Chapter
Perception for Detection and Grasping

E. Guerra, A. Grau, A. Pumarola, A. Sanfeliu

Abstract This chapter presents a methodology for the detection of the crawler used in the project AEROARMS. The approach consisted on using a two-step progressive strategy, going from rough detection and tracking, for approximation maneuvers, to an accurate positioning step based on fiducial markers. Two different methods are explained for the first step, one using efficient image segmentation approach; and the second one using Deep Learning techniques to detect the center of the crawler. The fiducial markers are used for precise localization of the crawler in a similar way as explained in chapter four. The methods can run in real-time.

1 Introduction

Detection and grasping of objects, like a crawler for inspection, is a very important task in aerial manipulation. This chapter presents a methodology for the detection of the crawler used in the project AEROARMS. The approach consisted on using a two step progressive strategy, going from rough detection and tracking, for approximation maneuvers, to an accurate positioning step based on fiducial markers. In the first step two types of methods are used: the first one based on detecting invariant features based on the appearance of the considered robot, through an efficient image segmentation approach; and the second one using Deep Learning techniques to detect the center of the crawler formulated as a background-object pixel-wise classification problem. The accurate positioning based on fiducial markers relies in building multiple marker detection over known augmented reality technologies, allowing to cross validate the different estimations and increase accuracy through least-squares optimization.

2 Crawler detection through monocular vision

Accurate detection and positioning of the target to be operated is a critical aspect in any robotic manipulation operation. In this section, the systems integrated to detect and localize a robotic crawler in an industrial environment are described and discussed. The robotic device considered is a crawler with magnetized wheels which adhere it to ferromagnetic pipes. This allows the crawler to travel along the pipes in order to scan them. Deployment of this kind of devices usually requires that the area is accessible to human personnel, which can be impossible or at least inconvenient in many circumstances. This disadvantages can be mitigated by performing the deployment operations from a multi-copter autonomous unmanned vehicle (UAV). At the same time, performing this operations using an autonomous robot require precise knowledge of the spatial relations with respect to the environment and the robotic crawler.

3 Rough detection and tracking for approximation maneuvers

One of the key tasks in approaching maneuvers is to detect and track the crawler from a far distance. There are various approaches for object detection: feature based, template based, classifier based and motion based. In feature based detection approaches the objects are models based on their appearance characteristics such as shape, size, or color [2], [7]. These methods are computationally efficient and can run in real time in embedded computers, which makes them desirable methods to exploit when appropriate. They are not appropriate in complex scenes where there are many objects with occlusions, shadings and similar shapes and colors (see Fig. 1). In these scenarios, the objects cannot be easily segmented which disables the use of shape and size descriptors. Similarly, if the scene contains multiple objects with the same color as the desired object the color is no longer a distinctive feature of the object.

Fig. 1 Sample industrial environment with heavy presence of pipelines. There are also present abundant reflections, shadows and other visual artifacts.

In template based approaches the object is detected by matching features between a template and the analyzed image. There exist two types of templates, fixed [13] and deformable [10]. Fixed frame matching methods can be used when object shapes are invariant viewing angle of the camera. The template position in the analyzed image is determined by minimizing the distance error between the template and various positions in the image. However, most objects shape change with respect to the viewing angle. In these cases, deformable template approaches are more suited. The detection is obtained by combining both global and local structures of the object parametrized by a deformation transform. Classifier based object detection methods [11], [19] are formulated as a background object pixel-wise classification problem. Deep learning classifier based methods are the current state of the art in object detection. They consist in training a network to regress a classification label for each pixel in the given image. Its main disadvantage is the prerequisite of having thousands of samples of the object to be detected in multiple scenarios, views and light conditions to train the network.

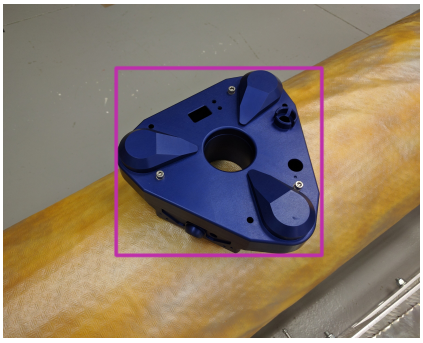

Fig. 2 Close up of the robotic crawler casing over a pipe in a synthetic environment, presenting varying color intensity, with no texture or invariant shape constraints.

The last category of object detection methods is based on motion features extracted as temporal changes at pixel or block level with frame differencing [14], optical flow [3] or Gaussian mixture based methods [1]. Although being the most widely used detection methods are prone to error with noisy videos and rapid camera position changes. For our particular problem of detecting a crawler with an embedded computer template based approaches are not well suited due to the high computational complexity of the template matching optimization problem. Also, a motion feature based approach would also fail due to the vibrations and fast acceleration of the drone itself. Then, given the fact that the crawler appearance is clearly distinctive with respect to a pipe the most appropriate methods are classified and featured based methods. Classifier methods, and in particular deep learning methods, would be the most appropriate method for far distance accurate positioning. We will explain briefly both methods that have been used.

The crawler shape appearance is not invariant to camera view position and it has not texture which disables the use of shape, size and texture features. However, it has a distinctive color with respect to refinery pipes (see Fig. 2). Hence, the first approach used for rough detection and pose estimation for approaching maneuvers is a color based segmentation method. Color based features are constant under viewpoint changes and computationally efficient to acquire. The developed algorithm considers each color component separately and transforms the image representation from RGB to HSL. Then, the image is thresholded to only capture the blue parts of the image and a connected components analysis is performed. In a perfect case scenario, there would be only one connected component, the crawler. To remove possible outliers each of the connected components is filtered based on its dimensions. Finally, once the object is detected its pixel coordinates are projected to the camera plane and the depth is obtained through the dimension of the object.

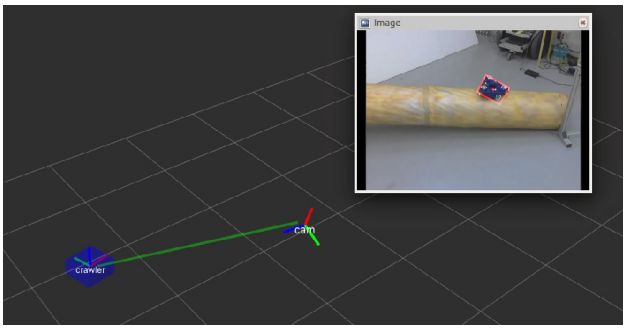

Fig. 3 Experiments on detection of the robotic crawler through visual inspection in a pipe.

The experiments show (Fig 3) that the proposed method is reliable at providing the crawler direction to navigate towards it. Notice that algorithm cannot provide a reliable depth estimation because the object shape is not in-variant to the camera viewpoint. However, this method is intended to be used in the approaching maneuvers, where the important information is the approaching direction, and not its accurate distance to the crawler.

The second technique uses deep learning technique to detect the crawler and it is formulated as a background-object pixel-wise classification problem. The method consists in training a network to regress a classification label for each pixel in the given image. Fig. 4 shows the architecture of the deep learning network.

Our method is based on regression problem using ConvNets [18]. Given an input image $\mathscr{I} \in \mathbb{R}^{HxWx3}$ (H is the height, W is the width, and 3 is the depth), the first step consists in extracting image features from a pre-trained network, in our case VGG [9]. Let us denote these features as $\Psi(\mathscr{I}) \in \mathbb{R}^{H^{'}xW^{'}xC}$. The image features are then fed into the $2D$ detection branch, which is responsible for estimating the $2D$ locations of the crawler center $\mathbf{u} \in \mathscr{U}$, where $\mathbf{u} = (u,v)$ is the set of all (u,v) pixel locations in the input image $\mathscr{I}$. The $2D$ location $\mathbf{u}$ is represented as a probability density

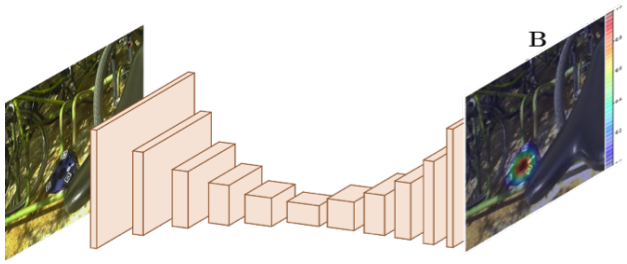

Fig. 4 Deep Learning architecture for detecting the crawler

Table 1 Symbols used in the development of rough crawler detection and tracking.

Definition	Symbol
3-channel $H \times W$ (height x width) resolution image: $\mathscr{I} \in \mathbb{R}^{H \times W \times 3}$	$\mathscr{I}$
Set of image features $\Psi(\mathscr{I}) \in \mathbb{R}^{H' \times W' \times C}$	$\Psi(\mathscr{I})$
Set of all pixel locations (u,v): $\mathbf{u} = (u,v)$	$\mathbf{U}$
Crawler center in image $\mathscr{I}$	$\mathbf{u}$
Probability density map (belief) for $\mathbf{u} = (u,v)$	$\mathbf{B} \in \mathscr{R}^{HxW}$
Deep Learning cost function	L
Binary Cross Entropy	BCE
Mean Squared Error	MSE

map $\mathbf{B} \in \mathscr{R}^{HxW}$ computed over the entire image domain as $\mathbf{B}(u,v) = P(\mathbf{u_i} = (u,v))$, $\forall (u,v) \in \mathscr{U}$. The output $\mathbf{u} = (u,v)$ can be recovered as the following weighted sum over the belief map $\mathbf{B}$:

$$u = \frac{\sum_{(u,v) \in \mathscr{U}} u}{\sum \mathbf{B}} \tag{1}$$

$$v = \frac{\sum_{(u,v) \in \mathscr{U}} v}{\sum \mathbf{B}} \tag{2}$$

The model is trained to minimize the Mean Square Distance between the estimated and desired probability density maps. The model was trained with scenes in indoors, outdoors, different orientations and different illuminations and different scales.

There was also developed a second version of the crawler detection using Deep Learning techniques, which not only computes the probability of being the crawler in the image in a similar way as explained before, but also the probability to detecting the upper-left and the bottom-right locations of the crawler in a bounding box that circumscribe the it. This technique which architecture is shown in Fig. 5 performed better that the previous one.

The results obtained using this last architecture shown a result of 97% of success and in Fig. 6 can be seen some of the detection results.

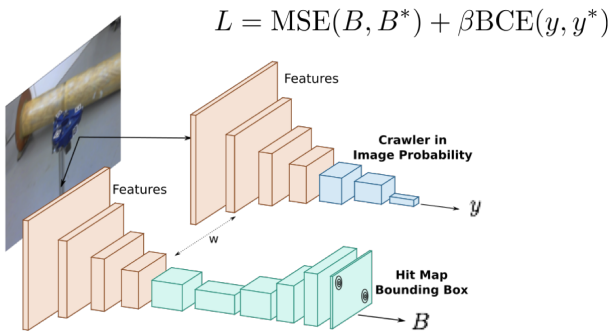

Fig. 5 Deep Learning architecture for detecting the crawler using image probability and bounding box detection

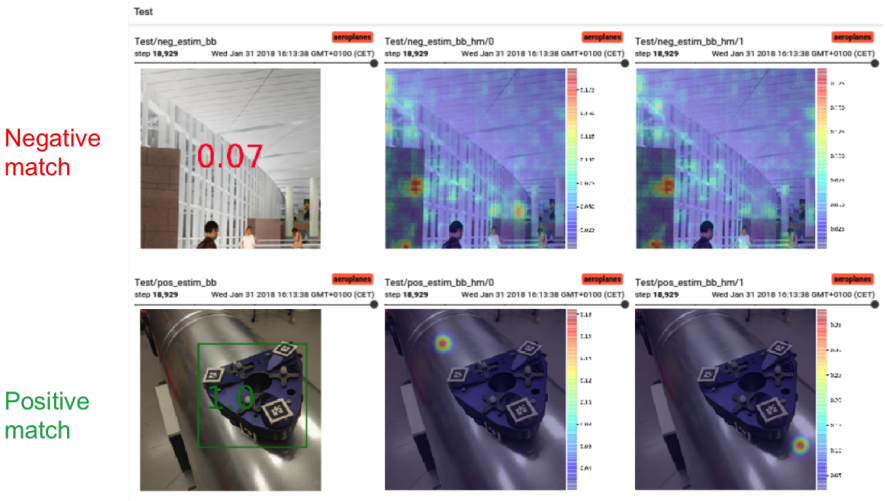

Fig. 6 Results of the crawler detection using image probability and bounding box detection

4 Accurate positioning and tracking for crawler operations

Precise estimation of the crawler pose is a mission critical information in order to successfully perform the logistics operations from the UAV. There are many techniques to detect objects through computer vision, but as described in the previous section, most of them present limited accuracy unless prior knowledge exist, or present excessive computational costs. On the other side, methods based on fiducial markers present the better trade-off between of ac-curacy and performance, especially when the computational power budget is restricted, and other advantages [16], like adding recognition capabilities. There are many libraries implementing different alternative algorithms, like Matrix [15], ARToolkit [8], ARTag [4], and

AprilTag [12], to name the most relevant. Some of the most usual are commented in [5], with an in-depth discussion on the different steps performed by each algorithm.

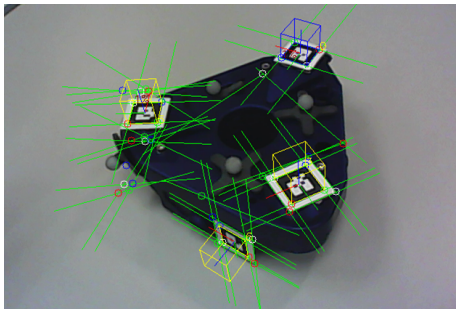

Fig. 7 Detail of the geometrical constraint model optimization process, based on ALVAR. The fiducial markers observed concurrently in a given image are annotated to perform LSE optimization calibration.

Thus, in order to guarantee the maximal accuracy, a positioning system based on fiducial makers was build. Several libraries were tested, including ARuco [6], ALVAR [17], and the works at [8] and [4]. The results obtained were in accordance with those reported in the literature, with ALVAR and ARuco showing greater accuracy and resilience to occlusions thanks to the ability to detect multiple markers jointly. Finally, the use of ALVAR enabled the introduction of a model learning process (see Fig 7), described in [17], which allows to learn a model describing the geometrical relations between the different markers. This process takes few minutes to complete, allowing to modify the setup and geometrical model.

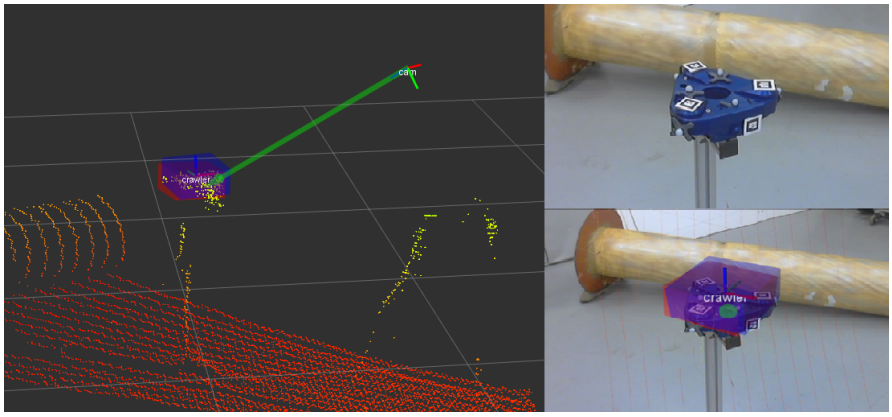

Fig. 8 Experiments with several sensors.Both the rough detection and the accurate positioning of the crawler are active: the red box is positioned according the accurate (marker based) estimation, while the blue box is positioned with the rough estimation

The experiments performed, seen in Fig 8, assumed a worst case scenario where the camera resolution was set at 640x480 pixels to minimize computational load. The results obtained where gave us a relative error in position estimation below 2% at distance interval between 1m and 1.2m, which was determined accurate enough, but produced spurious estimation in about 6% of the frames. most of the spurious estimations presented negative depth in the pose, which required the introduction of a filter to reject them. Reducing the distance below 1m increased the accuracy, and reduced the chances of spurious estimations to 3.4% or lower.

In the case that the fiducial markers are not flat, but are in a curve surface, then it is better to use other fiducial markers using Augmented Virtual Reality Markers, explained in a previous chapter of the Perception for Aerial Robotic Manipulation part.

5 Conclusions

In this chapter we have presented various approaches for detecting AEROARMS crawler robot. The two-steps strategy has been shown to be able to detect the crawler robot, minimizing the computational effort required at each phase of the approximation and operation. Though the rough detection step cannot provide depth information, it is informative enough to set a direction to allow the UAV traveling towards the crawler robot. We have tried two techniques to detect the crawler which are based on color and shape. One uses only color and the other one color and shape in an Deep Learning architecture. At the same time, the fiducial marker method proposed is able to detect the crawler with accuracy, and thanks to the model learning procedure, operations related to acquiring or modifying the fiducial model of the crawler or any other robot can be performed quickly.

References

1. Robert Bodor, Bennett Jackson, and Nikolaos Papanikolopoulos. Vision-based human tracking and activity recognition. In *Proc. of the 11th Mediterranean Conf. on Control and Automation*, volume 1, 2003.
2. J. Chiverton, X. Xie, and M. Mirmehdi. Automatic bootstrapping and tracking of object contours. *IEEE Transactions on Image Processing*, 21(3):1231–1245, 2012.
3. Navneet Dalal, Bill Triggs, and Cordelia Schmid. Human detection using oriented histograms of flow and appearance. In *Computer Vision ECCV 2006*, Lecture Notes in Computer Science, pages 428–441. Springer, Berlin, Heidelberg, 2006.
4. M. Fiala. ARTag, a fiducial marker system using digital techniques. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 590–596 vol. 2, 2005.
5. M. Fiala. Designing highly reliable fiducial markers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(7):1317–1324, 2010.

6. S. Garrido-Jurado, R. Muoz-Salinas, F. J. Madrid-Cuevas, and M. J. Marn-Jimnez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6):2280–2292, 2014.
7. Zhenjun Han, Qixiang Ye, and Jianbin Jiao. Online feature evaluation for object tracking using kalman filter. In *2008 19th International Conference on Pattern Recognition*, pages 1–4, 2008.
8. H. Kato, M. Billinghurst, I. Poupyrev, K. Imamoto, and K. Tachibana. Virtual object manipulation on a table-top AR environment. In *Proceedings IEEE and ACM International Symposium on Augmented Reality (ISAR 2000)*, pages 111–119, 2000.
9. Yann Lecun, Urs Muller, Jan Ben, and Eric Cosatto. Off-Road Obstacle Avoidance through End-to-End Learning. In *NIPS'05 Proceedings of the 18th International Conference on Neural Information Processing Systems*, pages 739–746, 2005.
10. X. Liu, L. Lin, S. Yan, H. Jin, and W. Jiang. Adaptive object tracking by learning hybrid template online. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(11):1588–1599, 2011.
11. Yi Liu and Y. F. Zheng. Video object segmentation and tracking using psi;-learning classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 15(7):885–899, 2005.
12. E. Olson. AprilTag: A robust and flexible visual fiducial system. In *2011 IEEE International Conference on Robotics and Automation*, pages 3400–3407, 2011.
13. J. Pan, B. Hu, and J. Q. Zhang. Robust and accurate object tracking under various types of occlusions. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(2):223–236, 2008.
14. Fatih Porikli and Alper Yilmaz. Object detection and tracking. In *Video Analytics for Business Intelligence*, Studies in Computational Intelligence, pages 3–41. Springer, Berlin, Heidelberg, 2012.
15. J. Rekimoto. Matrix: a realtime object identification and registration method for augmented reality. In *Proceedings. 3rd Asia Pacific Computer Human Interaction (Cat. No.98EX110)*, pages 63–68, 1998.
16. Andrew C. Rice, Robert K. Harle, and Alastair R. Beresford. Analysing fundamental properties of marker-based vision system designs. *Pervasive and Mobile Computing*, 2(4):453–471, 2006.
17. Sanni Siltanen, Mika Hakkarainen, and Petri Honkamaa. Automatic marker field calibration. In *Proc. Virtual Reality International Conference (VRIC2007), Laval, France*, pages 261–267, 2007.
18. K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *in arXiv preprint: 1409*, page 1556, 2014.
19. X. Zhang, W. Hu, W. Qu, and S. Maybank. Multiple object tracking via species-based particle swarm optimization. *IEEE Transactions on Circuits and Systems for Video Technology*, 20(11):1590–1602, 2010.