadrotor and images with 640 × 360 at 30 fps are sent to the onboard computer. Through image processing, our software will calculate the correction velocities (angular and linear) and send them to the Pixhawk through the serial port using the MAVLink protocol.

A pilot with Radio Controller will also be able to take control of the quadrotor at any time. The quadrotor has a long-range radio 900Mhz that communicates with the GCS (Ground Control Station), which is possible to configure some parameters and get real-time telemetry data. The laptop running the GCS also has ROS, which makes possible to control and debug our software that runs on the embedded computer.

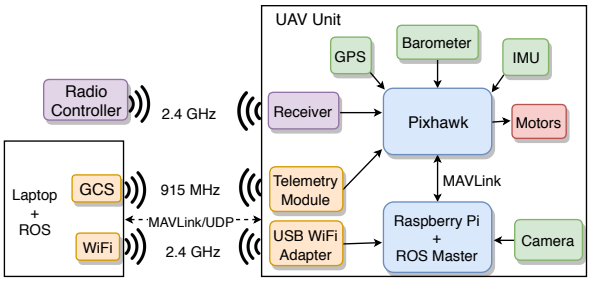

Figure 1: Hardware architecture, adapted from [12]

Figure 2: Developed UAV quadrotor

3.2 Mission Description

One of the main tasks in the inspection of transmission lines is the capture of images of the network components that are in the tower. In order to enable autonomous data capture, it is necessary for the UAV to be able to move between the towers for the execution of its mission. While transmission companies have data on the geographic coordinates of the towers, this data is for reference only and not its exact position. Além disso, o próprio GPS pode conter imprecisões e até mesmo ficar indisponível em algumas cituações. Therefore, making the flight considering only GPS data is not enough. In this project, visual information of the transmission

lines will be used as feedback to control the route of the UAV in the path between the towers.

For the beginning of the mission, the UAV will start above the initial tower and go through an initialization phase, to identify the transmission lines. The objective of the UAV will be to carry out a route that follows the lines, keeping its orientation parallel with the transmission line and minimizing the perpendicular distance between the UAV and the transmission line. Figure 3 illustrates this situation. On the left, we have the representation of the UAV and its field of view and the right, the orientation gap between the UAV heading and the center line, and the perpendicular distance between the UAV and the center line.

Our control strategy does not take into account the current position and orientation of the UAV, only its position regarding the transmission line, which can be identified by the image processing. However, as will be seen in Section 2, the output of our program are linear and angular velocity commands, which are sent to the flight controller via MAVLink. In order for Pixhawk to implement these commands, it needs to have UAV speed information, so we will use the GPS signal. Other techniques for obtaining the current speed can be used, such as the image processing itself. There is hardware that can be integrated with Pixhawk that performs this type of processing, and it could also be done on the embedded computer, but this is out of the scope of our project.

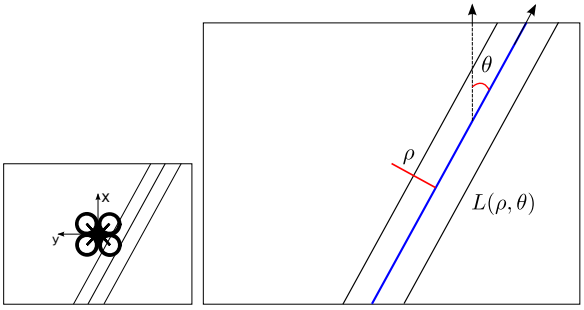

Figure 3: Representation of the image plane and the features extracted from the power lines

4 Results and Discussion

4.1 System Architeture

This project was developed using the ROS middleware and programmed in C++. With ROS it is possible to create a distributed network of processes, called nodes, that can communicate with each other through messages. The concept of publisher and subscriber is used. Each node informs which topics it wants to publish data to and which it wants to subscribe. Each topic has a name and an associated data type. There is a Master node that allows nodes to discover others on the network.

Our architecture consists of four nodes. A node is responsible for being the camera driver and sending images to the node that will handle the processing. The Image Processing Node will identify the transmission lines from the background and will get the center line.

This line will be sent to the main node responsible for the control of the position and heading of the UAV. In the main loop, the error of rotation and translation will be used as feedback for a PID controller. Speed commands will be sent to Pixhawk through mavros, a ROS node that implements the MAVLink protocol. mavros node is also responsible for converting between the coordinate system used by ROS and Pixhawk.

On the laptop running the GCS it is possible to have a ROS client. With it you can receive information about the nodes that are running on Raspberry Pi and also send some command, such as starting or stopping the execution of the mission.

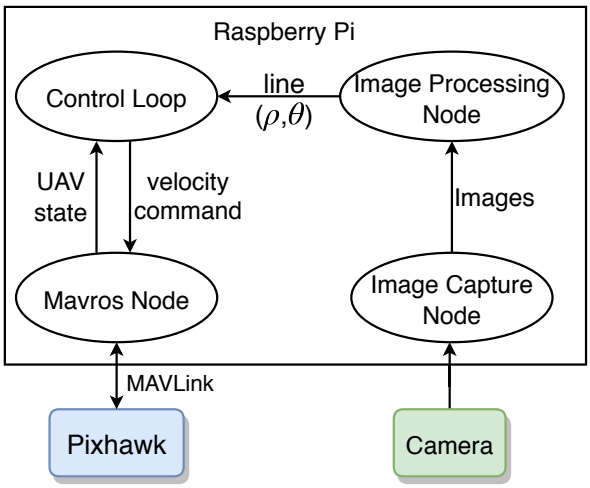

Figure 4: System architecture

4.2 Algorithm Implementation

4.2.1 Image Processing

OpenCV will be used for image processing. This is an open-source library with implementation of several algorithms for computer vision.

Each frame that arrives from the camera goes through a pre-processing. The image is converted to grayscale and then a media filter is applied for noise reduction. Then Canny filter is applied. It is responsible for identifying the edges in the image. This is a fundamental process because it reduces the amount of information present in the image and the processing time of the subsequent steps. The output of the Canny algorithm is a binary image, where the white pixels correspond to the edges of the image. Standard Hough Transform will be applied over this image to identify the straight lines.

With HT it is possible to identify any shape (line, circle, ellipse) that can be expressed analytically. In the case of straight lines, each point (x, y) in the image plane that belongs to a border will be mapped to a curve in the Hough space according to the Equation 1.

$$x\cos\theta + y\sin\theta = \rho. \tag{1}$$

For each point, all values of (ρ, θ) that satisfy the Equation 1 will have their frequency increased in Hough space. The parameters (ρ, θ) that has the highest amount of votes will be the corresponding to the line with more collinear points.

Just as it was done in [8], the HT output implemented by OpenCV needs to be modified to better serve our purpose. The parameter ρ, which represents the distance from the origin of the image (upper left corner) to the straight line, will become the distance from the center of the image to the line, using

$$\rho = \rho - \frac{W\cos\theta + H\sin\theta}{2} \tag{2}$$

where W e H represent the width and height of the image respectively. For the case of the center line, the value of ρ will be the error of the position of the quadrotor relative to the center of the image. The parameter θ will be in range $[0, \pi]$ and the desired setpoint for the UAV is to have θ fixed on zero. As there will be oscillations, around the point 0, we will have discontinuity and this will lead to an unstable control. The solution is to translate the value of θ to the range $[-\frac{\pi}{2}, \frac{\pi}{2}]$ according to

$$If\ \ (\theta \geq \frac{\pi}{2}) \quad Then: \begin{cases} \theta = \theta - \pi \\ \rho = -\rho \end{cases} \tag{3}$$

After extracting all the lines that are present in the image, it is necessary to identify those that are part of a transmission line. Our implementation takes into account the existence of a predetermined number of lines to be identified, which in our case is 3. Since the image does not change much between two consecutive frames, the line that has the most votes in HT and is close to the chosen line in the last frame will be chosen for the current frame.

$$\begin{cases} |\rho - \rho_{last}| < \rho_{max} \\ \text{and} \\ |\theta - \theta_{last}| < \theta_{max} \end{cases} \tag{4}$$

.

The other two lines are chosen in such a way that they are parallel to the first, $|\theta_i - \theta_1| < \theta_{max}$, and that does not refer to the same edge, $|\rho_i - \rho_1| > \rho_{min}$.

4.2.2 PID controller

Our algorithm will control the movement of the UAV only in the *2D* plane, not changing its height. For this two PID controllers will be implemented. One to correct the guideline of the quadrotor and another to correct its deviation perpendicular to the line. The mathematical expression of a PID controller is given by Equation 5

$$u(t) = K_p e(t) + K_i \int_0^t e(t)dt + K_d \frac{de(t)}{dt} \tag{5}$$

where a command $u(t)$ will be proportional to the error $e(t)$, the sum of this error over time and the rate of change of this error in the instant t. As can be seen in Figure 3, the error in the direction and position of the UAV in relation to the center line are the values θ, ρ, respectively. As presented in [8], the error in ρ tends to increase as the UAV moves in the direction that should decrease. This is due to the fact that the frame has to tilt to perform the position correction. This tilt moves the line in the plane of the image in the opposite direction, giving the impression that the distance has increased. We will use a gimbal to stabilize the field of view of the camera parallel to the ground in these situations.

4.3 Simulation

Simulations to test the ability to identify straight lines and mainly the control algorithm was made in the simulator Gazebo [13]. With the Gazebo simulator, it is possible to create different environments, simulate sensors and actuators. Within the developer community, you can find numerous sensors and actuator types to use on robots.

A simple simulation environment has been developed, as can be seen in Figure 5.

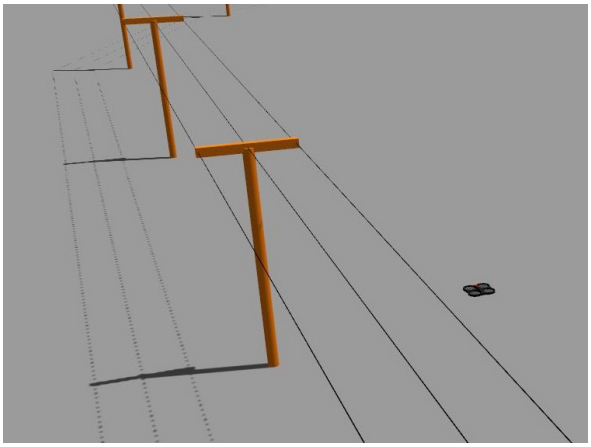

Figure 5: Simulated environment in Gazebo

In the Figure 6 you can see two steps of image processing performed on the same frame. At the top we have the image obtained after the application of the Canny filter and below the identification of the main line with the HT.

5 Conclusion

This paper has presented a Visual Servoing Control algorithm for power line inspection using a low cost quadrotor

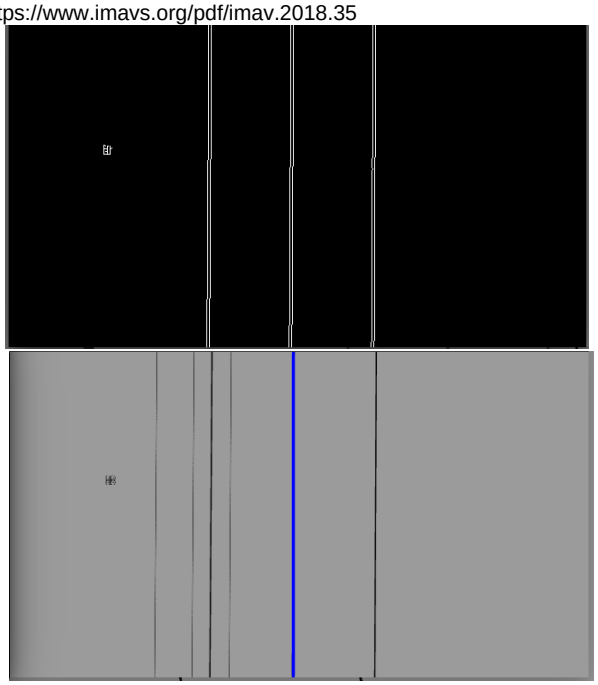

Figure 6: Edge map and Hough Transform applied in the same frame

UAV. The proposed method allows to fly with a quadrotor over the transmission line and to use visual feedback through the image processing to correct the trajectory. The project is being implemented using the ROS middleware and will be embedded on a Raspberry Pi 2.

Currently, testing is being done on the Gazebo robot simulator. The implementation is still simple and further improvements are needed until practical tests are performed.

In addition to making the simulation environment more realistic, we need to make the line detection more robust by using information from previous frames to reduce processing time and increase the accuracy of this step.

ACKNOWLEDGEMENTS

The authors would like to thank the funding institution: CAPES.

REFERENCES

[1] Leena Matikainen, Matti Lehtomäki, Eero Ahokas, Juha Hyyppä, Mika Karjalainen, Anttoni Jaakkola, Antero Kukko, and Tero Heinonen. Remote sensing methods for power line corridor surveys. *ISPRS Journal of Photogrammetry and Remote Sensing*, 119:10–31, 2016.

[2] A Pagnano, M Höpf, and R Teti. A roadmap for automated power line inspection. Maintenance and repair. *Procedia CIRP*, 12:234–239, 2013.

[3] Zhengrong Li, Yuee Liu, Ross Hayward, Jinglan Zhang, and Jinhai Cai. Knowledge-based power line detection for UAV surveillance and inspection systems. In *2008 23rd International Conference Image and Vision Computing New Zealand, IVCNZ*, 2008.

[4] Zhengrong Li, Yuee Liu, Rodney Walker, Ross Hayward, and Jinglan Zhang. Towards automatic power line detection for a UAV surveillance system using pulse coupled neural filter and an improved Hough transform. *Machine Vision and Applications*, 2010.

[5] Tang Wen Yang, Hang Yin, Qiu Qi Ruan, Jian Da Han, Jun Tong Qi, Qing Yong, Zi Tong Wang, and Zeng Qi Sun. Overhead power line detection from UAV video images. *International Conference on Mechatronics and Machine Vision in Practice*, pages 28–30, 2012.

[6] Steven Mills, Nabil Aouf, and Luis Mejias. Image based visual servo control for fixed wing UAVs tracking linear infrastructure in wind. In *Proceedings - IEEE International Conference on Robotics and Automation*, 2013.

[7] Jingjing Zhang, Liang Liu, Binhai Wang, Xiguang Chen, Qian Wang, and Tianru Zheng. High speed automatic power line detection and tracking for a UAV-based inspection. *Proceedings of the 2012 International Conference on Industrial Control and Electronics Engineering, ICICEE 2012*, pages 266–269, 2012.

[8] Oualid Araar and Nabil Aouf. Visual servoing of a quadrotor UAV for autonomous power lines inspection. *22nd Mediterranean Conference on Control and Automation*, pages 1418–1424, 2014.

[9] Pixhawk homepage: https://pixhawk.org/. Acessed: August, 2018.

[10] Ardupilot homepage: http://ardupilot.org/. Acessed: August, 2018.

[11] ROS homepage: http://www.ros.org/. Acessed: August, 2018.

[12] Rafael G. Braga, Roberto C. da Silva, Alexandre C.B. Ramos, and Felix Mora-Camino. Collision avoidance based on reynolds rules: A case study using quadrotors. *Advances in Intelligent Systems and Computing*, 558:773–780, 2018.

[13] Gazebo homepage: http://gazebosim.org/. Acessed: August, 2018.