

## SIMPLE NEAR-REALTIME CRANE WORKSPACE MAPPING USING MACHINE VISION

Mohammad S Rahman and Joshua Vaughan

Department of Mechanical Engineering  
University of Louisiana at Lafayette  
Lafayette, Louisiana 70503  
Email: joshua.vaughan@louisiana.edu

### ABSTRACT

Overhead cranes are widely used in industries all over the world. It is not easy to control the crane without vibration, increasing the likelihood of obstacle collision or other accidents. Even experienced crane operators make mistakes that cause loss of money and time. Some reasons for these incidents are limitations of the operator's field of view, depth perception, lack of knowledge of the whole workspace, and the dynamic environment of the workspace. One possible solution could be aiding the operator with a dynamic map of the workspace that shows the current position of the obstacles as well as probable areas of finding obstacles based on the previous positions of obstacles. This paper describes a simple method of mapping a crane workspace using machine vision.

### INTRODUCTION

In overhead cranes, the payload is suspended from a trolley that moves along the bridge. The bridge itself can also move, so that the crane can serve a large area. Figure 1 shows a typical bridge crane. Accidents in overhead crane operation remain a problem, causing damage and injuries [1]. According to Crane Inspection and Certification Bureau, 90% of all crane accidents occur due to human error [2]. Half of U.S. crane accidents that had injuries in 2009 resulted in fatalities.

Recent developments in crane vibration reduction research have greatly reduced the vibration of the crane, improving safety and efficiency. One of these techniques is input shaping [3], [4], [5]. It shapes the command input by convolving a sequence of



FIGURE 1: CRANE OPERATOR POSITIONING PAYLOAD

impulses with crane operator input. The convolved signal is then used as the reference command. This technique has proven useful for vibration reduction of cranes [6], [7], [8], [9].

Control techniques like input shaping reduce the probability of crane collisions with obstacles. However, lack of knowledge of the workspace and common limitations in human perception of distance and depth still limit the ability of the operator to safely move a crane through a workspace. One possible solution to these limitations is aiding the operator with a map of the workspace that depicts the current position of obstacles. The map could be extended to include information about past locations of

obstacles as well. The past locations could be an indication of the possibility of finding an obstacle at a given location in the workspace.

Research has been conducted to model the workspace in order to avoid crushing obstacles during operations of heavy equipment [10]. Vision systems have been used for mapping and navigation for mobile robots [11], [12], [13] and developing augmented reality workspaces [14], [15]. Simultaneous Localization and Mapping (SLAM) is a well established method for mapping an unknown environment, or updating a map within a known environment by mobile robots and autonomous vehicles while at the same time keeping track of their current locations [16], [17].

A machine vision system has been implemented on tower cranes [18], [19]. The sole purpose was to help the operator see the workspace clearly. Some researchers have used machine vision as a means of measuring the crane hook location for feedback control [20]. However, there was no attempt to map the crane workspace. This paper describes a novel approach of mapping the crane workspace in near-realtime using machine vision.

In the following sections, the object detection and mapping techniques are explained. First, the steps of image processing, background detection, obstacle detection, elimination of payload from map and stitching technique are discussed. Next, a method for combining current and older images to produce the final map is described. Then, the effect of changing the design parameters on the mapping performance is explained. Finally, the conclusions made from this research are discussed.

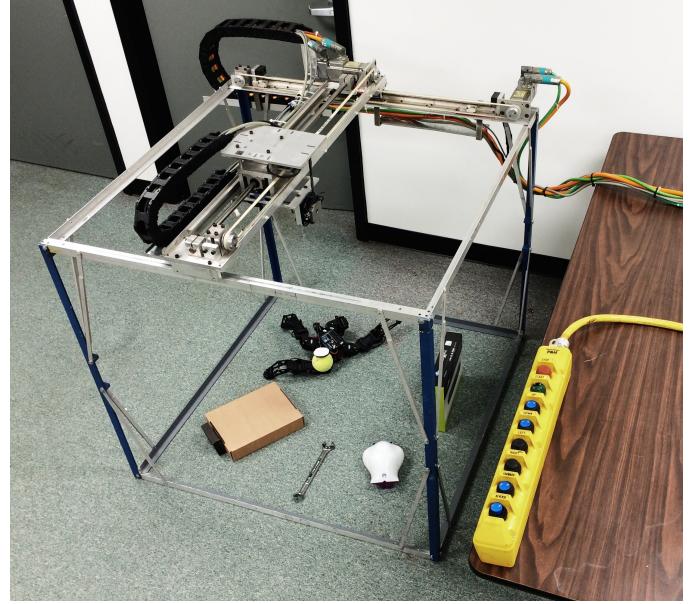
## Overview of Mapping

Figure 2 shows the crane used for the examples presented in this paper. The workspace of this crane is 1m x 1m x 1m. It is controlled using a Siemens PLC and is driven by Siemens AC servomotors.

For the mapping technique presented in this paper, a camera is mounted on the crane trolley. The camera automatically takes a picture at predefined points in the workspace. These images are then processed. The background is detected, and the crane hook is masked. Then, the obstacles are identified and a graphical representation of the workspace is generated.

An example map resulting from this process is shown in Figure 3. The map presented in this paper shows the current locations of the obstacles in red and past obstacle locations in yellow. The intensity of the yellow regions indicate how recently an obstacle was found at that location. For example, a high intensity of yellow indicates that an obstacle was at that location recently. This suggests that there is a higher probability of finding an obstacle there.

Figure 4 shows a flowchart of the entire process, each part of which is described in detail in the following sections.



**FIGURE 2:** SCALE CRANE USED FOR THE EXAMPLES IN THIS PAPER

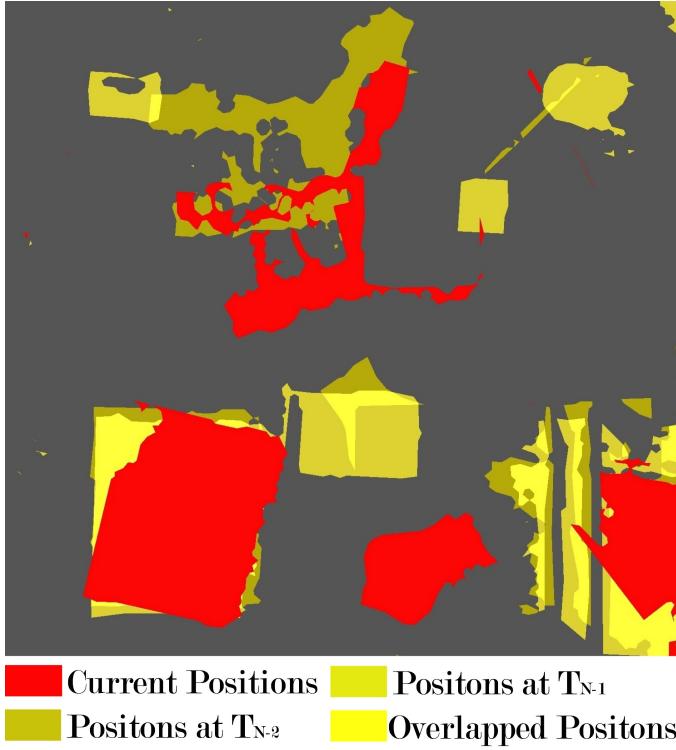
## Individual Image Processing

In order to eliminate noise in the images, it is necessary to smooth the images first. Gaussian blur and median blur are two of the most commonly used methods for blurring. They are also used in this work.

To separate the obstacles from the background, it is necessary to detect the background. If the background is known, a simple thresholding can be used to separate the background from the foreground. However, depending on the noise level and lighting, the background may vary significantly. Moreover, the reflections of the lighting source and the shadows of the objects can make it difficult to distinguish the obstacles from the background.

For the method used in this research, each of the three channels of the image is divided into ranges of equal length. Then, the pixels are scanned in each channel and the number of pixels that fall into each range are counted. The range where most of the pixels fall into is the background value range of that image in that channel; it is assumed in this work that the background occupies larger area than obstacles. In most cases, the background pixel values have significant variation. Instead of scanning each pixel, a number of pixels are skipped while scanning to get the correct background. Skipping pixel also saves computation power. The optimum number of pixels to be skipped depends on the lighting conditions and noise level of the workspace. The effect of number of pixels skipped on obstacle detection is discussed later.

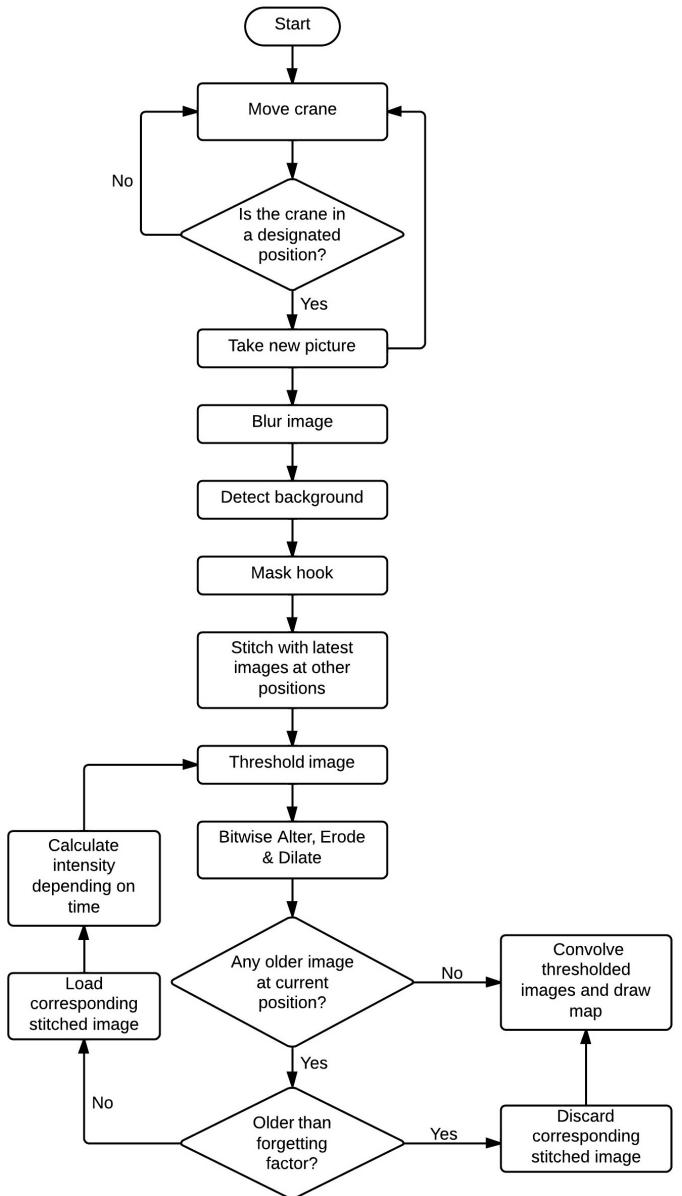
Because the camera can not capture the entire workspace in a single image, the individual images need to be stitched to-



gether. However, there will always be the crane hook/payload in the image, because the camera is mounted directly over the the hook. A simple solution is to cover that part of image with the average background color calculated, as shown in Figure 5. The part of the workspace area blocked by the hook can be recovered from neighboring images after stitching. The recovered workspace area is greater for a larger overlap between neighboring images.

After masking, the images are stitched together as shown in Figures 6 and 7. In order to stitch images, the key-points among them are matched. The larger the overlap between neighboring images is, the more key-point matches are found, yielding a better stitching performance. However, greater overlap means more individual images are required to cover the workspace. This takes more time and computation power to stitch.

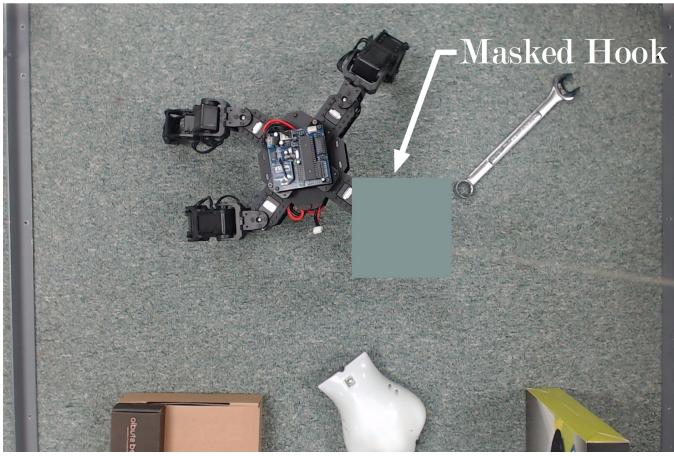
Next, all three channels of the stitched image are thresholded simultaneously through two three-channel scalars close to the background color. The range of these scalars depends on the degree of noise in the background. For example, a small threshold is sufficient for a low-noise and clear background, and gives clean image after thresholding, where every obstacle is perfectly detected. The resultant image after thresholding is a binary image, which should be bitwise altered and further refined using erosion and dilation to get rid of small blobs. Figure 8 shows



**FIGURE 4: FLOWCHART OF THE MAPPING PROCESS**

thresholded and bitwise altered images, and Figure 9 shows dilated and eroded images.

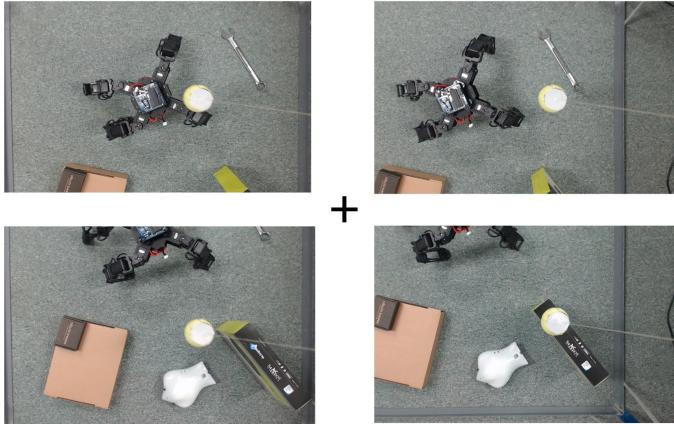
After erosion and dilation, the contours of the obstacles are extracted. Then, a polygonal curve is estimated using the Douglas-Peucker algorithm, and the contours are drawn, as shown in Figure 10.



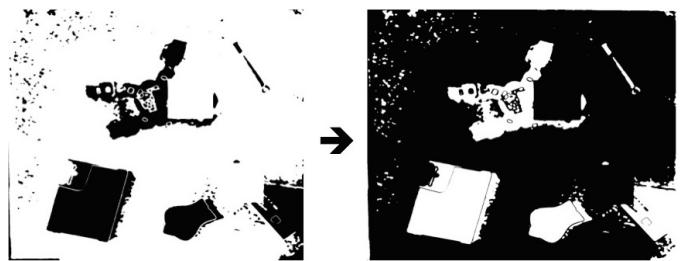
**FIGURE 5: CRANE HOOK MASKED**



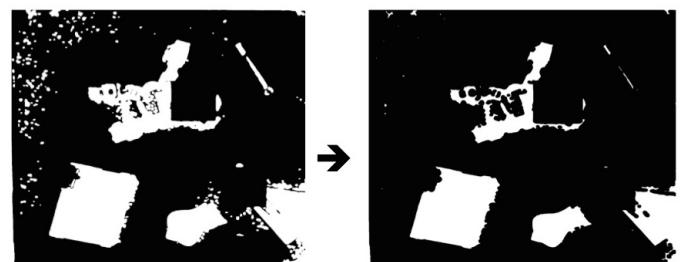
**FIGURE 7: STITCHED IMAGE**



**FIGURE 6: INDIVIDUAL IMAGES**



**FIGURE 8: THRESHOLDED AND ALTERED IMAGE**



**FIGURE 9: DILATED AND ERODED IMAGE**

## Mapping

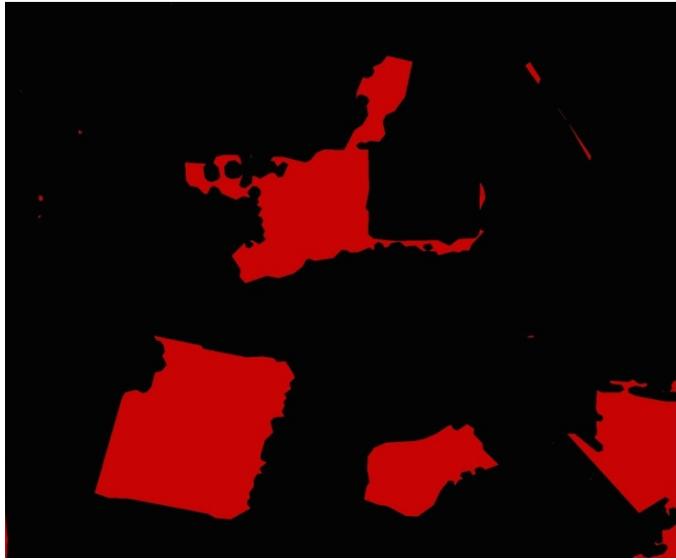
The mapping process presented in this paper takes the latest, as well as older positions, of the obstacles into account. The older positions of obstacles can be used as an indicator of the likelihood of there being an obstacle at that location in the future.

Each time a new picture is taken by the camera at a particular position, an individual map is generated by stitching that picture with the most recent pictures at other positions. Then, a final map is created by convolving the latest map with older individual maps.

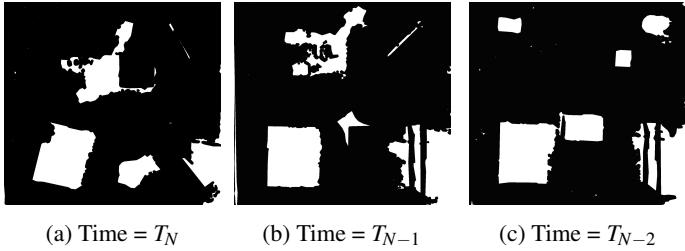
The individual maps are convolved in a way that the latest map is shown in red to indicate certainty of finding an obstacle. Older maps are shown in yellow, the intensity of which decreases linearly with time when that maps were generated. Relatively recent maps are more likely than older maps to indicate an obstacle position. The intensity decreases with time to show the decreased

probability of finding an obstacle at that location. The slope of linearity determines how fast the individual maps lose weight as they get older.

Using multiple older images at a given location provides additional information. If there is an overlap in the yellow regions



**FIGURE 10:** IMAGE AFTER CONTOUR IS DRAWN

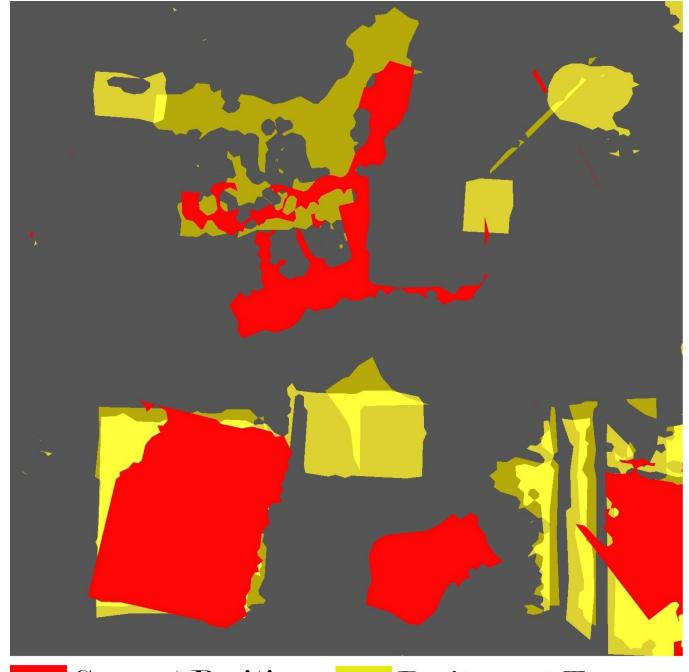


**FIGURE 11:** IMAGES AT THREE DIFFERENT TIMES

from past images, the overlapped area is brighter than in the individual maps. The overlapped area indicates there has been an obstacle present at these overlapping locations in the workspace at multiple times. This suggests that the probability of finding an obstacle there in the future is greater.

If an individual map is too old, it is forgotten. A memory factor determines when an individual map is forgotten. The bigger the memory factor is, the older an individual map can be and still taken into account for generating the final map.

As an example, individual maps at times  $T_N$ ,  $T_{N-1}$ ,  $T_{N-2}$  are shown in Figure 11, where  $N$  is number of images. The resulting map is shown in Figure 12. In the map presented here, the red areas show the most recent positions of the obstacles. Medium-bright yellow areas show next to the most recent positions, and the dark yellow areas show the oldest positions of the obstacles. The overlapping areas between the older obstacle positions are shown by the brightest shade of yellow.

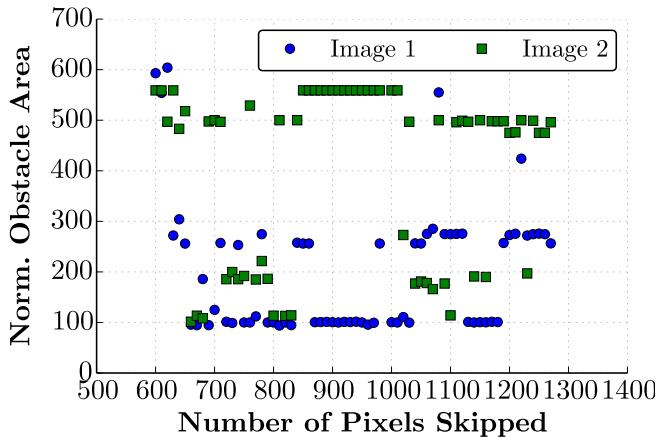


**FIGURE 12:** FINAL WORKSPACE MAP

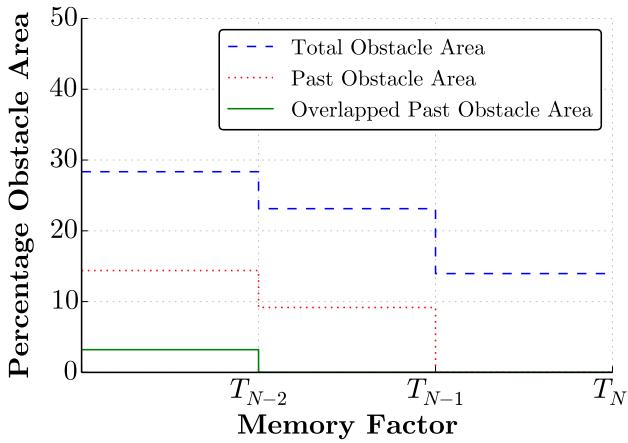
### Analysis of Algorithm

Although this mapping technique is very promising, there are some problems. The background detection method described is sensitive to noise and lighting. The thresholding is also sensitive. If there is an object which has the same color as the background, it will go undetected. If the background pixel has a wide range of values, the thresholding range has to be bigger, which means there will be a greater chance of missing obstacles. In addition, it is difficult to find the optimum value of the threshold range for a particular workspace. Depending on the time of the day, the lighting condition changes in the workspace, which makes adjustment necessary.

The mapping performance depends on the accuracy of background detection. Figure 13 shows the effect of number of pixels skipped for background detection. The map which was most representative of the actual obstacles in the workspace was set as a benchmark. In Figure 13, the area of detected obstacles is shown as a percentage of the obstacle area detected in the benchmark map as a function of the number of pixels skipped. For first image, there are three ranges of number of pixel values skipped for which 100% of the optimum obstacle area is detected. For second image, the range is reduced to two small ranges, which makes it difficult to find the optimum number of pixels skipped for 100% obstacle detection. This is a major weakness of this



**FIGURE 13:** PERCENTAGE OF OBSTACLE DETECTED VERSUS NUMBER OF PIXEL SKIPPED



**FIGURE 14:** OBSTACLE DETECTED AS A PERCENTAGE OF TOTAL WORKSPACE AREA VERSUS MEMORY FACTOR

algorithm.

Figure 14 shows the effect of memory factor on obstacle detected as a percentage of total workspace area. The total obstacle area, past obstacle areas and the overlap between past obstacle areas are shown. Obstacles occupy more area if the memory factor is bigger. In this example, when the memory factor is greater than  $T_{N-2}$ , all three individual maps are taken into account. The area occupied by obstacles is greater. When the memory factor is less than  $T_{N-2}$  but greater than  $T_{N-1}$ , the oldest of the three individual maps is forgotten. In this case, both the total obstacle area and past obstacle area are smaller. If the memory factor is less than  $T_{N-1}$ , only the most recent individual map is taken into

account, the other two are forgotten. The total obstacle area is the smallest in this case, and there is no past obstacle area shown in the map in yellow.

## Conclusion

In this paper, a new approach of crane workspace mapping in near real-time using machine vision was presented. An algorithm for processing individual images and combining them into a complete map was shown. A technique to show areas where there were obstacles recently, along with current positions of the obstacles was also described. The effect of changing the design parameters, such as number of pixels skipped for obstacle detection and the choice of memory factor, on the mapping performance was also discussed.

## REFERENCES

### REFERENCES

- [1] J. E. Beavers, J. R. Moore, R. Rinehart and W. R. Schriver “Crane-Related Fatalities in the Construction Industry,” *Journal of Construction Engineering and Management*, Volume 132 Issue 9, September 2006.
- [2] Crane Inspection and Certification Bureau, n.d., from <http://www.cicb.com/blog/posts/with-crane-related-injuries-on-the-rise-don-t-become-another-statistic.html>
- [3] O. J. M. Smith, “Feedback Control Systems,” New York: McGraw-Hill Book Co., Inc., 1958
- [4] N. C. Singer and W. P. Seering, “Preshaping command inputs to reduce system vibration,” *Journal of Dynamic Systems, Measurement, and Control*, Volume 112, pp. 76-82, March 1990.
- [5] W. Singhose, N. Singer, and W. Seering, “Time-optimal negative input shapers,” *J. of Dynamic Systems, Measurement, and Control*, Volume 119, pp. 198-205, June 1997.
- [6] N. Singer, W. Singhose, and E. Kriikku, “An input shaping controller enabling cranes to move without sway,” *ANS 7th Topical Meeting on Robotics and Remote Systems*, Volume 1, Augusta, GA, 1997, pp. 225-31.
- [7] K. Sorensen, W. Singhose, and S. Dickerson, “A controller enabling precise positioning and sway reduction in bridge and gantry cranes,” *Control Engineering Practice*, Volume 15, no. 7, pp. 825-837, July 2007.
- [8] A. Khalid, W. Singhose, J. Huey, J. Lawrence and D. Frakes, “Study of operator behavior, learning, and performance using an input-shaped bridge crane,” *IEEE International Conference on Control Applications*, Volume 1, pp. 759-764, September 2004.
- [9] D. Kim and W. Singhose, “Performance studies of human operators driving double-pendulum bridge cranes” *Control*

*Engineering Practice*, Volume 18, Issue 6, pp. 567-576, June 2010.

- [10] C. Kim, C. T. Haas, K. A. Liapi, and C. H. Caldas, "Human-Assisted Obstacle Avoidance System Using 3D Workspace Modeling for Construction Equipment Operation," *Journal of Computing in Civil Engineering*, Volume 20, Issue 3, pp. 177-186, May 2006.
- [11] D. Murray and C. Jennings, "Stereo Vision Based Mapping and Navigation for Mobile Robots," *1997 IEEE International Conference On Robotics And Automation*, Volume 2, pp. 1694-1699, April 1997.
- [12] F. Endres, J. Hess, J. Sturm, D. Cremers and W. Burgard, "3-D Mapping With an RGB-D Camera," *IEEE Transactions on Robotics*, volume 30, no.1, February 2014.
- [13] C. Stiller, J. Hipp, C. Rossig, A. Ewald, "Multisensor obstacle detection and tracking" *Image and Vision Computing*, Volume 18, Issue 5, pp. 389-396, April 2000.
- [14] G. Klein and D. Murray, "Parallel Tracking and Mapping for Small AR Workspaces," *6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, pp. 225-234, November 2007.
- [15] G. Klein and D. Murray, "Parallel Tracking and Mapping on a camera phone," *8th IEEE International Symposium on Mixed and Augmented Reality*, pp. 83-86, October 19-22, 2009.
- [16] "Simultaneous Localization and Mapping," S. Thrun and J.J Leonard, 2008, *Springer Handbook of Robotics*, Springer Berlin Heidelberg, Part E.
- [17] M.W.M.G. Dissanayake, P. Newman, S. Clark, H.F. Durrant-Whyte, M. Csorba, "A solution to the simultaneous localization and map building (SLAM) problem," *IEEE Transactions on Robotics and Automation*, Volume 17, pp. 229-241, June 2001.
- [18] A. Shapira, Y. Rosenfeld and I. Mizrahi, "Vision System for Tower Cranes," *Journal of Construction Engineering and Management*, Volume 134, Issue 5 pp. 320-332, May 2008.
- [19] J. Yang, P. Vela, J. Teizer and Z. Shi, "Vision-Based Tower Crane Tracking for Understanding Construction Activity." *Journal of Computing in Civil Engineering*, Volume 28, Issue 1, pp. 103-112, January 2014.
- [20] Y. Yoshida and K. Tsuzuki, "Visual Tracking and Control of a Moving Overhead Crane Load," *9th IEEE International Workshop on Advanced Motion Control*, pp. 630-635, 2006.