

Digital Image Processing Explorer

Rosen P. Spirov

Department of Electronics
Technical University of Varna
Varna, Bulgaria
rosexel@abv.bg

Neli S. Grancharova

Department of Electronics
Technical University of Varna
Varna, Bulgaria
nelly2000@abv.bg

Abstract—The present paper contains information on development FPGA explorer system for digital image processing in real time. The paper presents a realization of Kalman filter for navigation. In the mobile tracking, system dynamic objects developed a simplified architecture, generating compensation change the dynamics of the observed object in response, expressed in the development of control explorer platform

Index Terms—Digital Image Processing, FPGA, Kalman Filter, Robots, Servo Control, TV.

I. INTRODUCTION

The object detection and tracking has been an active research area for a long time because it is the initial important step in many different robotics applications, such as video surveillance, face recognition, image enhancement, video coding, and energy conservation. Thus, we turn every possible integer multiplication into integer shifts and additions digital image processing. Any PLD can be used, but the PLD the system was designed with is the Cyclone II EP2C35F672C6N on the ALTERA DE2 board that utilizes a 50 MHz clock [1]. Each current video frame was captured by the camera and sent to the FPGA's decoder chip via a composite video cable. The camera's analog image information is converted into an ITU656 in standard digital stream on the DE2 development board.

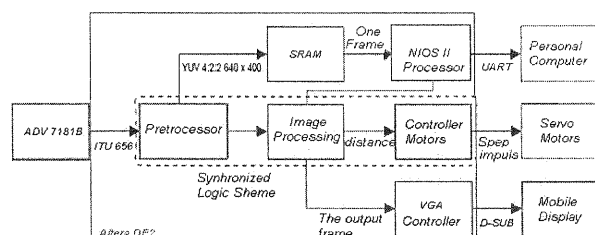


Fig 1 Schematic of FPGA image processing and control system

This stream is used in three ways:

- ☐ It controls the auto-tracking camera's left, right, up, and down operation.
- ☐ It provides a vision sharing system and JPEG of the wireless transfers.
- ☐ It provides the in-vehicle display.

This image capture module pre-processes the image by modifying the image size, removing interlacing mode, and calculating the frame buffer memory address. To acquire the distance between the target and the launcher, visual information is needed for the computation of the positions [2]. After the video signal was processed in different modules in Verilog, the final output passed through the VGA driver to be displayed on the VGA monitor. The GPIO outputs of the board controlled by impulses PWM drivers, switching power supply circuits of stepper motors.

II. THE IMAGE PROCESSING

A. The FPGA object detection processing

The center of the object is then marked up with a crosshair, and the UART is used to move the robot either left or right, with the speed of the robot increasing with increasing distance from the object center to the center of the frame. Finally, the marked-up image is copied using the DMA controller to the SRAM buffer in the VGA controller. For each input frame, first divided the whole frame into small blocks - 16x16 pixels in final implementation. For the first frame, simply use the center of the screen as the object center. To decide whether a pixel has the same color as the object, we separately compare each pixel's R, G, and B with the target's value within a predefined threshold. Afterwards, if this block belongs to the target object based on if the majority of its pixels are of the same color as the object. This is done for every block and so to know which blocks belong to the object. After that, it averages the positions of the highest, lowest, leftmost and rightmost blocks to calculate the center of the object.

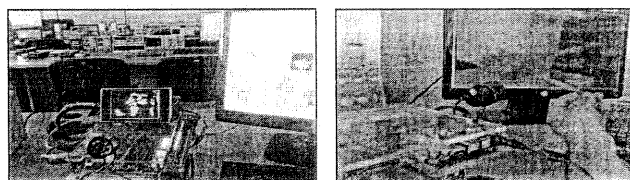


Fig 2 The object detection

The center value is saved and used for finding blocks of the object in next frame. By processing the image block by

block, we reduce the number of membership decisions from one pixel a time to one block a time. Applying morphological filtering including erosion and whole filling would, firstly reduce the background noise and, secondly, fill in missing pixels of the detected faces regions. After each group of detected pixels became one connected region, connected component labeling algorithm was applied.

The next step is looking for the connected regions whose areas were of 26% of the largest area and store their corresponding indices in face. In follow step the image that contained the connected regions found is output. The final stage was to determine object location.



Fig.3. The object detection, when there was a moved man

B. The control processing units of positioning system

The originality of the idea is to perform a simple movement, based on the coordinates of the current and preceding enclosing rectangles, as determine how to move the system. Therefore, that at the end only is used x- center of the current rectangle. This value is determined by finding the differences between left and right x coordinates of enclosing rectangle, tearing them by moving down one bit, subsequently adds this value back to the left x - coordinate. In the mobile tracking system dynamic objects developed a simplified architecture, generating compensation change the dynamics of the observed object in response, expressed in the development of control in the period-modulated pulses to the driver of the driving engines on carrier's platform [3].

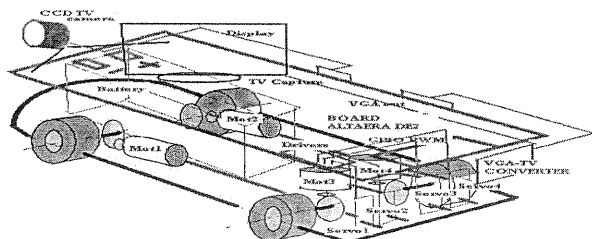


Fig.4. The project of FPGA Explorer

Management scheme based on control of the left and right input and output ports, as is shown on fig.6.

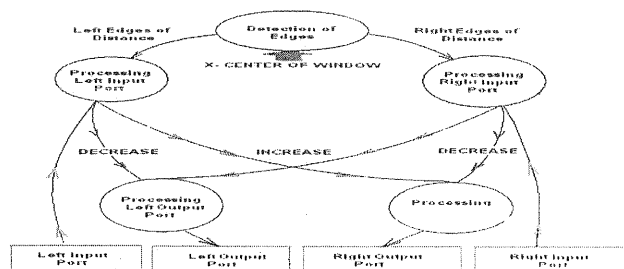


Fig.5. The control

The left input node is connected to the left output in limiting function by supplying the pulse. In the same way they are also attached from the right input to the right output by logic for generating pulses of variable period of the PLL of the FPGA to generate shorts pulses. Right input node is connected with the counter left starting with increase function by wider period impulses. Respectively for the link of Left input to Right output unit's connection of the control logic. The input logic unites accounted left and right distance between the input X center and the center O of the screen. Assigned to control pulses at a frequency corresponding to the duration of the signal input and proportional width to control the servomotor of output port [4].

By increasing or decreasing, the width of the control pulses are controlled output unites. Motor control provides control as the module is plugged with its inputs to the outputs of the above-described logic. Observe the condition of two pairs of reversible cyclical counters. Counting the first set performed with scan outputs of control units and set direction, separately carried out 100 clock cycles of the control logic. With each cycle other set counter increases its content, and process of increasing by equivalence. The second counter counts 64 cycles by controlling the outputs of the registers and ends with a check. In this finite state machine are set 7 different levels to coordinate the various movements, turning to the left, move left, a little to the left, move right, lower right move, move right, turning to the right. with a control register which is the output of the module. The table 1 illustrates the response of the control.

TABLE 1

Control	The reaction
0	FULL LEFT, Act in motion both LEFT counter
1	Small LEFT, Act in motion both LEFT counter
2	Smallish LEFT, Act in slow less-motion
3	Forward, Act in forward motion by left + right counters
4	Smallish RIGHT, Act in slow less-motion
5	Small RIGHT, Act in slow-motion
6	FULL RIGTH, Act in motion both RIGHT

Controlling this logic is implemented as a module in FPGA Cyclone II board Altera De2. The realized PWM is a digital reversible counter that restarts a specified time interval. The pace of work is variable. For each clock pulse by which the counter decreases its content, the output of the PWM is a log. For each clock pulse, a counter, which increases its content by one-step, the output of the PWM, is the log. 1. By changing the frequency of the next step and change the pulse width of the PWM control the speed and direction of rotation of the motor according to the standards for servo control [5].

Power servomotors are controlled through driver of programmable GPIO outputs on board Altera DE2 connected to PWM outputs from FPGA. As power, engines are used *Parallax Servo & Towerdservo*.

From the output jack GPIO_0 on board Altera DE2 control every each engine separately. Mobile small display platform, visualize survey camera on an FPGA board. The images are

presented in black & white range and the target is a formal contour object /FCO/ in color. By calculating the center of FCO and shift it to the center of the visual field to determine the velocity of the object. With corrective impulses to control servomotors positioned FPGA platform by matching the centers of the visual field and FCO. Therefore, the system accurately determines and monitors the movements of the object in real time. The objects are examined at different, increasing, resolutions until it is possible to separate the multiple regions in the objects belonging to the occlusion. Furthermore, to detect the identities of the objects in the current frame beginning from the identities of the objects in the previous frame, a graph matching algorithm is performed.

The basic idea is to provide the new incoming information into the current background image [6]. Therefore transformation from pixel coordinates to real world coordinates is required, as is shown in fig.6.

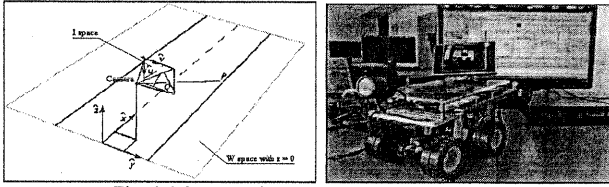


Fig.6. The Mapping Processing and FPGA system

That is an Inverse Each intermediate node represents a sub region of the whole region, and its attributes are the position and size of the sub region bounding box and average colour of that sub region.

III. THE MOVEMENT CONTROL ON ADAPTIVE KALMAN FILTER

For navigation, the experimental data are processed with recurrent Kalman filter. The determination of the matrix $K(k+1)$ requires the use of a matrix $n \times n$, where n is the projects of vectors of measurements $z(k)$. The important feature of the filter is represented in that the matrix $K(k+1)$ is not depend on the vectors of measurements $\{z(1), \dots, z(k+1)\}$ and the interference $\{w(1), \dots, w(k+1)\}$.

Therefore, it can be calculated a priori and its elements to be stored until the start of the processing of information. Then, in the process of treatment with the vector of measurements $z(k+1)$ calculated the matrix $K(k+1)$ making a new assessment $\hat{x}(k+1)$. This enables to perform processing of another measurement in real time. To see the model of the movement of a dynamic object set with expressions process $\{w(k), k=0,1,2,\dots\}$ is a Gaussian white series with zero mathematical expectation and constant dispersion Q ; and the process $\{\eta(K+1), K=0,1,\dots\}$ s white Gaussian series and averaging zero dispersion R . Summary Kalman filter equations will acquire type:

$$\hat{x}(k+1) = E\hat{x}(k) + K(k+1)[z(k+1) - E\hat{x}(k)] \quad (1)$$

$$K(k+1) = \frac{EV_e(k) + Q}{EV_e(k) + Q + R} \quad (2)$$

$$V_e(k+1|k) = EV_e(k) + Q \quad (3)$$

$$V_e(k+1) = \frac{R(EV_e(k) + Q)}{EV_e(k) + Q + R} \quad (4)$$

When initial conditions $\hat{x}(0) = m_x(0)$. They are assigned specific values of parameters such as: $E=1$, $V_e(0)=100$, $Q=25$, $R=15$. The previous equations will change as follows:

$$V_e(k+1|k) = V_e(k) + 25 \quad (5)$$

$$K(k+1) = \frac{V_e(k) + 25}{V_e(k) + 40} \quad (6)$$

$$V_e(k+1) = \frac{15 \cdot (V_e(k) + 25)}{V_e(k) + 40} = 15 \cdot K(k+1), \quad \kappa = 0, 1, \dots \quad (7)$$

In table 2 is shown the values of scalar variables for the first seven steps.

TABLE II.

K	Scalar variables		
	$V_e(k+1 k)$	$K(k+1)$	$V_e(k+1)$
0	-	-	100
1	125	0.893	13.393
2	38.393	0.719	10.786
3	35.786	0.705	10.570
4	35.57	0.703	10.549
5	35.551	0.703	10.549
6	35.549	0.703	10.549
7	35.549	0.703	10.549

The last rows of the table show the existence of limits for these values. Limit of $\bar{V}_e$ is obtained by assigning a value of $V_e(k+1) = V_e(k) = \bar{V}_e$ in the last expression. Then:

$$V_e(k)(V_e(k) + 40) = 15V_e(k) + 375; \quad (8)$$

$$\bar{V}_e^2 + 25\bar{V}_e - 375 = 0 \quad (9)$$

Graphical data results appear respectively as follows in fig.7:

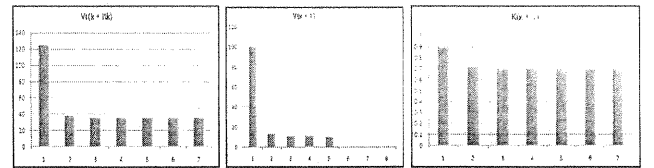


Fig.7. The diagrams of experimental values

$\bar{V}_e \geq 0$ is assumed that there is only one root of this quadratic equation, and that is $\bar{V}_e = 10.55$ then the estimated rate of transmission of the Kalman filter can be determined as follows:

$$\bar{K} = \frac{\bar{V}_e + 25}{\bar{V}_e + 40} = 0,703 \quad (10)$$

Kalman filter is established in the state $k \geq 5$ and described by the equation:

$$\begin{aligned} \hat{x}(k+1) &= \hat{x}(k) + 0.703(z(k+1) - \hat{x}(k)) \\ &= 0.297 \hat{x}(k) + 0.703 z(k+1) \end{aligned}$$

This equation illustrates very well the contribution to the "new" assessment $\hat{x}(k+1)$ in the "old" assessment $\hat{x}(k)$ and the new measurement $z(k+1)$. The Error and deviation calculated from the reference values can be determined as follows:

$$\Delta x = \frac{1}{K} \sum_{i=1}^K |x_{\text{exper}} - x_{\text{calcul}}| \quad (11)$$

Where K - is the number of measurements. Similarly determine Δy .

The next figures show all results by calculated, experimental simulated and Kalman data of determinations in axis OX and respectively OY.

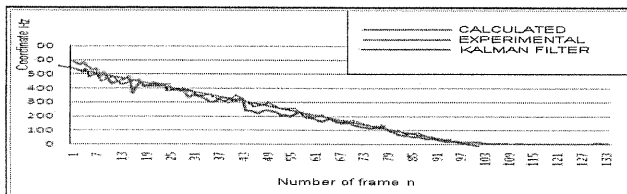


Fig. 8. The results of determination on axis OX

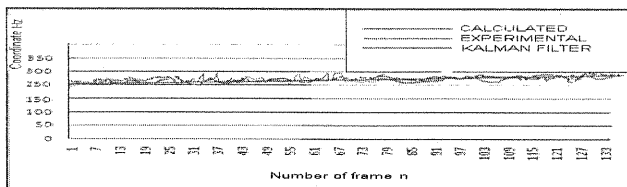


Fig. 9. The results of determination on axis OY

The realized idea is to provide the new incoming information into the current background image. To acquire the distance between the target and that launcher, visual information is needed for the computation of the positions. The RTL project by FPGA is build in Altera Quartus II Tools, as shown in fig.10.

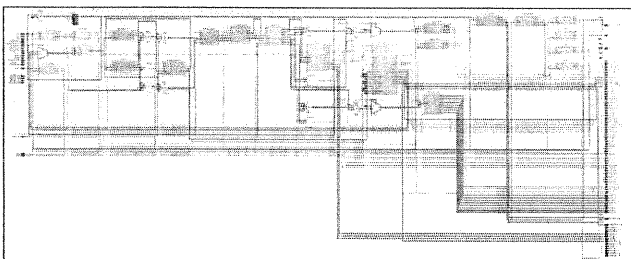


Fig. 10. The RTL project on FPGA

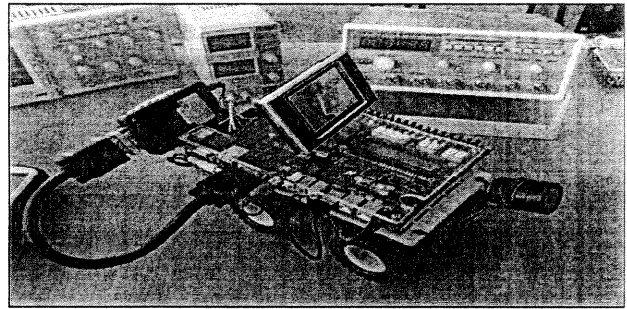


Fig. 11. The FPGA Digital Image Processing Explorer

IV. CONCLUSION

The Image Processing Toolbox provided in MATLAB allowed the process of developing and testing the algorithm to be more efficient. Object detection and tracking has been an active research area for a long time because it is the initial important step in many different applications, such as video surveillance, face recognition, image enhancement, video coding, and energy conservation.

In the mobile tracking system, the dynamic objects developed a simplified architecture, generating compensation change the dynamics of the observed object in response, expressed in the development of control in the period-modulated pulses to the driver of the driving engines.

The presented realizations of Kalman filter use for navigation. In the mobile tracking, system dynamic objects developed a simplified architecture, generating compensation change the dynamics of the observed object in response, expressed in the development of control on explorer platform

REFERENCES

- [1] Altera "DE2 Development and Education Board" DE2 Package, or available online at: <http://www.altera.com>
- [2] Alexander S.T. Adaptive Signal Processing Theory and Applications - N.Y.: Springer - Verlag, 2006.
- [3] Advanced Microcontroller Final Projects, or online at: <http://people.ece.cornell.edu/land/courses/ece5760/FinalProjects>
- [4] Kaufmann M, Machine Vision: Theory, Algorithms, Practicalities. 2004.
- [5] Limb J.O, J. A. Murphy, Measuring the speed of moving objects from television signals // IEEE Trans.-1975.-Vol. COM 23.-№4.-pp.411-414
- [6] Spirov R., Practical Object Tracking System on FPGA. 5-th Intern. Conference on Communications, Electromagnetic and Medical Application - CEMA'2012. Athens, Greece. ISSN 1314-2100