Journal of Korea Multimedia Society Vol. 12, No. 6, JUNE 2009(pp. 794-803)

Real-Time Container Shape and Range Recognition for Implementation of Container Auto-Landing System

Li Wei[†], Eung Joo Lee[††]

ABSTRACT

In this paper, we will present a container auto-landing system, the system use the stereo camera to measure the container depth information. And the container region can be detected by using its hough line feature. In the line feature detection algorithm, we will detect the parallel lines and perpendicular lines which compose the rectangle region. Among all the candidate regions, we can select the region with the same aspect-ratio to the container. The region will be the detected container region. After having the object on both left and right images, we can estimate the distance from camera to object and container dimension. Then all the detect dimension information and depth inform will be applied to reconstruct the virtual environment of crane which will be introduce in this paper. Through the simulation result, we can know that, the container detection rate achieve to 97% with simple background. And the estimation algorithm can get a more accuracy result with a far distance than the near distance.

Key words: container detection, line detection, rectangle detection, stereo vision, connected component labeling.

1. INTRODUCTION

With the development of modern economy of the whole world, containers are wildly used to transport all kinds of goods. In order to manage the containers efficiently and land containers full automatically and rapidly. Many researchers have been taking more attention to container tracking and management technology, traditional method for tracking and managing containers is based on the image identifier techniques and manual data input. In this method, the information of container gathered by the video is transmitted into the information process center; the information of goods is input into the information process center manually, it's lower identification speed, high error rate, and non-real time.

So for solving the problem, in this paper we will propose a system that detect and land the containers automatically. Container contains the line feature such as: parallel lines and perpendicular lines. There are many method that use Hough Transform (HT) and rectangle detection for this specific problem. Lagunovsky and Ablameyko [1] proposed a rectangle detection technique based on line primitives. First, linear primitives are extracted, line segments, which are grouped in straight lines. The length and orientation of these straight lines are compared and used to detect quadrangles, that are further approximated by rectangles. The approach of using Hough transform and rectangle detection used in many problems. Lin and Nevatia [2] studied the problem of rectangle/ parallelogram detection in aerial images. Their technique is based on line detection, and selection of line segments within a range of values (determined by maximum

※ Corresponding Author : Eung-Joo Lee, Address : (608-711) Department of Information & Communications Engineering, Tongmyong University, Busan, Korea, TEL : +82-51-629-1143, FAX : +82-51-629-1129, E-mail : ejlee@tu.ac.kr

Receipt date : Mar. 4, 2009, Approval date : June 11, 2009

† Department of Information Communication Engineering, TongMyong University, Korea
(E-mail : li_wei1983@naver.com)

†† Department of Information Communication Engineering, TongMyong University, Korea

and minimum building sizes). Given an initial line segment, anti-parallel lines are searched. A pair of anti-parallel lines is used to define a search region, where the remaining two sides of the rectangle are searched. Tao et. al [3] proposed a primitive-based approach for extracting rectangular buildings from aerial images. In their approach, edge elements are found and linear elements are extracted using a splitting arithmetic. Start-point, end-point and orientation of each linear element are used to detect parallel lines, and pairs of parallel lines are used to form rectangular primitive structures. Finally, these primitives are merged to form rectangles. Zhu et. al. [4,5] used a different approach to detect rectangular particles in cryo-electron microscopy images, by proposing a *Rectangular Hough Transform* (RHT). If the sides of a rectangle are known, the RHT uses a 2-D accumulator array to detect the center and orientation of the rectangle. By using the method we can get a fast and good result. And the framework of the system is as shown in Fig 1.

The rest of this paper is composed of the following parts: section 2 will discuss the container detection method, the method will use the hough transformation. In the section, we will discuss the hough transformation and lines detection in detail; section 3 will show the container distance estimation algorithm, in the section, we firstly introduce the stereo vision theory, and based on the theory, we proposed the method to estimate the distance of container; section 4 will introduce the remodeling method, based on the method we will create a virtual environment which is reconstructed from the real environment, and the position of containers in the virtual environment will keep real-time with the real containers. Finally we will give the experiment result and conclusion in the final part of the paper.

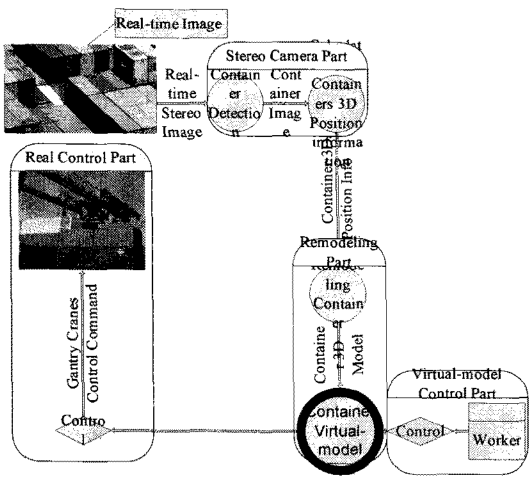


Fig. 1. Framework of the Auto-landing System

2. CONTAINER DETECTION IN HOUGH SPACE

As we can know that, container is a kind of regular geometrical one, it has kinds of line feature in the captured image, such as: parallel lines, perpendicular lines, and the scale of each container have a fixed value. According this, to detect container, we can firstly detect line information in the captured image, and then detect all the parallel lines and perpendicular lines from the lines result. After that, the cross points can be detected and we can select the candidate rectangle regions. Finally, the container region can be confirmed by selecting the rectangle region which with the correct scale. A flowchart of the container detection method is as shown in Fig 2.

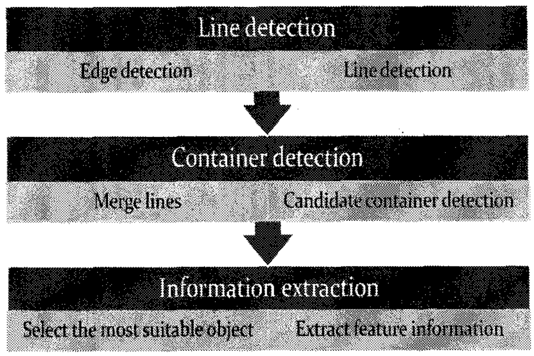


Fig. 2. Flowchart of Container Detection Method

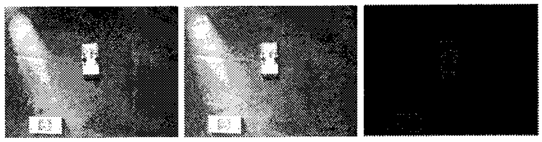
2.1 Lines Feature Detection in Hough Space

In order to detect container lines in images, one must be able to detect a group of pixels that are on a straight line or a smooth curve. This is what a Hough transforms (HT) is supposed to do [6,7]. The simplest case of Hough transform is a Hough linear transform. According to Hough theory, in the image space, the straight line can be described as y = mx + b and is plotted for each pair of values (x, y). However, the charactistics of that straight line is not x or y, but its slope m and intercept b. Based on the fact, the straight line y = mx + b can be represented as a point (b, m) in the parameter space. The two dimensions of the accumulator array would correspond to quantized values for m and b. For each pixel and its neighborhood, the Hough transform algorithm determines if there is enough evidence of an edge at that pixel. If so, it will calculate the parameters of that line, and then look for the accumulator's bin that the parameters fall into, and increase the value of that bin. By finding the bins with the highest values, typically by looking for local maxima in the accumulator space, the most likely lines can be extracted, and their geometric definitions read off. The simplest way of finding these peaks is by applying some form of threshold, but different techniques may yield better results in different circumstances-determining which lines are found as well as how many. Since the lines returned do not contain any length information, it is often next necessary to find which parts of the image match up with which lines. The equation of the line can be written as:

$$r = x \cdot \cos\theta + y \cdot \sin\theta \qquad (1)$$

It is therefore possible to associate to each line of the image, a couple (r, Θ) which is unique if $\theta \in [0,\pi]$ and $r \in R$, of if $\theta \in [0,2\pi]$ and r≧0. The (r, Θ) plane is sometimes referred to as Hough space.

In order to detect the candidate lines from the container captured image, we firstly detect the edge information from the image. And for getting the edge image, we will smooth the image by using the Gaussian filter [8] to remove the noise in the image, and then use Canny algorithm [9] to extract all the edges from the no-noise image. This step can also be named pre-process approach, and the result is as shown in Fig 3.



(a) (b) (c)

Fig. 3. Pre-process Approach Result: (a). Source Image; (b). Gaussian Filter Result and (c). Canny Method Result

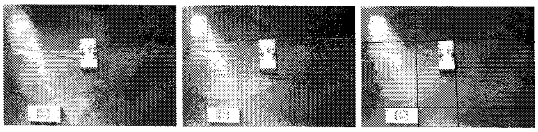
The Hough linear transformation will be applied to the result image of pre-process, the result contains many lines feature. But what we needs is the parallel lines and the perpendicular lines for confirming containers. And as we knows that the parallel lines have the same slope and perpendicular lines have the deviation value of π/2, so line slope angle can be used to classify the parallel lines and perpendicular lines. According to this, we can calculate the angle of each line, if any pair of lines has the same angle we will consider them as the parallel lines and remain them as the candidate lines. The algorithm is as shown following:

```
X = {}
FOR i = 0 : 1 : nLines
  FOR J = 0 : 1 : nLines
    {(L[i], Ox) = angle1
    {(L[j], Ox) = angle2
    if (angle1 - angle2) < threshold
  End
End
```

And if an pair of lines has a deviation value of π/2, we can consider them as the perpendicular lines and remain them as the rectangle lines. The detection result is as shown in Fig 4.

2.2 Container Lines and Corner Detection

We know that, each rectangle is composed of



(a) (b) (c)

Fig. 4. Parallel and Perpendicular Lines Detection Result: (a). Source Image; (b). Parallel Lines Detection Result and (c). Perpendicular Lines Detection Result

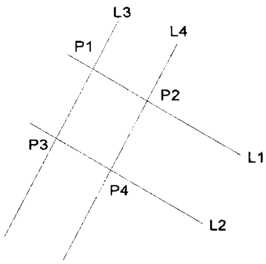
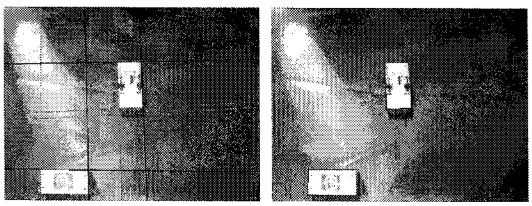


Fig. 5. A Diagram to show Rectangle Component

parallel lines and perpendicular lines, the line is segmented by the cross points. As seen in Fig. 5, L1 and L2 is a pair of parallel lines and L3 and L4 is a pair of parallel line, the two pairs of line are orthogonal. We can see that the four lines generate four cross points: P_1, P_2, P_3, P_4, and a rectangle is confirmed by the four points which we can called $P_1P_2P_3P_4$, and outside line segments of the rectangle can be removed finally.

The rectangles detection result can be seen in Fig 6.

There is a global regulatory standard for container size which can be seen from Table 1, any container in the world, must be according to the standard.

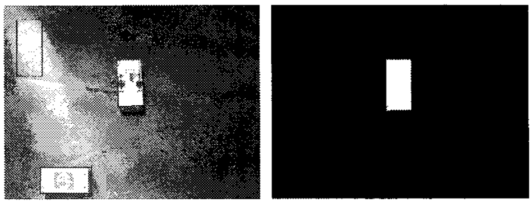


(a) (b)

Fig. 6. Rectangle Detection Result: (a) Lines Detection Result and (b) Rectangles Detection Result

Table 1. Container Size Global Regulatory Standard

Parameter		20 feet	
Material		Steel	
Unit		mm	Feet- Inches
Size	Long	6058	19-10 1/2
	Width	2438	8
	Height	2438	8



(a) (b)

Fig. 7. Container Objects Detection Result, (a) Candidate Rectangles and (b) Detected Container Regions

In our algorithm, we can utilize the size standard, as seen from the table, long of container is 6058, width and height of container have the same size: 2438, and scale of the container is 2.48 (6058:2438). So for all the candidate rectangles, we can calculate the size scale, if scale of a rectangle is about 2.48, we can assume that it is a container region and we can remain it, else it is a noise region which will be removed from the result image. Detail of the algorithm is as shown following:

```
X = {}
FOR i = 0 : 1 : nRegions
  if(Long/With ≈ 2.48)
    X = ContainerRegion
  else
    X = No ContainerRegion
END
```

Result of the approach is as shown in Fig 7.

3. CONTAINER DISTANCE ESTIMATION USING STEREO CAMERA

Recently, stereo vision technology has been widely used for object measuring, because stereo vision has the advantage that it is able to obtain an accurate and detailed 3D representation of the environment around an intelligent vehicle, by passive sensing and at a relatively low sensor cost. It is similar to the human vision system, the stereo vision algorithm uses a coarse to fine approach to perform the regional correlation. The resulting disparity map represents the correspondence between the points in the left and right images. For a detailed description of the stereo algorithm see, for instance, reference [10,11]; herein, starting from the knowledge of the related disparity, we concentrate on the computation of the relative depth.

Fig. 8 shows the configuration being considered: let P be the projection on the stereo plane of a point Ps in the space. We define the function K as:

$$\kappa(\alpha,\beta,\gamma,\delta)=\frac{\tan(\alpha-\gamma)\bullet\tan(\beta+\delta)}{\tan(\alpha-\gamma)+\tan(\beta+\delta)} \quad (2)$$

where α and β are the vergence angles, ɣ =arctan(x_l/F_l) and δ=arctan(x_r/F_r) define the position of two corresponding points on the image planes, $x_r=x_l+D$ (D is the know disparity), and F_l and F_r are the focal lengths of the left and right camera measured in pixels, respectively. It is easy to prove that the depth Z referred to in the stereo coordinate system is:

$$Z=B\bullet\kappa(\alpha,\beta,\gamma,\delta) \quad (3)$$

where B is the baseline length. From the knowledge of the vergence angles α and β, and the

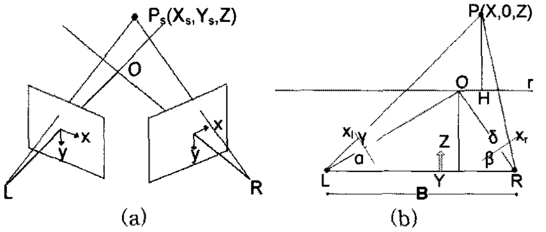


Fig. 8. Schematic representation of the stereo coordinate system: (a) the configuration in space is shown-the y axes of the cameras and of the stereo coordinate system are parallel and orthogonal to the plane defined by the two optical axes; (b) The projection of the point Ps = (X, Y, Z) on the plane Y = 0 (the point P) is shown

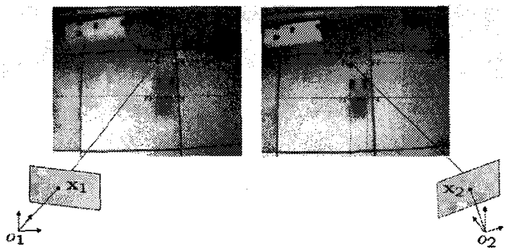


Fig. 9. Container Image of a Pair of Parallel Stereo Cameras

angular disparities ɣ, δ provides a depth measure with respect to the interocular baseline. As expected, the distance from the fixation point OK depends only on the vergence angles α and β; in fact:

$$Z_{fp}=\mathrm{B}\bullet\mathrm{K}(\alpha,\beta,0,0)=\mathrm{B}\bullet\frac{\tan(\alpha)\cdot\tan(\beta)}{\tan(\alpha)+\tan(\beta)} \quad (4)$$

In the frontal section, containers can be detected exactly, and we can apply the algorithm on both the left and right cameras to detect containers. Thus we can get two positions for each container, one is in the left image and the other is in the right image, as we can see from the Fig 9. P_1~P_4 is container position in left image and P_1'~P_4' is the position in right image.

The stereo camera which we used is consisted of a pair of parallel cameras. So in equation (3) α=β =90°, and thus the K equation can be changed as:

$$\kappa(90,90,\gamma,\delta)=\frac{-ctg(\gamma)\cdot ctg(\sigma)}{ctg(\gamma)-ctg(\sigma)} \quad (5)$$

And here: ctg(ɣ)=f/X_L and ctg(δ)=f/X_R, thus, the K equation can be transformed into:

$$\kappa(90,90,\gamma,\delta)=\frac{-\frac{f}{X_L}\cdot\frac{f}{X_R}}{\frac{f}{X_L}-\frac{f}{X_R}}=-\frac{f}{X_R-X_L} \quad (6)$$

The distance from the object point can be represented as:

$$d_{fp}=|Z_{fp}|=|\mathrm{B}\bullet\mathrm{K}(90,90,0,0)|=B\bullet\frac{f}{X_R-X_L} \quad (7)$$

And here:

d_{fp} : distance from point *ith* to camera.

f : camera focus.
B : base line (distance between two cameras).
X_L : container position in left image
X_R : container position in right image

4. CONTAINER DIMENSION CALCULATION AND REMODELING

Container dimension information include: color, texture, label, perimeters and line information. The information can be obtained by using container detection method. But we cannot get the real dimension information just through the detection algorithm. Because the scale of container have been reduced through the capture approach. As Fig. 10 shows that,

In order to get the real size information (long, width) of container from image, we should know the scaling of the capture approach. In the pre-section, container distance has been calculated according to the stereo vision theory. Similarly the real position of an object oriented point can also be gained by using the stereo camera as equation (7) and (8) represented that:

$$x_{fp} = X_L \frac{f}{X_R - X_L} \tag{8}$$

$$y_{fp} = Y_L \frac{f}{X_R - X_L} \tag{9}$$

So, for measuring container size, we can firstly calculate the original points of container 4 rectangle vertexes (P_1, P_2, P_3, P_4) in left image and (P'_1, P'_2, P'_3, P'_4) in right image.

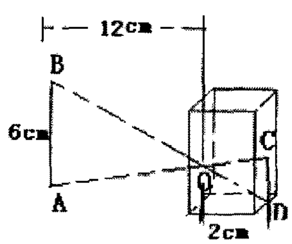


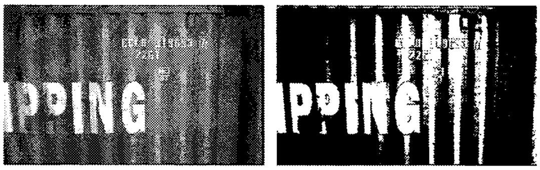
Fig. 10. Camera Capture Approach for Capture Container

After a multitude of tests on horizontal and vertical line segmentation, we can describe the important properties of the characters in real scene images with a resolution of 768x576 captured from a video camera as follows.

a) The character sizes of vehicle and container ID numbers in the captured image are in the pixel range from 16 to 64, and the width of the character strokes is greater than or equal to 3 pixels. These are the input parameters of the whole system.
b) The contrast between the gray-levels of the characters and the background is greater than that of other noncharacters and the background. The characters are either black on a white background or white on a black background.
c) The characters in the images are composed of both short vertical and horizontal line segments.
d) The non-character regions are composed of long horizontal and vertical line segments; long horizontal line segments but short vertical line segments; or long vertical line segments but short horizontal line segments.

So, if container region is confirmed from the image, we can determine that the characters in a real-life image will be located in the regions with both short horizontal and vertical line segments and with relatively high gray-level contrast. Since the characters are assumed to have high contrast against their background. The adjacent line segments will have relatively high gray-level contrast. We consider the properties of a group of characters in the scene image in two dimensions (horizontal and vertical directions). The characters can be distinguished from the background more easily and thus more accurate character segmentation can be achieved. The container label detection result is shown in Fig. 11.

The container label has been extract by the frontal method, the only job we should do for next is to employ a recognition machine to recognize the



(a) (b)

Fig. 11. Container Label Detection Result: (a) Source Image and (b) Label Detection Result

label with purpose of getting the container register information. In the system, we used neural network based recognition machine to recognize the label, when it acts as the label recognition module. Basically, it is a general feed-forward multi-layer perceptron with one hidden layer, trained by the back-propagation algorithm [12]. Number of nodes composing the input layer, the hidden layer and the output layer are 256, 100 and 35 respectively.

The input to the neural network is the character pattern extracted by the schemes described in earlier sections. The normalized label is represented in a 16 by 16 gray-level pixel-matrix. Each input node receives the state of one pixel, whose value is either 0 or an integer within the range 128 to 255.

At the output layer, the 35 output nodes are associated with the 35 alphanumeric character categories to be recognized. The state of an output node can be any floating-point value in the range [0⋯1]. Given an input character pattern, the score of each output node resulted from the network operation describes the likeliness fro the input to be regarded as the category associated with that node.

All the texture, color, label and size information of container have been detected in the system, we can apply these features to remodel the containers in the virtual world, the coordinate which describes container position is important. In our system, we selected the crane center bottom point as the original point of the coordinate, selected the horizontal direction as x-axis, as diagram in Fig. 12 shows that.

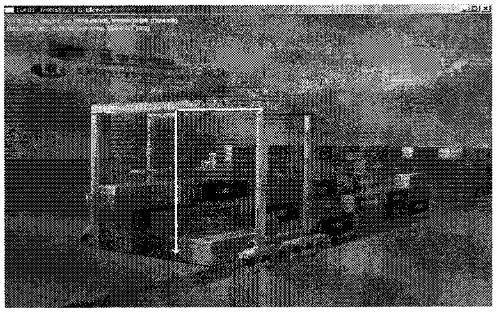


Fig. 12. Coordinate of Container Virtual Environment

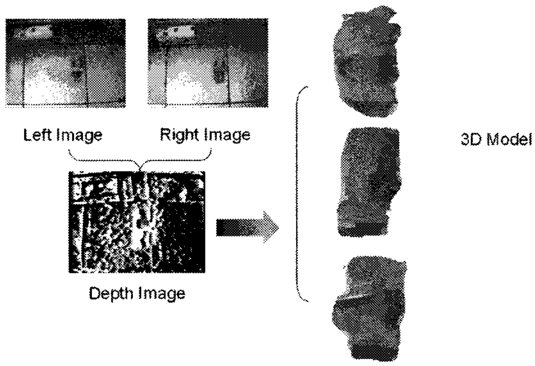


Fig. 13. Container Remodeling Result

Container position in real world coordinate will be transformed into the coordinate and have a corresponding position. At the position, we then create a rectangular block with the same scale of container, and attach the texture, color, label of real container to the rectangular block, thus a virtual model of the container can be created. Because the virtual coordinate is linear transformed from the real world, the crane control worker can just see the virtual environment to control the crane. The remodeling result is as shown in Fig. 13.

5. SIMULATION RESULTS AND COMPARISON

We have made simulation on a mini model of Port to evaluate efficiency of the proposed system. The mini model is composed of:

- Stereo camera. (Bumblebee, IEEE-1394)
- A mini crane system.

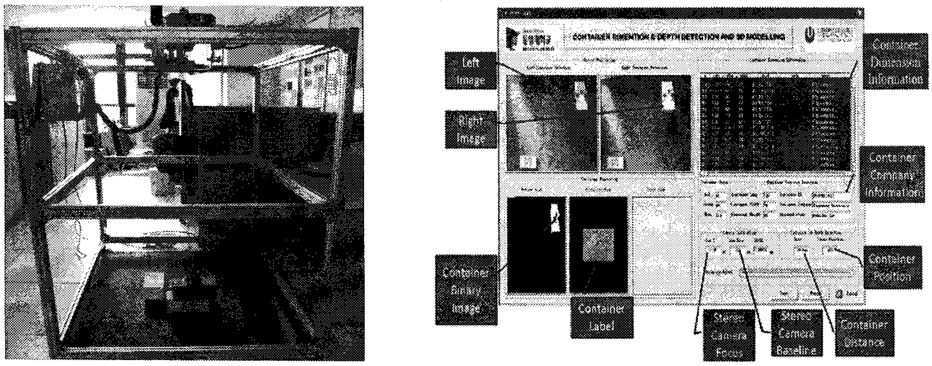


Fig. 14. Frame of the Container Automatic Landing Simulator

• Container Models

Frame of the simulator is as shown in Fig 14.

Any high precision measurement system will depend on all components: mechanical, optical, electronic, and algorithmic. Since we wanted to avoid such issues as optical distortion, shading problems, background problems, our simulation results serve to illustrate and compare the methods, rather than to give absolute accuracy figures.

We use simple background with black color for detecting container, the image what we obtained is with size of 800 * 600, the inter parameter specification of the stereo camera is as shows in Fig. 15.

Depth estimation result of container can be seen from Table 2, the result is shown from far to near. In the table, we also give the real distance of the container and the errors. And from the result, we can see that with the container moving far from the stereo camera, the errors will become greater than greater. It is due to the problem baseline length of the stereo cameras, the narrow-baseline camera configuration makes possible to find stereo correspondence automatically for every pixel, but a serious drawback is its low accuracy with long distance.

Table 2. Depth Estimation Result

Real Depth (m)	Estimated Depth (m)	Errors (m)
2	1.98	0.02
1.8	1.85	0.05
1.7	1.8	0.1
1.5	1.7	0.2
1.3	1.69	0.39
1	1.3	0.3
0.55	0.76	0.21

Fig 16 shows a diagram to describe result of the real distance, estimated distance and the errors.

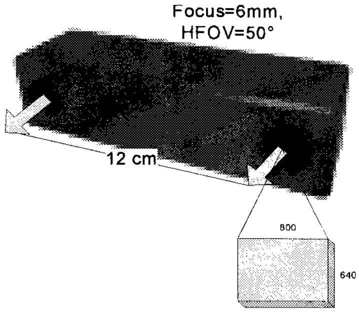


Fig. 15. Inter Parameter Specification of the Stereo Camera

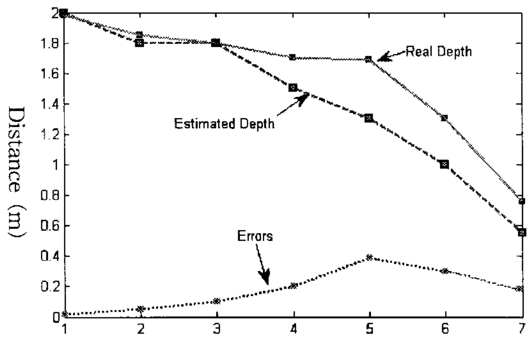


Fig. 16. Distance Result

The red line represents the real distance, the blue line represents the estimated distance and the green line represents the errors.

For measuring container distance, the conventional methods usually use the RFID based location technology as presented by Zhengwu Yuan and Dongli Huang [13], in their paper, they used the moving passive RFID and GPS equipment into shipping container to make the cargos and container monitored and located even if the container is loaded, stored, hoisted and transported. And another method is to use the point laser sensor to detect the distance information. The two kinds of conventional methods can get more accurate results when compare with the proposed vision based method, but the RFID based method need to improve the container with RFID device. And GPS equipment also needs to make a change on container. This makes it difficult to apply to all the containers that be used currently. But using the stereo camera based system. We need not to improve the container with any change. It can detect any containers currently.

6. CONCLUSION

In this paper, we have presented the system which automatic detect container and land container by using stereo camera. Based on the stereo vision theory, the container can be detected according to its' dimension feature such as: width-height scale, color, texture. In the system, we used the width-height scale and texture information to confirm the container region. Then we use stereo equation to calculate the real distance and position information of container. We then send it to the crane to control it move to the corresponding position to catch the container. On the other way, the texture, depth information will be applied to remodel the container in a virtual environment. The environment is reconstructed from the real environment, which is used for the worker.

We have made simulation based on the PORT simulator. The simulation result shows that the proposed algorithm can performs well on detecting container and depth information in the real-time system. However, it still can be influenced by the illumination, and because of the low quality and low accuracy of stereo vision, the depth information estimation algorithm will need to be improved further.

ACKNOWLEDGMENT

This research was supported by Ministry of Knowledge and Economy, Republic of Korea under the ITRC (Information Technology Research Center) support program supervised by IITA (Institute for Information Technology Advancement (IITA-2009-C1090-0902-0004).

REFERENCES

[1] D. Lagunovsky and S. Ablameyko, "Straight-line-based primitive extraction in grey-scale object recognition," *Pattern Recognition Letters*, Vol.20, No.10, pp. 1005-1014, Oct. 1999.

[2] C. Lin and R. Nevatia, "Building detection and descriptionfrom a single intensity image," *Computer Vision and ImageUnderstanding*, Vol.72, No.2, pp. 101-121, 1998.

[3] W.-B. Tao, J.-W. Tian, and J. Liu, "A new approach to extract rectangle building from aerial urban images," *International Conference on Signal Processing*, pp. 143-146, 2002.

[4] Y. Zhu, B. Carragher, F. Mouche, and C. Potter, "Automatic particle detection through efficient hough transforms," *IEEE Transactions on Medical Imaging*, Vol.22, No.9, pp. 1053-1062, Sep. 2003.

[5] Y. Zhu, Carragher, and C. Potter, "Fast detection of generic biological particles in cryo-em images through efficient hough trans-

forms," In *International Symposium for Biomedical Imaging*, pp. 205-208, 2002.

[6] J. Illingworth and j. Kittler, "A Survey of the Hough Transform," Computer Vision, Graphics and Image processing Vol.44, pp. 87-116, 1988.

[7] A. V. Nguyen, Y. Y. Choo and E. J. Lee, "Container Dimension Detection and 3D Modeling based on Stereo Vision," MITA 2008, pp. 207-210, Chiang Mai, Tailand, July 3-5 2008.

[8] Deng, G. and Cahill, L.W., "An adaptive Gaussian filter for noise reduction and edge detection," Nuclear Science Symposium and Medical Imaging Conference, Vol.3, pp. 1615-1619, 31 Oct.- 6 Nov. 1993.

[9] Kakarala R. and Hero A. O, "On achievable accuracy in edge localization," ICASSP-91, Vol.4, pp. 2545-2548, 14-17 Apr. 1991.

[10] A. Scheenstra, A. Ruifrok, and R. Veltkamp, "A Survey of 3D Face Recognition Methods," International Conference on Audio-and Video-based Biometric Person Authentication, Vol. LNCS 3546, pp. 891-899, NY, USA, July 2005.

[11] Li Wei, E.J. Lee, "Improved Multi-pose 2D Face Recognition Using 3D Face Model with Camera Pose Estimation Approach and nD-PCA Recognition Algorithm," IEEE Conference on ICCIT '08, Vol.2, pp. 728-736, Nov. 2008.

[12] Rumelhart D.E., McClelland J.L. (1986), "Parallel Distributed Processing," MIT. Press.

[13] Zhengwu Yuan and Dongli Huang, "A Novel RFID-based Shipping Containers Location and Identification Solution in Multimodal Transport," Electrical and Computer Engineering, pp. 267-272, 4-7 May. 2008.

Li Wei

Li Wei received his B. S. in Dalian University of Light Industry in China (2002 - 2006), and M. S in TongMyong University, Korea (2006-2008), and now (2008 - 2009) his is a Ph. D. student of in Korea. His main research interests are in image processing computer vision Biometrics and face recognition.

Eung-Joo Lee

Eung-Joo Lee received his B. S.,M., S. and Ph. D. in Electronic Engineering from Kyungpook National University, Korea, in 1990, 1992, 1996, respectively. In March 1997, he joined the Department of Information Communication Engineering of Tongmyong University, Busan, Korea, as a professor. His main research interests are in image processing, computer vision, and Biometrics.