

# Design and Implementation of Intelligent Vehicle Control System Based on Camera Sensor

Shubra Deb Das, Yang Jue, Muktadir Rahman Chowdhury, Mary Cindy Ah Kioon

USTB Smart Car Lab

University of Science and Technology Beijing (USTB)

Beijing, 100083, China

email: shubra.debdas@gmail.com, yangjue@ustb.edu.cn, alvychy@yahoo.com, cindyak86@gmail.com

**Abstract**— In this paper we propose an intelligent vehicle system design with MK60N512VMD100 as the micro-controller unit and CMOS image sensor to obtain image information of the track. Our proposed comparator circuit instead of an A/D conversion to obtain the binary image enables us to achieve a bigger field of view and a higher processing speed. A unique algorithm to extract track information and to successfully extract the black guide line is devised. We also design a PID closed-loop control to adjust the rotate speed of driving electromotor and direction of steering electromotor. The results indicate that our design proposal is feasible. The intelligent vehicle could successfully achieve a high speed, while maintaining an excellent stability.

**Keywords**- ARM processor; Sensor; Comparator; Line tracking; PID

## I. INTRODUCTION

An intelligent vehicle is a system composed of various functions that include, but not limited to, context-aware decision making and automatic drive. To implement these functions, we need basic knowledge of automatic control, pattern recognition, sensor technology, electronics, computer science, mechanical engineering. The objectives of this project are (1) to design and develop a line following autonomous robot, (2) to accurately detect the black guide line and (3) to control steering angle and motor speed with the help of path recognition.

## II. SYSTEM ARCHITECTURE

Here is a brief overview of how the intelligent vehicle's line following mechanism works: the CMOS image sensor captures the image of the track and outputs PAL-D signal, which then gets processed by MAX941 and LM1881. LM1881 is a video sync separator that extracts timing information including composite, vertical sync and odd/even field data from the signal (PAL). MAX941 conducts binary image conversion. The outputs from these two chips are received into MK60N512VMD100; it then controls the motor and servo based on the above mentioned information. A rotary encoder has been used to measure the speed of the car. The input capture function of MK60N512VMD100 has been used to obtain speed and distance information by reading pulse. To control servo and motor, PD control and PID control have been used respectively. Through PWM, the drive circuit adjusts the power of the motor. The block diagram of system architecture is as shown in Fig. 1.

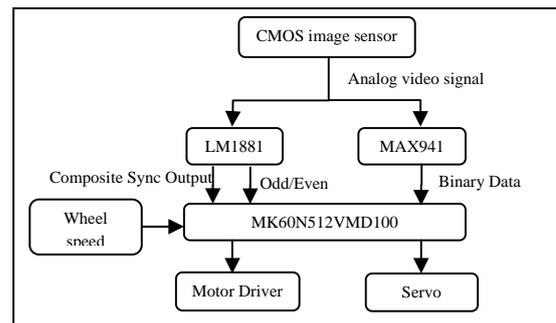


Figure 1. Block diagram of the system

## III. SENSOR CHOICE

### A. Camera

CMOS sensor can be implemented with fewer components, uses less power and provides faster readout than CCD's. To collect track information, we chose to use OV5116P Single IC CMOS monochrome camera with PAL analog output which is a complete black and white CMOS Video Camera chip. Its array size is  $352 \times 288$  and its effective photosensitive area is of  $312 \times 215$  pixels. PAL-D has television standards with 625 lines per frame, 25 frames per second and 50 fields per second.

### B. Wheel speed sensor (Rotary Encoder)

A rotary encoder is an electro-mechanical device, attached to a rotating object (such as wheel or motor) to measure rotation. We used Nemicon corporation manufactured (18mm diameter  $\times$  18mm long, 500ppr) OME-A incremental rotary encoder. The outputs of the encoder have been used as part of a PID feedback control system to determine rotational velocity. The sensor provides pulse sequence of the wheel speed change to 74HC74 D-type flip-flop, which provides TTL waveform to MCU timer input capture function.

## IV. HARDWARE DESIGN

### A. Main Control Unit (MCU)

MK60N512VMD100 is a K60 series MCU, which is a core chip of Cortex-M4 series. The memory of K60 is expandable from 32 KB flash memory/ 8 KB RAM to 1MB flash memory/ 128 KB RAM with optional 16 KB cache, for optimizing bus bandwidth and flash execution performance.

### B. Voltage Regulator

Throughout the system, the +5V power consumption is small and hence in order to reduce power supply ripple, we considered using a linear voltage regulator. TPS7350 is a micro-power low-dropout linear voltage regulator with good circuit protection against over-current, overvoltage and reverse voltage. Using this chip and some minimal external components, we can create an efficient voltage regulator circuit, as shown in Fig. 2.

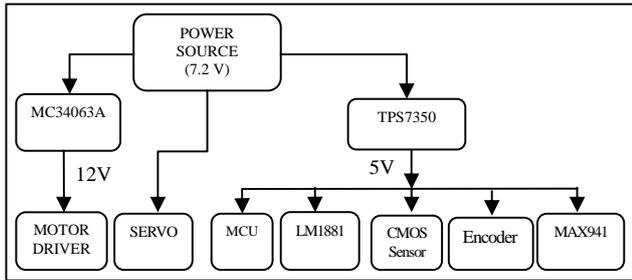


Figure 2. Power system

### C. Video sync separator circuit

To separate the video signal we have used LM1881 Video sync separator. In order to filter noise and matching impedance, the video signal is connected to pin no. 2 of LM1881 through a filter circuit. A low-pass filter must be added to avoid high frequency interference.

### D. Comparator circuit

Since there are only two colors (black and white) in the track, the gray scale image does not need to be of high resolution. By adjusting the threshold value, the gray-scale image is converted into a black-and-white image.

In a binary image there are only two possible values for each pixel (0 or 1). The optical system forms the detected object's image onto the CMOS photosensitive sensor, and the corresponding output voltage value directly represents the grayscale value of the image, there will be a difference in voltage value at the black guide line due to the change between white and black. MAX941 comparator converts the analog video signal into TTL level, which is directly received into the micro-controller's Enhanced Capture Timer Module (ECT) interface. The channel can capture edge setting (rising or falling edge) that appears on the interface, as shown in Fig. 3. Here, BC, DE segments respectively correspond to black line and sync head. By means of the information reflected by AB and CD segments, we can get the real-time location of the car relative to the black line. This information is the key to a high-speed, yet stable car.

We also connect an RC low pass filter circuit to the MAX941 chip circuit to prevent "burr" interference at the borders of the black guide line. The biggest advantage of this method is that general I/O operation is faster than A/D conversion, thus making the improvement of high resolution possible.

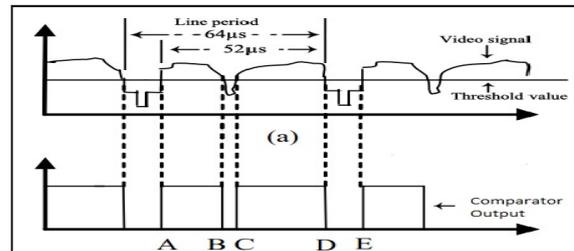


Figure 3. TTL level received onto the ECT module[1]

### E. Driving motor circuit

Motor driving circuit is implemented with the use of discrete-component manufactured DC motor reversible bipolar bridge. The power components are constituted of four N-Channel power MOSFET (IRF540), rated operating current can easily reach more than 100A, greatly improving the torque and speed of the motor. The motor driver consists of the following components: PWM input interface, logical commutation circuit, dead-time control circuit, power circuit, power-MOSFET gate driver, bridge power driver, buffering protection circuit.

## V. SOFTWARE DESIGN

The software design platform for our intelligent car's system is IAR Embedded Workbench IDE, where we use C as implementation programming language. The code development and debugging were all made using this same platform. The system's software design consists of the following modules: (1) MCU initialization, (2) real-time path detection (image processing), (3) servo control, (4) motor control, (5) interrupt of speed acquisition and (6) PID control algorithm. The flow chart of the program is as shown in Fig. 4 (a).

### A. Characteristics of the original image

After converting the original image into a binary image, we get a two-dimensional matrix where each element corresponds to a pixel. The line period is 64 µs; each line of the image can collect up to 220 pixels. In order to increase the width of the frame, we have used a 2.8mm wide-angle lens. The use of the comparator circuit instead of A/D conversion greatly improved the processing speed and due to this, the MCU can now process 95 lines out of 288 visible vertical lines (extract one line in every three lines), its field of view reaches up to 200 cm. Finally, the array size is IMAGE\_DATA [95] [200]. A record of transaction points is made for each line and stored in a two 2D array (rising edge and falling edge respectively). Track identification can be done by traversing the rising and falling edges.

### B. Line Extraction

The aim of the line extraction algorithm is to find the valid center for each line. The line period is 64µs where line blanking interval is 12µs. During this interval, the captured line can be processed, hence there is no need to allocate extra time for image processing. As shown in Fig. 4 (b), the basic principle of the black line detection algorithm is as follows:

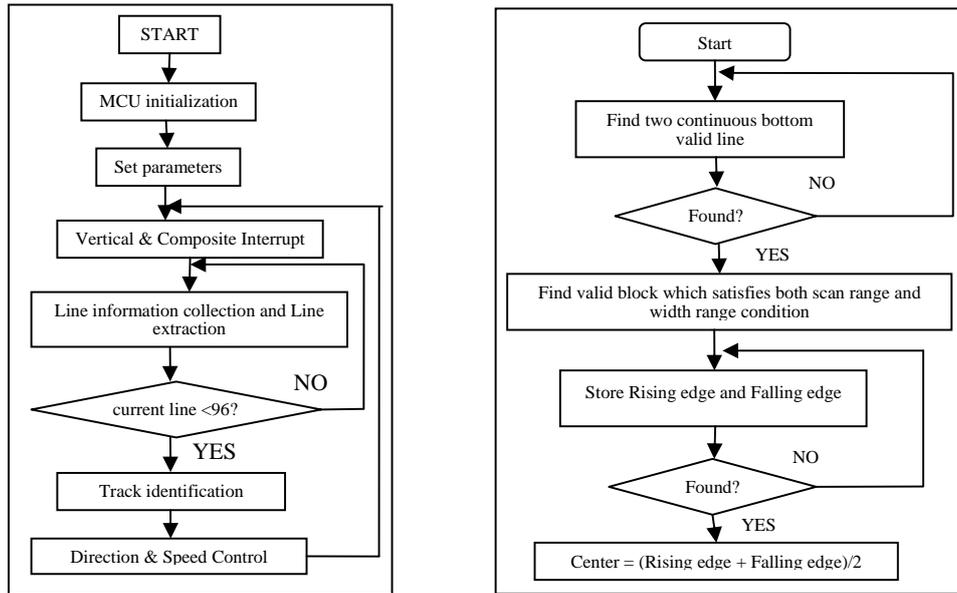


Figure 4. (a) Architecture of the program (b) Line extraction algorithm

1) We used near to far method as distant black lines are not as stable as near black lines. A direct progressive scan of the original image is performed and according to the set threshold value we extract the black-and-white transaction point (falling and rising edge).

2) The black line’s width has a range. A black block with a width value within this range is considered to be valid. It is then possible to filter the interference of black blocks with widths that are out of this range.

3) Using the continuity feature of the black line, we determine the scanning range for the current line based on the center of the previous valid line. We can hence save time as the points of a row do not need to be all scanned. By doing this we can infer the approximate position of the valid center.

VI. CONTROL

The average value of all the valid centers for a frame is used to control the direction and the line number of the farthest valid center is used to control the car’s speed. To make the vehicle run stably in different situations and to remove disturbing factors during the driving process, PID closed-loop control algorithm is used.

A. Servo control

The average position can be calculated based on a weighted average formula, as shown in (1).

$$X = \frac{\sum_{row=0}^{ROW\_MAX} (X[row]*row)}{\sum_{row=0}^{ROW\_MAX} row} \tag{1}$$

where X is the calculated weighted average center, X[row] is the calculated center for that specific row, and ROW\_MAX (0-94) is the number of valid rows. X and the corresponding reference angle values of servo PID constitute a linear relationship. Since the servo motor reaction itself is very slow and the integration parameter (I) of PID will weaken the rotating performance even more, we use a PD controller. In the control process, when the black line is not along the middle of the car, it generates a deviation (offset), which can be calculated as shown in (2). The proportional parameter (K<sub>p</sub>) takes an immediate regulatory role in order to reduce the deviation such that the mobile robot is made to follow the black line.

$$OFFSET = (Target\ position - X) \tag{2}$$

where the value of Target position is 100.

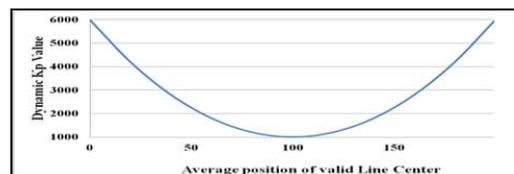


Figure 5. Center Position and Dynamic Kp two-function curve

The derivation regulator (D) reflects the rate of change of the system. Instead of taking different PD parameters in different shapes (such as straight, “S” shaped and curve line), we used a dynamic tuning process where Kp changes according to offset as shown in Fig. 5. A constant value for differential coefficient (Kd) is set, which reduces vibration and gives a dynamic response capability to servo control system.

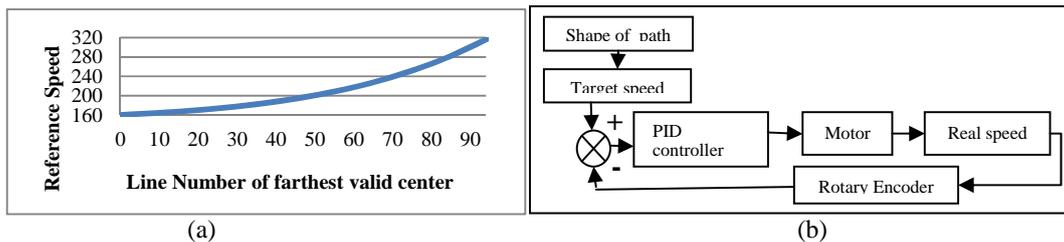


Figure 6. (a) Two-function curve relationship between Black line center position and speed (b) Block diagram of motor PID control

**B. Motor control**

For stable and fast motor control, the line number of the farthest valid center is used. Since the CMOS sensor’s field of view is 2m long, the car can start to decelerate 2m away from the curve line. It will similarly accelerate very fast in straight and “S” shaped line. After repeated testing, we find that the line number of the farthest valid center and PID reference speed values constitute a quadratic relationship as shown in Fig. 6(a). In this motor control system, there are two sensitive situations: (1) when the car is entering a curved line from straight line, we need speed deceleration, and (2) when the car is entering straight line from curved line, we need speed acceleration. In order to improve the overall average speed, the system error has to be adjusted very fast and smoothly. This paper uses incremental PID control algorithm, as shown in Fig. 6 (b).

**VII. EXPERIMENTAL RESULTS**

An experiment was setup to test the line-following and speed performance of the mobile robot. The CMOS sensor and hardware design retrieved a clear image of the track and an accurate binary image. Our line extraction algorithm was also successful in detecting an accurate scanning range for the valid center of each line, as shown in Fig. 7(a). The calculated valid center for each line was also found to be correct as shown in Fig. 7(b). After implementing weighted average position and PD control algorithm, our approach optimizes the trajectory of the car. Instead of following the black guide line, the mobile robot has been made to follow the line with the shortest path. For example, while turning it will take the shortest path as in Fig. 8 (b) and in the “S” shaped track, it will go almost straight as shown in Fig. 8 (a). The PID algorithm used in the motor control was also found to be very efficient. The car could accelerate and decelerate at a highly dynamic rate, enabling it to finish the track in the shortest possible time.

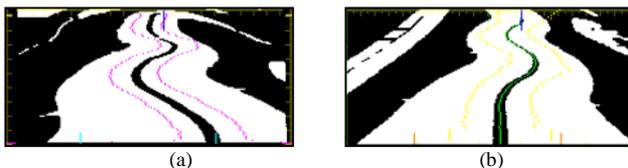


Figure 7. (a) Purple lines indicate scanning range, (b) Green line shows the center of the track after line extraction

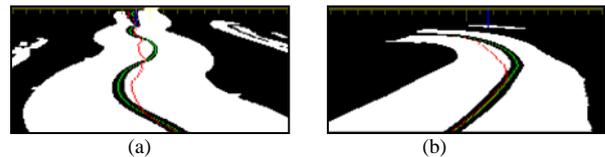


Figure 8. (a), (b) red line represents trajectory (shortest path) taken by the car

**VIII. Conclusion**

This paper presents a hardware and software architecture for an autonomous line following robot. The hardware has a high processing speed because the microcontroller’s CPU frequency is up to 100MHz and the use of the analog comparator circuit to convert the binary image reduces CPU processing burden. We also defined an algorithm that would enable the intelligent car to have a high response rate, i.e. accelerate fast in straight paths and decelerate fast during turns. Our test results indicate that the car is able to run smoothly on any kind of black guide line with an average speed of  $3.4ms^{-1}$  and maximum speed of  $3.7ms^{-1}$ , with fast speed control response and high stability. The proposed architecture was also used by USTB Smart Car Lab for the 4<sup>th</sup>, 5<sup>th</sup>, 6<sup>th</sup> and 7<sup>th</sup> Freescale Cup Competition China, of which it was the final winner.

**ACKNOWLEDGEMENTS**

This research has been supported by China National High-tech R&D Program (863 Program), 2011AA060404 and the USTB Smart Car Lab of University of Science and Technology, Beijing (USTB).

**REFERENCES**

- [1] Liu Jin, Qi Xiao hui, & Li Yong ke, Simple Design and Implementation of a Path Information Acquisition System Based on Camera, *Electronic Engineering & Product World*, 2008, 151-152.
- [2] Yan Jian Hong, Wu Kai Ya, *Graph Theory and Its Algorithm*, University of Science and Technology of China press, 2003.
- [3] Park K.H , Bien Z, Hwang D.H. A study on the robustness of a PID - type iterative learning controller against initial state error, *International Journal of systems science*, 1999, 30(1), 102-135.
- [4] Yin Yi Xin, Tao Yong Hua, *PID Control and Its Application*, China Machine Press, 1998.
- [5] Zhuo Qing, Huang K. S., Shao Bei Bei, *Learn to Make Smart Car*, Beihang University Press, 2007.