

# FPGA를 이용한 고속 영상처리보드의 개발

류형규\*, 박홍민\*\*, 이승환\*, 김태훈\*, 신천우\*, 김병수\*\*\*

\* 동명정보대학교 정보통신공학과

\*\* 인제대학교 의용공학과

\*\*\* (주) 현대자동차 상용제품개발연구소

## Development of the Real-time Imaging Processing Board Using FPGA

Ryu Hyung-gue\*, Park Hong-min\*\*, Lee sung-hwan\*, Kim Tae-hun\*,

Shin Chun-woo\*, Kim Byung-soo\*\*\*

\* Department of Information and Communication Engineering, Tong-Myung Univ.

\*\* Department of Biomedical Engineering, In-Je Univ.

\*\*\* Commercial Vehicle Test & Research Team, HMC.

### ABSTRACT

In this study, the basic image-board and algorithm has been developed to extract a road lane by modeling the driving process. The high speed processing enables an image capture, processing and prompt decision making. In order to high speed processing ASIC like FPGA was designed and integrated in one board system. The algorithm enabling road driving must recognize a straight and bend edge separately. The high speed image processing board using FPGA can be used in real-time decision making system for road driving and in the machine vision under bad working environments like a coal mine. And it also can be used in the safety control system in subway and in image input system of CCTV and CATV by designing the board to meet various user's needs.

### 1. 서론

인간의 기능을 보조할 수 있는 시스템의 개발에 대한 필요성에 따라 지각능력을 보조하고 대신하기 위해서, 고속영상처리보드를 이용한 도로주행시의 차선이탈 정도를 알아보려고 한다. 여기에서 중점을 둔 것은 시스템의 디버깅을 사용자가 수정을 할 수 있도록 하고, 고속 및 정확도를 유지하는데 그 초점을 맞추고 있다.

기존의 영상보드의 경우, 하드웨어가 시스템에 내장되어 있어 화상처리 알고리즘이라든지 user용의 logic 설계는 단지 소프트웨어를 ISA나 PCI를 통해 구현하다보니 저속일뿐만 아니라, 사용자가 하드웨어를 디버깅하기가 불가능하고 속도가 느린 단점을 가지고 있다. 또한 사용된 하드웨어 제어방식은 signal delay라든지 알고리즘의 변경에 많은 제약이 따른다.

본 연구에서는 이러한 기존방식의 단점을 극복하기 위해 FPGA를 이용하여 사용자 위주로 하드웨어를 수정, 변경, 보완이 가능하도록 하고, 주 프로세스와 분리된 독립적 모듈로 동작하는 고속영상처리보드의 개발을 목적으로 한다.

## 2. 연구방법

고속영상처리 보드의 개발을 위해서는 먼저 CCD camera와 같은 입력신호를 통해 화상입력부, 신호처리부, FPGA 전처리부, 실제 시스템에서 요구되는 알고리즘 제어부, 화상 Display등으로 구성된다. CCD camera의 입력신호 경우, 영상입력부에는 1차원적인 신호에서 수직, 수평동기를 분리하는 동기분리부를 포함시켰고, 신호처리부에는 A/D 입력단에 고해상도의 신호를 전달하기 위한 증폭, 필터, 위상처리회로등을 포함할 수 있도록 한다.

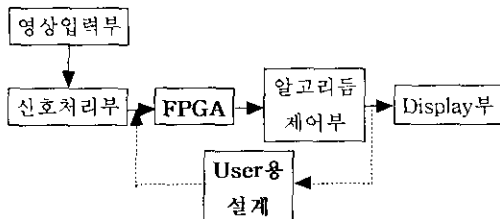


Figure 1. 고속영상처리보드의 블록다이아그램

알고리즘의 개발 및 설계는 사용자가 필요로 하는 신호를 규명하여 고속처리를 위한 적절한 edge처리, noise제거, 필터등과 함께 설계하고 기본적인 모듈을 설치한다. 여기에서 기본적인 화상처리 알고리즘과 DSP는 FPGA(Field Programmable Gate Array)를 이용하여 logic으로 구현한다[1]. 실제 ASIC 설계의 경우, 설계 및 검증기간이 짧고 설계 변경이 쉽고 빠르며 적은 비용으로 실현할 수 있다는 장점 때문에 고속영상처리보드의 설계에서 도입하였다. 또한 영상신호처리를 위해 필요한 필드의 판별, 메모리의 timing제어 등에 대한 연산처리시간은 다음과 같다.

- ㉞ 카메라로부터 들어온 영상을 메모리에 입력
- ㉟ 1차영상저장 : A/D로부터의 데이터를 메모리에 저장  
60Hz(one field, 17ms)
- ㊱ 2차영상저장 : 1차 메모리의 영상을 차선추출 알고리즘을 거쳐 2차측으로 이동(㊸)한뒤, 2차 메모리에 저장(㊹)한다.

㊸ edge추출을 통한 차선인식

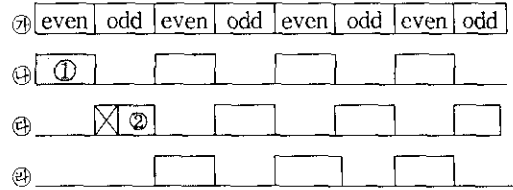


Figure 2. 고속영상처리보드의 연산처리 타이밍다이아그램

최종적으로 얻은 데이터 값을 고속의 마이크로프로세서 또는 일반 serial port를 이용하여 필터와 직선·곡선추출 알고리즘을 거쳐 차선을 추출한 뒤 시스템의 용도에 맞게 제어하고 화면재구성을 설계한다.

### 2-1. Input Image/Signal Processing

현행 NTSC방식(interlace)에 의한 카메라 영상신호에서 고속의 logic 설계를 위해서는 필드의 판별, 수직·수평의 동기분리회로, Video용 op-amp나 A/D convertor로 연결되는 신호처리회로등이 필요하다. 신호처리회로에는 영상신호에 맞는 임피던스를 매칭시키고, 고속·저전력을 위한 회로, A/D단을 full scale로 맞춰 해상도를 높이기 위한 회로, 고속의 신호흐름으로 생길 수 있는 신호유출 및 잡음을 제거하기 위한 필터등을 설계하였다.[2]

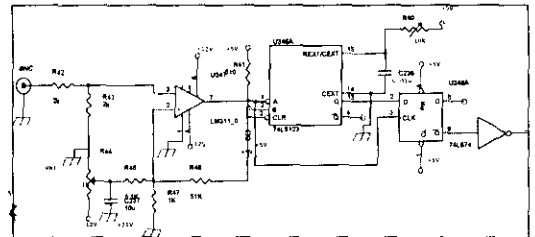


Figure 3. 동기분리회로

2-2. FPGA pre-Processing

FPGA에서는 입력단을 통해 A/D를 거쳐 들어온 디지털 영상신호에서 edge 추출을 하기 위해 mean filter, 미분필터, memory access, external interface connection, 영상신호의 판독과 디스플레이에 대한 중앙제어 등의 기능을 내장하도록 설계한다.[3]

즉, 영상에서 생길수 있는 불필요한 자료 즉 잡음을 없애기 위해서 median filter를 이용하여 제거하고, 차선의 edge를 인식하기 위해 영상에서 edge point를 찾아야 하는데, 이를 위해 1차미분 Sobel Mask를 사용한다.[4]

이렇게 획득된 data에서 1차적으로 유효한 edge를 추출하고, RISC와 같은 고속의 프로세스에서 최종 차선을 추출하도록 설계하는데, 워낙 고속의 data flow가 발생하므로 메모리의 access time 및 data writing time의 marching이 중요하다.

본 연구에서는 user용으로 I/O pin이 많은 FPGA chip을 이용하여 사용하고, 차후에는 메모리의 확장을 통해 복잡한 logic도 설계·수정이 가능하도록 하였다.

2-3. User interface

영상입력회로와 1차 edge추출을 위해 FPGA를 통해 얻은 소량의 데이터를 고속의 RISC에서는 사용자용 프로그램이나 알고리즘을 이용해 더욱 정확하고 안정된 차선을 얻을 수 있도록 coding이나 설계가 가능하도록 하였다. 추출된 차선이나 알고리즘의 결과물 외부로 display 하는데에는 Instruction에 따라 lframe 입·출력, 연속 frame 입·출력등을 결정할 수 있다.[7] 그리고, 메모리의 Read/Write는 chip select후 어드레스의 유효시간내에 데이터의 입·출력이 끝날 수 있는 충분한 시간을 확보할 수 있도록 RISC와의 경로를 확보하여 설계한다.

전체 256×256의 원영상에서 상하좌우 차선이 위치 가능한 유효한 영역에서 디스플레이 모듈조정이 가능하도록 하고, TRNR/L(turn signal right/left), STP(stop lamp), SPD(speed sensor), ALRM(alarm switch on/off), BB(브레이크), CC(클러치)등 차량과의 interface를 통해 실제 제어를 담당한다.

3. Least Square Algorithm

도로주행시의 차선추출을 위한 알고리즘에서 중요한 것은 직선·곡선도로에서 유효한 edge point들만을 추출하는 것이다. 그리고, 선택된 edge point를 가지고, 직선 또는 곡선 형태로 차선을 그을수 있어야 하는데, 이를 위해서 Least Square 방식을 이용해서 처리한다.

Least square 방식은 영상에서 각 점들의 기울기의 평균을 구해서 점들의 공통점을 구하기 때문에 직선이나 곡선의 라인을 그을 수가 있다. 이렇게 되면, 차량의 현재 위치, 차선의 이탈여부를 판단할 수 있게 되는 것이다. 또한 차선 주위의 유효한 edge point와 불필요한 edge point의 판단은 slope equation을 이용하여 다음과 같이 선택한다.[5]

- ① 첫번째 edge point와 다음 edge point와의 기울기를 각각 반복해서 계산한다.
- ② 각 점들의 기울기를 구한 후 ±0.4의 오차범위내에서 이 값들의 평균을 구한다.
- ③ 각각의 edge점들중 기울기의 범위안에 들어가는 것이 기준값 이상이면, 일단 edge point로 인식한다.

이렇게 적합한 edge point를 추출한 뒤, Least square 방식을 이용하여 직선이나 곡선의 라인을 구한다.[6]

$$g(x) = a_0 + a_1x + a_2x^2 + \dots + a_Nx^N \quad (1)$$

여기에서 각각의 자료들로부터 얻은 곡선의 편차는,

$$r_i = y_i - g(x_i), \quad i=1, 2, \dots, L \quad (2)$$

이고, L은 데이터의 개수이다. 편차의 제곱의 합은 다음과 같다.

$$R = \sum_{i=0}^L (r_i)^2 \quad (3)$$

R을 최소화하기 위해서 다항식의 계수에 대한 R의 편도함수를 0으로 놓으면,

$$\frac{\partial R}{\partial a_k} = 0, \quad k=0, 1, 2, \dots, N \quad (4)$$

와 같이 된다.

실제로 계수에 대한 방정식을 행렬로 표현하면,

$$\begin{bmatrix} 6.00 & 3.30 & 2.21 \\ 3.30 & 2.21 & 1.60 \\ 2.21 & 1.60 & 1.22 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 7.5400 \\ 4.8440 \\ 3.5102 \end{bmatrix}$$

이다. 이를 통해 역급수의 계수들은 다음과 같다:

Power n	Coefficient $a_n$
0	0.587114
1	0.059102
2	1.729537

따라서, 일반적인 직선과 곡선을 추출하기 위해 사용된 least square equation은 다음과 같다.

$$y = 0.587114 + 0.059102x + 1.729537x^2$$

#### 4. 연구결과

본 연구에서는 FPGA를 이용한 logic의 설계 및 interface를 구현함으로써 전체처리시간을 단축시켰고 더욱 정확하고 안정된 고속의 영상처리 결과를 얻었다. 즉, FPGA에서의 sobel mask 및 mean filter등을 통한 pre-processing과 RISC의 least square method를 통한 각 단계별 화상처리 및 차선추출이 30ms이내에 완료되었다(Figure4,5,6,7).



Figure 4.  
Original Image(256x256)

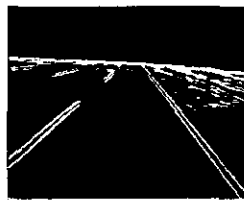


Figure 5.  
Sobel Mask Image

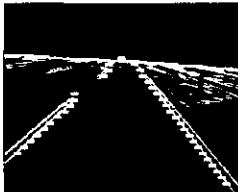


Figure 6.  
Least Square Image



Figure 7.  
Line Reconstruction

#### 5. 결론

FPGA를 이용해 하드웨어를 설계 및 디버깅하고 알고리즘을 pre/post processing의 모든 과정에 이식 가능하도록 고속영상처리보드를 설계하였다. 따라서 기존에 문제시 되었던 속도, 알고리즘의 수정, 하드웨어의 제어등에 관한 처리가 가능하다. 또한 도로주행시 장애물 및 비포장도로 인식, 기후나 환경의 변화에 따른 알고리즘의 보완등도 결국 사용자에 따라 설계나 제어가 가능한 것이다.

결과적으로 환경의 복잡성으로 인해 더욱 정확한 장애물인식이나 경로의 감지를 위한 센서의 개발과 데이터 분석이나 신경망의 이론등과의 접목을 통하여 앞으로 ALV (Autonomous Land Vehicle)에 대한 연구가 진행된다면 IVHS (Intelligent Vehicle Highway System)으로의 발전도 기대할 수 있을 것이다.

본 연구를 통해 개발된 고속영상처리보드는 여러용도로 이식(설계, 변경, 적용)이 가능하므로 실시간처리 시스템에 부합될 것으로 예상된다.

#### REFERENCE

- [1] Oppenheim/Schafer, Discrete-time signal processing, 大英社, 1996.
- [2] 畔津明仁, 基礎からの映像信號處理 CQ출판사, 1992.
- [3] Allen M. Dewey, Analysis and Design of digital systems with VHDL, PWS, 1997.
- [4] RANDY CRANE, Simplified approach to Image processing, 홍릉과학出版社, 1997.
- [5] 홀리오 산체스, IBM PC를 위한 그래픽디자인과 애니메이션 기법, 東逸出版社, 1992.
- [6] SHOICHIRO NAKAMURA, APPLIED NUMERICAL METHODS IN C, 大英社, 1996.
- [7] HITACHI, SuperH RISC engine SH7032, SH7034 CD-ROM, 1998.