http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 8 (1): 39-48, 2009
ISSN 1812-5638

Application of the Two-Axle Robot Tracing Object System with Multithread Control Technology

[1,3]K.C. Chen, [2]K.S. Hsu, [3]M.G. Her and [4]M.C. Chiu
[1]Geometrics Research Center, Chia Nan University of Pharmacy and Science,
[2]Department of Applied Geoinformatics, Chia Nan University of Pharmacy and Science, Taiwan 717, Republic of China
[3]Department of Mechanical Engineering, Tatung University, Taiwan 104, Republic of China
[4]Department of Automatic Control Engineering, Chung-Chou Institute of Technology, Taiwan 510, Republic of China

Abstract: The main aim of this study is to promote the efficiency of a control system using a multithread digital control design. In this system, the management of a computer's input and output information is handled appropriately by the program language. The multithread digital control design is used in the robotic arm's tracking system. The advantage of this multithread digital control design is to activate each procedure running simultaneously when the transient overload of the information's input and output in the control system occurs. Therefore, the time run in the multithread system will be shorter than that run in a traditional single thread system in which each procedure is lined up for running. In this study, case studies of multithread application used in image tracking and robot control are introduced. The results reveal that the speed of the tracking system can be improved by using the multithread technique under an immediate procedure plan.

Key words: Multithread, image tracing, feedback control

INTRODUCTION

As we know, a basic control system used in a computer is a flexible real-time controller. An important technique used in this system is multiple communicating loops for various executing cycles. A single loop periodically actuated by a counter is regarded as a thread; various loops performed concurrently in a control system are called a multithread.

The multithread controller is one kind of a multi-speed and digitized system. When a multi-rate controller is implemented on a computer-architecture, several problems may occur due to the limitation of computation and communication resources. In fact, there are phenomena that exist in the programmed net system that depend on the loop to conditional branches or random delays (Nilsson *et al*., 1998).

Because of multiple computation activities performed at the same time, the irregular time lags for data transmission and processing occur. In order to solve this problem, an appropriate technique in precisely manipulating the procedure of all computation activities is necessary. Therefore, in the multiple-control field, the development of a novel thread becomes an essential issue for researchers. In this study, in order to perform a multithread control and to solve an irregular time lag in a image-dealing system, a case study of procedural planning manipulated by a programming language to appropriately acquire and export the image in a trigger mode is proposed by Hu and Michel (1999) and McKenna *et al*. (1999). Each thread is responsible for its work in transmitting and processing new data individually. To construct a real-time control system, two steps are necessary. One is the establishment of an adaptive control performed within a specific time period. The other is to design an appropriate working-flow control to initiate the multithread controller. In order to assure the stability and performance of the system, the judgment of a threshold for the thread is required when the data is being acquired. However, up to now, the selection of a thread's threshold is still based on the experience rule.

Recently, most research work of the multithread technique is applied in the information processing of the information engineering field. They are widely used in data transmission and data operating systems. However, the application of the multithread technique in a control system is rare. Palopoli *et al*. (2002) applied the

Corresponding Author: Ko-Chun Chen, Geometrics Research Center, Chia Nan University of Pharmacy and Science, Taiwan 717, Republic of China

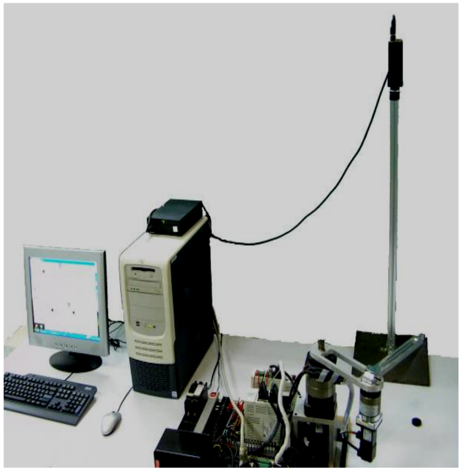

Fig. 1: The facility of a object tracking system in a two-axle robot

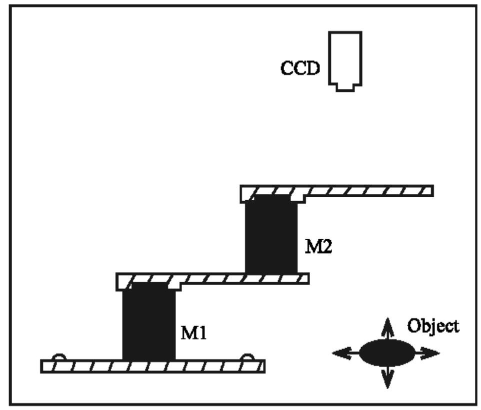

Fig. 2: CCD set up on the top of the system platform

multithread technique in the inverted pendulum system. By using a visual feedback system, the angle of the inverted pendulum is acquired so it may forward a gain to the actuator for further physical control much like the feedback control for a bicycle. Real-time control systems can be achieved by a multithread technique. In this study, the multithread control technique is adopted in a robotic visual tracking system. A case study of a visual tracking system for a two-axle robot in conjunction with the multithread control shown in Fig. 1 is proposed. Here, the system includes a two-axle robotic controller, a visual apparatus system and a programming interface. Moreover, the system is integrated and organized to fulfill the expected stability and performance.

SYSTEM DESIGN

Consequently, a variety of the system's control speeds between the single thread and the multithread are shown in the chart. The results indicate that the tracking speed in a two-axle robotic system can be improved using a multithread technique.

The Borland C++ Builder used in arranging the system's processes is a superior tool used with a visualized language that can be linked with a dynamic data base, graphic, application program and driven program, etc. The multithread function for actuating all processes concurrently can be fulfilled by linking up with both the video capture board and motion control card in conjunction with a system control program created by the Borland C++ Builder.

System introduction: The system hardware has a grayed CCD, a real-time video capture board (RTV-24), a servo motion control card (PCI-8134) and a two-axle robotic system platform. The whole visual-tracking system with a two-axle robotic arm is completed by setting up the man/machine interface and system control program written by Borland C++ Builder.

This system has a grayed CCD (a vision import device) set up on the top of the system. It can look down vertically on the complete system platform. To meet two requirements in the visual tracking system, two kinds of models are developed. First model as indicated in Fig. 2, the CCD set up on the top of the platform can keep tracking the object on the plane and locking the specified location. Because of the shelter effect from robotic arm, a variety of locations will exist.

Introduction to hardware specification: As shown in Fig. 3, PCI-8134 is a servo motion control card which has 4 axis steppers and can provide a high frequency pulse. The speed of electric motor can be varied by tuning the pulse's frequency. Moreover, the PCI-8134 not only can reduce the noise interference at signal input/output, but also can export the feedback signal.

RTV-24 is a video capture board which has four channels. It has a real-time video output function which has a perfect control module so that it can quickly acquire visual information. The captured video can be one or more pictures. Moreover, the dpi and visual correction in illumination can be arbitrarily adjusted.

In regard to the first axis (M1) of the robotic arm, the devices we use are (1)the servo drive of MITSUBISHI MR-J2S -40A; (2) the servo motor of 3AC 129V/2.3A 400 W/3000 rev min^{-1}; (3) the decelerator ratio of 1/20; and (4) the axis length of 20 cm. For the second axis (M2) of the robotic arm, the related devices we use are: (1) the servo drive of MITSUBISHI MR-E-10A; (2) the servo motor of 3AC 105 V/0.7 A 100 W/3000 rev min^{-1}; (3) the decelerator ratio of 1/20 and (4) the axis length of 25 cm.

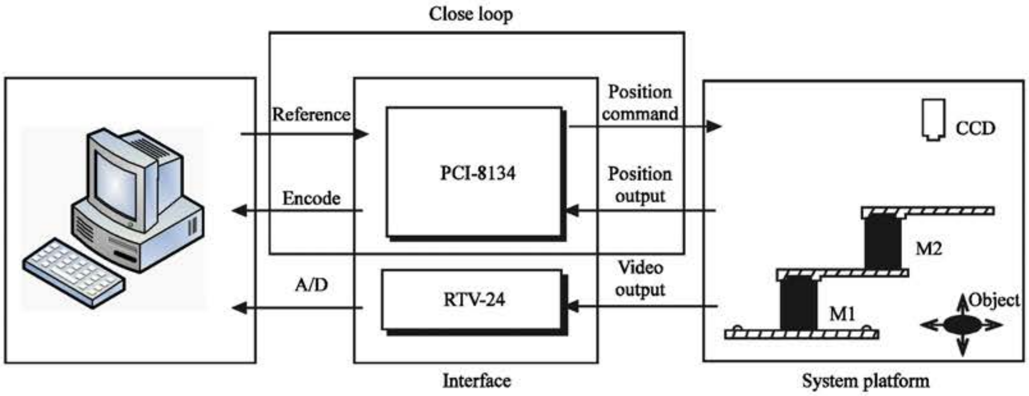

Fig. 3: Layout of the hardware structure

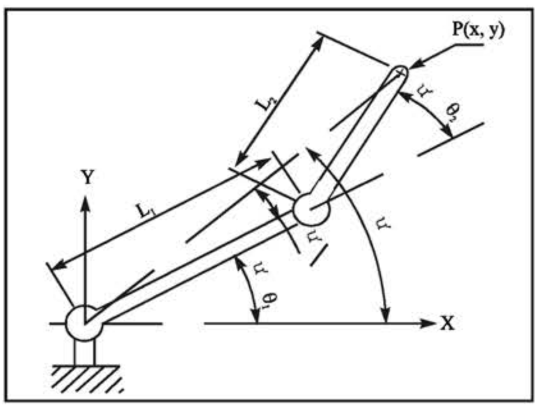

Fig. 4: Two degrees of freedom for a two-axle robotic arm

The two axels structure and manipulation method: The main concept of the visual-tracking system is to acquire the characteristic image of the object, to obtain the related coordinates and to activate the robotic arm to the specified location. To facilitate the kinetic motion in a robotic arm, the rotating angles of each axle is preset by using the inverse transformation of kinetic equations. Subsequently, the computer will demand the servo motors to rotate to specified angles so that the robotic arm can reach the calculated location. The two degrees of freedom for a two-axle robotic arm are shown in Fig. 4.

By using Inverse Kinetics equations, the related coordinates in both x axis and y axis are:

$$x = L_1\cos\theta_1 + L_2\cos\theta_1\cos\theta_2 - L_2\sin\theta_1\sin\theta_2 \quad (1)$$

$$y = L_1\sin\theta_1 + L_2\sin\theta_1\cos\theta_2 + L_2\cos\theta_1\sin\theta_2 \quad (2)$$

where, L_1 and L_2 are the lengths of two arms, θ_1 and θ_2 are the rotating angles with respect to two arms.

The rotating angles θ_1 and θ_2 can be obtained as:

Table 1: Required memory space with respect to various colors

Diagram picture element	Color format	Picture element depth (byte)	Required space (byte)
320×240	Grayscale	1	76800
320×240	Monochromatic	1/8	9600

$$\theta_1 = \tan^{-1}\left[\frac{y(L_1 + L_2\cos\theta_2) - xL_2\sin\theta_2}{x(L_1 + L_2\cos\theta_2) + yL_2\sin\theta_2}\right] \quad (3)$$

$$\theta_2 = \cos^{-1}\left(\frac{x^2 + y^2 - L_1^2 - L_2^2}{2L_1L_2}\right) \quad (4)$$

Image processing: By using the image processing technique, the image will be clearer or partially strengthened. The image color of grayscale or full-colors depends on the visual-capture apparatus. Here, a grayed CCD which is the grayscale format in the image is adopted in this study. For the characteristic data of 320 pixels in width and 240 pixels in height, the composition of the picture can be regarded as 320*240 pixels in which the pixel is the fundamental unit. The available color used in a pixel will be increased when the byte number of a pixel increases. Likewise, the required memory will be bigger.

The simplest and effective method in dealing with image processing is to simplify the depth of the image's color by using both the grayed processing and threshold technology to filter out the unnecessary color depth. Consequently, the location of an object can be obtained using a visual-tracking rule. The required memory with respect to various colors is shown in Table 1.

The threshold technology brightens or darkens the input image. Each image is classified as two kinds of gray values (0 or 1). The color of the image is therefore simplified as the foreground and background, where the foreground is the targeted image and the background is the unnecessary image. By using the threshold technology onto an image in which the variety of gray values around the edge of the object is small, the targeted image can be clearly identified by the observer.

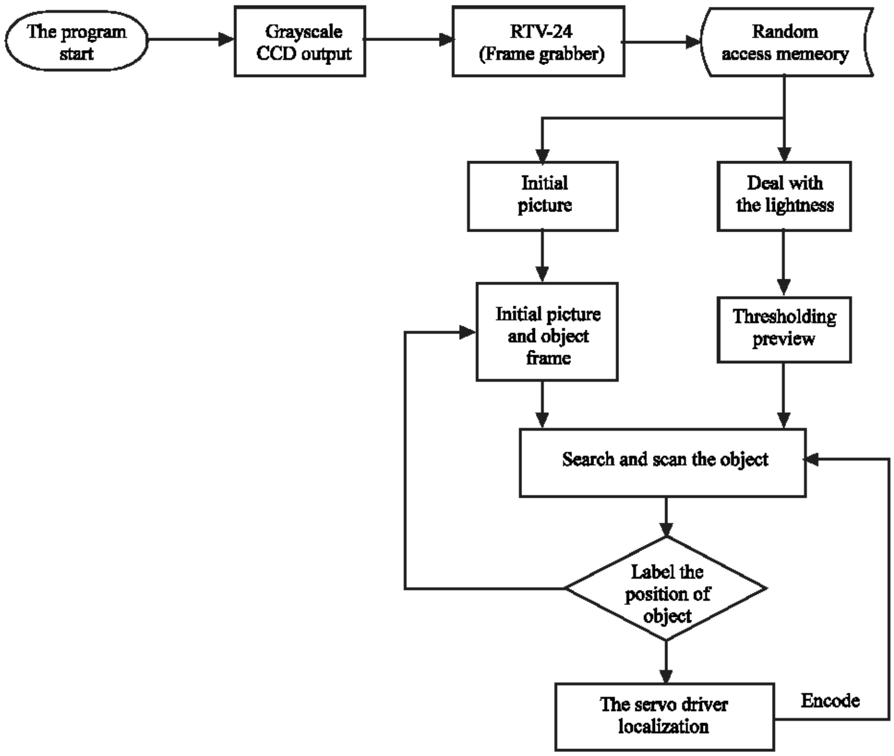

Fig. 5: Program structure for a two-dimension visual system

A comparison of the initial gray value and the cut-off gray value for a specified image is required. The cut-off gray value is adjustable. Because of influences on image recognition, the selection of an appropriate threshold value is essential. The targeted image will be filtered out if the selected threshold value is incorrect; therefore, threshold is called the process sieve for image processing (Tsechpenakis *et al*., 2004).

A program structure used in a two-dimension visual system is shown in Fig. 5. When the program is started up, the input image can be captured via RTV-24 video capture board of an image import device and transmitted to the computer memory. The acquired image will be shown in two kinds of picture frames on the pc. The first picture frame is used to demonstrate the initial image and the location of the targeted image after the visual-tracking process. The second picture frame is used to show the targeted image after the environmental luminance process. By using the environmental luminance process, the influence of the environmental lightness, which may worsen the recognition of the targeted image, will be reduced. In order to get rid of the background image after the threshold process, the left image should be the targeted image. Moreover, the location of the targeted image will be found using the scan-searching method. Subsequently, the location of the targeted image will be marked in the first picture frame. The related coordinates will be transmitted to the servo motor to move the robotic arm. Moreover, the encoded value will be fed back via the servo motion control card. The position comparison between the robotic arm and the targeted object will be continued until the targeted object is caught.

The vision tracing law: There are two important issues in visual-tracking; the speed and accuracy. A compromise between speed and accuracy is essential. The visual-tracking law is mainly used to distinguish the position of the object. A better visual-tracking law will shorten the searching time or provide a more accurate center position for the object.

An appropriate visual-tracking law will provide both a quick and efficient way to lock the position of the object. If a huge object is being searched for, the required time is lengthened (Klancar *et al*., 2004). Reducing the time for scanning and quickly acquiring the object or recognizing the object image within a complicated picture is crucial. Therefore, using the object's characteristic data, which may improve the recognition speed and accuracy in the visual-tracking system, is important.

Here, as shown in Fig. 6, we use the scanning method which starts at the left/upper and moves to the right/lower.

The scanning process will stop when an appropriate color with respect to a pixel is found. Without searching

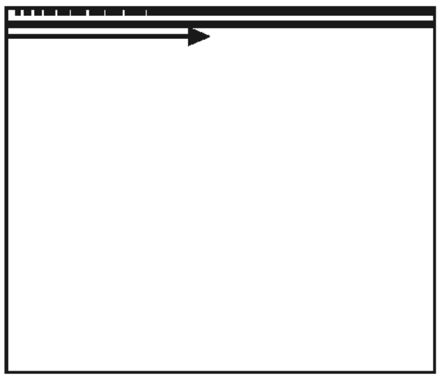

Fig. 6: Scan of rows

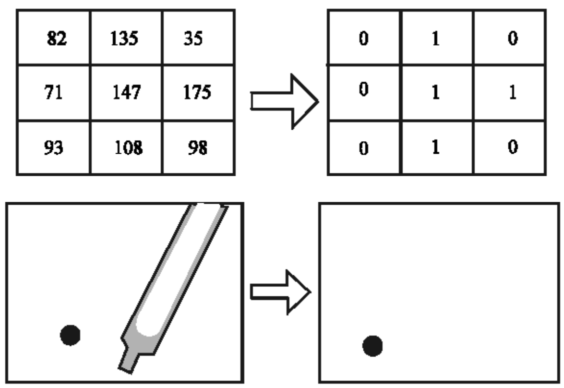

Fig. 7: Threshold transforming process

for other regions, time is small. Because the threshold value is preset as the gray value of the object, the object image can be separated from the background by the threshold process. Next, the object's dot distribution will illustrate the position of the object. As shown in Fig. 7, the gray value will be classified as 1 when the initial gray value is higher or equal to the threshold of 100. On the other hand, the gray value will be classified as 0 if the initial gray value is less than the threshold of 100.

The classification of images is necessary to capture the characteristic of the image, i.e., the distribution of the image print and the profile of the image grain. To obtain the color distribution of the image, the simplest way is to build up the histogram of the color. For example, similarity can be judged by a comparison of a grayed image before and after the threshold process in a histogram. Likewise, the characteristic of an image print can be acquired through the direction and reiteration of the image print. For instance, the print of various woods may be in a horizontal or a vertical direction. In this paper, the characteristic of color distribution for an object's image is adopted for the recognition of object position (Kaewtrakulpong and Bowden, 2003; Kim *et al*., 2001).

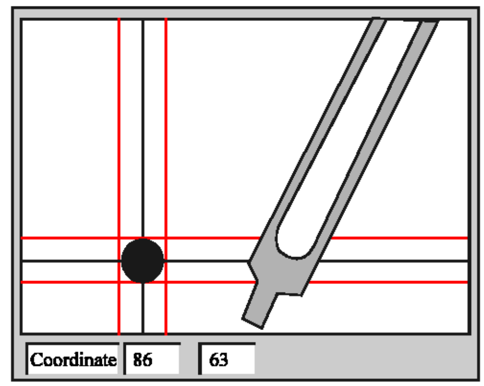

Fig. 8: Full-range picture scanning

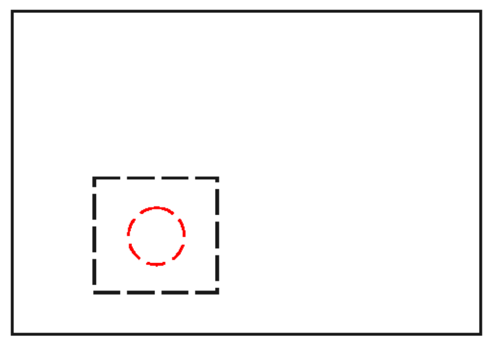

Fig. 9: Smaller range scanning (four times of the object's area)

The object scan law: A general judgment principle is used in this study. The complete image pixels are scanned to distinguish the accepted pixel in which the gray value can meet the characteristic of the object. All the qualified pixels are stored in the array temporarily. Subsequently, the maximum and minimum values in both the x and y axis can be obtained. The centre of an object can be acquired by the average of these values. However, even though this method can exactly obtain the center coordinate of the object, the time spent is longer. Therefore, scanning for other images will be discarded when the characteristic of the object is captured during the scanning process.

To increase speed and assure the accuracy during image recognition, a compound scanning rule-an improved scanning method-is introduced. As shown in Fig. 8, a full range of an image is preset for scanning for the first picture. After the image scanning process in conjunction with the characteristic of the object, a preliminary centre for the object is acquired. Subsequently, a new smaller range for the next scanning is preset. As indicated in Fig. 9, the region of the new scanned range is four times that of the object. The visual-tracking system will be actuated at the new scanned range when the object moves. Because the

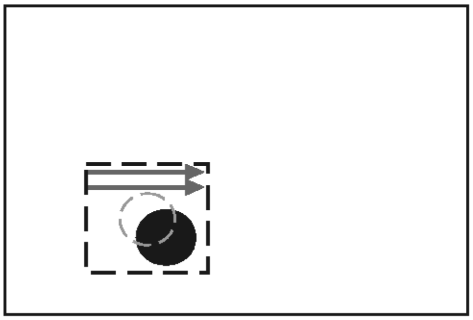

Fig. 10: Scanning of the object for two or more times

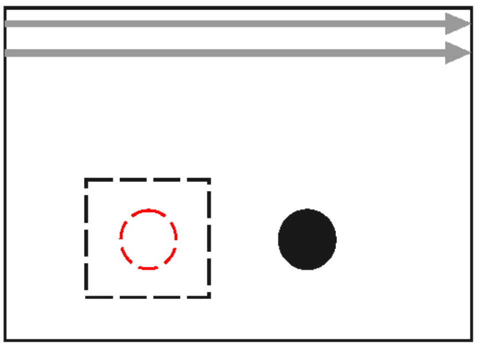

Fig. 11: Missing scan for the object

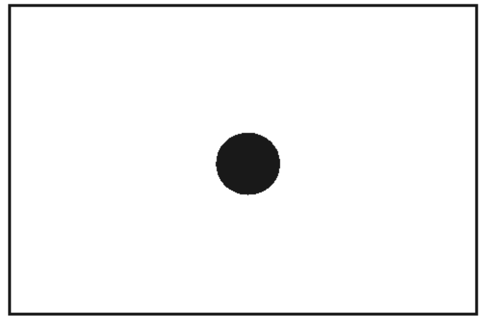

Fig. 12: Full pixels for a judgment scan (one pixel interval)

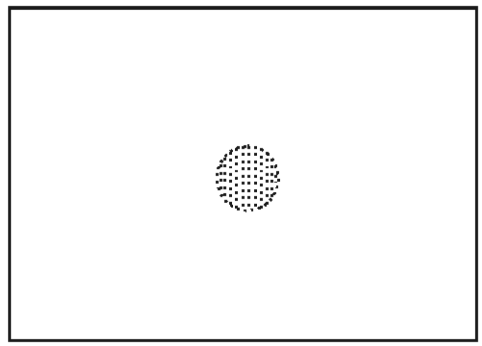

Fig. 13: Half pixels for a judgment scan (two pixels interval)

scanned region shown in Fig. 10 becomes smaller, the recognition speed increases. If the object is missed in the scanning procedure as shown in Fig. 11, a full-range picture scanning will be restarted that will find the new centre of the object (Tsechpenakis *et al*., 2004).

The calculated difference of the distance and vision light correction: Under certain circumstances, tracking speed is more important than accuracy. Therefore, a higher visual-tracking speed obtained by lowering precision is required. By enlarging the span of the scanning distance, speed can be improved. For example, a picture image of 320*240 pixels has 76800 pixels. For a one pixel interval scanning, 76800 pixels for judgment are required (Fig. 12). If there is a two pixel interval scan, the judgment is reduced to 38400 pixels, Fig. 13. The required judgment number with respect to the scan interval is shown in Table 2 (Tsechpenakis *et al*., 2004).

As discussed earlier, the object's dot distribution will be found by using the gray value as the threshold so that the image of the object is separated from the background image. Subsequently, the qualified pixels are saved in a temporary array. For normal lighting conditions, the total number of qualified pixels in the array is recorded first. For example, if the total number of pixels in the array is

Table 2: Required judgment number with respect to the scan interval

Picture element	Scan Interval	Ratio	Required judgment No. (byte)
320*240	1	1	76800
320*240	2	1/4	19200
320*240	3	1/9	8533

500 pixels, the allowance error value is 500±10 of the dot. If the lighting condition is higher, the total number of pixels in the array will be below 500 pixels. We then need to increase the threshold value that we may increase the array's dot number in the range of 500±10 pixels. However, the threshold value will be reduced when the lighting is darker. Consequently, the threshold can be adjusted to the environmental lighting condition (Navon, 2000).

SYSTEM STRUCTURE

Single thread structure: As shown in Fig. 14, a flow diagram of a single thread system is utilized to acquire consecutive images used by a grayed Camera Catch Device (CCD) and then transfer them to the computer memory by the video capture board. Subsequently, environmental luminance in deleting the unnecessary

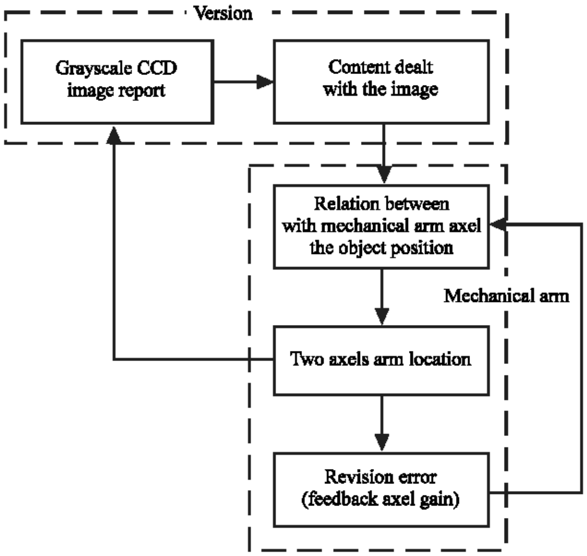

Fig. 14: Flow diagram of system flowchart

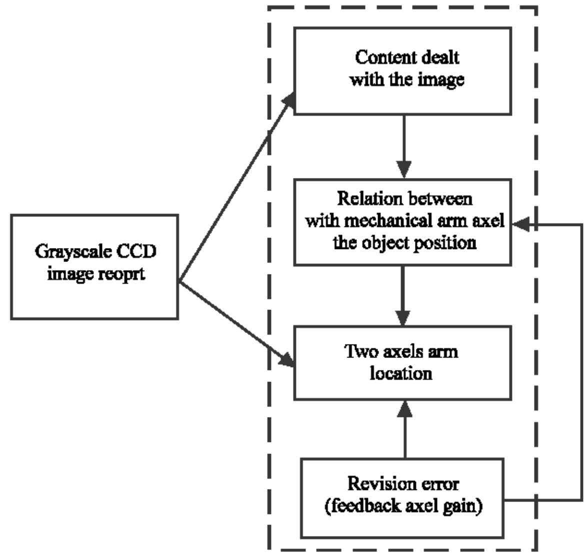

Fig. 15: Flow diagram of a multithread system

image data is performed. Next, the image-tracking algorithm is applied to the two-axle robot system to search for the location of a specified object. By using the kinetic relationship between the robotic arm and the object, the required actuated value of the servo motor is obtained. In order to achieve a real-time visual-tracking function, the image-tracking system will keep tracking the object, submitting the required actuated value of the servo motor and performing a position feedback control by a motion control card continuously while the object keeps moving.

Multithread structure: We know the single thread got four jobs to do. They is getting image and processing. Arm move to position with image. Servo motor counter feedback to adjust servo motor to indicated position. Calculate object position to arm position difference and every servo motor rotate degree.

As shown in Fig. 15, the multithread control process is established by the Borland C++ Builder. The program is initiated to actuate various processes (the environmental luminance process, the position calculating process between the object and the robotic arm, the orientation process of the robot and the feedback gain process) simultaneously when the threshold of multithread start-up condition is triggered. Moreover, in order to assure the stability and efficiency of data transmission in the visual-tracking system, the required running times with respect to each process are preset concurrently.

Symmetric multiprocessing: There are many operating systems which can support the multiplexing and the multiple CPU architectures on the market. The fundamental operation techniques are based on the multithread technique.

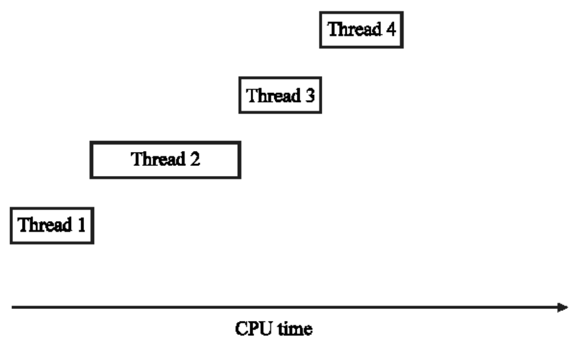

Fig. 16: The working for non-multithread operation

For a general single thread operation, the working processing is shown in Fig. 16. As indicated in Fig. 16, the tasks are lined up. The next task will be started when the front task is completed. For a multithread operation, the working process is depicted in Fig. 17. As indicated in Fig. 17, each task will be processed when they get into the task queue. The CPU is shared with all the tasks. By using the multithread working procedure, all the processes such as image processing, the process of position calculation and the actuating submission process, will be performed concurrently. Unless a priority is preset, the right to run shall be equal.

In the windows operating system, it seems that all the tasks are being processed. However, because the CPU resources are limited, the task will be terminated when the

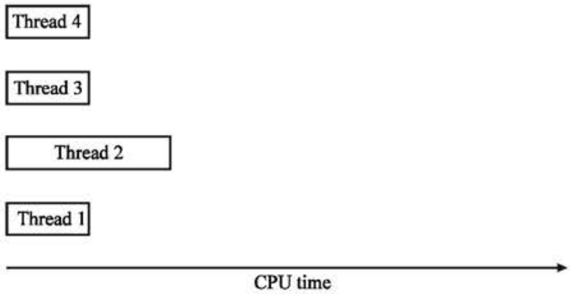

Fig. 17: The working for multithread operation

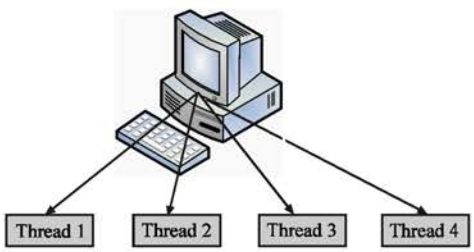

Fig. 18: The Scheme of multithread resources

preset time is up. Therefore, the CPU service will be shifted to the next tasks in turn. By quickly shifting the CPU service to the other tasks, the operating system will be multiplexing for all the processes. However, for the multithread operation, each thread will act as a single program. As indicated in Fig. 18, the CPU will be divided into several parts for each process. Each of the processes will be carried out individually and equally. No one process will be allowed to occupy the CPU service unless the priorities have been adjusted (Chou *et al*., 2006).

As shown in Fig.19, even though the multithread control actuates four kinds of thread processes at the same time, the internal data transmission will be ordered so that it meets the purpose of the visual tracking system. The image-acquisition process is started when the visual-tracking system receives the control command from the man/machine interface. Subsequently, by using the environmental luminance and the threshold technique, the object's position can be obtained so that it can calculate the feedback gain to actuate the servo motor. Thread 3, a data logging which is an independent process, will continuously feedback the robotic arm's position to thread 2 so that a location comparison between the object and robotic arm can be made. Moreover, the feedback encodes can be calculated and forwarded to the servo motor for further revision.

A comparison of position between the robotic arm and the object is continuously performed by receiving the

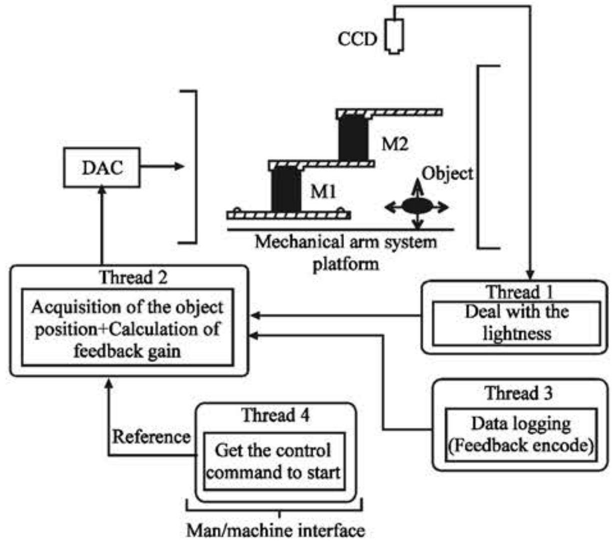

Fig. 19: An architecture of the multithread control system

encode value. The system can judge if the robotic arm reaches the object by calculating the feedback gain. At the same time, the system's error allowance will be preset. The robotic arm will stop moving when the gain is located within the error range. In case of a very small gain condition, the arm will be interfered with randomly. Therefore, presetting the error allowance in the system will assure that the visual-tracking system is stable (Palopoli *et al*., 2002; Conticelli *et al*., 1999).

RESULTS AND DISCUSSION

The multithread is most support on information techniques. We use the technique in automation control to improve control system effect. This technique used on vision of automation control system that we can improve system process time.

The multithread system need trigger thread relation and condition. That avoided the system dead lock. We define to this experiment terms. camera to object 90 cm, the diameter of object is 3 cm and 32 pixels. Put object to the center coordinate (0, 0) in the image of the object thing in experiment method. Let system find object, Arm move to object above. Then we move object 7 and 14 cm direction by X axis (Fig. 20). We record system active total time. Compare single thread and multithread of active efficiency.

The application of the multithread digital control in both visual-capture and signal-control reveals that the visual-tracking speed can be improved under the multithread system in conjunction with real-time schedule-planning. A variety of tracking speeds with respect to the

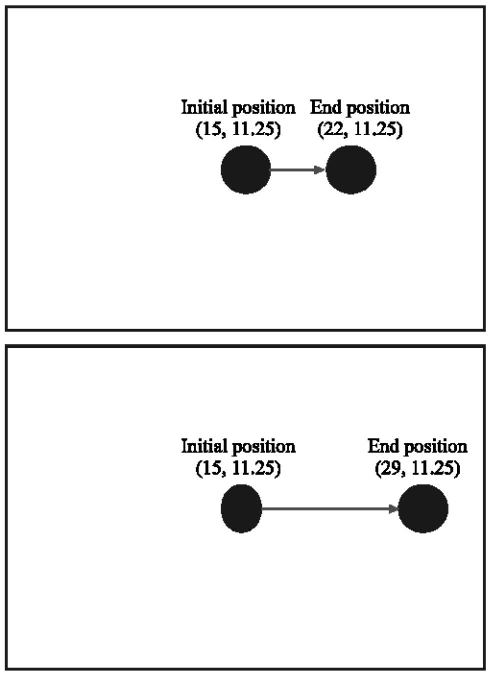

Fig. 20: Test path of object

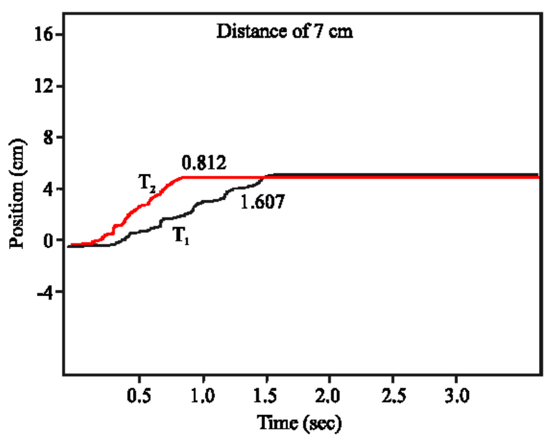

Fig. 21: The location response with respect to time for both the single thread and the multithread control system (Distance = 7 cm, Allowance = ±1 cm)

various initial distances between the robotic arm and the object is investigated. Similarly, a comparison of the tracking speed with respect to a single thread and multithread control is evaluated. To obtain the desired result, some parameters (such as the rpm of the servo motor, the number of the program's iteration loops and the iteration times, etc.) should be prefixed as the same condition in the experimental work.

As shown in Fig. 21, under the specified distance of 7 cm, the curve T_1 is the tracking speed in the single

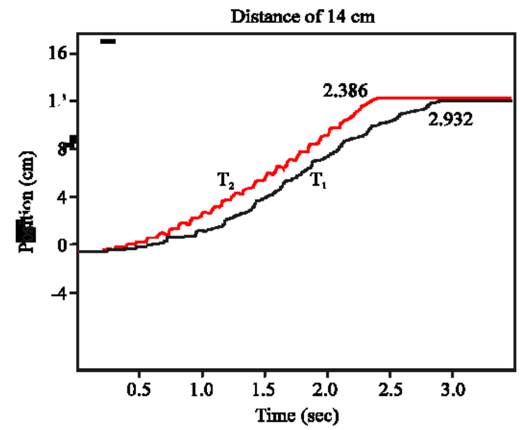

Fig. 22: The location response with respect to time for both the single thread and the multithread control system (Distance = 14 cm, Allowance = ±1 cm)

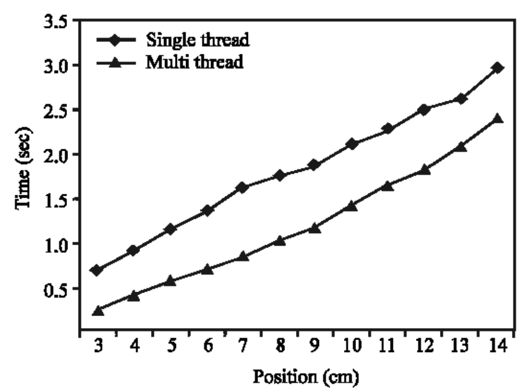

Fig. 23: The responses of running time with respect to various distances for both the single thread and multithread control system

thread control system. In addition, under the specified distance of 7 cm, the curve T_2 represents the tracking speed in the multithread control system. Likewise, as indicated in Fig. 22, under the specified distance of 14 cm, the curve T_1 is the tracking speed in the single thread control system. In addition, under the specified distance of 14 cm, the curve T_2 represents the tracking speed in the multithread control system. The results in Fig. 21 and 22 reveal that the tracking speed in the multithread control system is obviously higher than the single thread control system.

To appreciate the variety of running times between the single thread and multithread control, the responses of their running times with respect to various distances are shown in Fig. 23. As shown in Fig. 23, it is obvious that the required running time in the multithread control is shorter than that of the single thread control. Moreover, the response curve of the multithread control is more stable than that of the single thread control. Consequently, the multithread technique used in the visual-tracking system can improve the system's tracking speed.

CONCLUSIONS AND FUTURE WORK

For a control system, the executing speed of a multithread will be higher than that of a single thread. The results reveal that the multithread used in the visual-tracking system can improve the tracking speed. The multithread will be superior in both performance and stability by using an adaptive control which can divide the control system into several independent processes and arrange a flexible working sequence and running time. Moreover, the running time will be shortened. Therefore, the system will lean toward real-time control. In this study, instability in image-processing exists. The system will misjudge the image of the object when the background image is too complicated or a similar object rushes into the visual-working area. Therefore, to reach perfect image-recognition, a more effective scanning scheme is needed to overcome the above drawbacks.

REFERENCES

Chou, S.C., W.K. Lo and C.W. Lai, 2006. Information flow control in multithread applications based on access control lists. Inform. Software Technol., 48: 717-725.

Conticelli, F., B. Allotta and C. Colombo, 1999. Hybrid visual servoing: A combination of nonlinear control and linear vision. Robotics Autonomous Syst., 29: 243-256.

Hu, B. and A.N. Michel, 1999. Some qualitative properties of multirate digital control systems. IEEE Trans. Automatic Control, 44: 765-770.

Kaewtrakulpong, P. and R. Bowden, 2003. A real time adaptive visual surveillance system for tracking low-resolution color targets in dynamically changing scenes. Image Vision Comput., 21: 913-929.

Kim, M., J.G. Jeon, J.S. Kwak, M.H. Lee and C. Ahn, 2001. Moving object segmentation in video sequences by user interaction and automatic object tracking. Image Vision Comput., 19: 245-260.

Klancar, G., M. Kristan and R. Karba, 2004. Wide-angle camera distortions and non-uniform illumination in mobile robot tracking. Robotics Autonomous Syst., 46: 125-133.

McKenna, S.J., Y. Raja and S. Gong, 1999. Tracking color objects using adaptive mixture models. Image Vision Comput., 17: 225-231.

Navon, R., 2000. Process and quality control with a video camera, for a floor-tilling robot. Automat. Construct., 10: 113-125.

Nilsson, J., B. Bernhardss and B. Wittenmark, 1998. Stochastic analysis and control of real-time systems with random time delays. Automatica, 34: 57-64.

Palopoli, L., L. Abeni, G. Bolognini, B. Allotta and F. Conticelli, 2002. Novel scheduling policies in real-time multithread control system design. Control Eng. Practice, 10: 1091-1110.

Tsechpenakis, G., K. Rapantzikos, N. Tsapatsoulis and S. Kollias, 2004. A snake model for object tracking in natural sequences. Image Commun., 19: 219-238.