Detection and positioning of pipes and columns with autonomous multicopter drones

Edmundo Guerra[1], Rodrigo Munguía[2], Yolanda Bolea[1] and Antoni Grau[1]*

[1] Department of Automatic Control, Technical University of Catalonia UPC, Barcelona 08034, Spain; edmundo.guerra@upc.edu
[2] Department of Computer Science, CUCEI, University of Guadalajara, Guadalajara 44430, Mexico; rodrigo.munguia@upc.edu
* Correspondence: antoni.grau@upc.edu

Submitted: 23/March/2018

Abstract—A multimodal sensory array to accurately position aerial multicopter drones with respect to pipes has been studied, and a solution exploiting both LiDAR and vision sensors has been proposed. Several challenges, including detection of pipes and other cylindrical elements in sensor space, validation of the elements detected, and has been studied. A probabilistic parametric method has been applied to segment and position cylinders with LIDAR, while several vision based techniques have been tested to find the contours of the pipe, combined with conic estimation cylinder pose recovery. Multiple solutions have been studied and analyzed, evaluating their results. This allowed proposing an approach that combines both LiDAR and vision to produce robust and accurate pipe detection. The combined solution is validated with real experimental data.

Keywords: drone; cylinder positioning; LiDAR; vision-based sensing

1. Introduction

Automation of tasks, driven by technical development, has been gaining weight in industrial and civilian operations as part of complex system for a long time. The introduction of robotics (and other earlier advanced technologies) has frequently found strong opposition; but no matter the field, there has always been several factors pushing it: not only the increased efficiency, but also the better and safer conditions for human employees. In many areas, the introduction of these technologies has been delayed because there is still need for human capabilities hard to reproduce. One of these capabilities, intrinsic to human beings, is the generality and adaptability of human response, making them especially suited for supervisory and monitoring tasks.

Monitoring and maintenance tasks rely heavily in availability of the information, which can be obtained from remote/installation sensors, but many times require actual physical inspection of some elements. This is especially true for industry, where it would be impossible to sensorize all the elements/points which must be inspected or monitored at some time as part of the maintenance operations. An example of such elements would be pipes and canalizations, especially in heavy industries, where kilometers of pipes have to be periodically inspected.

These pipes and tubes are common structures not only in industry but also in urban environments, and can be frequently found in hard to reach areas. This poses a problem for the mentioned monitoring and maintenance operations, as those operations are commonly performed by human personnel or ground based unmanned vehicles (UGV) with great efficiency, become expensive and exceptionally risky in inaccessible areas. In such scenarios, operations with humans in high and/or hard to reach areas generally imply shutting off ordinary operation, building temporary scaffolds, and following complex safety protocols and procedures to minimize risks. In these situations, any opportunity to reduce the participation or risks taken by human personnel can have a great impact, both economically and in safety terms.

In this context UAV (Unmanned Aerial Vehicles) based solutions have started to appear. While UAV drones have been present for a long time, developments in MEMS (microelectromechanical systems) and battery technologies have produced an explosive growth of the field. This has made

them cheaper and easier to deploy, with a big research and development community supporting them, especially for the massively popular rotary-wing multicopters. This kind of UAV has already a strong presence in the audiovisual production and the surveying industry, and is gaining a foothold in other industries.

On the other side, the problem of locating pipes from robotic platforms is not new. There are many works that try to locate defects in pipes from the inside [1,2] which use robots that try to build a map of the pipe while they navigate inside the pipe. Those systems are based mainly in odometry and inertial measurement units to build the path and the map to locate themselves [3]. This is not the problem that author encounter in this research. In [4], authors present an on-board UAV visual system which tries to avoid collisions of such flying robots. The range visual detection varies depending on the detected elements but the accuracy in objects detection is not enough to locate the UAV with precision enough to obtain a good pose of the robot. In [5] another obstacle detection-equipped UAV is presented, the robot has visual systems, laser, barometer and ultra-sound on-board and, although they use a PTAM scheme [6] to locate, the excess in sensors makes that the accuracy in fusion gives large errors in location, at low heights the barometer is unreliable due to turbulences and at heights above 5 m the ultrasonic distance sensor drops out.

The paper presented describes the research and development performed to produce a solution to the problem of positioning an UAV with respect to pipes/columns/cylindrical elements found in the environment, using LiDAR, vision, or other sensors available to deploy in an UAV. The proposed architecture combines two different previously studied techniques to produce accurate and robust results. After a brief description of the general architecture used by the UAVs considered, presented in Section 2, the respective discussions of the studies performed with LiDAR and vision sensors are discussed in Section 3 and 4. In section 3, authors present the design of two different LIDAR-processing architectures: one design built initially to allow joining multiple LIDAR scans to produce denser data inputs for a RANSAC technique to be used; and a second lightweight approach, focused in performance. In section 4 a known state-of-the-art pose recovery method is implemented and studied, detailing the image processing used to detect the apparent contour of the pipes. After finding the strengths and weaknesses of the different approaches studied in Section 3 and 4, Section 5 describes the combined proposed approach, which uses the robustness of the LiDAR detection and segmentation, and the superior accuracy of the vision based positioning method. Experimental results are provided to benchmark the different techniques studied and to validate the proposed approach with real data captured with a hand-held sensor body emulating the configuration to be found in the UAV.

2. UAV architecture and properties

One of the most challenging aspects of robotics in the context of UAV is the weight constraint: the equipment deployable on board is twice limited by weight: the weight of the equipment itself, and the weight of the batteries required to power the device. This translates into very limited computational power deployable on board, even introducing additional SBC (single board computer). The chances of delegating computational efforts to other systems are also constrained by the range, bandwidth and latency of wireless communications; so the general assumption is to deploy anything needed at real-time performance on-board.

This affects the architecture of the robotic UAV, not only in hardware terms, but also from a high level architecture point of view. Thus, the common approach of deploying a single computing unit in the form of the FMU (Flight Management Unit) is ignored in favour of deploying and additional SBC. This additional computing unit will be responsible of all the hardware and processes not needed in the low level control loops to guarantee UAV stability and safety. The FMU will receive data from those sensors which require low computational power to process it (GPS, inertial and height sensors, etc…) and control the low level operation of the UAV. This way, the heaviest computational task, such as image processing, localization in maps, video streaming and communications, etc. are delegated to the SBC.

Figure 1 shows the architecture of the UAV drones considered in this work. Though the architecture was initially developed using Odroid SBC's (based on ARM processors, roughly equal to a high-end smartphone), the kind of computational power required by the proposed approaches required upgrading the hardware to a Intel NUC (Next Unit of Computing) device (same performance as a mid-to-high end laptop).

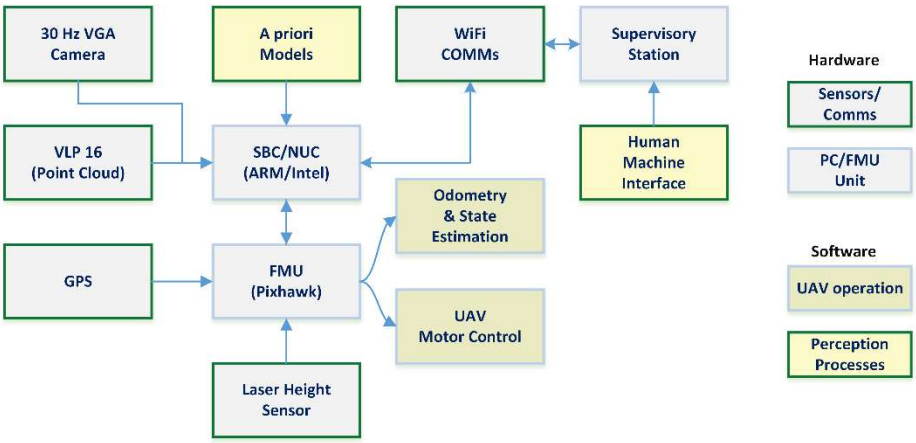


Figure 1. Architecture diagram with main sensors and processing hardware, noting some relevant processes for this work.

Under this architecture the FMU is still responsible of the odometry estimation, so the research and experiments have to account for the error characterization in these measurements.

3. LiDAR-based detection and segmentation of cylinders

Detection and positioning of pipes using LiDAR or similar range finder sensors is essentially a problem of shape detection in point clouds. There are many approaches to this problem, but they are generally based on five wide categories: edge based, region based, attribute based, graph based o model based methods. Each category shares a wide set of features, according to its procedures and strategies.

Edge based methods try to find the edges of a region of similar points, generally through identification of those points presenting a rapid divergence of the metric with respect to the neighbors. Some methods are based in gradient techniques [7], while other detect different edges and group them, producing scan lines representing surfaces [8]. Approaches like the latter one are suitable for only-range sensors, but produce weak results when the point-cloud density is uneven. On the other hand, region-based methods use local neighborhood information to build regions of points with similar features, and isolate regions according to the dissimilarity, thus growing regions instead of delimiting them as the edge-based methods. Though they have been reported to provide better results than edge-based methods, they have low accuracy determining the limits of the regions, and can require accurately seeds to start growing regions [9].

Methods based on attributes, like [10], work as a two-step process: in the first step an attribute or set of attributes is computed; and in the second step the data points are classified (commonly through clustering) according to the attribute. Though they are resilient and the clustering can be used to introduce clues, they are largely dependent on the attribute was chosen and its selection is not a trivial problem.

Graph-based methods read the whole point-cloud as a graph, with the simplest case matching each point to a node. They can produce very good results, as they can benefit from many techniques commonly applied to graph-based problems, like Markov Random Fields [11], k-nearest neighbor (kNN) [12], or conditional random fields (CRF) [13], to cite a few examples. The size of the cloud-point to be processed generally proves a weakness, as dense or semi-dense clouds are generally impossible to be processed in real-time with graph-based algorithms.

Model based approaches are mostly based on the Random Sample Consensus (RANSAC), technique [14]. The procedure is based on fitting geometric primitive models, and group points according to their proximity to the models. The RANSAC approach itself has been widely studied [15,16] in fitting problems, and given an adequate model and initial seed produces accurate results robustly.

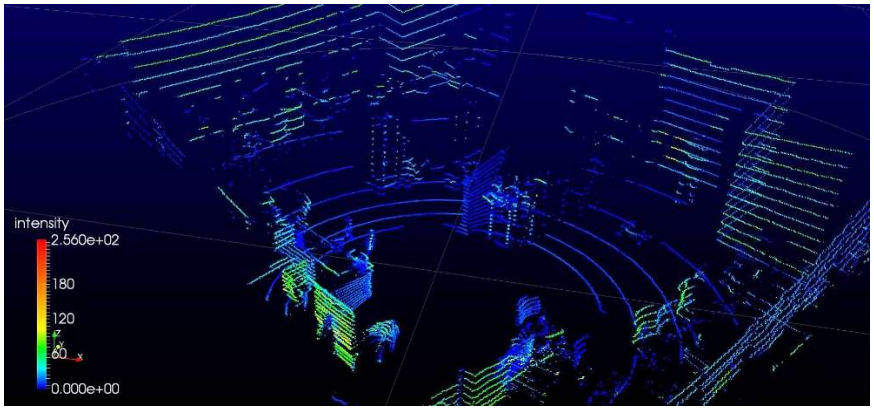
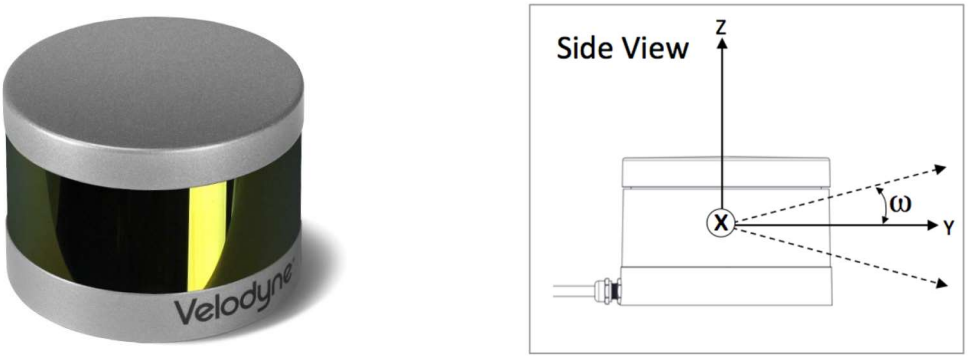


Figure 2. Characteristic sampling unevenness presented by most of the rotation-based LRF's scanner, like the VLP-16 used in this work.

Note that most of the popular laser range finder (LRF) sensors present a characteristic unevenness in sampling density and distribution, as they generally work by performing single or multiple parallel scans by rotating the range-finder element. As consequence of this operation the samples are quantized at some tens of coordinates along a limited subregion of the dimension/axis orthonormal to the scan plane, while the scan plane or half-planes are usually fully sampled, as seen in Figure 2. For example, the sensor used for this study, the Velodyne VLP-16, presents 16 scan lines distributed between +15º to -15º in azimuth, with 360º coverage each [17] (see Figure 3). This feature can impact the segmentation and positioning problem, especially in terms of accuracy depending on the relative orientation between the sensors and the objects, as it will be discussed.



(**a**) (**b**)

Figure 3. Sensor Velodyne VLP16: (**a**) appearance of the sensor; (**b**) sensor coordinate frame, with origin and scan witdh detail.

For our problem, in order to detect a pipe generalized as a straight homogeneous circular cylinder (SHCC) robustly, in a real-time scenario with the limited computational power deployable in an UAV, to determine their pose, a RANSAC-based segmentation approach was chosen, using a state of the art implementation [9]. Thus, assuming that there is a cylindrical pipe, which can be described as a SHCC, C, and a coordinate frame centered in the sensor L, with ${}^{\tilde{o}}_{L}T$ denoting the homogenous transformation from a world origin $\tilde{o}$ to this LiDAR frame, the RANSAC process tries to fit a SHCC model into the point cloud. This point cloud is referenced with respect to L, as the seven parameters of the model to be fit, namely: the coordinates of a support point for the axis of the SHCC

in frame L, $[x_p, y_p, z_p]$, a vector denoting the direction of said axis $[x_v, y_v, z_v]$, and the estimated radius r_c. A seed for the radius parameter can be provided, in the form of a range $[r_{min}, r_{max}]$, with the RANSAC procedure trying to force that r_c satisfies said range.

Proof of concept tests showed that the application of a raw RANSAC procedure to the data obtained from the LiDAR sensor was vulnerable to the unequal distribution of samples along the different dimensions of the sensor frame L, producing either false positives if the seed range for r_c was set with wide margins; or failing to find a cylinder C with parameters fitting the SHCC model. Because of this weakness, two different architectures were studied: a first one, designed to produce denser point clouds, exploiting the assumption that odometry measurements of the movements or the multicopter would be available, so an approximation to ${}^{\tilde{o}}_{L}T$ is available; and a second one, which would present a much better performance.

For the first architecture, the procedure starts with a scan joining step, where two or more of the point clouds scans produced are combined to produce an assembled point cloud. This operation is performed exploiting the capabilities to store and operate several buffers of time-stamped transformations and frames provided by ROS [18,19]. Note that this procedure is entirely reliant in the accuracy of the transformation ${}^{\tilde{o}}_{L}T$ and the sensing capabilities of the multicopter to optimize its performance. This is because as an ICP-derived [20] procedure, the scan joining process uses the transformation between the point clouds at different time instants as a seed. The main risk to this approach is correlated with the size of the assembled point cloud, as it grows linearly with the number of scans fused. If the assembled point cloud is larger than the size limit which can be robustly solved in real-time it again produces inaccurate model fittings or spurious detections. To avoid this, the point clouds are preprocessed to reduce the number of points considered into the RANSAC approach, treating them with a geometrical pass filter, a voxelization step, and a statistical filtering phase, assuming that for a point being part of a relevant surface it must lay in certain areas, and be near other points. Once the cloud has been filtered, the RANSAC procedure determines the model of a homogeneous circular cylinder described by an axis (a line with a support Euclidean point and direction vector) and a radius, by fitting the parametric model based on the neighbor surface normal of the data points.

This approach was tested indoors, with a false positive detection rate below 0.7%, and very accurate SHCC model parameter estimation, but presented two main weaknesses: firstly, the segmentation operation operated at an average rate of 0.73 Hz; and secondly, the indoor testbed used to simulate the odometry (estimated through motion capture with *Optitrack*®) produced an estimation with an accuracy beyond what it can be really expected during actual flight operations with on-board sensors. Introducing white noise into the odometry estimated with the motion capture system to simulate the actual accuracy that can be expected from real-time inertio-visual odometry approaches [21] produced a decrease in performance, with an average detection rate of 0.64 Hz. Still, the results obtained from testing this early architecture allowed to experimentally determine through human calibration the parameters to configure the RANSAC procedure and gave insight in how to produce a light-weight, faster approach.

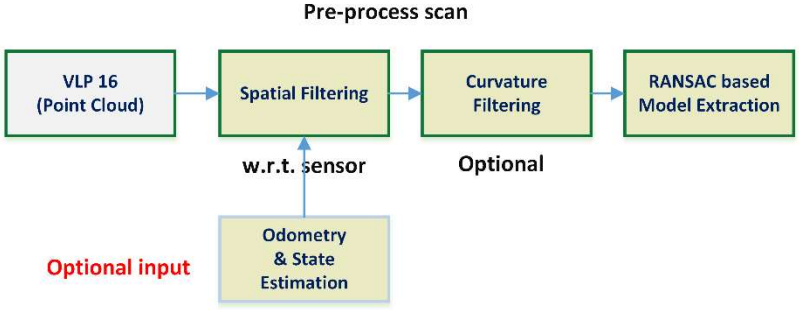


Figure 4. Optimized architecture for the RANSAC-based segmentation and positioning of SHCC in point clouds.

The second light-weight architecture (see Figure 4) presented several differences over the first tested: the cloud point joining process is removed, just like the statistical filter and the voxelization; and a new curvature-based filter was introduced. The design of the lightweight architecture allows a better adjustment of the RANSAC parameters, and to obtain a knowledge acquired testing the initial architecture. This meant that the new architecture was able to detect the desired SHCC with single point clouds, avoiding the scan joining step, as seen in Figure 4. This in turn removed the dependence on accurate odometry, with spatial filtering being generally done w.r.t. the sensor frame to remove the "shadow" of the UAV/rigid solid where sensor is attached. The statistical filter was removed as it was observed that it presented no relevant impact into the accuracy of the RANSAC procedure, neither to avoid fake positive nor improving accuracy. The voxelization process, though it had proved useful for dealing with dynamically sized cloud points, with the single point cloud approach it proved too expensive, as it is essentially a full resampling of the whole data.

Removing all these steps from the LIDAR pipeline allowed freeing computation time, thus a curvature filter was introduced. This filter is also a significant computational burden, but allows to work with fully unknown radius, removing the need for an initial r_c. This makes the prior knowledge completely optional, though it greatly reduces the chances of false positives, and can be used to disable the curvature filter increasing performance.

4.Vision-based detection and pose recovery of a cylindrical pipe

One of the main physical characteristics of pipes and tubes, in terms of vision-based perception and image, is the apparent contour, i.e., the edges presented: even when they present similar hue and texture as the background, the geometry of a pipe, as a SHCC, is noticeable (see Figure 7.b). Another important characteristic that can be usually detected and tracked is the material texture. Nevertheless, this saliency in terms of texture with respect to the rest of the environment may prove unreliable, as its detection can be largely affected by shadows, dynamic lightning, and other visual artifacts. These issues can be dealt with through computer vision techniques, but generally imply computationally expensive procedures, unsuitable for UAV deployment.

4.1. Pose recovery

Several vision-based approaches have tried to solve the pose estimation problem for cylinders from monocular images. In [22], several methods to estimate linear and quadratic primitives through analytics procedures are presented, focusing in the perspective inversion approach. In [23], a multistep process localizes each of the cylinder axis using a priori knowledge about the projection of the cross-sections, as described in [24], and use them to localize the cylindrical surface in the camera coordinate frame. More recently, in [25], the metric reconstruction of surfaces of revolution (SOR) was addressed combining the apparent contour and captures of cross-sections. Some of the geometrical properties and formulations described in [25] were also used in [26]. Later works, like ,have proposed solutions based in non-linear Levenberg-Marquardt optimization, though they tend to rely in multiple views and iterative solutions.

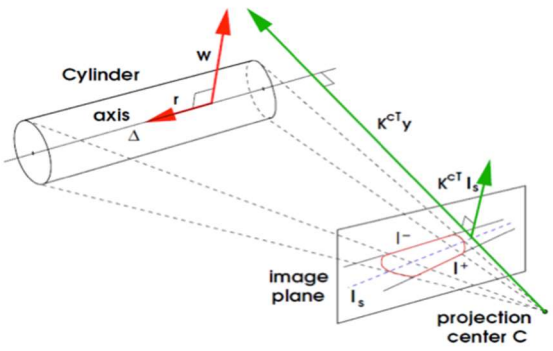


Figure 5. Projection of the apparent contours of a SHCC modelling a pipe in the image plane, with the camera projection center and the pose coordinates (courtesy of Doignon et al. [26]).

In [26], Doignon *et al*. present a pose recovery method for SHCC from the apparent contour in a single image. The apparent contour is assumed to be known as a pair of segments S_1 and S_2, with each one denoted as two points in homogeneous coordinates, S_{ia} and S_{ib}, for segments $i = [1\ 2]$. A closed-form solution to determine the pose between the axis of the SHCC and the camera scaled by the radius in Plücker coordinates [27] is given. This is achieved by formulating a matrix representing the degenerate quadratic defining the cylinder, which can be annotated as Plücker coordinates of the symmetry axis (see Figure 5). This formulation can be used in a conic-based pose fitting method, which can determine the pose exploiting the relations between the perspective projection and the pose parameters.

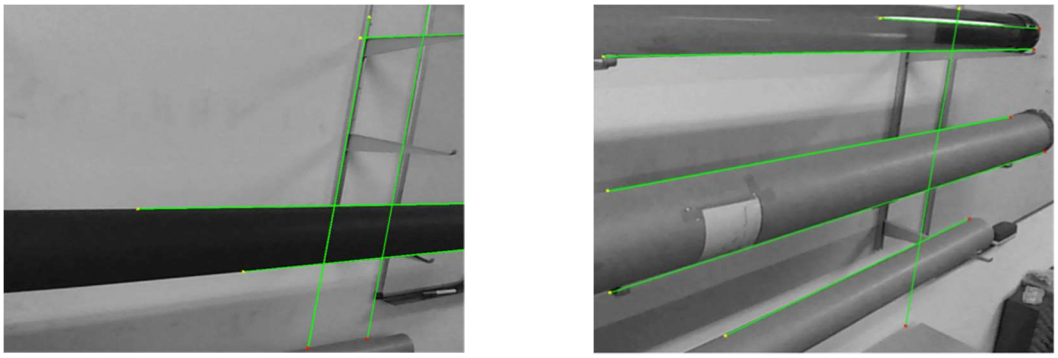
This solution was implemented to visually determine the pose of the pipe, as the closed-form solution described meant that the procedure could achieve real-time performance, as only a singular value decomposition operation was required to solve the optimization part of the method. Tests with synthetic datasets for apparent contours showed results consistent with those described in the original work. Indoor experiments were also successful, producing average relative error below 3.5% for depth estimation. Still, when the camera optical axis and the pipe axis become close to parallel, which constitutes a degenerate configuration, the method becomes inconsistent.

4.2 Apparent contour extraction

Several approaches were developed in order to extract the apparent contour of a pipe. A simplistic solution based in the Hough transform [28] was initially developed, where all the straight lines in a region of interest are detected and studied. During the initial indoor testing the probabilistic Hough transform based on Canny edge detector [29] with Otsu 's threshold [30] proved enough to achieve consistent binarization and edge detection (note that Canny is still widely known as *optimal detector* [31]), as seen in Figure 6. Note that in an uncontrolled environment, be it natural or artificial, there may appear multiple segments and pairs of them which may appear to be an apparent contour for a pipe, and they must be discriminated.

In order to initially find the apparent contour candidates, they were filtered to reject those shorter than a given threshold, and grouped by pairs according to the similarities in orientation and *closeness*. This closeness was defined as the number of approximately parallel segments between them, i.e., two edges of a same pipe or column should present a low number of other parallel lines between them. This step presents very challenging problems and scenarios, as seen in Figure 6: in subfigure (**a)** two different pair of lines could be interpreted as pipe contours, and it would require segmentation and/or scene interpretation techniques to solve the ambiguity; while in subfigure (**b)** both reflections and shadows modify the apparent contours of the pipes.

Thus, a priori knowledge was used to choose the apparent contour candidate to use in the method described earlier to recover the pose. This knowledge was introduced as geometric/model restrictions (i.e. approximately known orientation or position of the pipe), or through a human machine interface (HMI). Notice that using HMI knowledge to obtain priors required using accurate odometry transform the prior knowledge to the relevant coordinate frame of the camera. To add consistency to the method, once an apparent contour has been found and validated, a visual servoing tracking method [32] searches for it in successive frames, and only when there are inconsistencies the full detection is performed.



(**a**) (**b**)

Figure 6. Two samples of Hough transform pipe detection. Random elements can be easily present straight edges in structured environments: (**a**) two pair of lines detected which can produce two equally strong apparent contour candidates; (**b**) reflections and shadows may produce spurious apparent contours.

This implementation, including pose recovery, produced robust results in indoor environments in terms of detection, but presented poor performance around 8.64 Hz, while still being affected by multiple challenging issues in terms of computer vision (see Figure 6). A small battery of outdoor tests further revealed some critical weaknesses. Firstly, the global binarization process was not able to properly detect edges under natural uncontrolled lighting, especially when multiple/ambient light produce diffused shadows; the implicit assumption of presenting features similar to a bimodal image taken in the indoor case to use Otsu's thresholding was not useful in an uncontrolled environment. Additionally, the indoor structured environments presented easier to identify contours, usually presenting stronger edges with approximately known size and structure; thus able to be detected and identified with our assumed model. Finally, in the outdoor operation, the frame to frame contour tracking was unable to track the contour consistently, requiring to reintroduce prior knowledge in the case of the HMI.

A modified approach substituted the global binarization with two different local adaptive binarization approaches [32], but the performance achieved was too low to be useful, with 2.34 Hz on average at 640x480 pixels. In the end, the full binarization with Canny edge detection was removed in favor of introducing a line segment detector (LSD [33]). This final architecture, seen in Figure 7, improved the performance of the approach, working at an average 21.4 Hz, but still presented an unreliable contour detection step, as it is discussed further in the results section.

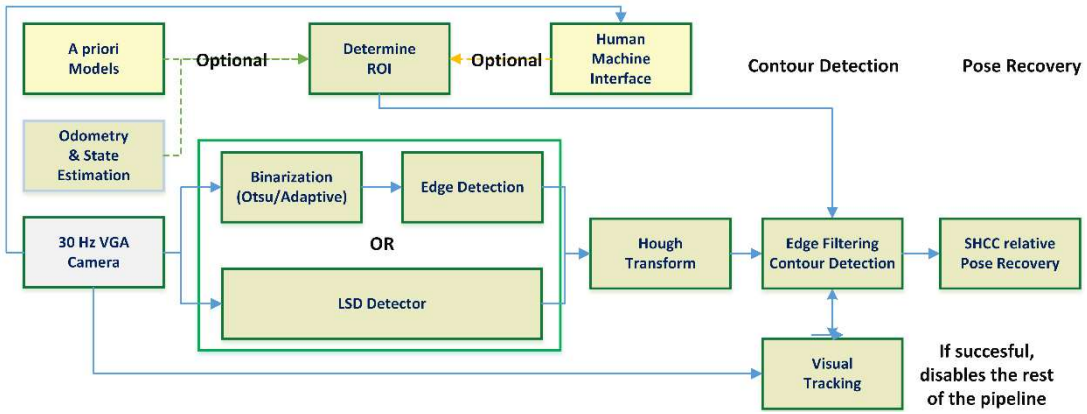


Figure 7. Final vision-based architecture for detection and pose recovery of pipes.

Notice that the final architecture proposed, in Figure 7, still uses prior knowledge, obtained through prior models and odometry, or with human interaction with the HMI. This allows determining the region of interest (ROI) to search for the apparent contour to improve the robustness

of the technique, while reducing the number of false positive detections. Another improvement of this architecture is the introduction of visual-tracking for the lines composing the apparent contour; so, once they are properly detected, if the frame-to-frame tracking is successful all the steps to detect and determine the apparent countour will be skipped in successive frames.

5. Integrated LiDAR segmentation and vision-based pose recovery

Earlier sections have discussed work developed with each of the available sensors in order to solve the problem of detection and pose recovery of a pipe with known radius. Of the studied approaches, using LiDAR and vision respectively, each one presented its own weaknesses and strengths. Our study showed that each of the approaches was stronger at one of the steps and noticeable weaker at the other task: LiDAR registration procedure achieved great robustness at the detection and segmentation task, while the vision based pose recovery presented great accuracy at higher rate, but with very weak detection results. These results led to the development of a combined approach to exploit the best features provided by each sensing technology.

The integrated method solves the problem in two different steps, working at different speeds with different sensors. Firstly, a RANSAC-based segmentation step, as described earlier, uses the point cloud data provided by the VLP 16 LiDAR to fit the SHCC model into the environment surrounding the UAV. This process works at an average 4.3 Hz, with an accuracy presenting dependencies w.r.t. the material and texture of the pipe to be detected, and specially to the relative position between the pipe axis and the sensors, as it will be discussed in section 6.

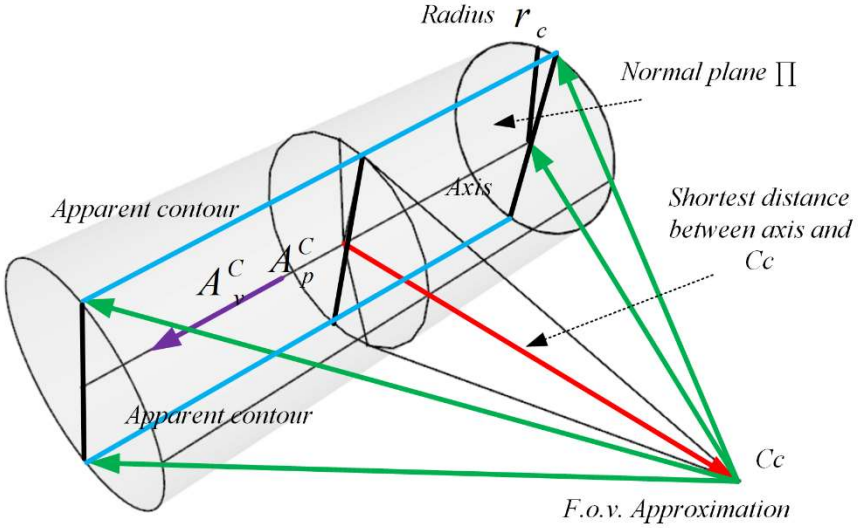


Figure 8. Detail of the projection into frame *C* operation, how the shortest segment (in red) between the axis crossing through A^C_p and the camera optical center C_c is used to build a normal plane to support the parallel lines used as apparent contour approximation (in light blue), and intersect with the predicted field of view of the camera (projected in green).

Once an estimation of the pipe axis pose is available as a point $A^L_p = [x_p, y_p, z_p]$ and a $A^L_v = [x_v, y_v, z_v]$ vector in the LiDAR frame, *L*, these are converted into the world coordinates using the transformation ${}^L_{\tilde{o}}T_k$, computed at the instant the laser scan was acquired, *k*. Once in the world frame *O*, the model of the pipe axis can be transformed into the camera vision frame *C*, using transformation ${}^{\tilde{o}}_{C}T_k$ or ${}^{\tilde{o}}_{C}T_{k+t}$, depending if it is assumed that the motion performed by the UAV during the time to process the LiDAR cloud point data is negligible, or it is relevant and possible to capture it with the odometry estimation available. Note that instead of ${}^{\tilde{o}}_{C}T_k$, it is possible to work directly with the transformation between the LiDAR sensor frame *L* and the camera vision frame *C*, ${}^L_C T_k$, if movement during the time interval *t* will not be considered in any case.

With the pipe axis translated to the relevant camera frame C, described through point ${}^{C}A_{p}$ and vector ${}^{C}A_{v}$, the shortest segment between the camera optical center pose ([0,0,0] in frame C) and the pipe axis is determined (see Figure 8). A plane $\prod$, normal to said vector is computed, and two lines lying on this plane, parallel to the pipe axis denoted by ${}^{C}A_{p}$ and ${}^{C}A_{v}$, at distance r_c are computed and considered as predicted apparent contour.

The predicted apparent contour is projected into the camera plane using the projection matrix of the calibrated camera sensor [34]. This allows determining a tightly bounded ROI to search for line segments in the image, and using strict criteria to accept or reject segments to use as image apparent contour.

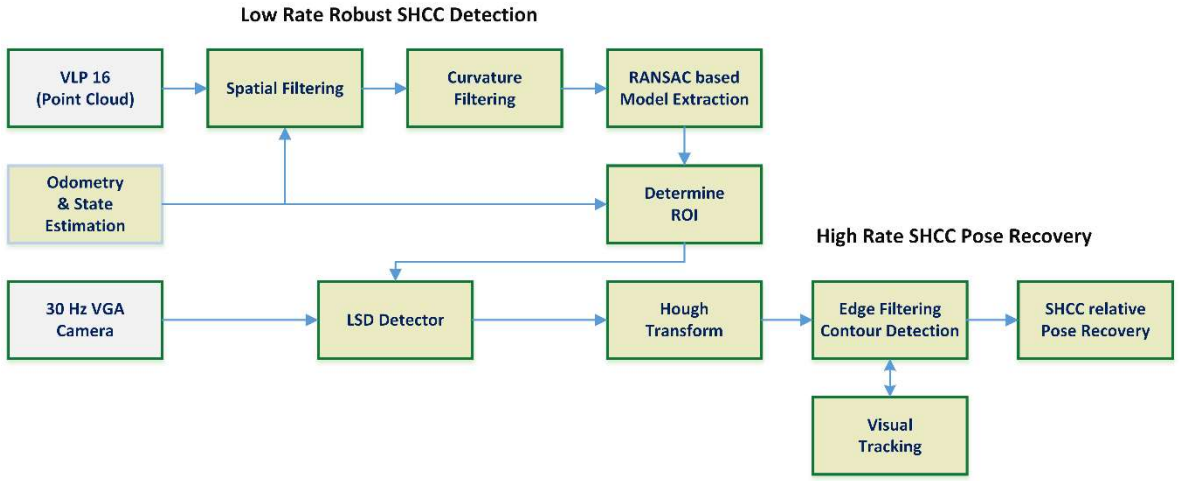
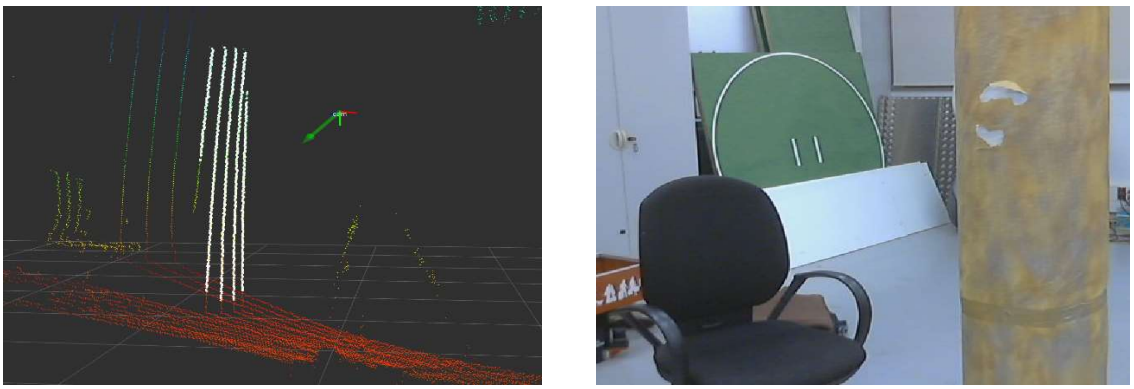


Figure 9. Integrated architecture combining the developed LiDAR and vision based pipelines.

Figure 9 shows the architecture diagram for the combined approach. The first row shows the LiDAR-based segmentation pipeline, starting with the point cloud data obtained from the VLP 16 sensor, and following the process shown in Figure 5, which provides robust detection of the pipe and an initial pose estimation. In the second row the step to convert the initial pose estimation produced by the LiDAR into a prior for the visual pose recovery is show. Note that in order to be able to use pose estimated by the RANSAC-based cylinder segmentation, an estimation of the state and odometry of the UAV/sensors rigid body is required, as the LiDAR segmentation and visual positioning pipelines work at different rates. Because of this, we cannot assume that the global position of the UAV/sensors rigid body will not vary and use the relative pose between the LiDAR detected cylinder and the UAV directly (as the frequency achieved is around 4.5 Hz the delay is around ~0.23 s), but we can assume that the odometry estimation provided by the FMU (as described in section 2, see Figure 1) will be locally accurate to transform the estimated line parameters into current camera coordinates. This data is then used in the *third row* of the architecture diagram, which details the visual pipe segmentation and pose recovery. Notice that although some measure of scene registration is still performed, the visual pipeline has been modified to use the data from the LiDAR detected pipe as a prior, so the processes and architecture described in section 4.2 are simplified and the apparent contour detection rate is greatly improved. These modifications remove the need for human feedback or accurate pipe priors, the only required that is the cylinder radius, with the pose recovery process remaining largely the same once the apparent contour is determined.

6. Experimental Validation

The proposed approach has been validated with real experimental data. Each of the different techniques and architectures was tested using the relevant sensors and ground truths. The experiments were performed over real data sequences captured (see Figure 10) through software provided by the ROS middleware.



(**a**) (**b**)

Figure 10. Samples of one of the indoor experimental sequences captured: (**a**) visualization of the VLP16 point cloud, with and RGB axis frame denoting the rigid sensor frame pose, and those point pertaining to the detected SHCC plotted in white; (**b**) view from the camera rigidly solidary to the sensor frame, the elements seen (pipe, chair) can be also observed in figure **a**.

The software developed was integrated into the ROS framework, and tested in a i7 laptop, at 2.5 GHz, running ROS Indigo over Ubuntu Trusty Tahr.

6.1. Experimental hardware setup

Two different hardware setups have been used to capture sequences tested with the developed techniques. Firstly, a multicopter drone platform, used as concept test, to check viability of flight with the increased weight and impact of vibrations and other disturbances introduced. An early image of the prototype target platform to deploy the developed software can be seen in Figure 11.a. A second hardware setup was developed in order to test and validate the different techniques developed without having to perform real flights, a standalone rigid frame was built to deploy the sensors, and operate them manually in indoor environments (see Figure 11.b). Working with the hand held sensor frame allowed us to easily study singular configuration and other cases of interest, and also permitted testing the approaches with data obtained inside and indoor motion capture system, providing a millimeter accuracy ground truth.

(**a**) (**b**)

Figure 11. Hardware Prototypes: (**a**) Early prototype UAV hexacopter deploying the sensor setup considered in this work; (**b**) hand-held sensor rigid body for experimentation in flight-denied areas (indoor laboratories, etc…).

In both setups, the UAV and the handheld frame, the Y axis of the VLP 16 was aligned parallel to the visual axis of the camera (commonly Z in camera frame according to literature). This meant that although there is no actual difference between X and Y axes in terms of LiDAR sensing capability, as during the capture the camera was pointed towards the pipe, the Y axis of the LiDAR became the depth from the sensor to the pipe, while the X axis mapped the pan or sidescrolling movements.

Thus, during the results discussion, those discussions referred to the Y axis of the LiDAR are actually related to the depth between pipe and sensor.

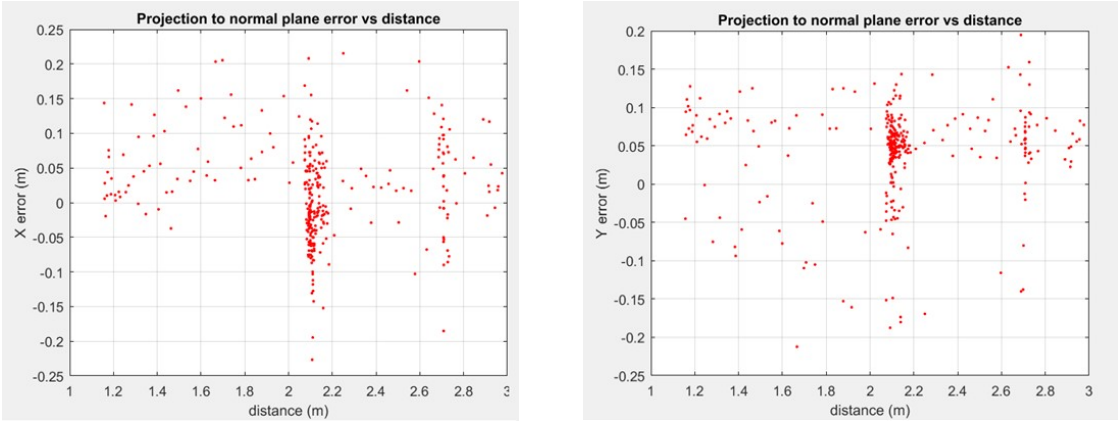
6.2. LiDAR detection and positioning results

To evaluate LiDAR segmentation robustness and accuracy several indoor tests were performed locating a vertical 0.5m diameter pipe; as these could be captured with an accurate ground truth. The first validation step was finding if the light-weight architecture without scan joining could achieve the same robustness, and how much better performance could be achieved. It was determined that the false positive rate was almost negligible for both (see Table 1), but at the same time, avoiding the scan joining step reduce greatly the computational effort. This is noticeable not only in the joining and pre-processing phases, but also in the RANSAC step, as the number of points introduced into the RANSAC method went down from an average of 19k to 8.5k, thus greatly alleviating the computational costs. The impact is evident in the average frame rates achieved by each method.

Table 1. RANSAC-based segmentation of a SHCC in point clouds, "joining scans vs. single scan"

Method	Initial size	RANSAC size	Avg. rate	False positives
Figure 3 Architecture	~60k points	19k avg. points	0.73 Hz	0.71%
Figure 4 Architecture	~30k points	8.5k avg.points	3.94 Hz	0.73%

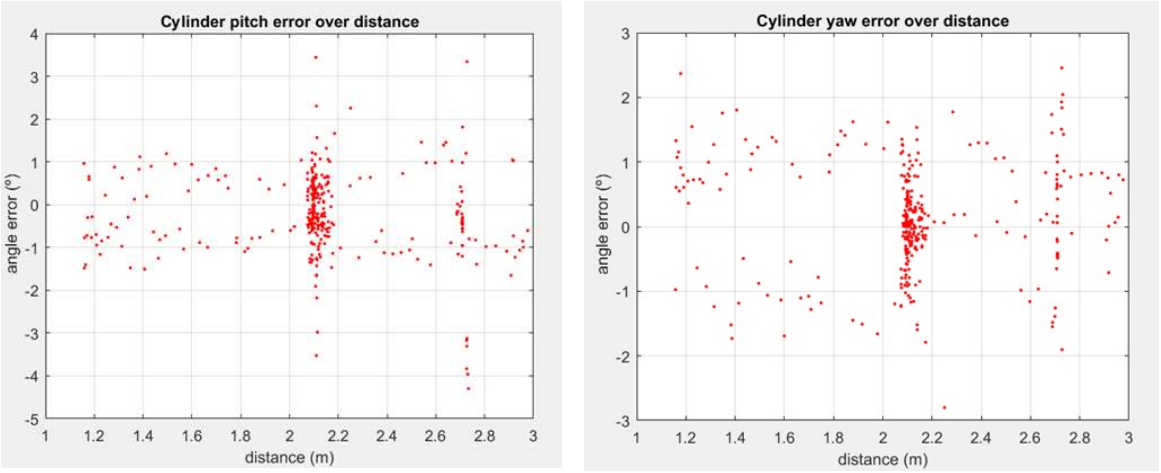
The impact of the distance and orientation between the pipe and the sensor was studied using the ground truth form the motion capture system. Figures 12 and 13 show the impact of distance in position and orientation estimation, respectively, for one of the experiments. In said experiment the rigid sensor frame was set a 3.5m distance from the pipe, then the distance was closed until ~1m, to later move away from it again. At around 2.10m the sensor frame was rotated in several axes, with multiple roll rotation around the line joining the LiDAR and the pipe axis. It is noticeable how in all the degrees of freedom the error is well bounded, and when studying the 2.10m point, as the most sampled distance, the error tends to follow a normal-like distribution (with a slight bias in the depth estimation, noted as Y axis with respect to plane XY plane of the LiDAR, per Figure 3.b).



(**a**) (**b**)

Figure 12. distance between LiDAR and SHCC vs position error in the plane XY of the LiDAR: (**a**) X position error in the XY plane of the LiDAR at the pipe axis intersection; (**b**) Y position error in the XY plane of the LiDAR at the pipe axis intersection.

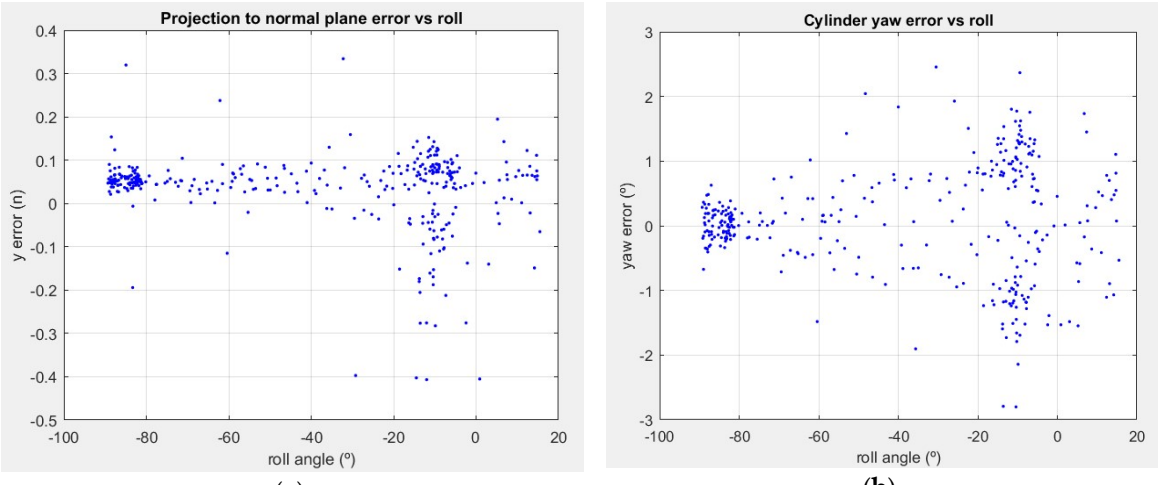
The study of the orientation error with respect to the distance shows (Figure 13) that it is well bounded around 1º for one axis, at Figure 13.a, with slightly more disperse results for the angle in the YZ plane (Figure 12.b). Notice that this angle is correlated with depth perception, and as such, it presents a slightly greater error, as it is noticeable in Figure 13.b.



(**a**) (**b**)

Figure 13. 2 D.o.F. orientation between estimated SHCC axis and actual vertical pipe, assuming as roll rotation around the pipe axis which extents along the Z axis of a coordinate frame: (**a**) pitch error, observed as the projected angle in a XZ plane containing the pipe axis; (**b**) yaw error, as a projected angle in a YZ plane containing the pipe axis.

The study of the sensibility of the SHCC estimation with respect to the orientation of the sensor showed a strong correlation between the roll along the Y axis of the sensor itself, and the *depth* related position and orientation components. The relevant results are shown in Figures 14.a and 14.b respectively. The low dispersion cluster with very low errors around 90º were produced, both for position and orientation in short distances, below the 2m marks, with the scan planes orthonormal to the floor and aligned to the pipe axis. The other big clusters are near horizontal orientations of the sensors, and present a much wider dispersion. This phenomenon was produced by the different detection rates, affected both by distance and orientation. As such, the approximately vertical orientation of the sensor, with scan lines almost parallel to the pipe, produces much more accurate results if the distance is close enough so that enough scan lines will hit the pipe, enabling detection of the SHCC through RANSAC. If the distance crosses the 2m mark, the accuracy drop slightly, but it is also prone to fail to find the SHCC in the point cloud.



(**a**) (**b**)

Figure 14. Depth positioning and orientation error vs sensor roll around LiDAR Y axis: (a) position error in pipe axis interception against the XY plane of the LiDAR, in the same XY plane; (b) yaw error, as a projected angle in a YZ plane containing the pipe axis.

6.3. Vision based countour detection and pose recovery.

The vision based approach was tested with the same indoor sequences, and some other outdoors sequences, which lacked ground truth or LiDAR information. The accuracy of the results obtained, in terms of pose recovery, was slightly worse than those reported by [26], with an average 4.2% relative errors, though this difference is probably produced by the different points of view studied, and other errors introduced by the methodology used.

In Table 2 statistics on the different approaches studied are displayed, showing the accuracy of the contour detection step, in terms of false positive rate (i.e. instances where no contour is present, or an incorrect contour is detected), and a general estimation of the performance of each method as the average rate achieved. Using HMI is excluded, as that method delegates contour detection to the human component. Only those based in approximate a priori knowledge about the pipe (i.e. the general orientation and initial distance), and odometry estimation are considered.

Table 2. Detection accuracy and performance of the visual detection of pipe apparent contours.

	Contour detection method	Indoor error rate[1]	Outdoor error rate[2]	Avg. rate
a	Canny + Hough Transform	23.45%	48.71%	8.64 Hz
b	**a** + Adaptive Binarization	17.29%	33.56%	2.34 Hz
c	LSD + Hough Transform	19.42%	29.38%	21.4 Hz
d	**c** + LiDAR based priors	1.32%	- [3]	19.78 Hz

[1,2] False positive rate (i.e. a contour not pertaining to the pipe or presenting wrong fitting is found).
[3] No outdoors data with LiDAR and image is available.

The purely edge attribute-based methods (a, b and c in Table 2) tested have been found unable to solve the general pipe contour detection problem in a fully satisfactory way, as seen in the high spurious detection rates. These results remove the pure vision-based approach to pipe contour detection onboard an UAV as an option, leading to the integrated LiDAR and vision method.

The method proposed integrating both LiDAR and vision (entry **d** in Table 2) presents the best detection rate, as the apparent contour is detected using as support the actual estimation of the pipe according to the LiDAR-based segmentation (which presented spurious detection rates below 1%). It is interesting how the performance of the vision based pipeline of the integrated method is slightly lower than that of the equivalent technique (entry **c** in Table 2) without LiDAR, though the most probably cause is the need added layer introduced by the data sharing and conversion between frames.

7. Conclusions

A methodology to accurately detect and recover the pose of a pipe (or any other cylindrical structural element) with respect to a robotic multicopter UAV has been developed. The proposed method combines LiDAR and vision to produce the best possible results in terms of robustness (as in ability to detect the pipe in complex environments and avoid false positives) and accuracy. This combined approach was the only solution which could solve the challenges without sacrificing either robustness or performance.

The initial studies tried to determine which of the available sensor devices, namely, monocular vision cameras or LiDAR, could provide a better solution to the detection and positioning challenges. These tests showed that none of the single-sensor solutions developed could provide an all-encompassing satisfactory solution. The LiDAR detection and positioning solutions were implemented based in RANSAC approaches, with two different developed architectures: one based in single LiDAR scan processing, and another one base in joining multiple LiDAR scans. The single scan architecture proved to be functionally as accurate as the approach with multiple scan joining, but presented a fivefold increase in performance measured as rate. This approach achieved very robust detection, with negligible false positives, but at a slow rate with average accuracy.

The visual pipelines developed were based in the pose recovery described in [26]. This required the detection of the apparent contour of the pipe, which proved to be a hard to solve challenge. Several edge-based methods were proposed and studied, with different degrees of success. The most successful unsupervised approach offered better results than the LiDAR approach in terms of pose recovery accuracy and speed, but with poorer detection rates.

Thus the integrated solution proposed uses the LiDAR to robustly detect the presence of the pipe and to produce an approximate estimation of its position, which in turn is projected into the image to use it as a seed to improve visual detection of the pipe. Once the pipe has been detected in the image, the apparent contour is extracted and used to recover the pose of an SHCC, considered the geometrical model of the pipe.

All the proposed methods have been tested with real experimental data acquired in a motion capture testbed, which provided the ground truth for a hand-held rigid frame deploying the sensors used, in a configuration analogous to the one that could be found in and UAV. Additional vision only sequences, captured with an actual multicopter, were used to test the vison based approaches as the differences between indoor and outdoor environments greatly impact their performance.

Acknowledgments: This research has been funded with EU Project AEROARMS project reference H2020-ICT-2014-1-644271, http://www.aeroarms-project.eu/.

References

1. Song, H.; Ge, K.; Qu, D.; Wu, H.; Yang, J. Design of in-pipe robot based on inertial positioning and visual detection. *Advances in Mechanical Engineering* **2016**, *8*, 168781401666767, doi:10.1177/1687814016667679.
2. Wang, Z.; Zhao, H.; Tao, W.; Tang, Y. A new structured-laser-based system for measuring the 3D inner-contour of pipe figure components. *Russian Journal of Nondestructive Testing* **2007**, *43*, 414–422, doi:10.1134/S1061830907060113.
3. Hansen, P.; Alismail, H.; Rander, P.; Browning, B. Visual mapping for natural gas pipe inspection. *The International Journal of Robotics Research* **2015**, *34*, 532–558, doi:10.1177/0278364914550133.
4. Zsedrovits, T.; Zarandy, A.; Vanek, B.; Peni, T.; Bokor, J.; Roska, T. Visual Detection and Implementation Aspects of a UAV See and Avoid System. In; IEEE, 2011; pp. 472–475.
5. Holz, D.; Nieuwenhuisen, M.; Droeschel, D.; Schreiber, M.; Behnke, S. Towards Multimodal Omnidirectional Obstacle Detection for Autonomous Unmanned Aerial Vehicles. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* **2013**, *1*, 201–206, doi:10.5194/isprsarchives-XL-1-W2-201-2013.
6. Klein, G.; Murray, D. Parallel tracking and mapping for small AR workspaces. In *Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on*; 2007; pp. 225–234.
7. Bhanu, B.; Lee, S.; Ho, C.-C.; Henderson, T. Range data processing: Representation of surfaces by edges. In *Proc. Int. Pattern Recognition Conf*; 1986; pp. 236–238.
8. Jiang, X. Y.; Meier, U.; Bunke, H. Fast range image segmentation using high-level segmentation primitives. In , *Proceedings 3rd IEEE Workshop on Applications of Computer Vision, 1996. WACV '96*; 1996; pp. 83–88.
9. Nguyen, A.; Le, B. 3D point cloud segmentation: A survey. In *2013 6th IEEE Conference on Robotics, Automation and Mechatronics (RAM)*; 2013; pp. 225–230.
10. Biosca, J. M.; Lerma, J. L. Unsupervised robust planar segmentation of terrestrial laser scanner point clouds based on fuzzy clustering methods. *ISPRS Journal of Photogrammetry and Remote Sensing* **2008**, *63*, 84–98, doi:10.1016/j.isprsjprs.2007.07.010.

11. Diebel, J.; Thrun, S. An Application of Markov Random Fields to Range Sensing. In *Proceedings of the 18th International Conference on Neural Information Processing Systems*; NIPS'05; MIT Press: Cambridge, MA, USA, 2005; pp. 291–298.
12. Golovinskiy, A.; Funkhouser, T. Min-cut based segmentation of point clouds. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*; IEEE, 2009; pp. 39–46.
13. Lafferty, J.; McCallum, A.; Pereira, F. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data.
14. Fischler, M. A.; Bolles, R. C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM* **1981**, *24*, 381–395, doi:10.1145/358669.358692.
15. Choi, S.; Kim, T.; Yu, W. Performance evaluation of RANSAC family. *Journal of Computer Vision* **1997**, *24*, 271–300.
16. Raguram, R.; Frahm, J.-M.; Pollefeys, M. A Comparative Analysis of RANSAC Techniques Leading to Adaptive Real-Time Random Sample Consensus. In *Computer Vision – ECCV 2008*; Lecture Notes in Computer Science; Springer, Berlin, Heidelberg, 2008; pp. 500–513.
17. Glennie, C. L.; Kusari, A.; Facchin, A. Calibration and Stability Analysis of the VLP-16 Laser Scanner. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* **2016**, *34*, 55–60, doi:10.5194/isprs-archives-XL-3-W4-55-2016.
18. Quigley, M.; Conley, K.; Gerkey, B.; Faust, J.; Foote, T.; Leibs, J.; Wheeler, R.; Ng, A. Y. ROS: an open-source Robot Operating System. In *ICRA workshop on open source software*; 2009; Vol. 3, p. 5.
19. Foote, T. tf: The transform library. In *2013 IEEE Conference on Technologies for Practical Robot Applications (TePRA)*; 2013; pp. 1–6.
20. Besl, P. J.; McKay, N. D. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **1992**, *14*, 239–256, doi:10.1109/34.121791.
21. Taketomi, T.; Uchiyama, H.; Ikeda, S. Visual SLAM algorithms: a survey from 2010 to 2016. *IPSJ Transactions on Computer Vision and Applications* **2017**, *9*, 16, doi:10.1186/s41074-017-0027-2.
22. Ferri, M.; Mangili, F.; Viano, G. Projective Pose Estimation of Linear and Quadratic Primitives in Monocular Computer Vision. *CVGIP: Image Understanding* **1993**, *58*, 66–84, doi:10.1006/ciun.1993.1032.
23. Puech, W.; Chassery, J.-M.; Pitas, I. Cylindrical surface localization in monocular vision. *Pattern Recognition Letters* **1997**, *18*, 711–722, doi:10.1016/S0167-8655(97)00077-9.
24. Puech, W.; Chassery, J. M. Curved surface reconstruction using monocular vision. In *1996 8th European Signal Processing Conference (EUSIPCO 1996)*; 1996; pp. 1–4.
25. Colombo, C.; Bimbo, A. D.; Pernici, F. Metric 3D reconstruction and texture acquisition of surfaces of revolution from a single uncalibrated view. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2005**, *27*, 99–114, doi:10.1109/TPAMI.2005.14.
26. Doignon, C.; de Mathelin, M. A degenerate conic-based method for a direct fitting and 3-d pose of cylinders with a single perspective view. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*; IEEE, 2007; pp. 4220–4225.
27. Bartoli, A.; Sturm, P. The 3D line motion matrix and alignment of line reconstructions. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2001. CVPR 2001*; 2001; Vol. 1, p. I-287-I-292 vol.1.
28. Duda, R. O.; Hart, P. E. Use of the Hough Transformation to Detect Lines and Curves in Pictures. *Commun. ACM* **1972**, *15*, 11–15, doi:10.1145/361237.361242.

29. Canny, J. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence* **1986**, 679–698.
30. Otsu, N. A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics* **1979**, *9*, 62–66, doi:10.1109/TSMC.1979.4310076.
31. McIlhagga, W. The Canny Edge Detector Revisited. *International Journal of Computer Vision* **2011**, *91*, 251–261, doi:10.1007/s11263-010-0392-0.
32. Marchand, E.; Spindler, F.; Chaumette, F. ViSP for visual servoing: a generic software platform with a wide class of robot control skills. *IEEE Robotics Automation Magazine* **2005**, *12*, 40–52, doi:10.1109/MRA.2005.1577023.
33. Grompone von Gioi, R.; Jakubowicz, J.; Morel, J.-M.; Randall, G. LSD: a Line Segment Detector. *Image Processing On Line* **2012**, *2*, 35–55, doi:10.5201/ipol.2012.gjmr-lsd.
34. Hartley, R.; Zisserman, A. *Multiple view geometry in computer vision*; Cambridge University Press: Cambridge, UK; New York, 2003; ISBN 0-521-54051-8.