Active Scene Text Recognition for a Domestic Service Robot

José Antonio Álvarez Ruiz*, Paul Plöger, and Gerhard Kraetzschmar

jose.alvarez@smail.inf.h-brs.de
{paul.ploeger,gerhard.kraetzschmar}@h-brs.de
University of Applied Sciences Bonn-Rhine-Sieg
Computer Science Department
Sankt Augustin, Germany

Abstract. We developed a scene text recognition system with active vision capabilities, namely: auto-focus, adaptive aperture control and auto-zoom. Our localization system is able to delimit text regions in images with complex backgrounds, and is based on an attentional cascade, asymmetric adaboost, decision trees and Gaussian mixture models. We think that text could become a valuable source of semantic information for robots, and we aim to raise interest in it within the robotics community. Moreover, thanks to the robot's pan-tilt-zoom camera and to the active vision behaviors, the robot can use its affordances to overcome hindrances to the performance of the perceptual task. Detrimental conditions, such as poor illumination, blur, low resolution, etc. are very hard to deal with once an image has been captured and can often be prevented. We evaluated the localization algorithm on a public dataset and one of our own with encouraging results. Furthermore, we offer an interesting experiment in active vision, which makes us consider that active sensing in general should be considered early on when addressing complex perceptual problems in embodied agents.

Keywords: Scene text recognition, active vision, domestic robot, pan-tilt, auto-zoom, auto-focus, adaptive aperture control.

1 Introduction

Beyond being a simple commodity, domestic service robots might open the doors to a more fulfilling life to the elderly and handicapped. However, for robots to perform assistance tasks under very complex and *dynamic* environments, great *robustness* and *flexibility* are required. For example, the agents should gain information about the environment, the agent's situation in it, and that of other agents. For this, perceptual processes have to turn the raw sensory data into higher-level representations. In this investigation, we work towards the acquisition of information from a source seldom exploited in robotics, text embedded in

* The Master's degree of José Antonio Álvarez Ruiz was funded with a CONACYT-DAAD scholarship.

images, commonly referred as *scene text*. Text is a valuable source of information because: **1)** It is readily available in human made environments **2)** Humans make extensive use of it **3)** It contains semantic information. In the end, text provides us humans with the information needed to identify and compare products at the supermarket, find our path at the airport, ordering at a restaurant, etc. A potential application of *Scene Text Recognition* (*STR*) in robotics is product identification, which is generally performed with some sort of appearance based classification. However, appearance based methods suffer of an *inherent lack of generalization* because the appearance of products changes over time and across vendors. If a robot could read text written on boxes and bottles, and understand it, it would result in a more general solution.

STR is a very challenging topic –not to be confused with traditional *Optical Character Recognition* (*OCR*)– because scene text is known to have a large intra-class variance in terms of font, color, layout, symbol repertoire, etc. and the presence of background clutter. Nevertheless, advances in *STR* are not only applicable in robotics, but also profitable by visually impaired and blind humans. Therefore, we consider *STR* to be an important research topic and hopefully, with this article, we raise more interest in it within the robotics community. Although we deal with a perceptual task, our approach diverts respect to the traditional and still often applied conception in computer vision that: sensation, perception and cognition are isolated processes previous to actuation. Under such paradigm, commonly referred as *passive vision*, the perceptual system is limited to operate using the raw data captured by the sensors "as is". This "sensation-followed-by-idea-followed-by-movement" lacks on "psychological adequacy" according to [6]: *"We begin not with a sensory stimulus, but with a sensorimotor coordination... In a certain sense it is the movement which is primary, and the sensation which is secondary, the movement of the body, head, and eye muscles determining the quality of what is experienced"*. Therefore, our robot does not obtain information by plain observation, but also by interaction and selection of stimuli using a *Pan-Tilt-Zoom* (*PTZ*) camera, in such a way, that the agent gains control of *"what to see"* and *"how to see it"* [13].

2 Related Work

Fibonacci search was introduced as an effective methodology for searching optimal focus values using the tenengrad operator in [11]. A system to optimize focus and aperture, based on a hierarchy of artificial neural networks (ANN), was described in [13]. In [14], a system for the extraction of low-resolution text was developed. The system first locates text areas and then uses a *PTZ* camera to capture and assemble a high-resolution mosaic of each region. The use of a polynomial zoom model is mentioned but no details were given. Recently, [17] developed a text localization system for a robot. That work is very relevant because it also includes the extraction of semantic information from the text using probabilistic models and textual web-search. [10] created a robot system with text reading capabilities for aiding visually impaired humans in naviga-

tion tasks. The robot, equipped with a *Pan-Tilt Unit* (*PTU*), was designed to read room numbers using a template matcher. However, the authors placed very strong assumptions, such as: possible characters are limited to numbers and to A-E characters. [18, 19] developed a text localization system based on *Discrete Cosine Transform* (*DCT*), and a text tracking system. A *PTU* was used to take a panoramic capture from which text is detected. In those publications active vision is limited to the capture of a mosaic to increase the *Field of View* (*FOV*), but no adaptive actions were performed. [4] introduced a set of features for text discrimination. Those features are calculated from *sub-regions (blocks)* embedded within the detection window that were found to exhibit a distinctive behavior for text. Several statistical measures were computed and combined from the blocks and used to train a *Cascade of Boosted Classifiers* (*CoBC*) with asymmetric adaboost as stage classifiers. [15] used a similar block layout to delimit regions from which *Histogram of Oriented Gradients* (*HOG*) features were extracted to train the first layers of a *CoBC*. The successive layers used *Local Binary Patterns* (*LBP*) and multi-scale *LBP*. [16] extended their previous work in [15] to use two *Conditional Random Fields* (*CRF*) to filter non-text connected components. An image operator called *Stroke Width Transform* (*SWT*), which proved to be very useful and yield better results than other *STR* methods was introduced in [7]. We introduced a *Connected Component* (*CC*) based *STR* system in [1, 3]. However, it performs poorly on low-resolution text. Besides, being a passive *STR* system, it is unable to adapt to different image acquisition conditions, which limits its usefulness in the real-world.

3 Text Localization

Text localization, i.e. to identify and delimit image regions that contain text, is generally the first step in *STR*. This task is difficult because of the large *intra-class variance* of text, lack of prior-knowledge on the scale, orientation, etc. and the presence of background clutter, which might generate similar visual stimuli as text, e.g. windows of a building, a fence, etc. To find text regions, we pass a sliding window through the image at each possible location and at different scales. At each location and scale, a set of discriminative features are extracted from the image inside the detection window. Then, a classifier uses the feature values to assign a *confidence score* to each detection window, higher scores indicate higher probability of text and conversely. Finally, a *confidence map* generated during classification is thresholded and smoothed, and the bounding rectangle of each remaining segment stands for a text region. Localization is a canonical example for *rare-event detection* [21], in which the expected amount of content for the positive class (*text*) is much smaller than for the negative class (*non-text*). Such conditions, along with the number of detection windows that need to be classified, also make of text localization a difficult problem. We use a *CoBC* [22] with *asymmetric adaboost* [21] as stage classifiers and *decision trees* [2, 20] of depth two as weak learners. This particular classification framework is specially well suited for problems with skewed class distributions. Besides, the *CoBC* is com-

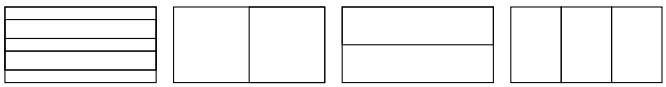


Fig. 1: Example of blocks, each inner rectangle represents a block. The outer rectangles represent the detection window.

putationally efficient in comparison to other methods because it performs the classification in stages of increasing complexity. In the initial stages, the stage classifiers are very simple and operate on low-dimensional feature spaces, and still they are able to discard a large amount of the non-text detection windows. Later stages, require of more complex classifiers operating over feature spaces of a higher dimensionality. However, thanks to the decision tree algorithm, features are calculated as needed, instead of pre-calculating the complete feature vectors. Unfortunately, these concepts will not be discussed in detail due to space constraints.

3.1 Feature Space

To train the *CoBC*, we used features that have been reported to perform well in our domain. The features are extracted from sub-regions within the detection window called *blocks* [4]. The blocks are arranged in such way, that the features extracted from them exhibit low entropy, see Fig. 1. A first set of features we use, introduced in [4], is based on mean and standard deviation values either of the intensity image, the intensity gradient magnitude or the x or y intensity gradients; we will refer to these features as "Chen". A second set of features is *HOG* [5], having a *HOG* per-block as defined in [15]. Roughly, the gradient orientations are decomposed in a set of bins. Then, each pixel within a block, casts an orientation vote in the corresponding bin in the *HOG* of that block. The contribution of each pixel is weighted by the magnitude of the gradient at that location. To minimize aliasing effects, we interpolate the values accumulated in each histogram. From now on, Chen and HOG features will be referred as "raw". Furthermore, the raw feature values are processed to transform them into *log-likelihood ratios* [4]. To this end, for each raw feature and possible combination of pairs of raw features, we estimate the conditional probability density of the text and non-text classes. In the case of feature pairs, this process creates new features from the raw features. To model each probability density distribution we use a *Gaussian Mixture Model* (*GMM*) trained with the *Expectation Maximization* (*EM*) algorithm. The optimal number of components per-mixture is estimated with the *Bayesian Information Criterion* (*BIC*) [8].

4 Active Vision Module

Images of poor quality can easily hinder the performance of a *STR* system, for example, due to a loss of contrast between text and the background, or lack

of spatial resolution for the task. To cope with this we developed three active vision behaviors, namely: **1)** *Auto-focus*, to prevent blur by defocus **2)** *Adaptive Aperture Control* (*AAC*), to widen the dynamic range of the capture; both aimed to retain the contrast of the text regions by optical means, and **3)** *Auto-zoom*, to acquire high-resolution images. The sensory system we use consists of a SONY camera model VFW-VL500 and a Directed Perception *PTU* model 46-17. The sensory space we consider is formed by the parameters: pan, tilt, zoom, aperture and focus. The active vision behaviors have three major components, either implicitly or explicitly: **1)** *Quality metrics*, that assign a quantitative value to the quality of an image **2)** *Actuation*, to change the configuration of the sensory system by manipulating its electrical and mechanical components **3)** *Search strategies*, to explore the configuration space of the sensory system and find desirable configurations.

Our active vision module has an *initialization phase* and a *recognition phase*. In the initialization phase, the camera is prepared to localize candidate text regions in the scene. First, we set the zoom to its minimum value to maximize the *FOV* and set the focus to its maximum value to produce sharp images of relatively distant objects. Furthermore, the *AAC* behavior corrects the camera aperture according to the illumination conditions of the scene. Once the sensory system has been set up for the scene, we store the current configuration of the sensory system. Afterwards, we capture a frame and localize text regions in it using the algorithm described in Sect. 3 to obtain a set of bounding rectangles of candidate text areas; most of which correspond to real text, and eventually some false positive regions. Finally, a priority calculated from the text confidence map is assigned to each candidate region. During the recognition phase, the regions are attended one by one in order of descending priority as follows. Each candidate region is centered and zoomed-in, in order to capture a high-resolution image. Then, the aperture is optimized again and auto-focus is executed to acquire a sharp image. After processing each region, the sensory system is set back to the previously stored configuration before the next candidate region is processed, see Fig. 2.

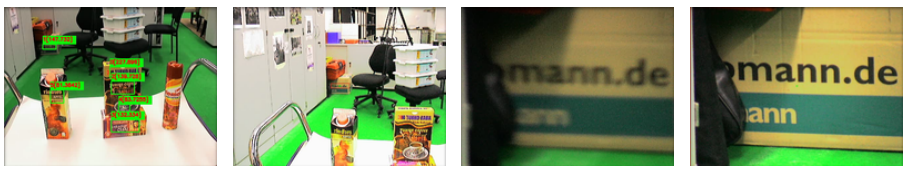


Fig. 2: Example of active STR. From left to right: Localization results after the initialization phase, centering candidate region 1 (a carton box at approximately 3 m w.r.t the camera, the top-left candidate region), after zooming-in, after AAC and auto-focus.

4.1 Auto-Focus

Real world cameras require of lenses to focus the light passing through the aperture against the *image plane*, where the imaging sensor is located. However, fixed lenses cannot adequately focus light coming from arbitrary distances. In general, given a lens with a focal length f, only objects located at a distance d_{out} in front of the lens will produce a sharp image on a plane behind the lens located at a distance d_{in}, called *focal plane*. Therefore, objects at different distances will have different focal planes. Light rays coming form objects that lie either closer or farther than d_{out} will be projected as a circle of radius r instead of a point over the focal plane. However, since the sensor resolution is limited, if the radius of a blur circle is small enough, the defocus effects will not be resolved by the sensor and will not be observable; this range of distances is known as *Depth of Field* (DOF). Auto-focus consist thus, of changing the distance d_{in} so that the focal plane of a certain object of interest is aligned with the image sensor. The acquisition of focused images is desirable in STR because text can be considered as *high-spatial-frequency* content, which is smoothed due to defocus, leading thus to weak intensity gradients. To optimize the focus, we measure the image quality using the *thresholded gradient magnitude operator* (also known as tenengrad operator) defined in Eq. 1, and use *Fibonacci search* to find the maximum [11].

$$Tenengrad(|\nabla\mathcal{I}|) = \sum_x \sum_y |\nabla\mathcal{I}|(x,y) \;\text{ for } |\nabla\mathcal{I}|(x,y) > \tau \;\; . \tag{1}$$

Where $|\nabla\mathcal{I}|$ is the magnitude of the intensity gradient and τ is a threshold [1]. For a certain DOF, the tenengrad operator will exhibit its maximum when an object in the *Region of Interest* (ROI) within that DOF is focused. Nevertheless, the tenengrad operator can have local maximum if the ROI contains objects at different DOFs. It is also important to know that the DOF decreases as the magnification increases, making it harder to focus magnified objects.

4.2 Adaptive Aperture Control

The aperture controls the amount of light entering the camera, and must be set accordingly to the the scene's illumination and structure. Otherwise, the captured image might exhibit a narrow dynamic range and thus poor contrast. Two extreme manifestations of this phenomena are *overexposure* and *underexposure*. Whereas for a fixed aperture and under certain conditions, underexposure can be corrected with the use of additional light sources, this does not occur with overexposure. However, the opposite also holds true, if there is not enough light in the scene, opening the aperture will not prevent underexposure on its own unless the robot illuminates the scene. In this investigation, we use the entropy of the intensity image as a quality measure [9] and Fibonacci search to maximize the entropy. In the end, well illuminated images generally present a more evenly distributed intensity histogram.

[1] We use temporal averaging and set $\tau = 0$ instead [11].

4.3 Auto-zoom

Digital cameras capture a 2D *discrete approximation* of a 3D continuum. The spatial resolution with which an image region is captured, depends on the number of pixels in the imaging sensor and the distance between the camera and the points in 3D space within that region. In general, low-resolution images are challenging to deal with in computer vision, and are often found when working in problems involving small structures[2] observed from a distant viewpoint. For this reason we implemented auto-zoom, which allows to acquire high-resolution images of a ROI. Our auto-zoom algorithm begins by centering the ROI in the camera frame, which is achieved through iterative correction of the camera external orientation using a PTU. At each step, the image of the ROI becomes a *template*. Then, a small pan and tilt correction towards the camera center is performed (0.1° each) and the improvement in centering is measured in vertical and horizontal direction by finding the ROI in a new capture using a cross-correlation based template matcher. Using the x and y displacement of the ROI respect its previous location and the pan and tilt angles, we estimate new pan and tilt commands. The necessity for an iterative solution arises because we assume an uncalibrated camera at this point. Later on, the zoom parameter value that would allow a capture of the highest resolution of the ROI can be found using Eq. 2.

$$z_{max}(z_0, R_w, R_h) = Z_v\left(\zeta M\left(z_0\right) \min\left(\frac{w}{R_w}, \frac{h}{R_h}\right)\right) \quad . \tag{2}$$

Where z_0 is the initial zoom parameter value, R_w and R_h are the width and height of the ROI respectively, Z_v is a polynomial model [14, 23] that maps magnification factors to zoom parameter values, M is a polynomial model that maps zoom parameter values to magnification factors, w and h are the width and height of the image respectively, and ζ is a multiplier on the resulting magnification. The optimal degree of these *cubic polynomials* was estimated using BIC. The regression was performed on 32 data points obtained by recording the corresponding zoom parameter value of 8 different magnification factors for 4 different calibration targets, where each calibration target was a black circle printed on a white sheet of paper. For each calibration target, the camera position and orientation was first manually set so that at zoom parameter 0, the target fits withing a square of known size overlaid in the camera images. Then, the zoom was increased so that the calibration target fits in a larger square and so on. The side length ratio of each of the squares respect to the smallest one corresponds to the magnification factor. Auto-focus was performed at each step but the focus parameter values were not used for regression.

5 Experimental Evaluation

We trained a *CoBC* of four stages, for which we first assembled training and validation image collections using images of the ICDAR train dataset [12] and

[2] Such as text written on a can.

Fig. 3: Normalized training images. On the left for the negative class, and on the right for the positive class.

hundreds of images of scenes such as parks, streets, kitchens, living-rooms, grocery products, etc. obtained from the Internet. With this, we attempt to capture the high variability of the text and non-text classes. Afterwards, we generated a set of normalized training and validation images for the positive and negative classes (see Fig. 3) from which feature vectors for the training and validation examples were extracted. The normalized images have the same width and height as the detection window (24×12 pixels) and resemble the detection windows classified by the *CoBC*. The training datasets per-stage were formed of 3,378 examples of each class and the validation datasets contained 2,218 examples of the positive class and 30,000 examples for the negative class. The feature space had 7,180 dimensions, formed by 160 raw features extracted from 20 blocks, which after being turned into log-likelihoods ratios over the individual and pairwise combinations of raw features produced additional 7,056 features. In order to avoid a higher dimensionality in the data, the raw feature combinations were only performed over features of the same kind. Finally, the confidence map threshold and the threshold of the last stage classifier were optimized on a validation set of images.

The localization algorithm was evaluated on the ICDAR dataset as well as on a dataset of images of grocery products (referred as grocery images). All of the grocery images are RGB images of 640×480 pixels captured at a distance of 60 cm w.r.t to the camera. The performance was measured in terms of pixel-wise precision and recall. In general, precision is defined as $p = \frac{|C|}{|E|}$, recall as $r = \frac{|C|}{|T|}$ and their harmonic mean $h = \frac{1}{\alpha/p+(1-\alpha)/r} | \alpha = 0.5$, where C stands for the correct detections, E for all detections, and T for the target detections according to the ground truth. Since our method does not implement word grouping and the ground truth consists of bounding rectangles of each word the precision estimate will result pessimistic. Nevertheless, word grouping is very hard to realize on low-resolution images and is of minimal practical use for our application. Moreover, we compare our algorithm against the *literate_pr2*[3] package placed in the public domain and available in the *Robot Operating System* (*ROS*) repositories. The literate_pr2 package was used with OCR validation disabled. The results of the evaluation are given in Table 1. Although our algorithm performed

[3] The algorithm is based on [7].

Table 1: Text localization results in terms of pixel-wise precision, recall and harmonic mean. Complementary, the average execution time per-input image is given.

Method	Dataset	p	r	h	time(s)
Presented method	ICDAR	0.68	0.59	0.63	6.08
	Grocery Images	0.66	0.77	0.71	1.58
literate_pr2	ICDAR	0.45	0.67	0.54	0.18
	Grocery Images	0.43	0.75	0.5361	0.03

Fig. 4: Examples of the performance of our localization method on the ICDAR dataset on top and on the grocery images at the bottom.

better in both datasets, it also resulted slower than the literate_pr2 package. We attribute this drawback to the small validation datasets used to create the *CoBC*. Some localization results for our method can be seen in Fig. 4. The recall of our method was poor in images of the ICDAR dataset in which only one or two characters occupy an entire image.

5.1 Adaptive Aperture Effect in the Localization Algorithm

To validate the usefulness of our active vision module, we devised an experiment intended to resemble one of many situations a robot can face under operation in the real-world. For this, we placed the camera at a distance of approximately 60 cm from a table in a room with normal indoor illumination. The camera sensory system was prepared as in the initialization phase described in Sect. 4. We call the resulting aperture *initial aperture value*, or *reference*. We observed that these aperture values produced well illuminated images of the scene (see Fig. 5a). Then, we turned-off the light in the room and turned-on a table lamp and made a new

capture called *passive*; note that the camera was still configured with the initial aperture value (see Fig. 5b). Finally, we performed AAC to optimize the camera aperture to the new illumination conditions and made a final capture called *active* (see Fig. 5d). We repeated the same process to make a series of captures of different products on the table. Each of the images (reference, passive and active) were captured in 4 variants: one shot, one shot with Gaussian filtering, and temporal average of 2 and 5 frames. Hence, the images exhibit different degrees of noise and blur to ensure that the different results are due to illumination. The performance of the localization algorithms in these images is depicted in Fig. 6.

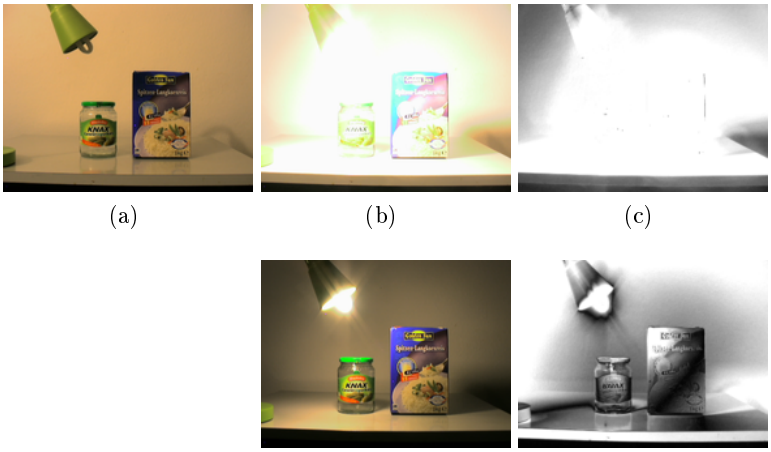
6 Conclusions

In this investigation we devised and evaluated an active STR system with text localization, auto-zoom, auto-focus and AAC capabilities. Our evaluation on a public dataset and on a new dataset gives evidence of the performance of our localization method. Moreover, we demonstrated how the ability to adapt to changes in the environment is crucial to the performance of STR systems. Since harsh acquisition conditions are often problematic in other similar tasks, we are convinced that active vision, and *active sensing* in general will play a crucial role in the development of robotics and should, whenever possible, be considered when working on difficult classification problems. In our experience, it is more effective to do so than to devise more complex passive perceptual systems. Further improvements to our system include the addition of a controllable external light source.

References

1. J. A. Álvarez Ruiz. Learning to Discriminate Text from Synthetic Data. In *Proceedings of the 15th RoboCup International Symposium*, 2011.
2. L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen. *Classification and Regression Trees*. Chapman and Hall/CRC, 1 edition, Jan. 1984.
3. T. Breuer, G. Giorgana Macedo, R. Hartanto, N. Hochgeschwender, D. Holz, F. Hegger, Z. Jin, C. Müller, J. Paulus, M. Reckhaus, J. A. Álvarez Ruiz, P. Plöger, and G. Kraetzschmar. Johnny: An autonomous service robot for domestic environments. *Journal of Intelligent & Robotic Systems*, 66:245–272, 2012. 10.1007/s10846-011-9608-y.
4. X. Chen and A. 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–II–373 Vol.2, june-2 july 2004.
5. N. Dalal. *Finding people in images and videos*. PhD thesis, Institut National Polytechnique de Grenoble, july 2006.
6. J. Dewey. The reflex arc concept in psychology. *Psychological review*, 3(4):357, 1896.
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, june 2010.

8. C. Fraley and A. E. Raftery. MCLUST version 3 for R: Normal mixture modeling and model-based clustering. Technical Report 504, University of Washington, Department of Statistics, 2006. (revised 2009).
9. R. Huber, C. Nowak, B. Spatzek, and D. Schreiber. Adaptive aperture control for image enhancement. In *Computer Architectures for Machine Perception, 2003 IEEE International Workshop on*, pages 7–11, may 2003.
10. K. Iwatsuka, K. Yamamoto, and K. Kato. Development of a guide dog system for the blind people with character recognition ability. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 1, pages 453–456 Vol.1, aug. 2004.
11. E. Krotkov. Focusing. *International Journal of Computer Vision*, 1:223–237, Feb. 1987.
12. S. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, R. Young, K. Ashida, H. Nagai, M. Okamoto, H. Yamamoto, and Others. ICDAR 2003 robust reading competitions: entries, results, and future directions. *International Journal on Document Analysis and Recognition*, 7(2):105–122, 2005.
13. C. Micheloni and G. Foresti. Active tuning of intrinsic camera parameters. *Automation Science and Engineering, IEEE Transactions on*, 6(4):577–587, oct. 2009.
14. M. Mirmehdi and P. Clark. Extracting low resolution text with an active camera for OCR. In *IX Spanish symposium on pattern recognition and image processing*, pages 43–48, 2001.
15. Y.-F. Pan, X. Hou, and C.-L. Liu. A Robust System to Detect and Localize Texts in Natural Scene Images. *The Eighth IAPR International Workshop on Document Analysis Systems*, pages 35–42, Sept. 2008.
16. Y.-F. Pan, X. Hou, and C.-L. Liu. Text Localization in Natural Scene Images Based on Conditional Random Field. *10th International Conference on Document Analysis and Recognition*, 0:6–10, July 2009.
17. I. Posner, P. Corke, and P. Newman. Using text-spotting to query the world. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 3181–3186, oct. 2010.
18. H. Shiratori, H. Goto, and H. Kobayashi. An efficient text capture method for moving robots using dct feature and text tracking. *Pattern Recognition, International Conference on*, 2:1050–1053, 2006.
19. M. Tanaka and H. Goto. Autonomous text capturing robot using improved dct feature and text tracking. *Document Analysis and Recognition, International Conference on*, 2:1178–1182, 2007.
20. T. Therneau and E. Atkinson. An introduction to recursive partitioning using the RPART routines. Technical report, Technical Report 61. URL http://www.mayo.edu/hsr/techrpt/61.pdf, 1997.
21. P. Viola. Fast and robust classification using asymmetric adaboost and a detector cascade. *Advances in Neural Information Processing Systems*, 2002.
22. P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, pages I–511–I–518, 2001.
23. R. G. Willson. *Modeling and calibration of automated zoom lenses.* PhD thesis, Carnegie Mellon University, Pittsburgh, PA, USA, 1994. UMI Order No. GAX94-19735.



(a) (b) (c)

(d) (e)

Fig. 5: AAC allows captures under different illumination conditions to be more consistent. Fig. 5a, image captured using the initial aperture, being the table lamp off. Fig. 5b, passive image captured with the initial aperture and the table lamp on. Fig. 5d, active capture with the table lamp on, after executing the AAC behavior again. Fig. 5c and Fig. 5e are the pixel-wise Euclidean distances in RGB space, between Fig. 5a and Fig. 5b, and Fig. 5a and Fig. 5d respectively. Brighter values indicate a larger Euclidean distance.

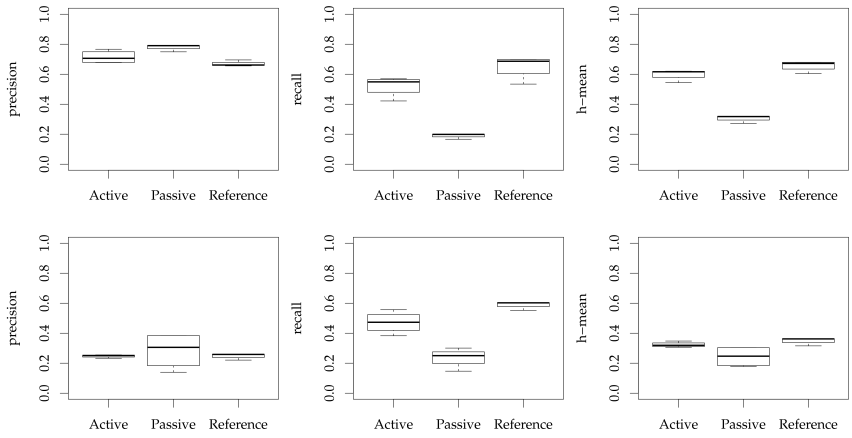


Fig. 6: A series of captures were made following the same procedure as in Fig. 5. On top, results of our algorithm. The literate_pr2 results are displayed at the bottom. From left to right: pixel-wise precision, recall and harmonic-mean. The recall of both algorithms decreases more significantly if the aperture is not adapted to the new illumination conditions.