Active Parameter Control for the Low Level Vision System of a Mobile Robot

G. Appenzeller, P. Weckesser, R. Dillmann

Institute for Real-Time Computer Systems and Robotics
Prof. Dr.-Ing. U. Rembold, Prof. Dr.-Ing. R. Dillmann
University of Karlsruhe, 76128 Karlsruhe, Germany

Abstract

Computer vision systems are today an important sensor for intelligent robotic systems. However, the design of a vision system that a robot can use as a fast and robust sensors in a complex, partially unknown and dynamic environment is still difficult. A main reason for this is that the parameters of vision systems are often adjusted by hand and remain static during the operation of the robot. In this paper we present an general architecture that adapts the the parameters of a segment based low-level vision system dynamically to increase its speed and robustness. Adaptation is done to a priori knowledge about the environment or to the sensor data itself. The architecture is implemented on a mobile robot using special hardware that allows real-time operation. Quantitative experimental data on its performance is given.

1 Introduction

Computer vision today has become an important sensor for robotic systems. However, the design of vision system that a robot can use as fast and robust sensors in a complex, partially unknown and dynamic environment is still difficult. Vision systems used in robotics usually consists of a static set of steps producing a symbolic representation of the scene as an output. In the last decade however, the concept of computer vision has changed towards a dynamic process [1]. A Computer vision system is now considered as a sensor whose use has to be actively planned and whose parameters have to be constantly adapted to a priori knowledge and the results obtained.

Active vision has developed architectures for the automatic control of the external degrees of freedom of image acquisition systems (e.g. pan, tilt, focus, vergence, zoom). The parameters of the feature extraction system (e.g. smoothing parameters, feature detection thresholds) itself however have so far remained static. Their control however is important, as research in computer vision systems in the last years, notably in the field of performance characterization, has shown. The selection of the parameters is often more significant for the results obtained by the system than the choice of the individual algorithms used.

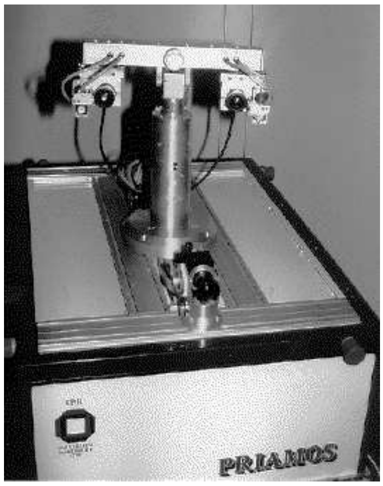

Figure 1: The KASTOR camera head on the mobile robot PRIAMOS.

While for some vision applications parameters can be adjusted individually for each scene by a human operator, this is not possible on mobile robots. If the performance of the system has to be fast as well as robust at the same time (robust against changing environmental conditions and to noise) it is necessary to constantly adapt the parameters of the system to the environment, the objects we try to locate and the results obtained. We therefore need an automatic control strategy for these parameters.

In this work we present a control architecture for parameters of a segment based computer vision system. The system is implemented on the the mobile robotic system PRIAMOS [2] (figure 1). PRIAMOS is a mobile robot that is used as a test-bed for the intelligent use of multiple sensor-systems. It is equipped with the active camera head KASTOR and special hardware allowing segment extraction in real time. The vision system is used for various robot navigation and exploration tasks [3].

We first discuss what parameters the vision system has and how they affect its overall performance, suggest a number of control strategies for them, present the design of the vision system and give experimental results.

2 Parameters and their Effect

The vision system used by the robot is a classic segment based vision system (for a complete description see e.g. [4]) as it is used for various applications. It is depicted in figure 2.

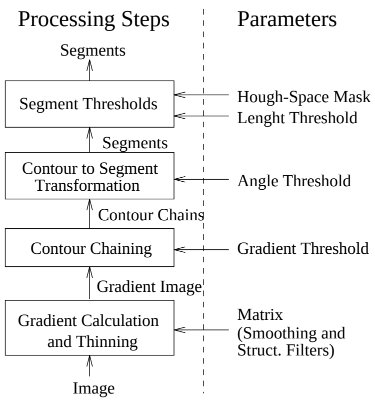

Figure 2: The segment extraction system.

On the left side of the diagram we can see the steps of the vision system. The input of the system are the images that are captured by the active camera head. In the next step the absolute value of the gradient and its direction is calculated for every image pixel and all gradient non-maxima are suppressed. In the third step all pixels with a gradient above a certain value are chained to contours. The contour chains are transformed into segments in step 4. Finally the segments are filtered by removing all segments that do not meet certain criteria. Usually the criterion used is a minimum length for the segments. We we will later suggest an enhanced approach using an additional hough-space matrix.

Before we can define a control strategy, we have to analyze how each parameter affects the overall output of the system.

The calculation of the gradient image is usually done by convolving the image with an edge detector (or filter). Common detectors are the finite Sobel and Canny detector or the infinite Deriche detector. Two of the most important parameters of edge detectors is their stability in the presence of high frequency noise (their smoothing strength) and how their response depends on the orientation of the edge (anisotropic edge detectors). We will treat each of these properties as a parameter of its own.

2.1 Smoothing Strength

As Canny [5] has shown the derivative of a Gaussian is a good approximation of an edge detection operator that is optimal according to his criteria. The only parameter that can be adjusted is the size of the Gaussian.

The effect of this parameter is best described as the smoothing strength. Convolving the image with the derivative of Gaussian is equal to convolving it with a Gaussian and calculating the derivative. We could there as well smooth the image and calculate the derivative of the smoothed image. The Gaussian will act as a low-pass filter and eliminate high-frequency noise making the calculation of the gradient stable. Additionally however it will suppress small details.

Usually the noise and contents parts of an image can not be distinguished by their frequency spectrum. Smoothing will always destroy noise and image details at the same time. The situation is even more complicated by the fact that what is considered noise will depend on the task the system has to perform. Signal processing defines everything as noise that is obscuring the desired information in the data. If we try to find a large shape (e.g. a door) any small details (e.g. door handle, hinges, small objects in front of the door) that cause the contour line of the door to break are noise, if we are trying to locate one of the small details (e.g. the door handle) they are important data. We therefore have to adapt the smoothing to the object we are interested in.

Witkin [6] has proposed the scale-space to describe the behavior of features under smoothing. The optimal scale space position (or smoothing strength) at which we should observe the feature however is difficult to predict. A low scale space position will not suppress other features sufficiently while a too high scale space position will make the feature disappear. Witkin has additionally shown that before the features disappear their location will become unstable. If high precision of the contours is required (e.g. for 3D reconstruction of the scene) it is therefore important to choose a scale space location significantly below this point.

Hardware and computational constraints often limit the maximum possible smoothing of the edge detectors. A way to obtain additional smoothing is to defocus the cameras of the camera head. This however is only effective to suppress noise (or details) that are contained in the image. Noise that is generated by the image acquisition process (e.g. camera noise) is not reduced. Defocusing the cameras only became applicable after the introduction of high-quality low-noise digital cameras.

2.2 Anisotropic Edge Detectors

Usually two operators are used to obtain the first derivative of the image in X- and Y-direction for each pixel. From these two values the direction and magnitude of the gradient are calculated. Depending on the edge detector used the response of the detector to edges with different orientations do not have to be the same. We can design edge detectors that give preference to edges with certain orientations. Canny [5] has shown that these anisotropic operators make edge detection more robust.

If we know the geometry of the object we are trying to locate in the scene we can determine what the orientation of its edges is. If the current task of the robot is to drive through a door it is sufficient to identify the two vertical edges of a frame. Therefore an edge detector that is only sensitive to vertical lines can be used. Other applications are to use artificial Landmarks with diagonal lines which are easy to identify if edge detectors are used that are only sensitive to them or to detect the vanishing lines in a corridor where the approximate angles are known.

There are two main reason why anisotropic edge detectors allow a more robust segment detection. First the classic gradient maxima thinning is unstable is situations as T-intersections of segments or edges with an angle of 22.5 degrees. The distribution of the gradient direction in both cases is random and torn contour chains due to thinning or chaining errors are frequent. The second reason is the anisotropic edge detection reduce the amount of data significantly as the experimental data in section 4.1 shows.

2.3 Gradient Threshold

After the calculation and thinning of the gradient image only those pixels are chained into contours whose absolute gradient value is above a certain threshold. This is done to prevent weak edges from generating segments.

A higher gradient threshold will lead to less segments and low-contrast segments are more likely to break. If the threshold is low, the segments we are interested in will be found. However a large number of other segments, some of them due to various types of noise, will also be detected. If the threshold is set very low even the quality of the strong segments deteriorates as the growing number of T-intersections causes segments to break.

Canny [5] proposed a hysteresis thresholding technique which is superior to simple thresholding but even more difficult to control and incompatible with our special hardware.

2.4 Segment Thresholds

The contour to segment transformation contains a thresholding operation in most vision systems. Typically one threshold is used to control the maximum deviation of a segment from a straight line before it is broken into two pieces. It is either implemented as an angle or distance threshold. For our application a static setting of this threshold has proven sufficient.

A second threshold specifies the minimum length a segment has to have. This threshold can be used to separate segments and white Gaussian noise [7] or to find only objects of a certain size.

We were looking for a fast but more complex filter that could not only separate segments by their length but also by their position and orientation. Two major possibilities are apparent:

Filtering in the **Image Domain**. By specifying a mask only the edges in some areas on the scene could be selected. Control of the desired orientation of the edges is however impossible and for some applications (e.g. finding all vanishing lines) no reduction of the segments is possible.

Filtering in the **Segment Domain**. Every segment can be characterized by its coordinates in the hough-space (angel and distance to origin). This allows the selective filtering of certain orientations, positions for all of our applications.

Hough-Space filtering is fast. For each edge only the angel and distance to the origin are calculated and the corresponding matrix position is checked. Figure 3a shows example matrices for the door experiment and the detection of vanishing lines.

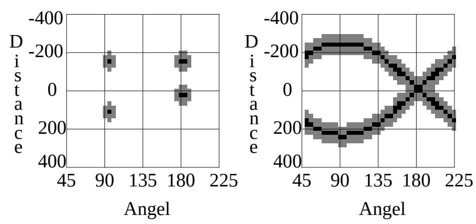

Figure 3: Hough-space filter matrices for door scene and vanishing point detection. An angel of 90^o corresponds to vertical edges. For the vanishing lines case only the absolute value of the distance has been used.

The location of expected segments is only known with a certain precision. The precision of the hough space filter can be easily increased by dilating the matrix with a structured element. In figure 3 the black matrix elements are the original predictions and the grey area surrounding them is the tolerance generated by dilation.

3 Control Strategy and System design

The parameters of the vision system are adapted to two different things. First to **a priori knowledge** about its environment, the robot and the task we perform. Second to the **sensor data** we obtain during the vision process.

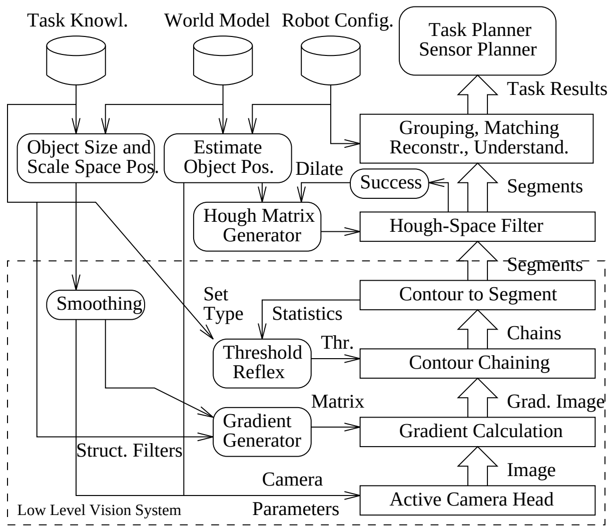

Figure 4: Control Architecture of the System

The overall design of the vision system is shown in figure 4. To the right we see the steps of the vision process. From the images that are captured by the active camera head the gradient is calculated, contours are chained, these are transformed to segments which are filtered by the hough-space filter. Finally the high level vision steps as matching the segments between the camera images, reconstruction of the 3D-scene and its evaluating the scene, are executed. On the top of the image one sees the interface of the architecture with the robot. To the right the sensor planner or task planner receives the results of the vision system and triggers new vision tasks. To the left the databases containing the knowledge of the vision system are shown. These are the constantly updated world model of the robot, the configuration of the robot and the task database containing knowledge about how certain vision tasks are executed. Below it is the control structure for the vision system. Arrows indicate the flow of information.

3.1 Use of Model and Task Knowledge

When the vision system is given a task to perform it usually has some information about what structures it will see in its environment. Predictions come from three main sources.

From the **World Model** the robot can obtain a list of segments in his environment and a geometric description of the object he is trying to locate.

The **Robot Configuration** includes all the parameters of the robot as its position, orientation and the parameters of the camera head including the calibration matrices.

With **Task Knowledge** we mean all knowledge that is associated with the specific task the robot is trying to perform. Such knowledge could be “doors are best located by their vertical segments with vertical edge detectors” or “door handles are small, use little smoothing to detect them”.

The vision process now works as follows. At the top level the world model and the robot position is used to estimate the relative position of the object and the active camera head is positioned towards it. Using this information and the calibration data of the cameras the segments of the object are projected into the three camera images. From the projection and the uncertainty of the robot position the hough space matrices are calculated.

The estimated best point in scale space where to detect the object is contained in the task database. We can not simply derive it from the geometrical size of the object as we have to consider the distance to other objects and their relative contrast. The segment length threshold too is currently stored with the task knowledge database. Its best setting depends on the likeliness of segments to break. The task database additionally contains data about what kind of edge detectors to use (e.g. diagonal for vanishing lines, anisotropic for exploration).

From the desired smoothing the setting for the focus of the camera head and the smoothing value for the gradient is derived. From this and the information about the use of anisotropic edge detectors the final filter matrices are calculated.

3.2 Adaptation to Obtained Data

A priori knowledge provides us with good initial settings for the parameters. However, as the robot often operates in an only partially known or changing environment, some parameters have to be readjusted during the vision process. If the light is switched off or objects the robot tries to locate have moved it has to adapt the vision system to the new conditions by changing the gradient thresholds or making his prediction of the position of the object more tolerant.

In order to derive control mechanisms for a parameter we first need a model how the parameter affects the results of the corresponding step of the system. From this model we can then derive an evaluation criterion which we can use in order to determine how we have to change the parameter. We have implemented dynamic control for two parameters, the gradient threshold and the hough space filter.

The optimal setting of the gradient threshold has been investigated repeatedly [7, 8]. All authors however assume that its purpose is to separate structure from noise for which a simple statistical model exists (e. g. white Gaussian). This is not applicable in out case as the main sources of segments of non-interest is not random noise but other image features.

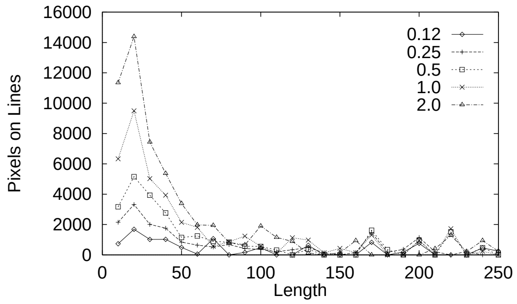

Figure 5: Segment histograms for different detection thresholds. Sum of 5 experiments.

Figure 5 shows the histograms of the number of detected pixels that belong to segments of a certain length (this is the same as the number of segments multiplied with their length) for different sensitivity (the inverse threshold) settings. The number of short (< 50 pixels) segments is about proportional to the threshold. This indicates that these segments are mainly due to noise, textures etc. but not to real image strctures of a particular size. The peaks at 170,200,220 and 240 pixels are the long structures we are interested in. They remain almost unchanged until they disappear completely at some threshold level. We thus propose three different control mechanisms:

Histogram Matching. If we know the object we are looking for we can calculate its segment histogram and change the threshold until the object histogram is part of the histogram we see.

Constant Number of Segments. If we do not know what object we are looking for we can not use the above technique. Assuming that all scenes contain roughly the same amount of structure we can use the reflex to keep the number of segments constant. This does not work well in scenes with many long lines. Long segments will tend to break if the threshold is too low, this will generate more segments, the reflex will lower the threshold further breaking even more long segments etc.

Segment Pixel Ratio. The above technique can be enhanced if we allow more short edges in images in which we have found large structures. A criterion that has proven successful is:

$$Pix(length < 100) = 2000 + \frac{Pix(length > 100)}{10}$$

Here $Pix(X)$ denotes the total number of pixels of all edges for which X is valid. The selection of the type of reflexive control loop is stored in the task database.

A second level of adaptive control is included in the hough filtering process. The histogram matching technique is used to verify if the segment extraction has failed. If this is the case the hough filter will change its tolerance (by dilating the matrix) and restart the vision process.

4 Experimental Results

As an experiment to illustrate the performance of the system we have chosen the task of locating a door in an indoor environment. This is a common task as the robot uses them as landmarks, has to determine if they are open and pass through them frequently. The scene where the door has to be located is displayed in figure 6a. Detecting the door in a fast and robust way have proven difficult in earlier experiments. Many small lines in the door frame create additional lines, small details (e.g. the hinges) cause the contours to break and many smaller structures around the door create additional lines that have to be distinguished from door fragments.

4.1 Predictive Adaptation

The door recognition process is initiated by the task controller of the robot. The vision system now uses task knowledge and knowledge about the geometry of the door to adapt its low level parameters. The task knowledge associated with the task is that the door is best found by its two long vertical side edges of the frame and that these edges are large compared to their environment. The gradient calculation matrices are therefore set to a maximum smoothing detector that will only detect vertical edges and, the active camera head defocuses the cameras slightly.

Using the geometric model of the door, the world model, an estimation of the robot position and the calibration matrices of the cameras it projects the expected segment position into the images. With this

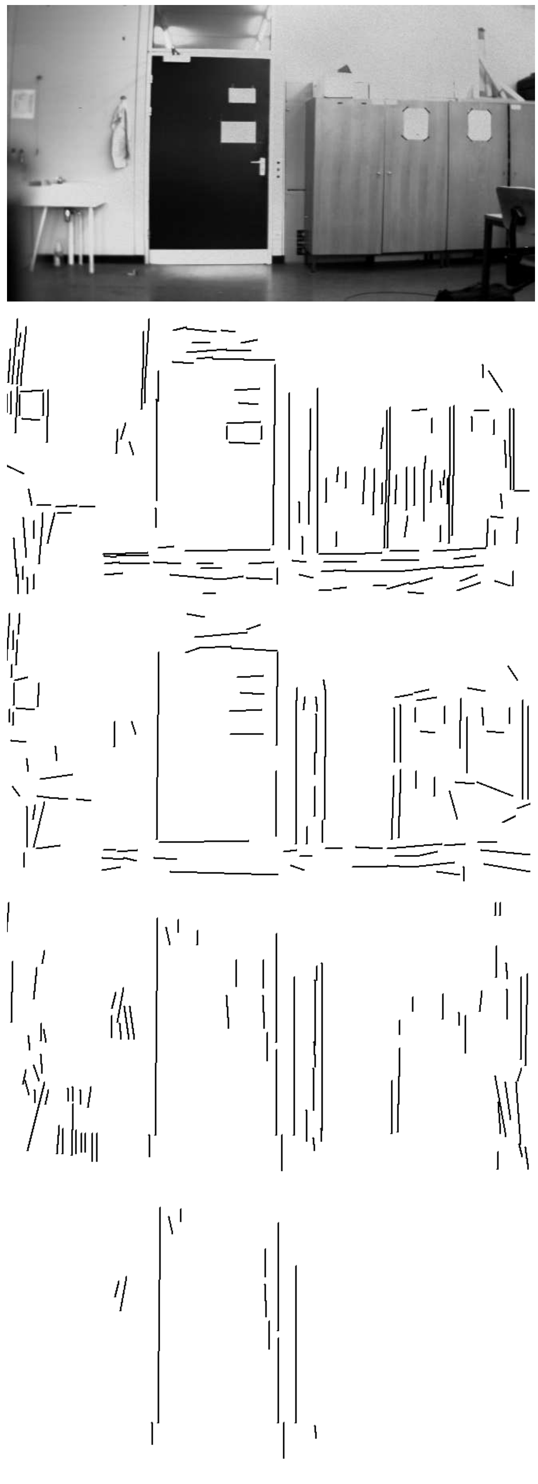

Figure 6: (a-e) The results of the vision system with different types of adaptation. (a) Real image (b) Edges without optimization (c) With scale space adaptation (d) With additional use of anisotropic edge detectors (e) after Hough-Space filtering

position and an estimation of the current location error it initializes the matrices of the hough-filter and positions the active camera head. Finally the image acquisition process is started.

In figure 6 the result of the acquisition process with different levels of adaptive control are shown. In this experiment the early error detection was switched off. Only segments with a length above 5 are shown. Fig. 6b shows the segments of the image without any adaptive control. The door contours are broken, segments parallel to it make identification of the door difficult and the image contains many segments of non-interest. In figure 6c smoothing detectors are used. The number of segments is reduced, the left door segment is no longer broken (as the hinges only cause a very weak deviation of the contour at this scale space level). However the left part of the door now has wide gap due to the melting of the door handle into the door contour. Figure 6d shows the effect of the additional use of vertical edge detectors. The number of segments is again reduced (as horizontal segments are no longer detected), the right door segment is now almost rejoined and most segments are slightly longer. The final figure 6e shows the segments that are passed to the upper levels if the hough-space filter is activated. An additional data reduction (depending on the precision of the estimation by a factor of 5-15) is visible.

To obtain more quantitative and objective results we have measured the effect of the optimization on the histogram of the detected segments. Figure 7 shows the results. The y-axis shows the number of pixels that belong to segments of this length. Each histogram shows the result of 10 independent measurements of the same scene. Without any adaptive control a large number of segments, most of them short and due to broken segments or noise, are found. With adaptive smoothing the number of short segments is reduced. However the number of pixels on very long segments (>200) remains constant. The additional use of anisotropic edge detectors further reduces the detection of short segments but increases the number of detected long segments. The edge detector that only detects vertical edges should only detect about half the number of segments as an anisotropic filter. The fact that it detects more segment suggests that it is possible to use a lower gradient threshold with anisotropic edge detectors. This further reduces the number of segments that have to be processed.

4.2 Continuous Control

In order to test the reflexive behavior of the gradient threshold we place the robot in front of the door and give the vision system the task to look for a large,

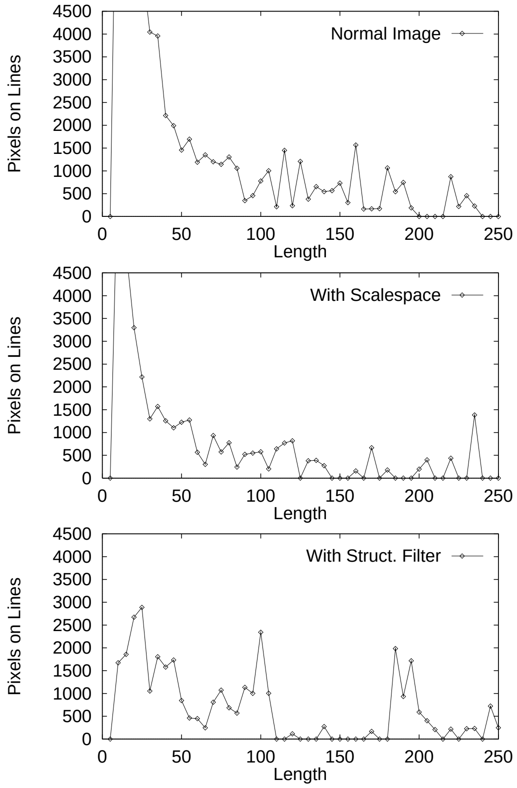

Figure 7: (a-c) Segment histograms of door scene with different types f adaptation. (a) No adaptation (b) Scale space adaptation (c) Additional anisotropic edge detector

high contrast structure. The threshold is initialized to a value which has proven to be successful in the past. Now the system is exposed to a strong change in contrast (by switching on all lights of the room). While the automatic gain control of the cameras compensates for some of the change the number of detected segments rises dramatically.

Figure 8a shows the segments obtained after switching the light on. The large number of detected segments does not only make identification of the desired ones extremely difficult but the large number of gradient pixels creates many T-forks and causes even segments of high-contrast segments to break. The robot was not able to identify the door in this situation. The results of the experiment with the threshold reflex beeing active is shown in figure 8b. The gradient threshold is adjusted automatically to a value that only high contrast segments remain and identification of the door segments if now easily possible.

A diagram of the dynamic behavior of the reflex is shown in figure 9. The upper diagram shows the value

Figure 8: (a-b) Segments of door scene after a change of contrast (a) without threshold reflex (b) with threshold reflex.

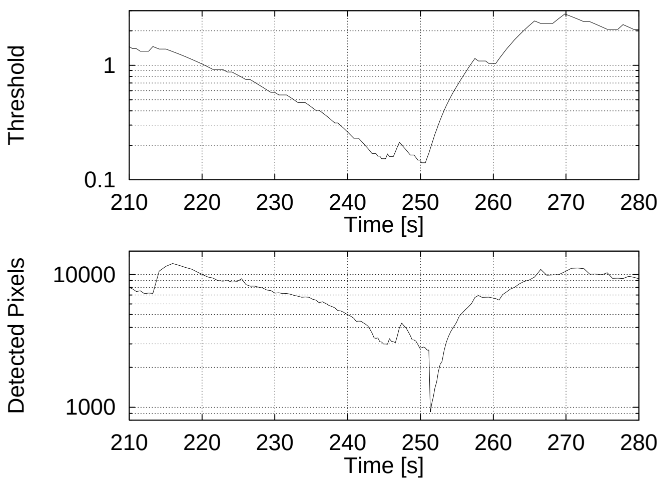

Figure 9: (a-b) Reflexive control of the detection threshold. (a) Threshold Value (b) Number of detected pixels

of the sensitivity (the inverse threshold) and the lower one the total number of detected pixels.

At the beginning of the experiment the room is relatively dark. The reflex has adjusted the sensitivity to a value around 2.0. At t=214 the light is switched on. The number of detected pixels increases by a factor of about 1.5. As it is now much easier to detect the door the sensitivity is slowly lowered to 0.15, It is interesting to note that we now only need to detect about half as many segments in order to find the same number of long segments. At t=252 the light is switched off. The number of found pixels drops and is slowly readjusted a value similar to the one at the beginning of the experiment.

The cycle time of the reflex varies between 1.0 (after switching the light on) and 0.15 (after switching it off). In our Experiments reaction of the reflexive control was damped as faster response proved to be unstable. In normal robot operation drastic changes in contrast do not happen frequently. While it is important that the robot is able to continue operation when they occur quick reaction is less important. The reaction time of the reflex is between 10 and 30 seconds. This is similar to the time humans need to adapt to significantly different lightning conditions.

5 Conclusion and Future Work

We have presented a general control architecture that automatically adjusts the parameters of a low-level vision system to a priori knowledge about the observed scene and the results obtained. The architecture has been implemented on a mobile robot and quantitatively evaluated. It has proven to make the feature extraction process, and thus the overall performance of the robot, more robust and faster. We have demonstrated the successful use of the reflexive control of low-level vision parameters with cycle times of less than 0.2 seconds and how this gives the robot the ability to use his vision sensor effectively in dynamic environments.

The use of the control architecture of low-level vision parameters is not limited to mobile robotics. The performance of any vision application that uses segments and has some knowledge about its environment the architecture can be used. In static environments or environments of which complete model is available (e.g. industrial robotics, surveillance) even better performance can be expected.

Several extensions of the work are currently under consideration. Currently the settings for the smoothing, the anisotropic edge detectors and the expected segment lengths are contained in the task database. A better method would be to calculate them directly from the geometric model of the object. This however requires a fast model of the effect of these parameters on the obtained images.

The reaction times for the dynamic control of the gradient threshold could be enhanced by a better control strategy. Additionally the system should remember the best threshold for a certain location and object to enhance the initial setting.

Acknowledgments

The work was performed at the Institute for Real-Time Computer Systems and Robotics, Prof. Dr.-Ing. U. Rembold and Prof. Dr.-Ing. R. Dillmann, University of Karlsruhe, 76128 Karlsruhe, Germany and at the LIFIA, Prof. James L. Crowley, Institut IMAG, 38031, Grenoble Cedex, France. The authors would like to thank Bernt Schiele for valuable discussions on the subject.

References

[1] J. Crowley and H. I. Christensen, eds., *Vision as Process.* Springer Verlag, Heidelberg, 1994.

[2] R. Dillmann, J. Kreuziger, and F. Wallner, “The control architecture of the mobile system priamos,” in *Proc. of the 1st IFAC International Workshop on Intelligent Autonomous Vehicles, Southampton*, 1993.

[3] P. Weckesser, F. Wallner, and R. Dillmann, “Position correction of a mobile robot using predictive vision,” in *International Conference on Intelligent Autonomous Systems, IAS'95* (U. Rembolt and R. Dillmann, eds.), Apr. 1995.

[4] O. Faugeras, *Three-Dimensional Computer Vision, a Geometric Viewpoint.* MIT Artificial Intelligence, The MIT Press, 1993.

[5] J. Canny, “A computational approach to edge detection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.

[6] A. Witkin, “Scale space filtering,” *Proc. Int. Joint Conf. Artificial Intell., held at Karsruhe, West Germany, 1983, published by Morgan-Kaufmann, Palo Alto, California*, 1983.

[7] V. Ramesh and R. M. Haralick, “Random perturbation models and performance charcterization in computer vision,” in *Computer Vision and Pattern Recognition*, pp. 521–527, 1992.

[8] G. Appenzeller and J. L. Crowley, “Experimental evaluation of vision systems and automatic parameter control,” in *Asian Conference on Computer Vision*, Dec. 1995.