# Active Text Perception for Mobile Robots

Martin L. Wyss[1] and Peter Corke[2]

*Abstract*— **Our everyday environment is full of text but this rich source of information remains largely inaccessible to mobile robots. In this paper we describe an active text spotting system that uses a small number of wide angle views to locate putative text in the environment and then foveates and zooms onto that text in order to improve the reliability of text recognition. We present extensive experimental results obtained with a pan/tilt/zoom camera and a ROS-based mobile robot operating in an indoor environment.**

## I. INTRODUCTION

The ability to read text in an environment provides humans with rich semantic information that is used almost unconsciously for high-level navigation tasks. The huge amount of such text in the world in the world attests to its utility for tasks such as place name (room 1107), place functional description (bathroom), navigation (street sign, building directory) or driving hints (speed limit, curve ahead). To date robots have not been able to access this source of semantic information and often rely on a second tier of informational infrastructure such as QR codes or RFID tags. However a widespread rollout of robot navigational infrastructure would be costly to implement and therefore quite unlikely to occur. A preferred solution is to allow robots to read the text that is already in place for human benefit.

The problem of reading so-called *text in the wild* is much harder than the problem of recognising text in a scanned document or book. Wild text has a wide variety of fonts and orientations sometimes varied on a character by character basis for artistic reasons. We can make no assumptions about the orientation of text, lighting can be variable and non fronto-parallel viewing can lead to character distortion and different sized characters within words.

To date there has been relatively little work on text reading for robots but what has been reported[9], [12], [13] is consistently limited by poor optical character recognition (OCR) performance on wild text, that is, the returned text string is a corrupted version of the world text. The reports also consistently agree that a significant contributor to this problem is poor image resolution — it has been repeatedly shown that as the text size in the image falls below a certain number of pixels performance degrades very rapidly.

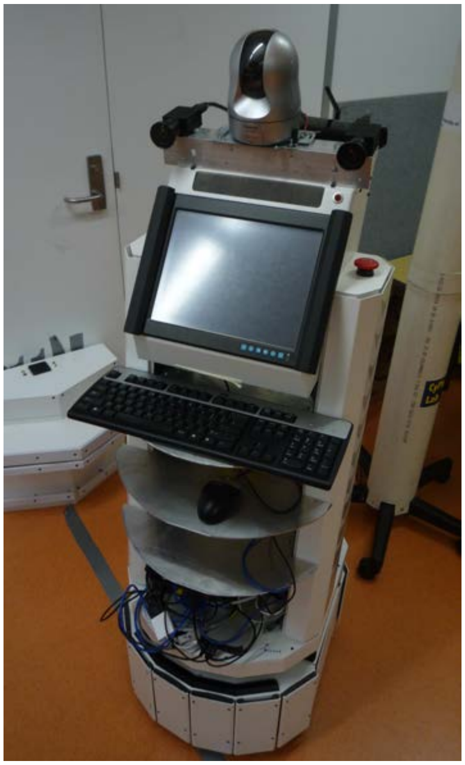


Fig. 1. Guiabot equipped with the pan-tilt-zoom camera.

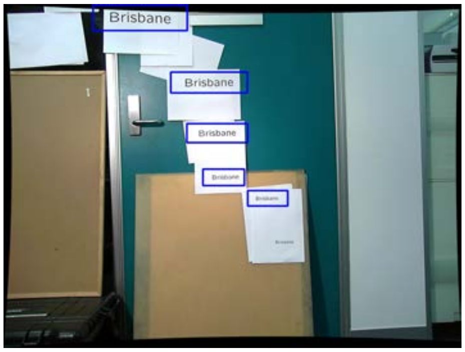


Fig. 2. Active vision text reading setup.

In this paper we describe and evaluate the first active perception system for mobile robot text detection. We use a pan/tilt/zoom camera to foveate on potential text regions identified in a wide-angle view of the scene. By applying the camera's optical zoom we are able to have words fill the image which allows for greatly improved OCR performance. With an $18\times$ optical zoom we are able to achieve a $3\times$ improvement in ability to correctly read wild text. We present a brief description of the novel active-vision algorithm but devote most of the paper to an extensive experimental evaluation of this system. The robot operates in an indoor work environment with modest light levels and is integrated

[1] Martin is with the Autonomous Systems Lab, ETH, Zurich

[2] Peter is with the CyPhy Lab, School of Electrical Engineering & Computer Science, Queensland University of Technology, Brisbane, Australia peter.corke@qut.edu.au

with onboard mapping and localization subsystems.

The next section reviews relevant prior work in text detection and recognition, active vision and robotic text reading. Section III describes the architecture of our system and the active-vision algorithm. In Section IV we describe detailed experiments that evaluate different aspects of our system's performance. Finally, in Section V we present our conclusions and discuss some areas for future work.

## II. PRIOR WORK

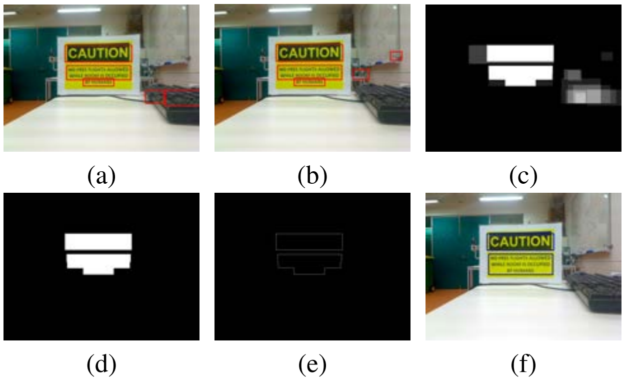
Robotic text reading is commonly divided into text detection (which parts of the scene contain text), text recognition (what is that text) and layout and spelling correction typically organised in a linearly *pipeline*. In order to progress the field the Int. Conf. on Document Analysis and Recognition (ICDAR) has run a series of robust reading and text localisation competitions in 2003, 2005 [2] and 2011 [3]. The latest competition revealed that the wining participant reached a recall for text detection of 62.5% and a precision of 83.0%. However when combined with text recognition, even after spelling correction, the best result was a recall of 41.2% of the ground truth words. We can conclude that the problem is hard but text recognition is harder than text detection.

The cue for text in a region of the image is presence of high image frequencies. Many approaches are based on the use of weak classifiers based on Haar-like features[5], as originally for the problem of face detection, which are combined using boosting to create a strong classifier. Chen et al. [6] extended this by defining additional features based on x- and y-direction image gradient. A different approach is the stroke width transform[7] which highlights close parallel edges that are indicative of text[9].

The most recent robust reading competition [3] indicates that optical character recognition (OCR) is still less robust than text detection. Reasons for a bad recall rate are small text size, the angle of observation and complex font styles. Mirmehdi et al. [10] and Posner et al. [12] both mention an optimal character height of at least 20 pixels for successful OCR. A commonly used OCR engine is the open-source package Tesseract[11] which has been used in the systems described by [9], [12] and [13]. To compensate for errors introduced at the OCR stage it is common to apply spelling correction, typically based on a database of words and the *edit-distance* distance metric[9], [12].

A variant from the common pipeline is PICT [13] which applies Histogram of Oriented Gradient (HOG) techniques to recognize individual characters and uses a mass-spring model to spatially organise them into words. They report more than twice the recall performance of Tesseract.

The concept of active vision for robotics dates back to [17], and can be considered to mimic the human eye which is continually explore the world. Peniak et al. [15] suggest that Mars-rovers should use active vision for obstacle avoidance. A recent paper [16] discusses active perception for robotic text reading but no robot-based results are provided. They achieved a precision of 68% and recall of 59% (harmonic mean of 63%) in 6.1 s.



(a) (b) (c)

(d) (e) (f)

Fig. 3. Stable bounding boxes detection. a) First image, b) Last image, c) Voting result, d) After threshold, e) Contours, f) Stable BB

Posner et al. [12] described an early system for text detection in natural images with a mobile robot and made available an open-source text reading pipeline based on a boosted classifier and the Tesseract OCR engine. They also investigated semantics of spotted text in order to answer high level image queries. [9] use a PR2 robot to read door signs for room numbers and occupant names in an office environment. After building a map the three dimensional location of observed text is placed into the map. Literate_PR2[18] is a ROS-based package for detecting and recognising wild text and has achieved a precision of 45% and recall of 67% (harmonic mean of 54%) in just 0.18 s.

## III. ARCHITECTURE & ALGORITHMs

Our active text reading is based on open-source components ROS, Literate_PR2 and Tesseract[11]. Our goal is to extract as much text from the world as possible, but we not (yet) attempting to infer semantic meanings about places from that text.

The approach is summarised in Algorithm 1. The robot starts by scanning its environment using the widest angle zoom setting and panning around the environment to cover a specified azimuth range with minimal overlap typically ±90 deg with respect to the direction of travel.

The system then recursively searches stable text bounding boxes foveating and zooming as it proceeds. When it encounters a bounding box of sufficient size, or at the zoom limit, OCR is applied. This recursive, depth-first, investigation of the image helps to minimise the motion of the camera and thus reduces the active perception time.

Working in an indoor environment with low light levels causes the camera to choose a high gain setting which results in noisy images. The bounding box detector is based on images edges and is therefore quite sensitive to image noise. We look for temporal consistency in the bounding boxes by examining a short sequence of $N_s$ images see Figure 3 (we chose $N_s = 7$). All pixels in all detected BB vote and the result (c) is thresholded at 80% of the maximum (d) from which contours parallel to the x- and y-axes are extracted (e).

Before exploring a stable bounding box the camera has to foveate on the centre of the box and then zoom. We deter-

**Algorithm 1** Text scanning

```
procedure ReadText
    for $\psi = \psi_{min} \to \psi_{max}$ do            ▷ pan across the scene
        ProcessBB
    end for
end procedure

procedure ProcessBB                                     ▷ Process a bounding box
    for all $b \in$ FindStableBB do
        if readable then
            text ← OCR($b$)
        else
            $\theta, \phi, \zeta \leftarrow$ ZOOM($b$)   ▷ determine pan/tilt/zoom
            MoveCamera($\psi + \theta$, $\phi$, $\zeta$)  ▷ move the camera
            $R \leftarrow$ ROI($b$)                       ▷ compute ROI
            if zoom possible then
                ProcessBB
            else
                text ← OCR($b$, $R$)
            end if
        end if
    end for
end procedure

function FindStableBB
    $V \leftarrow 0$                                     ▷ clear voting array
    for $i = 1 \to N$ do
        $I \leftarrow$ image                             ▷ acquire image
        $V \leftarrow$ UpdateVote($V$, $I$)              ▷ update votes
    end for
    $\{b_i\} \leftarrow$ FindBoxes($V$)                  ▷ Find boxes in voting array
return $\{b_i\}$                                         ▷ return list of bounding boxes
end function
```

mine the largest zoom that allows the box to fill the image vertically or horizontally and the pan/tilt angles necessary to centre it in the image.

If the box has a very different aspect ratio to the image we compute and apply a region of interest (ROI) mask to prevent spurious detections and reduce processing time. In some situations the camera is at a motion limit and the box cannot be centred and we use a ROI to mask out the relevant part of the image.

To embed text in the robot's map we use localisation data from the AMCL node and insert 2D text markers into the map. To reduce map clutter we filter text against a list of expected words. We also apply spatial clustering so that similar words at similar locations are placed into the map just once.

## IV. EXPERIMENTAL SETUP & RESULTS

For the experiments we used an Adept Guiabot mobile platform equipped with a SICK laser range sensor and ultrasonic sensors. The platform's low-level computer runs Ubuntu/ROS and is connected via an onboard Ethernet network to a MacMini/Ubuntu system that runs navigation and stereo vision software. A third computer to run the active text spotting system was placed onboard the robot and connected to its local network. A Panasonic WV-SC385 performs the active vision task, it is a dome IP/Ethernet camera with 18x optical zoom, pan-tilt, autofocus and depth from focus function. At its lowest zoom setting the view angle is 55°. The camera has a web interface, thus the camera parameters can be queried and set with HTTP get and post methods. The video stream was subscribed with the ROS package GSCAM.

|  | Recall Stable BB | Read correctly |
|---|---|---|
| LMMono10 (Serif) | 10% | 100% |
| FreeSans (Sans Serif) | 100% | 50% |
| LMSansQuot8 (Sans Serif) | 100% | 90% |

TABLE I

EVALUATED STATISTICS IN TEN RUNS OF THE ACTIVE VISION PROGRAM.

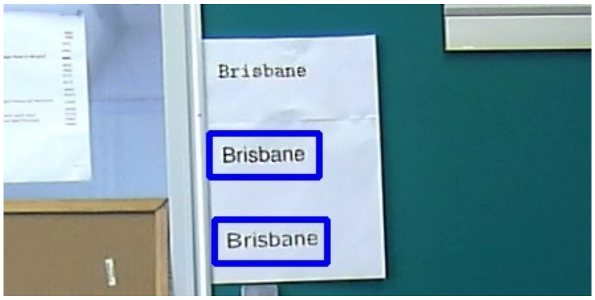


Fig. 4. Font evaluation with blue boxes indicating stably detected bounding boxes. Note that the top font was not detected.

We conducted a series of graduated experiments to evaluate the performance of bounding box detection, full text reading pipeline and the active text perception.

### A. Font Evaluation

As already mentioned the weakest part of the pipeline will be the OCR so we wish to remove this factor from our evaluation of the active text perception performance. We active this by choosing a font that is reliably read by OCR. We evaluated fonts from LibreOffice (LMMono10, FreeSans, LMSansQuot8) which are shown in Figure 4. The camera was placed 2 m away from the text and perpendicular to the text plane and the pipeline was run ten times. We evaluated bounding box detection performance

$$\text{Recall} = \frac{\text{Num relevant retrieved BB}}{\text{Total relevant BB}} \quad (1)$$

and OCR correct reading rate (CRR) performance is

$$\text{CRR} = \frac{\text{Num Correctly Read BB}}{\text{Num BB Retrieved}} \quad (2)$$

Results are shown in table IV-A and indicates that the LMSansQuot8 font was detected and read most consistently. This is a sans serif font with good spacing between the letters.

| Font size in Point | Font size in Pixels |
|---|---|
| 160 | 30 |
| 130 | 24 |
| 100 | 19 |
| 72 | 14 |
| 60 | 12 |
| 48 | 10 |

TABLE II

MAP FROM FONT SIZE TO EFFECTIVE PIXEL SIZE.

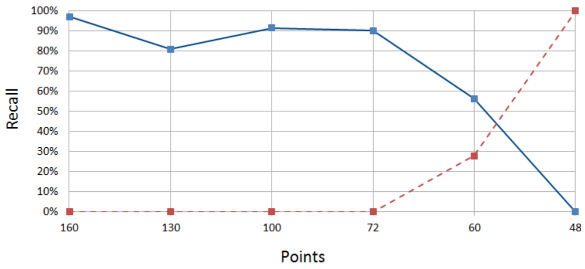


Fig. 5. The bounding boxes can be retrieved reliably (blue line) for font size greater than 48 points. The fraction of trials with no detection is given by the red dashed line.

### B. Bounding Boxes Recall as a Function of Text Size

The size of a character in the image depends on the font size and the distance of the text from the camera. To establish the minimal image character size (in pixels) for further experiments we evaluate bounding box recall for a range of font sizes between 160 and 48 points. The camera configuration was as above and the pipeline was executed 25 times for each font size. Bounding box recall is

$$\text{Recall} = \frac{\text{box area intersection}}{\text{biggest box}} \tag{3}$$

where intersection is the overlap between the computed box and a defined ground truth box so as to penalise incorrect box size estimates.

Figure 5 shows the average recall (solid blue line) and the number of trials with zero recall (red dashed line). We observe that the bounding boxes are reliably computed for fonts larger than 72 points, which accordingly to table IV-B correspond to 14 pixels in the image. For image size between from 14 to 10 pixels the recall becomes unstable and the number of trials with zero recall starts to rise. At 10 pixels no bounding boxes are retrieved. As a rule of thumb bounding boxes can be reliably computed for image character size of 13 pixels or more.

### C. Active Vision Text Reading

To evaluate the benefits of active vision system for text spotting we use the same setup as for the previous experiments but the OCR functionality is enabled. The evaluation is executed with and without zooming and the pipeline is executed 15 times for each case.

The results of the recall measure are presented in figure 6 and show that the correct recognition rate without zooming falls dramatically below 130 points or 24 pixels and is zero below 19 pixels. With zooming enabled, recognition performance is above 80% for all pixels sizes which spans a size range of nearly a factor of three. Without zooming the character height has to be at least 30 pixels but with zooming the minimum size can be as low as 12 pixels.

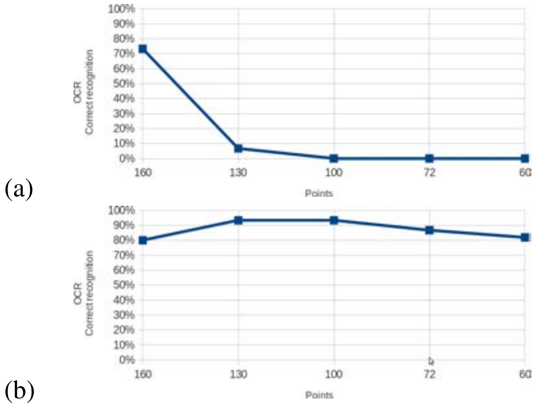


Fig. 6. OCR recognition rate with (a) and without (b) zooming.

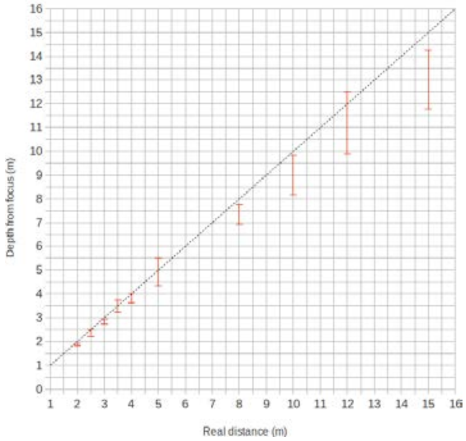


Fig. 7. True and actual depth from focus between 1.5 m and 15 m.

### D. Depth from Focus

The Panasonic camera publishes estimates of depth obtained from focus — according to the data sheet in the range 1.4 to 999.9 m. We were intrigued by the possibility of using this camera to not only recognise text but also to report its distance from the camera, which is important when placing text information into a map. We moved the text over a distance range 1.5–15 m from the camera which was kept zoomed onto the text.

Figure 7 shows the actual (dashed line) and reported distance with error bars. Up to 4 m the standard deviation is less than 25 cm but for distances of 12 m the deviation is very significant, more than 1 m. While somewhat disappointing the accuracy was sufficient for text on the walls of the hallway where the robot operated.

### E. Text Spotting with a Mobile Robot

This final experiment demonstrates that the developed algorithms and programs are able to perceive text along a

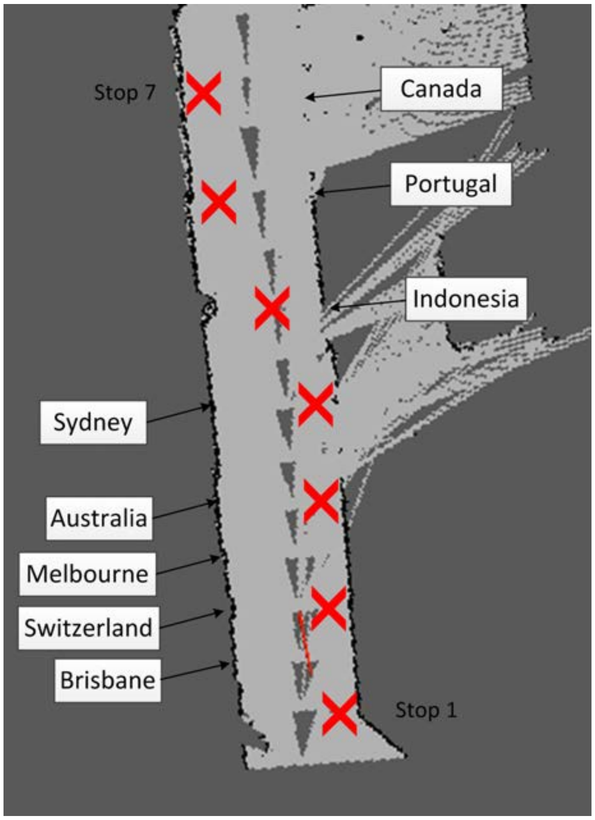
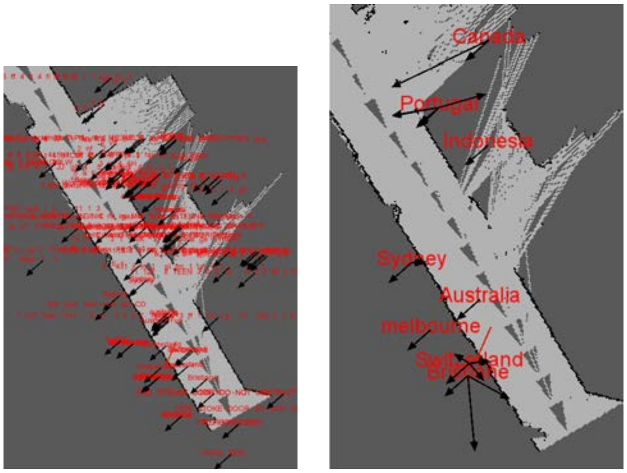


Fig. 8. Ground truth map from the setup.



Unfiltered map

Filtered map

Fig. 9. Text spotting results from the mobile robot

hallway from a mobile robot. To eliminate the effects of OCR weakness we placed words in a consistent font (100 point LMSansQuot8) along the hall way. We evaluate:

1) Recall of the expected BBs depending on the distance to the text.

$$\text{Recall} = \frac{\text{relevant retrieved boxes}}{\text{total relevant boxes}} \tag{4}$$

2) Correct text recognition rate depending on the distance to the text.
3) Average displacement of the markers

| | -2 | -1 | 0 | +1 | +2 |
|---|---|---|---|---|---|
| BB Recall | 53% | 100% | 100% | 100% | 47% |
| Correct read | 75% | 33% | 72% | 50% | 29% |
| AVG initial char size (pix) | 14 | 18 | 20 | 18 | 14 |
| Real dist. (m) | 2.69 | 2.06 | 1.80 | 2.06 | 2.69 |
| Angle | ±138° | ±119° | ±90° | ±61° | ±42° |

TABLE III

STABLE BB RECALL AND CORRECT READING RATE DEPENDING ON THE DISTANCE TO THE TEXT AND ANGLE.

4) Average scanning time.

Here the recall for the boxes is not calculated on ground truth intersection, since it is difficult to define this for a dynamic experiment. We consider that recall was 100% if the text was retrieved from within a box otherwise it was 0%.

The robot uses a pre-learned map (from gapping) and is programmed to stop every 2 m and scan the environment three times for text. A map of the environment is shown in Figure 8, the stops are marked with a cross and the boxes show the text which is placed in the world. The robot drives past the signs at a distance of 1.2–1.8 m.

All found text was placed into the *unfiltered map* and the keyword filter was used to remove existing environmental text — we are interested in the performance only against the words we posted on the walls. The robot's pan range was set to 300° degrees in order to spot text that the robot has passed.

The generated maps are shown in figure 9. The filtered map shows arrows that indicate the view direction to the word when it was detected. Most words are detected multiple times from different robot locations. The average displacement between text locations in the map and ground truth is 64 cm most of which we attribute to errors in distance obtained from depth from focus. Figure 10 shows the word 'Brisbane' being triangulated from three different robot poses.

Over the three scans the correct recognition rate of the signs is 87.5%. The reason for not reaching 100% is that the word "melbourne" was detected with a lower-case rather than upper-case first letter which we consider an error. If the scans are analysed separately then we achieve recognition rates of 75%, 75% and 62.5%. In table III the various performance metrics are binned against the distance from the text at the time of recognition. The real distance in meters and the angle is calculated from the driving direction (the robot's orientation is parallel to the corridor).

The recall is stable plus minus one meter around the robot. Further away the recall drops to about 50% and for a distance of 3 m nothing is detected anymore, since the average pixel size is about 11 pixels. The reason for the large sensitivity to distance is the influence of this single consistently misread word that falls into the larger distance bins. The recall of the expected bounding boxes 1.8 m from the robot was 100% and the correct recognition rate for these bounding boxes is 72.2The average performance times per scan is 165 s and the average time per stable bounding box is 15 s.

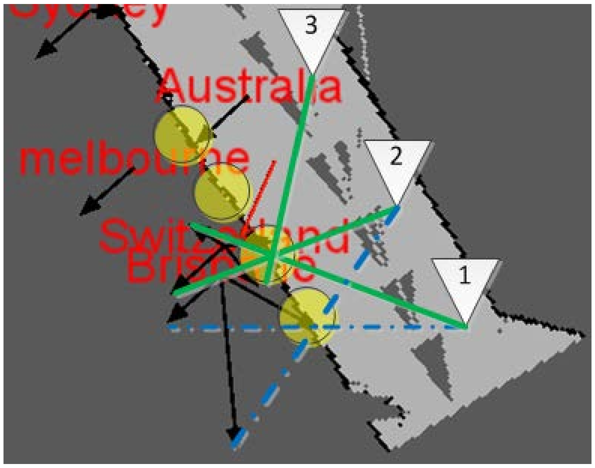


Fig. 10. The dotted lines point at the pose of Brisbane and the solid line of Switzerland. The black arrows lay on a line which goes through the robots pose and the ground truth image position.

The Recall and the correct recognition rate after three scans are good results. The average pixel size shows that without zooming none of the text could have been read, since the maximum average pixel size is about 20 pixels and our earlier experiment show that the pipeline does not recognize any text below 24 pixels. The benefit of zooming is real and significant.

The fact that Tesseract had problems reading the word Melbourne twice in the 100% bounding box recall region shows clearly that the OCR step is the weakest link in the framework.

## V. CONCLUSIONS

In this paper we describe an active text spotting system that uses a wide angle views to locate putative text in the environment and then foveates and zooms onto the text in order to improve the reliability of text recognition. We present extensive experimental results obtained with a pan/tilt/zoom camera and a ROS-based mobile robot operating in an indoor environment. The framework is able to spot stable text regions and detect text in the environment up to a couple of meters depending on the text size and font. The active vision system increases the likelihood of recognizing the spotted text correctly and we were able to achieve a $3\times$ improvement using active vision. We were also able to place observed text in the robot's map with an accuracy better than $1\,\mathrm{m}$. As in previous work we have found that weakest link in the chain is the text recognition or OCR function.

Prospective areas for future work include optimising the camera motion to reduce scan time and better early culling of extraneous bounding boxes. The task could also be paralleled with the active perception system queuing likely boxes to the OCR engine for post hoc analysis. It would also be useful to make the system operate with the robot moving. Early work shows that text triangulation is promising and this should be investigated further.

## ACKNOWLEDGMENT

The authors would like to thank ETH Zurich and Professor Roland Siegwart, head of the Autonomous Systems Lab, for making this work possible as well as Dr. Cédric Pradalier deputy head of the Autonomous Systems Lab for his support.

## REFERENCES

[1] S.M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, and R. Young, ICDAR 2003 robust reading competitions. In Proceedings of the Seventh International Conference on Document Analysis and Recognition, volume 2, pages 682-687, 2003.

[2] S.M. Lucas. ICDAR 2005 text locating competition results. In Document Analysis and Recognition, 2005. Proceedings. Eighth International Conference on, pages 80-84. Ieee, 2005.

[3] A. Shahab, F. Shafait, and A. Dengel, ICDAR 2011 robust reading competition challenge 2: Reading text in scene images. In Document Analysis and Recognition (ICDAR), 2011 International Conference on, pages 1491-1496. IEEE, 2011.

[4] J. Aloimonos, I. Weiss, and A. Bandyopadhyay, Active vision. International Journal of Computer Vision, 1(4):333-356, 1988.

[5] P. Viola and M.J. Jones, Robust real-time face detection. International journal of computer vision, 57(2):137-154, 2004.

[6] X. Chen and A.L. Yuille, Detecting and reading text in natural scenes. In Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on, volume 2, pages II-366. IEEE, 2004.

[7] B. Epshtein, E. Ofek, and Y. Wexler, Detecting text in natural scenes with stroke width transform. In Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, pages 2963-2970. IEEE, 2010.

[8] J. Canny, A computational approach to edge detection. Pattern Analysis and Machine Intelligence, IEEE Transactions on, (6):679-698, 1986.

[9] C. Case, B. Suresh, A. Coates, and A.Y. Ng, Autonomous sign reading for semantic mapping. In Robotics and Automation (ICRA), 2011 IEEE International Conference on, pages 3297-3303. IEEE, 2011.

[10] M. Mirmehdi, P. Clark, and J. Lam, A non-contact method of capturing low-resolution text for ocr. Pattern Analysis & Applications, 6(1):12-21, 2003.

[11] R. Smith. An overview of the tesseract OCR engine, In Document Analysis and Recognition, 2007. ICDAR 2007. Ninth International Conference on, volume 2, pages 629-633. Ieee, 2007.

[12] I. Posner, P. Corke, and P. Newman, Using text-spotting to query the world. In Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Con- ference on, pages 3181-3186. IEEE, 2010.

[13] K. Wang and S. Belongie, Word spotting in the wild. Computer Vision-ECCV 2010, pages 591-604, 2010.

[14] Y. Han, Imitation of human-eye motion-how to fix gaze of an active vision system. Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on, 37(6):854-863, 2007.

[15] M. Peniak, D. Marocco, S. Ramirez-Contla, and A. Cangelosi, Active vision for navigating unknown environments: An evolutionary robotics approach for space research. 2010.

[16] J.A.Á. Ruiz, P. Plöger, and G. Kraetzschmar, Active scene text recognition for a domestic service robot. 2012.

[17] R. Bajcsy, "Active perception," *Proceedings of the IEEE*, vol. 76, no. 8, pp. 966–1005, 1988.

[18] M. Zhu, "Literate_pr2.", `http://www.ros.org/wiki/literate_pr2`, 2012.