Chip-based Real-time Gesture Tracking for Construction Robot's Guidance

Ying-Hao Yu[a]*, Chun-Hsien Yeh[a], Tsu-Tian Lee[b], Pei-Yin Chen[a] and Yeu-Horng Shiau[c]

[a]Department of Computer Science and Information Engineering, National Cheng Kung University, Taiwan
[b]Department of Electrical Engineering, Chung Yuan Christian University, Taiwan
[c]Department of Electrical Engineering, National Yunlin University of Science and Technology, Taiwan
E-mail: yinhaun@gmail.com, ttlee@cycu.edu.tw, pychen@csie.ncku.edu.tw, shiauyh@yuntech.edu.tw

Abstract -

Mobile robots in automation construction have been designed for applications of craning, conveying, excavating, and floor polishing. These robots nowadays are equipped with various sensors to detect environmental information and finish tasks autonomously. When robot's navigating path needs to be rescheduled, the supervisor of robot can duly interrupt system and then redefines a new route for robot. In addition to robot remote control by radio signals, using digital camera to receive instructions from supervisor's gestures is also effective and can avoid the drawback of networked data routing. In this paper, we propose a gesture tracking system by simulating a traffic light baton to guide a differential drive robot in construction site. Here a real-time moving object detection first tracks supervisor's waves (gestures) with digital camera. Next the system determines guiding direction and steering angles based on fuzzy logic. All of our designs are implemented in single FPGA chip for operating under rigor environments. The experimental results demonstrate that proposed gesture tracking system is accurate and promising for chip-based gesture guidance on construction robots in the future.

Keywords -

Gesture Tracking; Moving Detection; Fuzzy Logic; Supervisory Control; FPGA

1 Introduction

Tracking pre-defined paths is one of the most important capabilities of mobile robots. The paths of robots are generally constructed in various configuration of tags or environmental features so that robots can detect and compare with onboard virtual maps. These paths and maps have to be installed and recorded beforehand in order to guarantee that robot can run on expected paths [1-4].

However, due to the changeability of construction sites, path tracking algorithms are inapplicable with pre-defined paths. For example, an indoor floor cleaning robot will bump into obstacles once the house fittings are changed. Although the robot designed with heuristic algorithm can learn and update maps after every new bump [3], this kind of control scheme in construction sites is unworkable in that heavy and large robot will cause serious safety issues to workers and facilities.

Besides, the algorithm called simultaneous localization and mapping (SLAM), which can recognize indoor environmental features, might overcome aforementioned problem by replacing heuristic algorithm with digital camera. Recently, the SLAM can achieved real-time processing by using algorithms of simpler training, improved classifying, steady-state operations, and high-end computers [4], [5]. The problem is that SLAM working under extremely changeable environments still remains a challenge to be resolved. Any failure of tracking or computing delay from SLAM will cause a disaster by lost robots.

In the construction sites, mobile robots have been designed for various applications of craning, conveying, excavating, road and tunnel inspecting, serving, and tiling machine [6-11]. The navigating paths of robots are usually determined by humans' volition. For instance, the path of a road inspecting robot could be randomly rescheduled depending on the obstacles, traffic conditions, and cracks found by the supervisor. Similarly, optimal manoeuvre and parking location of a mining haul truck is also dynamically determined by supervisor for relative position to excavator, turning space, loading of the truck, and terrain of mine. Consequently, a semi-automatic control system is assumed to be more suitable than a fully automatic system in construction sites.

A typical semi-automatic system in robotics is the supervisory control. This kind of system is composed of a supervisor and robot. The supervisor controls the performance and progress of assignments and duly interrupts and reschedules operations of robot. For the role of robot, it just needs to perform routine tasks with pre-defined control modes [10]. Such control mechanism might look inferior to a complete automatic system, but a safe working environment with heavy robot, therefore, can be secured from accident.

In this paper, we simulate a real-time hardware-based gesture tracking system for guiding a differential drive robot. The gesture tracking design on robot can update navigating path accordingly by detecting supervisor's gestures (waves). Comparing with traditional gesture recognitions, the palm tracking is practically replaced with coloured traffic light baton in construction sites. Such design avoids interferences of palm's tracking, which will break down under a crowded or nonideal illuminated environments. In addition, the range of a wave is generally defined as an approximative value in real-life operations. This makes fuzzy logic a suitable choice in our system to interpret the scale of a waving range. Finally, by considering the rigor environments of construction sites, all of our designs were realized in a single field programmable gate array (FPGA) chip in order to conform to criteria of lower power, cost, and installation dimension.

This paper is arranged as the follows. In the Section 2, gestures (waves) tracking algorithms based on moving object detection will be represented. Resource usages of FPGA chip design and demonstration will be shown in the Section 3. A short discussion is arranged in the Section 4, and the conclusion of paper is drawn in the Section 5.

2 Gesture Tracking Algorithms

Before starting gesture tracking off, traffic light baton's image from digital camera is first detected by our previous works of hardware-based real-time demosaicking and moving object detection [12], [13]. Once supervisor sweeps the traffic light baton, moving image will be immediately marked by our moving object detection and then tracked by hardware-based gestures tracking algorithms as the follows.

2.1 Tracking a Traffic Light Baton

We assume only a traffic light baton with specific colour that will be detected by robot's camera. The gravity point P of baton's moving marks can be derived by designating a rectangular tracking frame on marks. Each image picture produces one gravity point, and every five gravity points can determine the waving direction of traffic light baton by linking up first and last points.

Besides, detected gravity points sometimes need to be examined on monitor amongst waves. An additional sorting problem of gravity points arises from the different sequences between detecting and displaying gravity points. For example, as shown in Figure 1, if P_0 has to be first shown on monitor, P_1 will appear in next picture and so on. Thus five gravity points will not appear on monitor in one picture unless a sorting mechanism is designed as the follows:

1. Load the gravity points P_0 to P_4 into the shift registers R_0 to R_4 in turn, where the P_0 is the first tracked gravity point and P_4 is the last one.
2. Examine gravity points' coordinates (x, y) on image sensor array with registers pairs (R_4, R_3) and (R_2, R_1). The point that will first show on monitor should change to the left-hand position in each register pair.
3. With the similar operation, examine the register pairs of (R_3, R_2) and (R_1, R_0) continually and swap data as step 2.
4. Repeat steps 2 and 3 until the order of all gravity points in registers corresponding with the displaying sequences of monitor. Thus the most significant bit of R_4 will be the first point showing on the monitor, and the last point showing on monitor will be the least significant bit R_0.
5. Display gravity points by reading out shift registers' data from the most significant bit in order.

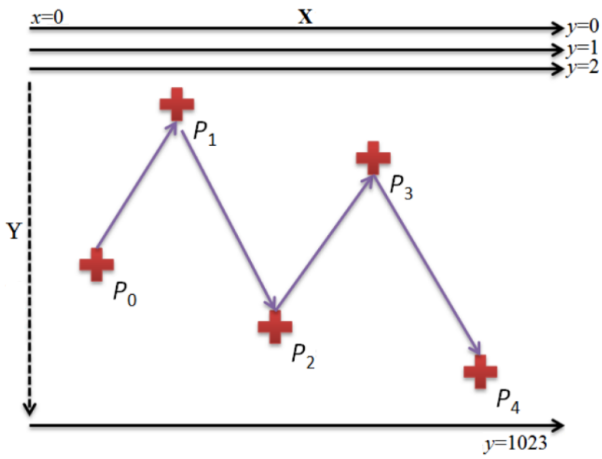


Figure 1. Possible distribution of gravity points on monitor

2.2 Confirmation of a Gesture

After determining a wave's direction, the next step is to confirm a valid gesture (wave) of supervisor. In this process, the first three waves initialize a new tracking and then the system confirms gestures by every wave. This mechanism can be achieved by recording wave directions in registers, as shown in Figure 2. In Figure 2(a), the registers L and R respectively denotes the left or rightward wave. The content of both registers will be (1,0) if traffic baton sweeps from the right- to left- hand side, and the (0,1) condition is for reverse direction. The operations in vertical direction are similar to the horizontal direction.

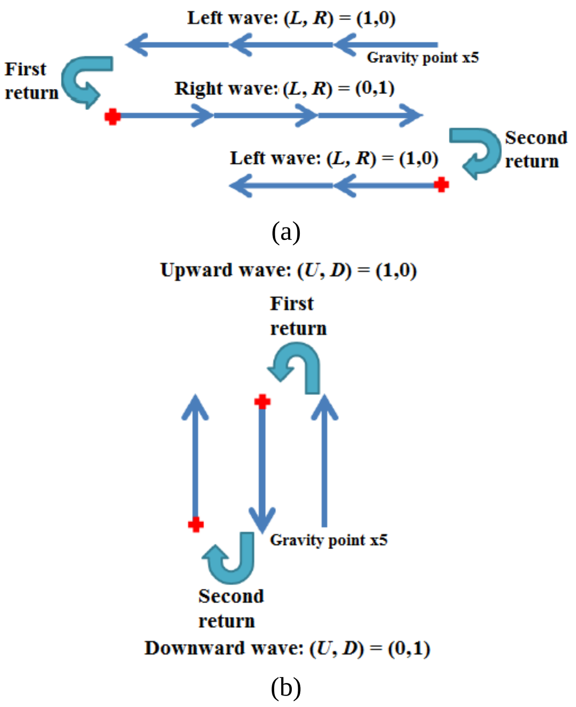


Figure 2. Registers' conditions for: leftward wave (a) and upward wave (b)

As shown in Figure 2(b), the register U of vertical waves represents upward direction and D is for reverse one.

Meanwhile, additional shift registers are necessary for counting the number of waves in different directions. The Table 1 shows an operation of two shift register groups (LX_2, LX_1, LX_0) and (RX_2, RX_1, RX_0) to record the leftward waves. The number of waving leftward was recorded in (LX_2, LX_1, LX_0) and waving rightward was in (RX_2, RX_1, RX_0). For the leftward waves, the registers (LX_2, LX_1, LX_0) were first filled with three waves after initialization and then (RX_2, RX_1, RX_0) were inhibited. Consequently, the system after initialization only allowed LX_2 to be continually cleaned and refilled. This mechanism is also similar to the waves in other directions.

The final process of tracking a gesture is to determine the initial point of a wave. As humans' behaviour, the same gesture will be confirmed if the initial points of waves are similar. Once the location of initial point is out of a range, the record of initial point will be reset and a wave of reverse direction might be considered. For the case of leftward waves, the traffic light baton sweeps from the right- to left-hand side, and the initial point CX will be determined by coordinate x on image sensor and the registers RX_0 and RX_1 as:

$$CX = \begin{cases} \dfrac{x_{RX0} + x_{RX1}}{2}, & if \quad x_{RX1} > 0 \\ x_{RX0}, & if \quad x_{RX1} = 0 \end{cases} \qquad (1)$$

Table 1. Data variation of shift registers for leftward wave

	LX_2	LX_1	LX_0	RX_2	RX_1	RX_0
Initial state	0	0	0	0	0	0
1st (return)	0	0	1	0	0	0
2nd	0	0	1	0	0	1
3rd	0	1	1	0	0	1
4th	0	1	1	0	1	1
5th	1	1	1	0	1	1
6th	0	1	1	0	1	1
7th	1	1	1	0	1	1

Operations of initial point for leftward waves are illustrated in Figure 3. A starting off area is defined by CX and errors E. Three cases in this figure can be discussed for:

Case 1:
The x_{RX0} is out of starting off area on the left-hand side. Tracking system with this case will still recognize a leftward wave, but the initial point CX will be updated and gradually moved to the left-hand side.

Case 2:
The x_{RX0} is within the range of starting off area. The leftward wave is confirmed with last CX.

Case 3:
Latest x_{RX0} locates on the right-hand side of starting off area. System with this case will immediately clean all waves' records in registers and then a new rightward wave might be confirmed depending on upcoming waves.

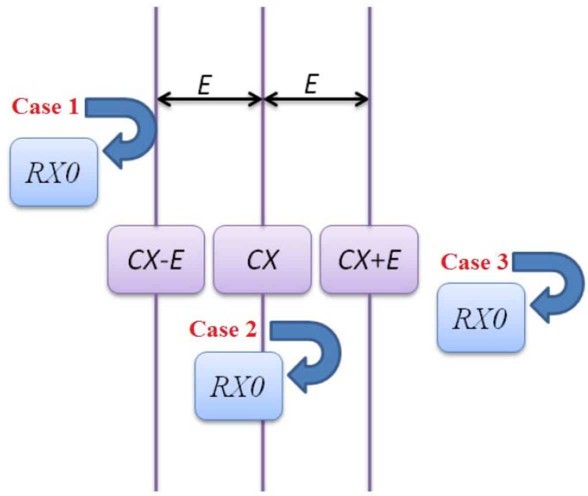


Figure 3. Operations of starting off area for leftward waves

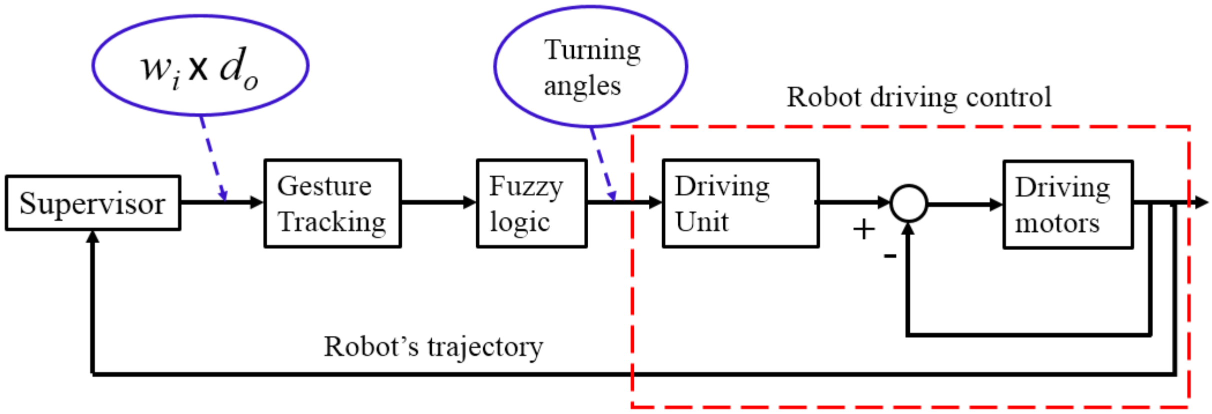


Figure 4. Steering angles control by using fuzzy logic to guide a differential drive robot

2.3 Guiding Robot with Fuzzy Logic

This subsection represents the determination of steering angles for a differential drive robot. The design difficulty arises from the magnification of camera lens that causes misjudgement with the same waving range at different locations. Consequently, guiding a robot by actual spatial scale on camera is impractical.

To resolve such problem, magnification of camera lens has to cooperate with fuzzy logic in proposed system. According to magnification of lens,

$$M = \frac{h_i}{h_o} = -\frac{d_i}{d_o} \tag{2}$$

h_o: height of object

h_i: height of projected image

d_o: object distance

d_i: image distance

the distance between traffic light baton and camera lens (d_o) can be derived from baton's height (h_o), projected image height (h_i), and baton's image distance (d_i). It can be seen that waving range w_i on monitor and d_o have an inverse proportional relationship, which approximates a constant of $w_i \times d_o$ for the same wave at different locations and helps us to resolve aforementioned misjudgement problem.

Figure 4 depicts the gesture control scheme for a differential drive robot. The supervisor assumed guiding a robot with expected driving directions. Proposed system on robot first tracked supervisor's gestures then derived steering angles by fuzzy logic. Here the inputs of fuzzy set are $w_i \times d_o$ cm^2 and the outputs of fuzzy logic are expected steering angles from supervisor. Steering angles from fuzzy logic will be translated into physical values by the driving unit of robot.

The fuzzy set of $w_i \times d_o$ is defined in the Figure 5. Here the triangular membership functions were adopted to denote "range zero (*RZ*)", "range small (*RS*)", 'range medium (*RM*)", and "range big (*RB*)" of $w_i \times d_o$ inputs. Triggered values of membership functions can then be expressed as:

$$\mu_{RZ}(x) = \begin{cases} \dfrac{440-x}{440}, & 0 \le x \le 440 \\ 0, & x > 440 \end{cases} \tag{3}$$

$$\mu_{RS}(x) = \begin{cases} \dfrac{x}{440}, & 0 \le x \le 440 \\ \dfrac{1320-x}{880}, & 440 < x \le 1320 \\ 0, & x > 1320 \end{cases} \tag{4}$$

$$\mu_{RM}(x) = \begin{cases} 0, & x < 440 \ \ or \\ & x > 2200 \\ \dfrac{x-440}{880}, & 440 \le x \le 1320 \\ \dfrac{2200-x}{880}, & 1320 < x \le 2200 \end{cases} \tag{5}$$

$$\mu_{RB}(x) = \begin{cases} 0, & x < 1320 \\ \dfrac{x-1320}{880}, & 1320 \le x \le 2200 \\ 1, & x > 2200 \end{cases} \tag{6}$$

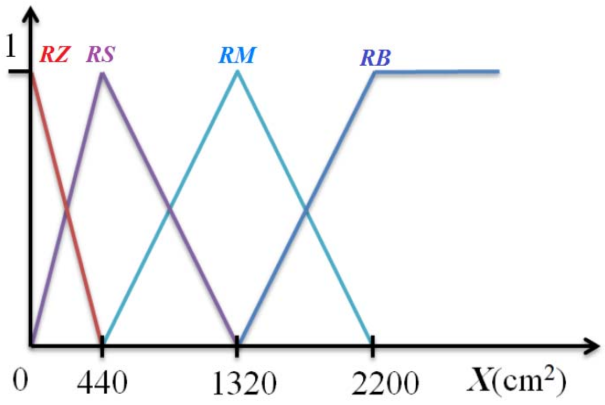


Figure 5. The fuzzy set of wave ranges

For the fuzzy rules, we defined *TZ*, *TS*, *TM*, and *TL* for the turning angles of "turn zero", "turn small", "turn medium", and "turn large" of the different drive robot. Each rule was respectively assigned with weight of 0, 1, 3, or 5, as shown in Table 2. Robot's steering scales to driving unit were assumed from 0 to 5 to control robot's two wheels in different speeds. Finally, as aforementioned discussion, the driving directions of robot were determined by the gravity points' tracking as supervisor's waves.

In the end, the algorithm of defuzzification was minimum inference engine [14] as:

$$\mu_{A'}(x) = \begin{cases} 1, & if \quad x = x^* \in X \\ 0, & other \end{cases} \tag{7}$$

where $\mu_{A'}$ is a fuzzy singleton of input, and triggered rule l and output $\mu_{B'}$ are,

$$\mu_{B'}(y) = \max_{l=1}\left[\min\left(\mu_{A^l}(x^*), \mu_{B^l}(y)\right)\right] \tag{8}$$

The actual output of steering angles was based on centre of gravity for singletons as [15]:

$$y^* = \frac{\sum_i \mu_{B'}(y_i) y_i}{\sum_i \mu_{B'}} \tag{9}$$

where y^* represents the outputs of steering angles by fuzzy, $\mu_{B'}(y_i)$ denotes the triggered values of membership function, and y_i is the weight of fuzzy rules.

Table 2. Fuzzy rules

Inputs	*RZ*	*RS*	*RM*	*RB*
Outputs	*TZ*	*TS*	*TM*	*TL*
Weights	0	1	3	5

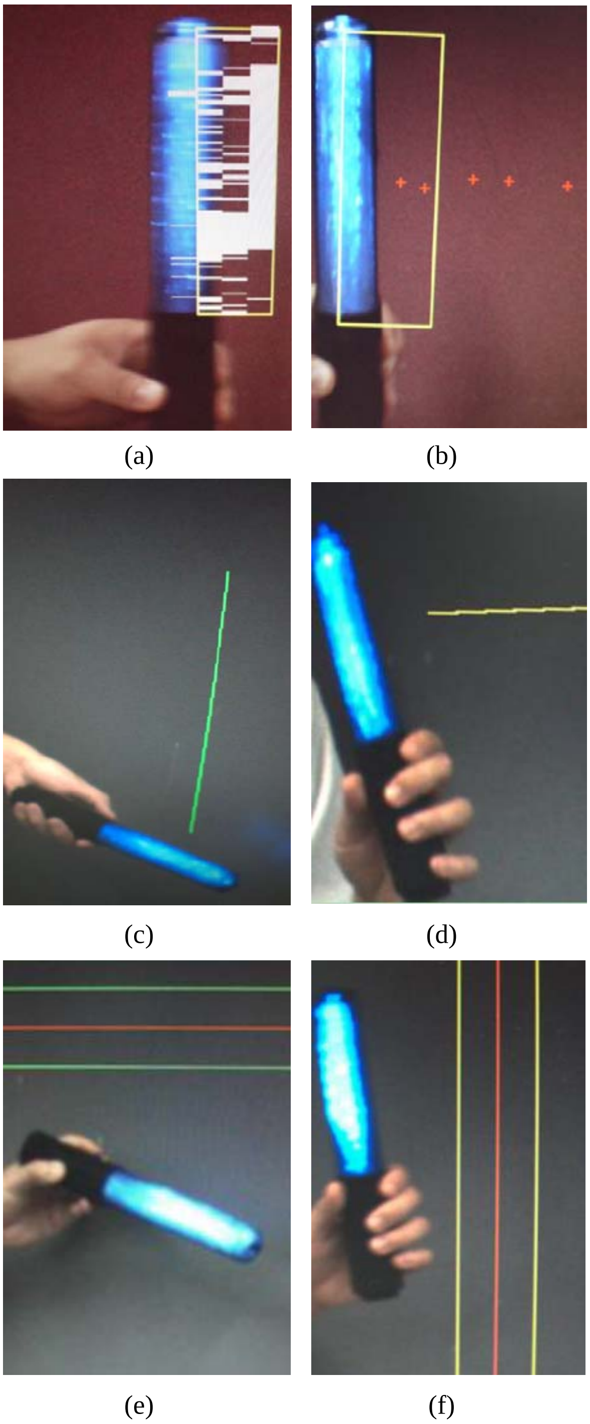
Table 3. Resources usages of proposed gesture tracking system in single FPGA

Designs	**LEs**	**%**
Total design	4,678	6.84
Full colour demosaicking	1,753	2.56
Moving detection	361	0.53
Gesture tracking	1,210	1.77
Fuzzy logic	1,354	1.98

3 Experimental Results

Proposed gesture tracking system is implemented in single Cyclone II 2C70 FPGA chip from Altera. Adopted digital camera module was set for 1280×1024 pixels resolution and pictures' frame rate were 12 fps, which is suitable for moving detection with walking speed. Total hardware resources usages of logic elements (LEs) are shown in Table 3. It can be seen that we only consumed 4,678 (6.84%) out of 68,416 LEs. This design involves the full colour demosaicking (2.56%), moving object detection (0.53%), gesture tracking (1.77%), and fuzzy logic (1.98%).

Figure 6 shows the snapshots of proposed gesture tracking. A sweeping traffic baton with blue light was first detected and marked with white stripes, and every stripe was composed of 32 pixels. Here the colour detection was simply designed by thresholds for baton's blue colour. Cluster of moving marks was real-time tracked by a rectangle on monitor, as shown in Figure 6(a). Based on the rectangular mark, gravity points of rectangles can be plotted as the crosses in the Figure 6(b). These crosses represented the trajectory of a wave. Similarly, as shown in Figures 6(c) and 6(d), traffic light baton's trajectories in vertical and horizontal directions were also represented by solid lines on monitor, and the marks of starting off areas are shown in the Figures 6(e) and 6(f).



(a) (b) (c) (d) (e) (f)

Figure 6. The snapshots of real-life gesture tracking on monitor

Finally, the errors of various waving ranges are listed in Table 4. In this test, the traffic light baton was vertically installed and moving horizontally on a rail in order to control the quality of waves. We tested different waving ranges for 10, 20, 30, 40, and 50cm while supervisor stood front camera from 1 to 3 meters away. Each test scenario collected 10 waves including initialization stage. It can be seen that errors of waves were efficiently controlled by a maximum of 15% and the most of these errors were ≤ 10%. Meantime, average errors were all under 10%. The accuracy of proposed system is sufficient to guide a robot by successively gestures.

4 Discussion

Using a traffic light baton to guide a robot is more practical than the palm tracking. Although guiding a robot by palm is natural, palm's detecting rate is usually deteriorated by nonideal illumination such as shadows and reflections. Moreover, additional algorithms for face recognition might also be required to track supervisor under a crowded environment. In contrast to palm recognitions, guiding robot with a traffic light baton can be realized with a simpler colour detection by using thresholds. Proposed real-time supervisory control is not only working well in the dark environments but also mitigates the interference from reflection.

Besides, the adequate and stable moving marks on traffic light baton are important to proposed system. It can be seen that accuracy of wave range depends on the number of moving marks, which represent the actual dimension of the traffic light baton. Accordingly, with 1280×1024 pixels resolution of image, there is a lower bound of traffic baton's height or length on monitor for 2cm. This limitation can be improved by a higher image resolution without increasing of computing delay [13].

Finally, due to the moving marks which are composed of stripes, slightly move at a further location will be thought as a static object by system. It leads invalid detection at first column by 3m away from camera and errors' fluctuations in each row, see the Table 4. Here a better detecting resolution can be achieved by decreasing the length of moving marks or with higher image resolution.

5 Conclusions

In this paper, we have proposed a practical gesture tracking for robot's guiding system, which is based on the real-time hardware chip designs. Essential gestures such as horizontal and vertical waves have been represented in paper. Our designs involve real-time demosaicking, moving detection, and colour detection by thresholds to detect the specific target, a traffic light baton. Mobile robot's steering angles are determined by fuzzy logic for different waving ranges. Proposed system also enables supervisor to guide robot flexibly with different distances from camera but without significant errors. Comparing with the traditional gesture guiding

Table 4. Horizontal waving errors with different distances away from camera

	Waving ranges				
	10cm	**20cm**	**30cm**	**40cm**	**50cm**
Maximum errors measured at 1m	10%	11%	10.67%	10.5%	9.6%
Maximum errors measured at 2m	6%	9%	6.67%	13%	12.4%
Maximum errors measured at 3m	N/A	15%	8.67%	9.5%	10.8%
Average errors measured at 1m	3.9%	6%	5.68%	6.65%	7.76%
Average errors measured at 2m	2.6%	3.95%	3.53%	9.85%	7.96%
Average errors measured at 3m	N/A	7.4%	3.97%	5.95%	9.6%

methodologies by using palm recognition, proposed system has the advantages of low-cost, real-time processing, miniature installation dimension, and a higher detection rate by using traffic light baton. Experimental results in this paper have demonstrated our gesture tracking system is promising for the supervisory control. The future work will focus on the implementation of gesture guiding system on mobile construction robots.

Acknowledgements

Supports from Ministry of Science and Technology, funded by the government of Taiwan under Grant NSC 99-2221-E-027-057-MY3, and Mr. Sheng-Jie Lin are gratefully acknowledged.

References

[1] D'Andrea, R and Wurman, P. Future challenges of coordinating hundreds of autonomous vehicles in distribution facilities. In *Proceedings of the International Conference on Technologies for Practical Robot Application*, pages 80–83, Woburn, USA, 2008.

[2] Tammet, T., Vain, J., Puusepp, A., Reilent, E., and Kuusik, A. RFID-based communications for a self-organising robot swarm. In *Proceedings of IEEE International Conference on Self-Adaptive and Self-Organizing Systems*, pages 45–54, Venezia, 2008.

[3] Tribelhorn, B and Dodds, Z. Evaluating the roomba: a low-cost, ubiquitous platform for robotics research and education. In *Proceedings of the International Conference on Robotics and Automation*, pages 1393–1399, Roma, Italy, 2007.

[4] Williams, B., Klein, G., and Reid, Ian, B. Real-time SLAM relocalisation. In *Proceedings of IEEE International Conference on Computer Vision*, pages 1-8, Rio de Janeiro, 2007.

[5] Kim, A and Eustice, R-M. Real-Time visual SLAM for autonomous underwater hull inspection using visual saliency. *IEEE Trans on Robotics*, 29(3):719–733, 2013.

[6] Chang, Y. C., Hung, W. H., and Kang, S. C. A fast path planning method for single and dual crane erections. *Automation in Construction*, 22:468–480, 2012.

[7] Navon, R. Process and quality control with a video camera, for a floor-tiling robot. *Automation in Construction*, 10:113–125, 2000.

[8] Tseng, Y. H., Kang, S. C., Chang, J. R., and Lee, C. H. Strategies for autonomous robots to inspect pavement distresses. *Automation in Construction*, 22:1156–1172, 2011.

[9] Yu, S. N., Jang, J. H., and Ham, C. S. Auto inspection system using a mobile robot for detecting concrete cracks in a tunnel. *Automation in Construction*, 16:255–261, 2007.

[10] Tiwan, R., Jeremy, K., and George, D. Bucket trajectory classification of mining excavators. *Automation in Construction*, 31:128–139, 2013.

[11] Osch, M., Bera, D., Hee, K van., Koks, Y., and Zeegers, Henk. Tele-operated service robots: ROSE. *Automation in Construction*, 39: 2014.

[12] Yu, Y. H., Kwok, N. M., and Ha, Q. P. Color tracking for multiple robot control using a system-on-programmable-chip. *Automation in Construction*, 20:669–676, 2011.

[13] Yu, Y. H., Ha, Q. P., and Kwok, N. M. Chip-based design for real-time moving object detection using a digital camera module. In *Proceedings of the International Conference on Image and Signal Processing*, pages 1993-1997, Tianjin, China, 2009.

[14] Wang, L. X. *A Course in Fuzzy Systems and Control*. Prentice-Hall, USA, 1997.

[15] Jantzen, J. *Foundations of Fuzzy Control*. John Wiley & Sons, USA, 2007.