The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

Camera Auto–Exposing and Auto–Focusing for Edge–Related Applications using a Particle Filter

Thuy Tuong Nguyen and Jae Wook Jeon, *Member, IEEE*

***Abstract*— The use of edge–related applications is important in the field of computer vision. These applications help robots with understanding their surrounding environments; the lane or wall detection system is one of the most popular applications. Numerous studies have recently been conducted for enhancing the capabilities of robotic vision, but they typically lacked the applications that were related to coping with the environmental changes of the scenes. In this paper, we propose a method that integrates a particle filter into the process of tracking the camera's parameters (the exposure and the focus) to find the captured frame with the high edge quality. The relationship between the current sequence of frames and the previous sequence was given no consideration when all the possible parameters were scanned. Our work attempts to find that relationship and to increase the speed of the camera system. The edge results are evaluated with using a line detection algorithm – that is known as the Standard Hough Transform. A test method is applied to analyze the correctness of the line detection results. Furthermore, we propose the entropy of the Sobel gradient method for measuring the image sharpness and its contrast when the exposure and focus of a digital camera are changed. Our experimental results show that our method can be applied in real–time systems because of its low computational requirements.**

I. Introduction

Acquiring well edge images from a camera under various environmental conditions is necessary for further processing, because the images are then used as input of many of the computer visual applications. Research has been conducted on the automatic adjustment of camera exposure and focus, and image contrast. The work in [1] presents an approach that is based on a multi–layer feed–forward network with a back–propagation learning rule for automatic illumination correction. This can be used for scene enhancement and for improving object tracking. The neural network learns about the mapping between two illumination conditions, which are the unknown and the canonical illuminations. The neural network then applies this transformation to every pixel in the successive frames that were captured by the camera. This work provides the experimental results of hand tracking by using a condensation tracker, and a contrast transformation that is applied for color correction.

This research was performed as part of the Intelligent Robotics Development Program, one of the 21st Century Frontier Research and Development Programs funded by the Ministry of Commerce, Industry and Energy of Korea.

Thuy Tuong Nguyen is with School of Information and Communication Engineering, Sungkyunkwan University, Suwon, Korea (email: ntthuy@skku.edu).

Jae Wook Jeon is with School of Information and Communication Engineering, Sungkyunkwan University, Suwon, Korea (email: jwjeon@yurim.skku.ac.kr).

A system for traffic–sign detection and classification is shown in [2]. The work is divided into three stages: first, the detection of the information about the edges of the image by using the Hough Transform; second, the traffic-sign classification by using a neural network; and third, tracking by using a Kalman filter. The information about the edge–images in the first stage is obtained with using the Canny operator that is applied in a dynamic method that uses two thresholds. This method depends on the histogram distribution of an image. The threshold levels are taken over the region with a wider distribution. Consequently, it is possible to use the same algorithm under good visibility conditions or under poor illumination.

The aforementioned two works can mainly process single images or a sequence of frames. They merely consider the data from the images without paying attention to the parameters of the camera, even though they used camera systems to track hand gesture and to monitor traffic signs. The latter two works offer robust methods, but they may fail under particular or extreme illumination conditions such as over–exposure and under–exposure. If the scenes have regions that are too bright or too dark, then the features in these scenes cannot be correctly detected. There have been recent studies on coping with extreme illumination conditions. The idea proposed in [3] is for improving the quality of the acquired images with non–optimal exposure. This approach analyses the CCD/CMOS sensor Bayer data or the corresponding color–generated image. It adjusts the exposure level according to a camera–response–like function after identifying specific features. The techniques of exposure adjustment that are described in this paper are primarily designed for mobile sensor applications. Similarly, [4] provides fast and accurate auto–exposure capabilities for the digital–still cameras. The algorithm in [4] begins with an electronics–centric auto–exposure approach. This adjusts the exposure on the scene to alter the brightness of the image to the appropriate level. The lighting condition of the scene is identified using the main object and the background. The algorithm calculates different weights for images' areas that contain the objects and the background until the brightness of the scene reaches the proper exposure.

An optimal statistical measure for camera focus and exposure is provided in [5]. Sharpness and contrast are the two components that express the image quality. They can be directly translated into camera focus and exposure. In [5], measures are developed to independently adjust the focus and the exposure independently. The statistical measures that are computed with a gray–level histogram are the mean, the

978-1-4244-6676-4/10/$25.00 ©2010 IEEE

standard deviation, the entropy, the percentage of pixels, and the absolute central moment (ACM). Experimental results show that the ACM is the best measure of the image quality; if the image has the best sharpness and contrast, then it will yield high value image quality. The ACM was examined and proven to be an excellent measure; however, there were only a few test patterns, and they were not sufficient for concluding that the ACM is a good measure.

Achieving adequate auto–focus (AF) depends on the design of a good focus measure. Investigations of the focus measures that are presented in [6] showed that it is very difficult to achieve an AF function with only one focus measure. The AF algorithm that is proposed in [6] contains two different stages of focus measures: the discrete variance FM_{var}, and the energy of Sobel gradient. The FM_{var} is used for finding the vicinity of the focus at the first stage, while the energy of Sobel gradient is used for obtaining the best focused images at the second stage.

A focusing algorithm that is based on non–uniform sampling is proposed in [7] and [8]. Incorrect focus results occur when the focusing window is too big or too small in a typical automatic–focusing system. The algorithm that is proposed in [7] and [8] adopts the non-uniform sampling and the threshold gradient to obtain higher resolution at the image center and to obtain a wider field of view. Non–uniform sampling is applied to the digital image at the first step. A derivative–based algorithm is then used as a measuring method for finding the best focused image.

Most of the related studies attempted to inspect all the obtained images or all the possible parameters. This process takes a long time to determine the best sharpness and contrast image; even the measuring method that is applied to image quality is simple and it requires low computational resources. This paper presents an auto–exposing and auto–focusing method with using a particle filter (PF) for the edge–related applications. The PF is integrated with the process of tracking camera parameters (the exposure and focus parameters) to find the captured frame with the high edge quality. In the case of scanning all possible parameters, the relationship is not considered between the current and previous sequences. Our work attempts to find that relationship and increase the speed of the camera system.

Edge quality is usually high when an image has good sharpness and contrast. The results of the edges that were detected from the high sharpness and high contrast image are evaluated through the Standard Hough Transform, which is a line detection algorithm. A test method is applied to analyze the correctness of the line detection results. This correctness information forms the basis for judging the quality of the edge. Furthermore, we proposes the entropy of Sobel gradient method to measure the image sharpness and contrast at the time when the exposure and focus of the digital camera are changed. Experimental results showed that our method can be applied in real-time systems because of its low computating requirements.

This paper is organized as follows. In the next section, the entropy of Sobel gradient method is presented. Section III briefly describes a particle filter that is used for tracking the exposure and focus levels. Section IV presents the image evaluation for sharpness and contrast. Section V addresses auto–exposing and auto–focusing with using a particle filter. Section VI presents our experimental results. This paper is drawn to a conclusion and future work in Section VII.

II. Entropy of Sobel Gradient

This section presents a method for measuring the image sharpness and contrast when the exposure and focus of a digital camera are changed. Statistical measures of gray–level distribution have been based on the mean, variance, or entropy [5], but they are not particularly useful, and they are inconsistent. Therefore, we did not directly apply any of these measures to the captured images.

A gradient method was employed to measure the image sharpness and its contrast with respect to the edge feature. The Sobel gradient has the advantage of its smoothing effect when it is applied to many spatial gradient operations. It tends to make the derivative operations less sensitive to noise. Let $I(x,y)$ be the original image, and $I_s(x,y)$ be the image that is convoluted with a Sobel window $s(x,y)$. We then have

$$I_s(x,y) \quad = \quad s(x,y) * I(x,y). \tag{1}$$

We measured the effect of the exposure and the focus parameter changes on images with respect to the edge feature. When acquiring a series of images at different exposure and focus levels, we picked the image with the best sharpness and contrast based on entropy of the Sobel gradient image $I_s(x,y)$. An intuitive understanding of information entropy relates to the amount of uncertainty about an event that is associated with a given probability distribution. Entropy can serve as a measure of ”disorder”. A higher entropy value of Sobel gradient denotes better image sharpness and contrast. According to [9], the entropy is defined as:

$$h(X) \quad = \quad \sum_{k=1}^{n} p(x_k) \log_2 \frac{1}{p(x_k)}, \tag{2}$$

where X is a discrete random variable with possible outcomes x_1, x_2,... x_n, and $p(x_k)$ is the probability of the outcome x_k. The outcome is a gray level in the gray scale image, and its probability is calculated by

$$p(x_k) \quad = \quad \frac{n_k}{n}, \tag{3}$$

where $k = 1, 2, ...L$, L is the total number of possible gray levels in the image, n is the total number of pixels, and n_k is the number of pixels that have the gray level x_k. Consequently, the entropy of Sobel gradient is defined as $h(I_s(x,y))$.

The entropy itself is inconsistent when it is used for measuring image sharpness and contrast, so we have to take the gradient of the image before calculating its entropy. This method is particularly useful in algorithms relating to detecting edges.

III. Particle Filter

Particle filters [10], [11] enable the robust tracking of moving objects in a cluttered environment. They are used for nonlinear and non–Gaussian problems that focus on the detection and tracking of moving objects. In our work, a particle filter is used as an algorithm to track the positions of the camera's intrinsic parameters, when we control them, in order to adapt to the changes of the environmental conditions. The parameters are on the exposure and focus levels.

In our system, the target state at time index k is defined as

$$\mathbf{x}_k \triangleq [x_k, y_k, \dot{x}_k, \dot{y}_k]^T. \tag{4}$$

Here, x_k and y_k are the positions of the camera exposure and focus levels. Components $\dot{x}_k$ and $\dot{y}_k$ are the velocity components in the x and y axes at time k. The components x_k^i and y_k^i of the ith sample $\mathbf{x}_k^i$, $i = 1, 2, ...N$, where N is the number of particles that are drawn from $p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)$ in the sampling importance resampling (SIR) filter, are generated as follows:

$$\begin{aligned} x_k^i &= x_{k-1}^i + \dot{x}_{k-1}^i + u_{k-1,x}^i \\ y_k^i &= y_{k-1}^i + \dot{y}_{k-1}^i + u_{k-1,y}^i \end{aligned} \tag{5}$$

Here, $u_{k-1,x}^i$ and $u_{k-1,y}^i$ are zero–mean Gaussian processes with standard deviations σ_x and σ_y, respectively.

The weight $\tilde{w}_k^i$ of sample $\mathbf{x}_k^i$ is calculated as

$$\tilde{w}_k^i = \frac{1}{\sqrt{2\pi}\sigma} exp\left(- \frac{1 - \gamma[p, q]}{2\sigma^2} \right), \tag{6}$$

where σ is the variance of a zero–mean Gaussian process, and

$$\gamma[p, q] = \sum_{u=1}^{m} \sqrt{p^u q^u} \tag{7}$$

is the Bhattacharyya coefficient of the sample histogram p and the target histogram q, and u is the bin index of the histogram. The similarity of the histogram of the sample to the target histogram (the appearance model) and the "motion" measurement of this sample are weighted using alpha blending to obtain an integrated similarity measure. The appearance model is updated by the histogram of the sample with the maximum weight [12].

The components $\dot{x}_k^i$ and $\dot{y}_k^i$ of sample $\mathbf{x}_k^i$ are drawn from $p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)$ as displacements from the previous position to the estimated position of a moving level, in accordance with the following expression:

$$\begin{aligned} \dot{x}_k^i &= (\overline{x}_k - \overline{x}_{k-1}) + 0.5(\overline{x}_k - \overline{x}_{k-1}) u_{k-1,\dot{x}}^i \\ \dot{y}_k^i &= (\overline{y}_k - \overline{y}_{k-1}) + 0.5(\overline{y}_k - \overline{y}_{k-1}) u_{k-1,\dot{y}}^i \end{aligned} \tag{8}$$

where $u_{k-1,\dot{x}}^i$ and $u_{k-1,\dot{y}}^i$ are zero–mean Gaussian processes with unit standard deviations. Components $\overline{x}_k$ and $\overline{y}_k$ are the means of position components, x_k and y_k, respectively, and they are determined as follows:

$$\begin{aligned} \overline{x}_k &= \textstyle\sum_{i=1}^{N} w_k^i x_k^i \\ \overline{y}_k &= \textstyle\sum_{i=1}^{N} w_k^i y_k^i \end{aligned} \tag{9}$$

where w_k^i is the normalized weight of sample $\mathbf{x}_k^i$.

IV. Image Evaluation for Sharpness and Contrast

In our work, the result of the edges that were detected from an image with high sharpness and contrast qualities was evaluated through a line detection algorithm. We utilized the line extraction algorithm to the edge–image which is applied the Canny edge detector. Many algorithms have been proposed for extracting the line features from 2–D images. There are two popular approaches: the Split–and–Merge algorithm [14] and the Hough Transform [15], [16]. The Split–and–Merge algorithm has several desirable properties; especially, its exploitation of the local structure, whereas the Hough Transform algorithm considers only exploiting the global structure. The Hough Transform is chosen in this paper because it is robust to noise, and it is successfully used with intensity images.

The basic Hough Transform that known as the Standard Hough Transform (SHT), has established itself as the default technique for the straight line Hough Transform evaluation. Its popularity comes from its robustness to noise and simple algorithmic implementation. In SHT, all the points (x, y) were calculated by using the following equation that is expressed in polar coordinates:

$$\rho = x \cos\theta + y \sin\theta, \tag{10}$$

to obtain ρ as θ changes successively in the parameter space. Having the values (ρ, θ), the SHT algorithm then looks for the accumulator cells into which the parameters fall. The SHT then increases the values of those cells. After some edge pixels considered to constitute a straight line have been accumulated, the distribution of the parameter space is shaped like a butterfly. The most likely straight line in the image domain is represented by the highest value in the parameter space. The present accumulator has peaks that may have values that are greater than the specified threshold. The simplest way to find these peaks is to compare the values of the peaks with the threshold value. We can draw straight lines from these peaks.

The quality of the detected edges is not intuitively evaluated, but it can be evaluated through the Standard Hough Transform algorithm, with which we can count the number of detected lines and the number of correct lines. The evaluation was executed on real–world images and their corresponding ground-truth data, which are considered as criteria. The two correctness measures are defined as follows:

- The number of the possibly detected lines by the algorithm ($N_{det.}$).
- The number of correct lines ($N_{cor.}$), which is an important parameter of correctness. In order to count the lines, we must decide if two close lines are near each other in an allowed range. We base this on the precision of ρ and θ. The condition to increase the total number of correct lines is

$$(|\rho_i - \rho_j| \leq \Delta\rho) \wedge (|\theta_i - \theta_j| \leq \Delta\theta), \tag{11}$$

Fig. 1. Sony EVI–D100 CCD video camera.

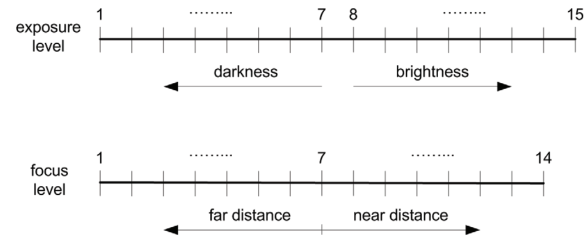


Fig. 2. A 1–D representation of the exposure and focus levels.

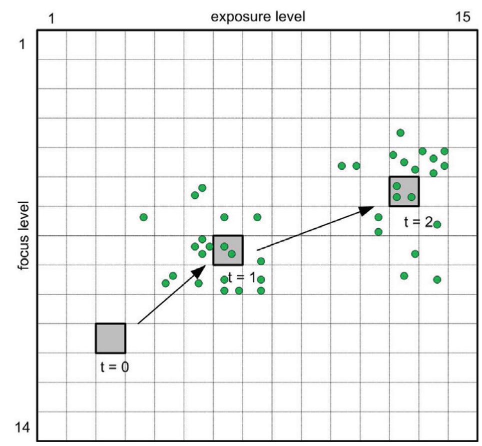


Fig. 3. A 2–D representation of the exposure and focus levels with the process of finding suitable frames by using particles.

where ρ_i, θ_i are ρ, θ of the ith detected line; ρ_j, θ_j are ρ, θ of the jth true line; and $\Delta\rho$, $\Delta\theta$ are the resolutions of $\rho\theta$–space.

V. Camera Auto–Exposing and Auto–Focusing

Our method works on gray level frames that are captured from the camera where the exposure parameter is continuously changed. We used a Sony EVI–D100 CCD video camera (see Fig. 1) and a frame grabber board. The EVI–D100 camera was connected to the frame grabber and to a PC through its RS–232C port, through which all settings of the camera were made. The gray scale that are images captured by the frame grabber board had a resolution of 640×480 pixels. This camera supports many useful commands such as panning, tilting, zooming, focusing, exposure compensating, etc. [13]. We adjusted only the aperture, exposure, and focus parameters. The aperture parameter was set to obtain the deepest depth of field (DOF). Instead of only focusing on an object or a shallow DOF, we attempted to detect the background and the other objects in the scene. The exposure compensation parameter includes 15 positions to change, and the focus parameter contains 14 levels. The lowest position of the exposure parameter represents darkness, and the highest position represents brightness. To obtain the best focus frame, the focus value must be the lowest when the object is farthest from the camera, and the focus value must be highest when the object is nearest to the camera. Fig. 2 shows 1–D representation for the exposure and focus levels. Furthermore, the camera also allowed us to control the backlight, which was used to increase the brightness. However, the backlight can over-expose some regions in the captured images, so we did not use it.

We have a total of 210 levels of exposure and focus parameters. As our initial step, all possible images were captured with different camera parameters. The entropy of Sobel gradient was applied to every image in a sequence of the captured images to find the best sharpness and contrast image with respect to the edge feature. The exposure and focus levels were saved as criteria for the second sequence of the captured frames. They were used as input to the particle filter (PF). The PF bases on these parameters to scatter particles with respect to the standard deviations. The processes of measurement and prediction in the PF algorithm were repeated over the sequences of the captured frames. The results of each image sequence were evaluated through edge detection. Fig. 3 demonstrates the initial step ($t = 0$) when applying the entropy of Sobel gradient, and the next two steps with the particles being scattered. At times $t = 1$ and $t = 2$, the particles (the solid small circles) were distributed to surround the position that was obtained from the previous step with different weights. The current exposure and focus levels were estimated based on the particles.

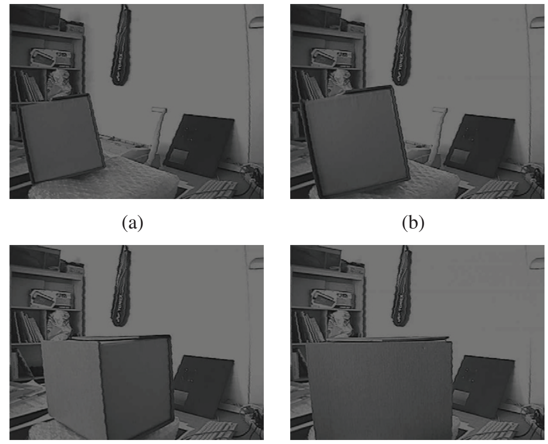
VI. Experimental Results

In this experiment, we used the Sony camera and the frame grabber board, as was described in the previous section. The PC was an AMD Athlon 64 X2 Dual Core Processor 3800+ 2.00 GHz with 2 GB RAM.

The experiment was run in an indoor environmental setting under normal light conditions. The frames were taken from a scene at a corner of our laboratory. Fig. 4 shows the frame that represents the scene of the experiment. For the purposes of this test, we put a desk lamp in front of the camera. One object is close to the camera, while several objects are far from it. All these things are used to simulate the environment where a mobile robot with a mounted camera moves back and forth. All the tests that are described in this section were performed with using this scene, and we have the following six cases for the experiment:

1) *The normal case*: We did not pan/tilt the camera, change focus, or compensate exposure in this test case. All camera parameters were set to their default values. The automatic exposure mode was turned off. The aperture, exposure, focus and zoom parameters

Fig. 4. The scene of the experiment.



(a) (b)

(c) (d)

Fig. 5. The four sample frames of the scene with an object for testing focus changes.

were all set to their initial values, and there was no movement or sudden light.

2) *The pan/tilt case*: This case was based on the normal case. We pan/tilt the camera in order to examine our method's ability to adapt to the changes in the scene. For simplicity, we only pan the camera to the left for which the angle of one pan action is approximately 10 degrees.
3) *The test–focus case*: Based on the normal case, we put an object (a box) in front of the camera, and then we moved the box from the farthest distance to the neareast distance. Fig. 5 shows four sample frames that were captured from the scene with the object placed near the camera. We hoped to see the change of the camera's focus level as the camera adapted to the object's movement.
4) *The test–exposure case*: Similar to the test–focus case, this case is based on the normal case. We used a desk lamp to create over-exposed regions in the scene. The test scene is the same as that in Fig. 4, but the lamp is turned on or off to control the light. Adjusting the exposure parameter in the camera reduces the brightness of the captured frames when the lamp is turned on, and it reduces their darkness when the lamp is turned off. Thus, image sharpness and contrast are both maintained.
5) *The pan/tilt and test–exposure case*: We experimented with panning and with changing the light of the environment simultaneously.
6) *The test–focus and test–exposure case*: We moved an object from the farthest distance to the nearest distance to the front of the camera, and then we simultaneously turned on or off the desk lamp to control the light. This test case clearly reflects the changes in the focus and exposure levels.

We tested each case with 100 captured images. Ground–truth data are built along with the test images in order to evaluate the edge results. We used four methods to find the best sharpness and contrast image:

1) *No exposure compensation and no focus change*: all camera parameters were set to their default values; the exposure compensation and focus parameters modes are set to manual.
2) *Auto–exposure of the camera*: the camera was configured with the automatic exposure mode.
3) *Auto–focus of the camera*: the camera was configured with the automatic focus mode.
4) *The scanning of all possible parameters*: the exposure and focus levels were successively changed. We captured a frame with each pair of parameters. After 210 captured frames, we found the best frame with using the entropy of the Sobel gradient method.
5) *Applying a particle filter*: our proposed algorithm.

Table I shows the edge evaluation results for six test cases and four methods of finding the best image. The input parameters that were used in all experiments are as follows:

- Canny edge detection: the low threshold $Thr_{low} = 50$, and the high $Thr_{high} = 200$.
- Line detection (the Standard Hough Transform): vote threshold $Thr_{vote} = 100$ with accumulation of 1; θ resolution $\Delta\theta = 1.15$ degrees; ρ resolution $\Delta\rho = 1$ pixel; the Hough space size: $N_\theta = \pi/\Delta\theta$ and $N_\rho = (2(I_{width} + I_{height}) + 1)/\Delta\rho$; and the local maxima window size is 5×5.
- Particle filter: the number of particles $N_{particle} = 40$; $\sigma_x = 2$ and $\sigma_y = 1$.

The average number of detected lines $\overline{N}_{det.}$ and of correct lines $\overline{N}_{cor.}$ are shown in columns of Table I. The number of detected and correct lines was obviously lower than that in the other four methods in the case where there was no exposure compensation and no focus change. In particular, the number of correct lines was more important than the number of detected lines in assessing the clear edges. The automatic exposure and the focus mode the were implemented in the camera were executed for comparison with other methods. These two methods achieved adequate processing time results because they were directly implemented on the

TABLE I

RESULTS OF EDGE EVALUATION FOR THE CASE OF NO EXPOSURE COMPENSATION AND NO FOCUS CHANGE, AUTO–EXPOSING, AUTO–FOCUSING OF CAMERA, SCANNING OF ALL POSSIBLE PARAMETERS, AND APPLYING A PARTICLE FILTER

	No ExpCom, no focus change		Auto–exposing of camera		Auto–focusing of camera		Scanning of all possible parameters		Applying a particle filter	
	$\overline{N}_{det.}$	$\overline{N}_{cor.}$	$\overline{N}_{det.}$	$\overline{N}_{cor.}$	$\overline{N}_{det.}$	$\overline{N}_{cor.}$	$\overline{N}_{det.}$	$\overline{N}_{cor.}$	$\overline{N}_{det.}$	$\overline{N}_{cor.}$
Normal	5.2	4.2	13.8	8.2	8.4	6	16.2	9.6	15.6	9.2
Pan/tilt	13	7.8	25.1	11	14.6	7.6	27	13.3	25.22	13
Test–focus	4.9	4.2	11.4	8	4.9	3.7	14.2	9.78	13.11	8.89
Test–exposure	4.8	3.9	5.3	4.2	4.9	3.8	8.33	6	6.11	5
Pan/tilt and test–exposure	4.3	3.2	10.59	7.24	3.7	3	11.4	6.67	11.56	7.33
Test-focus and test–exposure	7.7	5.7	19.9	9	9.9	6	20.7	8.56	17.78	9.22

hardware. Their processing time is lower than 1 ms; however, their results of edge evaluation are not as good as those of the method that involves scanning all the possible parameters and the particle filter based method.

The method of scanning all possible parameters yielded good results in most test cases, but the results in the latter two test cases were worse than the result of applying a particle filter method. This is because there is no relationship between the two successive sequences of the images in the method of scanning all images. It takes an average of 2542 ms to process 210 images of a sequence in the case of scanning all images. In the results our proposed method, the first four cases are approximated to the method of scanning all images. Two other cases of applying a particle filter gave slightly better results than the other methods, because the measurements from the previously captured frames were used. Furthermore, the particle filter significantly contributes to reducing the computation time of finding the best image in comparison with the method of scanning all images: 107 ms and 2542 ms. The advantage of applying a particle filter is most clearest in complicated test cases, which represent real environments.

VII. CONCLUSION AND FUTURE WORK

We described a method for integrating a particle filter with using the processes of tracking the exposure and focus camera parameters to find the captured frame with high edge quality. The average processing time per frame for an image size of 640×480 pixels is 107 ms. This processing time can be reduced by implementing the algorithm on specific devices instead of a computer, which is what we used. The strengths of our method are emphasized in the complicated test cases that simulate common situations in real environments. Our experimental results showed that our method can be applied in real–time systems because of its low computational requirements.

For future work, we plan to investigate the results in an outdoor environment. We hope to build a mobile robot with a mounted pan/tilt camera and apply our method to a lane detection system. Moreover, the camera aperture parameter was not exploited in this paper. Thus, we also plan to automatically change the aperture to assist robots with following the tracking objects in a constant depth of field.

REFERENCES

[1] A. Nayak and S. Chaudhuri, "Automatic illumination correction for scene enhancement and object tracking," *Image Vis. Comput.*, vol. 24, no. 9, pp. 949–959, 2006.

[2] M. A. Garcia–Garrido, M. A. Sotelo, and E. Martin–Gorostiza, "Fast traffic sign detection and recognition under changing lighting conditions," in *Proc. IEEE Int. Conf. Intell. Transp. Syst.*, pp. 811–816, 2006.

[3] G. Messina, A. Castorina, S. Battiato, and A. Bosco, "Image quality improvement by adaptive exposure correction techniques," in *Proc. Int. Conf. Multimedia and Expo*, vol. 2, pp. 549–552, 2003.

[4] J. Liang, Y. Qin, and Z. Hong, "An auto–exposure algorithm for detecting high contrast lighting conditions," in *Proc. Int. Conf. ASIC*, pp. 725–728, 2007.

[5] M. V. Shirvaikar, "An optimal measure for camera focus and exposure," in *Proc. Southeastern Symp. Syst. Theory*, pp. 472–475, 2004.

[6] J.–H. Lee, K.–S. Kim, B.–D. Nam, J.–C. Lee, Y.–M. Kwon, and H.–G. Kim, "Implementation of a passive automatic focuing algorithm for digital still camera," *IEEE Trans. Consum. Electr.*, vol. 41, no. 3, pp. 449–454, 1995.

[7] K. Zhu, W. Jiang, D. Wang, X. Zhou, and J. Zhang, "An effective focusing algorithm based on non–uniform sampling," in *Proc. IEEE Int. Workshop VLSI Design Vid. Tech.*, pp. 276–279, 2005.

[8] K. Zhu, W. Jiang, D. Wang, X. Zhou, and J. Zhang, "Using non–uniform sampling to make automatic focusing result correct," in *Proc. IEEE Int. Symp. Microwave, Antenna, Propagation and EMC Tech. for Wireless Commun.*, vol. 2, pp. 1264–1267, 2005.

[9] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379–423 and 623–656, 1948.

[10] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online non–linear/non–Gaussian Bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, 2002.

[11] B. Ristic, S. Arulampalam, and N. Gordon, "Beyond the Kalman filter, particle filters for tracking applications," *Artech House*, 2004.

[12] S. Fleck and W. Strasser, "Adaptive probabilistic tracking embedded in a smart camera," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, vol. 3, pp. 134–134, 2005.

[13] Sony Corporation, "EVI–D100/D100P Color Video Camera, Technical Manual," available at http://kino.iteso.mx/∼ivan/redes/videoconferencias/manuales/sony evi100 user.pdf, 2001.

[14] G. A. Borges and M.–J. Aldon, "A split–and–merge segmentation algorithm for line extraction in 2–D range images," in *Proc. Int. Conf. Pattern Recognit.*, vol. 1, pp. 1441, 2000.

[15] P. V. C. Hough, "Method and means for recognizing complex patterns," *U.S. Patent 3069654*, 1961.

[16] R. O. Duda and P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures," *Communications of the ACM*, vol. 15, pp. 11–15, 1972.