Closed-Loop Control in Image Processing for Improvement of Object Recognition

Sorin M. Grigorescu, Danijela Ristić-Durrant, Sai K. Vuppala, Axel Gräser

Institute of Automation, University of Bremen, Otto-Hahn-Allee NW1, 28359 Bremen, Germany (Tel: +49-421-218-3490 / 7523; e-mail: grigorescu, ristic, vuppala, ag @iat.uni-bremen.de)

Abstract: This paper presents the employment of a novel idea of inclusion of feedback control at different image processing levels to improve the robustness of object recognition. The proposed approach is intended to cope with object image color uncertainty that arises from changes in the illumination conditions during image acquisition. The focus is on the control of quality of binary segmented image, which represents the input to higher processing levels, feature extraction and recognition, of the image processing chain. The main idea behind this is that introduced closed-loops drive the current segmented image to the image of the desired, reference, quality so that higher processing levels are provided with reliable input image data. In this way, the reliability and robustness against external influences of the overall object recognition system is improved. The specifics and benefit of closed-loop control in image processing are considered throughout the presentation of the closed-loop object recognition in the robotic system FRIEND II. The benefit of the closed-loop image segmentation in service robotics is considered through the demonstration of results achieved for recognition of the object of interest from a working scenario of the system FRIEND II.

1. INTRODUCTION

A majority of image processing applications concern object recognition and typically consist of sequentially arranged processing steps: preprocessing, image segmentation, feature extraction and recognition. In this sequentially arrangement, the reliability of the higher processing levels is strongly dependent on the correctness of the low-processing levels. Nonetheless, in a traditional open-loop image processing systems the parameters of high-level algorithms are commonly designed under the assumption that the inputs are of good quality. However, in real-world applications, numerous external influences including illumination conditions, imaging system and imaged objects characteristics and processing parameters, often lead to the low-processing level results of poor quality. The progress through the image processing sequence of an uncertainty introduced by a poor quality result may cause unreliability of the higher processing steps, regardless of how well designed they are. Furthermore, due to the absence of feedback between the higher and lower levels, the low-level processing is performed regardless of the requirements of the subsequent steps, which leads to low-robustness against the external influences of the overall open-loop system.

The improvement of robustness of image processing is an emerging field of research, which has been growing for the past two decades (Deshmukh [2005]). However, in spite of the number of published papers, there is still no general method for determining the processing parameters at different processing levels which will lead to reliable object recognition results in the presence of the numerous external influences that can arise in real-world applications. In contrast to the majority of robust image processing methods, which are based on the adaptation of the processing parameters to different vision conditions in feedforward actions (Abutaleb [1989]), in this paper the use of feedback information on processing result to adjust the processing parameters is considered. The main idea behind this is to change the processing parameters in a closed-loop manner so that the control error, which is defined as the difference between a reference value and the current image processing result, is driven to zero. The motivation comes from the fact that feedback control systems have a natural ability to provide robustness against system uncertainty, which is a fundamental concept in control theory.

The employment of control techniques in image processing makes it a new control application field. This raises a lot of questions regarding the specific features of an image processing system from a control perspective, choice of the actuator and controlled variables, control action design, etc. These questions are considered in detail and their answers are addressed both generally and within the context of specific gray level image processing applications in (Ristić [2007]). In this paper, the idea of inclusion of feedback control for improvement of image processing is extended to the color object recognition in service robotic applications. Service robotic systems such as the system FRIEND II (*Functional Robot arm with frIENdly interface for Disabled people*), which has been developing at the Institute of Automation of the University of Bremen (Volosyak [2005]), are intended to support the user in daily life activities. Because of this, the object recognition must be robust enough to work effectively in different lighting conditions that arise during the day. In (Vuppala [2007]), first results of the improvement of robustness of the object

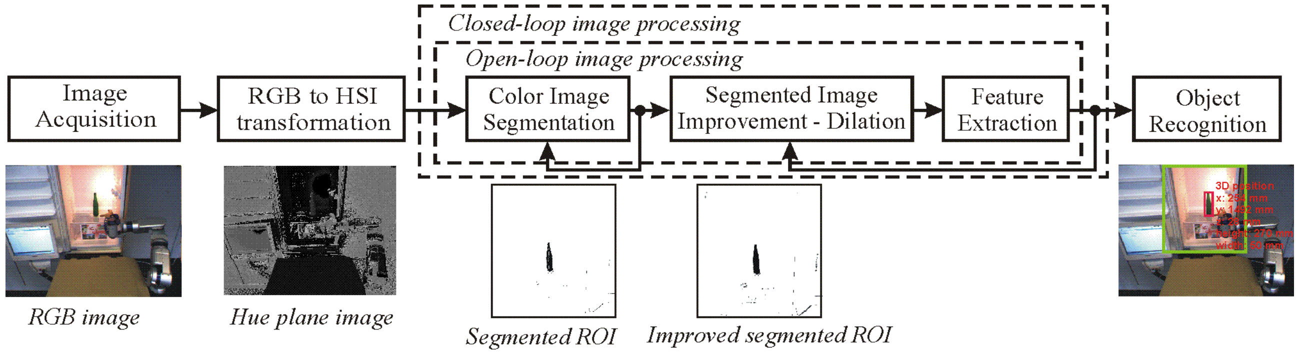

Fig. 1. Open-loop and closed-loop object recognition in the robotic system FRIEND II.

recognition, implemented in the system FRIEND II, by inclusion of closed-loop image segmentation were described. The main idea behind the closed-loop segmentation is that the closed-loop drives the current segmented image to the image of the desired, reference, quality so that higher processing levels, feature extraction and recognition, are provided with reliable input image data. In this paper, further improvements of the closed-loop object recognition in the robotic system FRIEND II obtained by using more appropriate "actuator variable-controlled variable" pairs and by processing the image *Region of Interest* (ROI) rather than whole image, are presented.

The paper is organized as follows. The Machine Vision Framework within the FRIEND II and the open-loop color based object recognition are presented in Section 2. The design and implementation of two closed-loops aiming at control of segmented image quality are given in Section 3. The performance evaluation of the proposed closed-loop image segmentation method is presented in section 4.

2. THE VISION FRAMEWORK OF THE ROBOTIC SYSTEM FRIEND II

2.1 System architecture

The organization of the FRIEND II robotic system is built using a CORBA (*Common Object Request Broker Architecture*) based software architecture for interconnecting the various processing layers, e.g. task planning, machine vision, motion planning, Human Machine Interface (HMI) etc. (Prenzel [2007]). Within this architecture an important role plays the Machine Vision Framework. It has to provide reliable visual information to be used by the task planning module of the FRIEND II system for controlling the manipulation of objects in the working environment. The vision hardware consists of a Bumblebee Stereo Camera system (PointGrey [2007]) connected to the computer dedicated to running the image processing algorithms. The stereo camera system is mounted on a frame-rack behind the user, above its head, and views the scene in front of the robotic system including the manipulator and the tray which is mounted on the wheelchair in front of the user. The acquired Bumblebee Cameras images are represented using the RGB (*red, green, blue*) color model.

Due to the small focal length of the used cameras, which implies a large field of view, not only objects of interest are in the image. To cope with this problem the Machine Vision Framework benefits from the FRIEND II control architecture by getting the information on the image ROI. Because the location of the object of interest in a particular scenario is approximatly known, e.g. "object is in the fridge" or "object is in the microway", the vision algorithm is applied on the ROI determined either using information on 3D position of the container (e.g. fridge) or using information from the HMI. In the first case, a marker based object recognition for 3D localization of the container is used. Then, by mapping from 3D to image plane and using a priori known dimensions of the container, the image ROI is determined. In the second case, the user of the FRIEND II system uses the HMI to select the ROI in the image (Prenzel [2007]).

2.2 Open-loop object recognition

The starting point for the improvement of the image processing algorithms is the understanding of the conventional open-loop method used for object recognition and the problems encountered in it. The open-loop sequence of operations, in the object recognition system of FRIEND II, is presented in Fig. 1. The images in Fig. 1 are from the "beverage serving" scenario aiming at serving the user with a drink from a bottle located in the fridge. In this scenario, the task of the manipulator is to fetch the bottle from the fridge. For this, the vision system has to reliably extract the 3D position of the object of interest, the bottle, so that robust autonomous manipulator action is provided. 3D object localization is strongly dependent on the result of object recognition, that is on the result of the feature extraction and object detection in the image ROI.

The object recognition method used is based on color information and uses the HSI (*hue, saturation, intensity*) color model. The reason for using this color model is that it stores the color information in only one gray level plane, the *hue* plane. The acquired RGB images from the stereo cameras are converted at the preprocessing stage into HSI images which are further feed to the image segmentation module. The image segmentation module consists of two parts: the *color segmentation* block and the *segmented binary image improvement* block.

The color segmentation algorithm thresholds the hue plane of the HSI color space. The color of an object is represented by pixels values from a particular interval in the hue image. According to this information the hue image is thresholded to foreground (black pixels 1) and background (white pixels 0). For further explanation of the algorithm let us define the hue image as a function $f(x, y)$ which can take values in the interval $[0, 255]$ and the color of an object as a subinterval $C_l \in [0, 255]$. The thresholding is:

$$t(x, y) = \begin{cases} 1, & \text{if } f(x, y) \in C_l \\ 0, & \text{if } f(x, y) \notin C_l \end{cases} \quad (1)$$

where $f(x, y)$ is the pixel value at hue image coordinates (x, y). For the sake of clarity an object color class C_l in the following is referred to as the *object thresholding interval* $[T_{low}, T_{high}]$. T_{low} and T_{high} are the low and high thresholding boundaries. The reference object thresholding interval is determined by off-line manual thresholding of the so-called *reference image* aiming at extracting as much object pixels as possible. The reference image is the image of the FRIEND II working environment taken at a specific artificial illumination.

Due to different reflection and shadows durring image acquisition it happens that not all objects pixels are segmented as foreground pixels even though the uniformly colored object is thresholded with the reference thresholding interval. Bearing in mind that and the definition of a good segmented image as one which contains "full" and well shaped segmented object region, it turns out that the binary segmented image $t(x, y)$ has to be improved so to obtain the "full", compact, object region. In the presented system, this improvement is achieved using morphological dilation. The dilation operation increases the area of foreground pixels while covering the "holes" in the segmented regions. The dilation operator takes two inputs. One is the binary image to be dilated and the other is the so-called structuring element. The structuring element is nothing but a matrix consisting of 0's and 1's. The distribution of 1's determines the shape of the structuring element and the size of the pixel neighborhood that is considered during the image dilation. The structuring element is shifted over the image and at each image pixel its elements are compared with the set of the underlying pixels according to some predefined operator. As a result, basically, a white background pixel turns to black foreground pixel if there are black pixels in its neighborhood that are covered by the 1's of the structuring element. The effect of "filling" the segmented regions by dilation strongly depends on shape and size of the structuring element as well as on the number of performed dilations.

Binary improved image is given further to the feature extraction module. The feature extraction module extracts the features of segmented regions of connected pixels needed for further object classifying and recognition. Firstly, the boundary of segmented regions are detected and descriptors of the segmented foreground pixels inside the boundary are calculated. In the presented system Hu moments are used. Hu moments are invariant coefficients proposed by (Hu [1962]) which are derived from moments of the image region. In case of a digital image intensity function $f(x, y)$, the moment of order $(p + q)$ is:

$$m_{pq} = \sum_x \sum_y x^p y^q f(x, y) \quad (2)$$

where x and y are pixel coordinates in the considered image region. The central moments μ_{pq} are defined as:

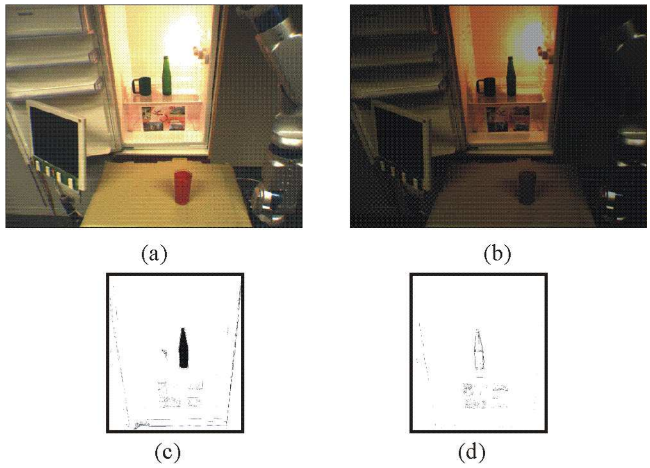

Fig. 2. Segmentation of the green bottle object using a reference object thresholding interval. Artificial (a) and daylight (b) illuminated scene. (c) and (d) corresponding segmented ROIs.

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q f(x, y) \quad (3)$$

where $p, q = 1, 2, 3, ...,$ $\bar{x} = \frac{m_{10}}{m_{00}}$, $\bar{y} = \frac{m_{01}}{m_{00}}$. In case of a binary image, $f(x, y) = t(x, y)$ defined by (1). In the presented system, for object recognition, two Hu moments are used:

$$I_1 = \eta_{20} + \eta_{02}, \; I_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \quad (4)$$

where η_{pq} is the normalized central moment:

$$\eta_{pq} = \mu_{pq} \cdot \mu_{00}^{-1-\frac{p+q}{2}}. \quad (5)$$

Moments (4) are calculated for each segmented region of connected pixels and are combined in the Euclidean distance:

$$d_r = \sqrt{(I_{r1} - I_1)^2 + (I_{r2} - I_2)^2} \quad (6)$$

where I_i and I_{ri}, $i = 1, 2$, are, respectively, measured moments of segmented region and reference Hu moments of the object of interest. The reference Hu moments are calculated from the so-called *ground truth image* which is obtained off-line by manually thresholding and dilation of the reference image until "full" compact weel shaped object region is obtained. If d_r is smaller than a predefined value than the segmented region is recognized as the object of interest.

As seen from equations (2)-(5), Hu moments take into account all segmented pixels of a region. Therefore, in order to get a reliable recognition result for the object of interest it is of crucial interest to have a good input segmented image to the feature extraction module. However, due to external influences the open-loop image processing system can not provide good segmented image in all working conditions. As said before, in the open-loop object recognition method the object thresholding interval $[T_{low}, T_{high}]$ is determined off-line from the reference image. This implies that this reference thresholding interval

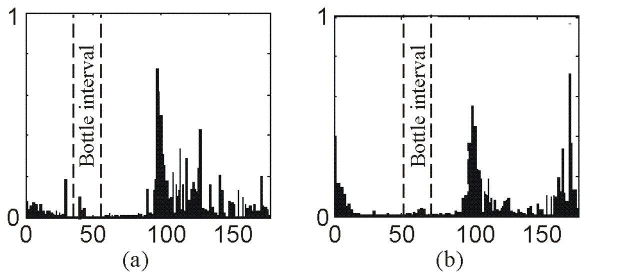

Fig. 3. Histograms of the hue planes of the images in Fig. 2(a) and 2(b) respectively, overlaid with the object threshold interval.

is applied on-line to all the acquired images. Because of variable illumination conditions, a constant thresholding interval will give reliable results only when the light resembles the one present during the off-line acquistion of the original reference image. This phenomenon can be seen in Fig. 2 which shows the segmentation result for two images of the same scene acquired in different illumination conditions. The segmentation was done using the reference thresholding interval for the green bottle object. Hence, in order to properly segment the object of interest the thresholding interval must be adjusted to the current illumination conditions. Manual tuning of the thresholding and, further, dilation parameters is time-consuming and rather meaningless for autonomous robot system. In the next section, a method for automatic parameters tuning is proposed.

3. CLOSED-LOOP EXTENSION OF IMAGE PROCESSING FOR OBJECT RECOGNITION

The above presented open-loop recognition system is extended with two closed-loops. As shown in Fig. 1, the first closed-loop is realized as the feedback between the quality of segmented image ROI and the segmentation, i.e. thresholding, parameter and the second one is the feedback between the feature extraction result and a dilation parameter.

3.1 Thresholding closed-loop

For choosing the actuator and controlled variable in the first closed-loop the characteristics of both the hue plane and the corresponding binary segmented image are to be investigated. The histograms of the hue planes of the images in Fig. 2(a) and 2(b), representing the same scene imaged in different illumination, are shown in Fig. 3. As it can be seen the thresholding object interval in the image taken in daylight conditions is shifted to the right with respect to the thresholding interval of the same object in the image captured in artificial illumination conditions. Bearing in mind that the actuator variable should be one that influences the segmented image quality, it turns out that the *increment* to be applied to the low and high thresholding boundaries of the object thresholding interval is a good choice of the actuator variable.

For the choice of the controlled variable to be used together with the chosen actuator variable, the segmented image quality has to be analyzed. From Fig. 2(c) and 2(d) it can be seen that a measure of image quality is linked to the

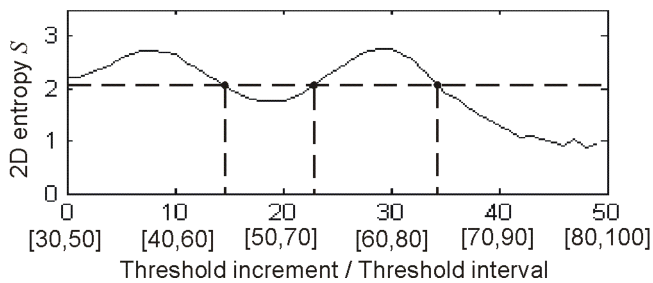

Fig. 4. 2D entropy S of segmented pixels vs. threshold increment, i.e. thresholding interval.

number and spatial distribution of foreground black pixels. Because a good segmentation implies foreground pixels to form a compact, well shaped segmented object region, beside the number of segmented pixels a measure of their connectivity is a natural candidate for the controlled variable. The used connectivity measure, introduced by Ristić [2006], is the so-called two-dimensional (2D) entropy of the segmented pixels defined as:

$$S = -\sum_{i=0}^{8} p_{(1,i)} log_2 p_{(1,i)} \tag{7}$$

where $p_{(1,i)}$ is the relative frequency, that is, the estimate of the probability of occurence of a pair $(1,i)$ representing the black segmented pixel 1 surrounded with i black pixels in its 8-pixel neighborhood.

The entropy S can be also considered as a measure of disorder in a binary segmented image since, as demonstrated in (Ristić [2007]), *the higher the 2D entropy S, the larger the disorder (noise, breaks) in a binary image is.*

In order to investigate the input-output controlability when considering variable thresholding interval as the input and the entropy S as the output, the thresholding of the ROI in the image shown in Fig. 2(a) was done. Having in mind that the object color interval is known *a priori* it is meaningful to apply the control algorithm only on a subinterval of the hue plane pixel values. Through off-line experiments with a number of images acquired in different illumination conditions, it was found that the shades of green can reside in the subinterval $[30, 100]$. Hence, the thresholding interval was set to an initial state $[T_{low}, T_{high}] = [30, 50]$. To this interval the increment $u = i$ was added as $[T_{low} + i, T_{high} + i]$. For each segmented image corresponding to the increment $i \in [0, 50]$, the 2D entropy S of the ROI was calculated. The resulting characteristic is presented in Fig. 4. As it can be seen, the entropy S is sensitive to the chosen actuator variable across its effective operating range. Also, it is clear that each input value is mapped to at most one output value and that it is possible to achieve the minimum of S by changing the thresholding boundaries. The satisfaction of these prerequisites for successfull control action to be performed demonstrates the pair "threshold increment - 2D entropy S" as a good "actuator variable - controlled variable" pair.

Bearing in mind the available reference value r of the entropy S determined off-line from the ground truth image,

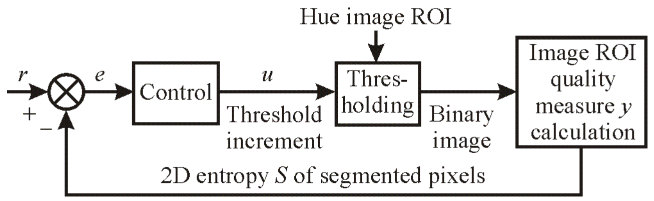

Fig. 5. Block diagram of the thresholding closed-loop.

the inclusion of classical error-based control algorithm is suggested. The block diagram of the closed-loop thresholding is shown in Fig. 5. The discrete PI controller is implemented in the following velocity form:

$$\Delta u(k) = K_P\left(\Delta e(k) + \frac{1}{T_I}e(k)\right) \tag{8}$$

where k is the discrete time, $e(k)$ is the control error($e(k) = r - y(k)$), $y(k)$ is the controlled variable (2D entropy S) and $u(k)$ is the actuator variable (the thresholding boundaries increment). The main idea behind the implemented closed-loop is that properly tuned K_P and $K_I = \frac{K_P}{T_I}$ gains gives thresholding interval which provides that the 2D entropy S is driven to a given reference value r, which corresponds to good segmentation meaning extraction of as much object pixels as possible.

Due to the nonlinear characteristic in Fig. 4 one problem encountered is that the reference value of S can be achieved for more than one input value. As shown in Fig. 4, the reference $r = 2.05$ is achieved for three threshold intervals. For two of them, $[51, 71]$ and $[66, 86]$, the resulted binary image ROIs are shown in Fig. 6(a) and 6(b). Evidently, only one thresholding interval yields the result corresponding to the good segmentation. This happens because the entropy S deals with the relative frequency of pair $(1, i)$. As seen in Fig. 6(c) and 6(d), the appeard relative frequencies can be the same but the distribution of pair $(1, i)$ differs. To overcome this problem, the probability distribution of pair $(1, i)$ is checked after the measured output reaches the reference value. If the relative frequency of the pair $(1, 8)$ is the largest the result is accepted as good since it means the majority of segmented pixels surounded with 8 segmented neighborhod pixels and so full segmented region. The largest value of some other pair, like $(1, 2)$ in Fig. 6(d), means that the segmented object region is broken. In such case the controller is restarted with different initial conditions, $u(0) \neq 0$, and the control action is repeated until desired segmentation result is achieved.

Using the proposed closed-loop the result of the thresholding process of an image from the FRIEND II scenarios is driven automatically to the desired reference result independent on the illumination conditions.

3.2 Dilation closed-loop

Although the segmented image ROI resulted from the thresholding closed-loop is of good quality, as explained in subsection 2.2, to obtain a "full" and well shaped object region, the segmented image ROI has to be improved. The purpose of the improvement is to "fill" the holes still present in the segmented ROI. In the presented system, this is done by the included dilation closed-loop. The

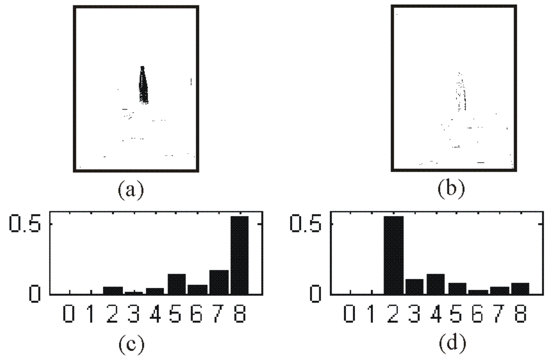

Fig. 6. Binary segmented image ROIs corresponding to different thresholding intervals. (a) good and (b) bad segmentation result. (c) and (d) corresponding normalized histograms of distribution of pair $(1, i)$.

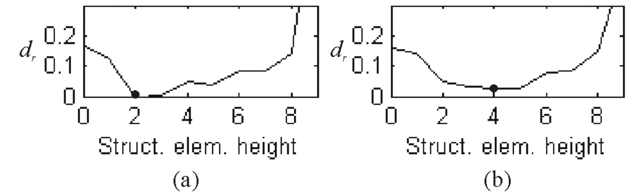

Fig. 7. Euclidean distance d_r vs. height of the dilation structuring element in the case of reference (a) and alternative (b) image from the FRIEND II environment.

actuator (input) variable in this closed-loop is the height of the dilation structuring element. The controlled (output) variable is the shape of the segmented object of interest expressed by the Hu moments (4), i.e. by the Euclidean distance (6). Bearing in mind that the Euclidean distance d_r measures the closeness of the segmented object Hu moments to their reference values, it turns out that the desired value of d_r is equal to zero. In order to investigate the input-output controllability of the dilation process, the input-output characteristic shown in Fig. 7 is considered. As it can be seen, it is possible to achieve the global minimum of d_r when changing the input variable across its operating range. In real-world applications, when dilating the segmented image corresponding to the image different from the reference one, the global minimum of d_r is not equal but it is very closed to zero as shown in Fig. 8(b). Due to the input-output characteristic having global minimum the control action based on extremum searching algorithm (e.g. hill-climbing) is suggested, as shown in Fig. 8.

4. PERFORMANCE EVALUATION

In order to evaluate the effectiveness of the proposed closed-loop segmentation its performance is compared with the performances of two segmentation methods: a traditional open-loop segmentation consisting of thresholding and dilation steps, and closed-loop segmentation presented in (Vuppala [2007]). In contrast to the closed-loop methods, which use feedback information on the processing results to adjust the processing parameters at particular processing level, the open-loop method uses constant reference parameters of both thresholding and

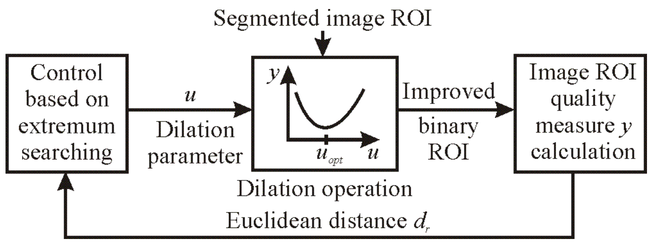

Fig. 8. Block diagram of the dilation closed-loop.

dilation operation. These parameters are determined off-line, as discussed above, by manual thresholding and dilation of the reference image. The novelty in the presented closed-loop segmentation with respect to one presented in (Vuppala [2007]) is processing the image ROI rather than the whole image and consequently using the measure of quality of ROI rather than of whole image as feedback variable. Also, the novelty is using the feedback information on feature extraction result to adjust the dilation parameter for improvement of binary image quality and consequently of feature extraction.

In order to evaluate the performances of the considered segmentation methods, the Euclidean distance (6) was used as performance criterion. A set of images of the FRIEND II environment in the "beverage serving" scenario were taken in different instances of time during the day. Illumination condition during the image acquisition was ranging from the bright artificial lighting to the dark day lighting. Each captured image was segmented using the three tested segmentation methods. For each segmented image the distance measure (6) was calculated after the extracting Hu moments as relevant features of the segmented bottle object region. The results are shown in Fig. 9. As it can be seen, the Euclidian distance calculated from segmented images obtained by open-loop as well as by closed-loop segmentation of bright images is almost equal to desired zero value. This means that all three considered segmentation methods give good segmentation result for the images captured in lighting conditions similar to the reference ones. This is an expected result even for the open-loop method since the used constant processing parameters are determined off-line by manual segmentation of the reference image. However, the performance of the open-loop segmentation, in contrast to closed-loop methods, degrades significantly with the changing of the illumination conditions. Evidently, between the two closed-loop methods, the one presented in this paper is of notably better performance than the method presented in (Vuppala [2007]). This indicate the importance of using directly the feedback information on feature extraction result to adapt the dilation parameter for improvement of feature extraction reliability rather than using the measure of dilated segmented image quality as feedback variable. This confirms the need of interaction between the higher and lower processing levels in an image processing chain. However, as evident from Fig. 9, even the proposed closed-loop method gave bad object segmentation result in images captured in very dark illumination condition. But, this bad result can be considered irrelevant for the robustness evaluation of the proposed method. Namely, applications of the rehabilitation robotic system FRIEND II are considered to be indoor. For that reason the condition of dark

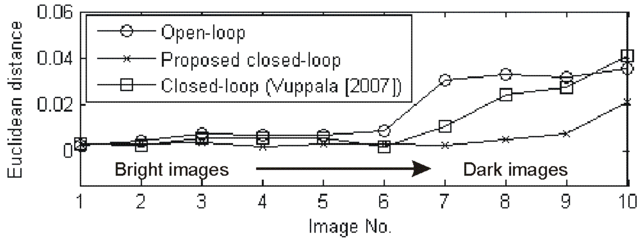

Fig. 9. Performance measure of the object segmentation in images captured in different illumination conditions ranging from bright to dark.

illumination can be avoided since the system FRIEND II operates always either in the very bright daily light conditions or in bright artificially light conditions.

5. CONCLUSIONS AND OUTLOOK

This paper consideres the benefit of the use of control techniques in image processing, representing a new control application field. The usefulness of the closed-loop image segmentation is demonstrated through the results on object recognition in the robotic system FRIEND II. Presented results on comparison of the performance of proposed closed-loop method with the traditional open-loop object recognition demonstrate the need for the feedback adaptation of processing parameters to different external influences. Also, the importance of the choice of an appropriate "actuator variable-controlled variable" pair when implementing closed-loop image processing for improvement of object recognition is demonstrated.

REFERENCES

A. S. Abutaleb. Automatic thresholding of gray-level pictures using two-dimensional entropy. *Computer Vision, Graphics and Image Processing*, 47/1, pp. 22–32, 1989.

Bumblebe digital stereo vision camera. *Point Grey Research Co.*, Vancouver, Canada, Feb. 2005.

K.S. Deshmukh and G.N. Shinde. An adaptive color image segmentation. *Electronic Letters on Computer Vision and Image Analisys*, 5/4, pp. 12–23, 2005.

M.K. Hu. Visual pattern recognition by moment invariants. *IRE Trans. on Information Theory*, pp. 179–187, 1962.

O. Prenzel, C. Martens, M. Cyriacks, C. Wang and A. Gräser. System controlled user interaction within the service robotic control arhitecture MASSiVE. *Robotica, Special Issue*, 2007.

D. Ristić. Feedback structures in image processing. *PhD. thesis*, to be published, 2007.

D. Ristić and A. Gräser. Performance measure as feedback variable in image processing. *EURASIP Journal on Applied Signal Processing*, volume 2006, pages 61–70, 2006.

I. Volosyak, O. Ivlev and A. Gräser. Rehabilitation robot FRIEND II - the general concept and current implementation. *in Proc. 9th ICORR*, pp. 540–544, Chicago, 2005.

S.K. Vuppala, S.M. Grigorescu, D. Ristić and A. Gräser. Robust Color Object Recognition for a Service Robotic Task in the System FRIEND II. *in Proc. 10th ICORR*, pp. 704–713, Netherlands, 2007.