

DSP 기반의 실시간 이동물체 검출 및 추적

Real-time Detection and Tracking of Moving Objects Based on DSP

이욱재*, 김양수*, 이상락*, 최한고*

Uk-jae Lee*, Yang-su Kim*, Sang-rak Lee*, Han-go Choi*

요약

본 연구는 무인 영상감시를 위한 이동물체의 실시간 탐지 및 추적을 다루고 있다. 고정 카메라로부터 획득한 영상을 사용하여 영상 내의 모든 이동물체를 검출하고 이동물체를 에워싸는 사각형 박스로 물체를 표시하여 추적한다. 추적방법은 TI DSK6455 키트와 키트 상에 연결된 FPGA 보드로 구성되는 임베디드 시스템에 구현하였다. DSP 키트에서는 이동물체의 검출과 추적을 위한 영상처리 알고리즘을 처리하며, 영상획득과 표시를 위해 설계된 FPGA 보드에서는 라인별로 영상을 읽고 DMA 데이터 전송 방식으로 DSP 프로세서로 영상 데이터를 보내며 또한 처리된 영상 데이터를 VGA 모니터로 보낸다. 실험결과에 의하면 이동물체의 추적은 만족스럽게 동작하였으며, 추적속도는 320x240 영상 해상도에서 초당 30 프레임의 속도로 수행하였다.

Abstract

This paper describes real-time detection and tracking of moving objects for unmanned visual surveillance. Using images obtained from the fixed camera it detects moving objects within the image and tracks them with displaying rectangle boxes enclosing the objects. Tracking method is implemented on an embedded system which consists of TI DSK6455 kit and the FPGA board connected on the DSP kit. The DSP kit processes image processing algorithms for detection and tracking of moving objects. The FPGA board designed for image acquisition and display reads the image line-by-line and sends the image data to DSP processor, and also sends the processed data to VGA monitor by DMA data transfer. Experimental results show that the tracking of moving objects is working satisfactorily. The tracking speed is 30 frames/sec with 320x240 image resolution.

Keywords : Moving Object, Detection and Tracking, DSP, Real-time, Image Processing

I. 서론

현재 우리는 멀티미디어의 시대를 살고 있는데, 멀티미디어란 문자, 음악, 영상 등 여러 가지 매체를 한 대의 컴퓨터나 단말기에 조합해 정보를 전달하는 기술 또는 전달 시스템을 말한다. 다양한 멀티미디어 매체 중에서도 인간은 환경에 대한 정보를 인식할 때 90% 이상이 시각에 의존하기 때문에 영상이 차지하는 비율은 다른 매체들 보다 상대적으로 높다. 최근에 아파트나 공공건물의 지하주차장, 은행의 무인 자동화 창구, 산업현장이나 백화점 등에서 보안과 범죄예방에 대한 관심이 크게 증대되면서 보안관련 영상처리 제품의 필요성과 수요가 급격히 증가하고 있는 실정이다. 이러한 보안 시스템에서는 무인 감시 카메라를 이용하여 외부 침입자를 식별하고, 이동 상황을 연속으로 감시하여

이동물체를 추적할 수 있는 영상처리 시스템으로서 기존의 개인용 컴퓨터를 기반으로 한 시스템을 탈피하여 실시간 동작을 강화하기 위해 임베디드 시스템 형태로 개발되고 있다.

영상신호로부터 물체의 움직임을 추적하는 여러 방법이 제안되어 왔으나 대부분이 계산량이 많은 수학적 접근법을 사용하여 시뮬레이션을 수행하거나 적용된 알고리즘이 복잡하다[1-3]. 특히, 명암과 색깔 정보를 포함한 많은 양의 영상 데이터를 처리해야하는 알고리즘[4], 영상처리 후반부에 복잡한 영역 라벨링(Region Labeling), 클러스터링을 적용한 알고리즘[5], 그리고 반복연산이 많은 Hidden Markov Model, Kalman Filter 등의 복잡한 확률 및 통계적 접근방법을 적용한 알고리즘[6, 7] 등은 임베디드 시스템으로 구현하기에는 적합하지 않다.

임베디드 시스템 기반의 실시간 이동물체 추적을 위해 본 논문에서는 비교적 계산량이 적으면서 용이하게 이동물체를 검출할 수 있는 움직임 에너지 접근방법(Motion

* 금오공대

투고일 : 2010. 5. 14 수정완료일 : 2010. 8. 19

계재확정일 : 2010. 10. 29

Energy Approach)을 바탕으로 물체를 추적하는 기존 알고리즘[8, 9]을 수정하고, 움직임 에너지를 구하는 방법[10]을 더욱 간략화 하였다. 특히, 기존 연구에서 움직임 영역을 배경영역으로부터 분리하고 다수의 이동물체의 각 경계영역을 추출하기 위한 방법으로 영역 라벨링, 경계선 추출(Boundary Extraction) 등이 제시되었으며[5], 이 경계선 추출방법에는 Connectivity Tracing, Contour Following, Edge Linking 등의 알고리즘을 사용하였다[11]. 그러나 이러한 방법은 알고리즘이 복잡하고 많은 계산 량 때문에 실시간 이동물체 검출 및 추적 알고리즘으로는 적당하지 않다. 이를 해결하기 위해 본 논문에서는 영상을 다중 분할한 후, 분할된 영역에 대해 수평/수직 방향 투영방법을 이용하여 다수 이동물체의 최소 경계영역을 추출하는 알고리즘[10,11,12]을 적용하였다.

이동물체 검출 및 추적기능을 임베디드 시스템으로 구현하기 위해 본 논문에서는 DSP 기반의 시스템을 이용하여 실시간 동작을 확인하였는데, 시스템 구현에서 사용된 DSP 장치로는 TI사에서 개발한 DSK 6455 키트와 직접 설계한 FPGA 보드를 사용하였다. DSP 키트에서는 제안된 영상처리 알고리즘을 코딩하였고, FPGA 보드에서는 카메라로부터의 영상획득과 처리된 영상내용을 모니터로 전송을 수행하는데, 영상 데이터 전송 및 수신은 DMA 방식으로 동작하도록 설계함으로써 DSP 프로세서의 로드를 최소화하였다. 제작된 시스템을 사용한 실험결과에 의하면 카메라에서 획득한 영상으로부터 움직이는 물체, 즉 이동물체를 모두 검출하여 추적하였으며 추적된 결과를 모니터에 표시함으로써 실시간 추적을 확인할 수 있었다. 영상획득, 영상처리, 영상표시 등을 포함한 추적속도는 320x240 영상 해상도에서 초당 30 프레임의 속도로 수행하였다. 이 속도는 CCD 센서의 영상획득이 초당 30 프레임으로 고정되어 있기 때문이며, 실제 영상추적은 더 빨리 수행될 수 있다.

II. 이동 물체 검출 및 추적

움직임 에너지 기반의 다중 이동물체 검출에는 여러 방법이 사용되고 있으나 대표적인 방법으로는 연속적으로 획득하는 영상에서 현재 영상과 이전 영상과의 차영상을 통해서 이동물체를 검출하거나, 현재 영상과 배경 영상과의 차영상을 통해서 검출한다[3]. 현재영상과 이전영상과의 차를 이용하는 경우 이동물체가 일시적으로 정지하는 경우에는 정지하는 이동물체를 검출할 수 없다.

따라서 본 연구에서는 일시적 정지물체의 검출과 이동물체의 실시간 추적을 구현하기 위해 배경영상과의 차영상을 이용한 물체검출 알고리즘을 구현하였다. 이 방법은 초기에 배경영상을 획득하고 난 후 연속적으로 현재의 영상을 획득하여 영상차를 통하여 이동물체를 추적하는 방법으로 특별한 추가정보 없이 이동물체를 검출할 수 있는 장점이 있다. 그림 1은 본 연구에서 사용한 다중 이동물체 검출 및 추적 알고리즘 흐름도를 보여주고 있다.

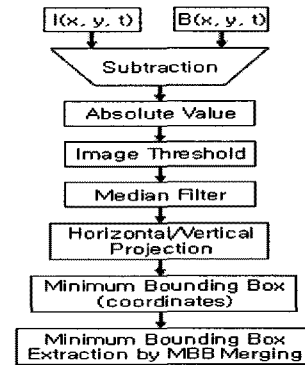


그림 11. 다중 이동물체 검출 알고리즘 흐름도
Fig. 1. Algorithm flow for detection of multiple moving objects

이동물체 검출을 수행하기 위해서는 먼저 영상 내에서 움직임을 감지하고 움직이는 물체를 배경영역으로부터 분리(Segmentation)시켜야 한다. 이를 구현하기 위해서 최초에 획득한 배경영상과 현재 획득한 영상과의 차영상을 구한 후 적절한 임계값으로 이진화하여 이진(Binary) 차영상을 구한다. 이진 차영상은 물체의 유무 및 위치에 대한 정보를 갖고 있으며, 차영상 내에는 이동물체의 외곽성분들이 존재하게 되는데 이 외곽성분을 제거하기 위하여 공간영역 기반 처리방법인 미디언 필터링(Median filtering)을 수행한다. 다음으로 이진 차영상을 M×N 개의 다중 영역으로 분할한 후, 각 다중 영역별로 수평과 수직 방향으로 투영한 결과를 통해 움직임 영역 내에 속한 다중 영역에서 한 개씩의 작은 MBB(Minimum Bounding Box)를 구한다. 이러한 작은 MBB들은 이동물체의 크기에 적합한 합병조건에 따라서 전체 이동물체로 합병하여 물체를 검출한다.

2.1 배경 영상과의 차연산 및 이진 영상화

현재영상과 배경영상과의 차연산을 통하여 얻어진 차영상을 적절한 임계값으로 이진화하여 움직임 에너지를 추출하고 동시에 움직임 영역과 배경 영역을 분리한다. 배경영상과의 차연산에 의해 영상 내 움직임 물체를 추출하는 방법은 이동중이거나 정지한 물체를 모두 찾아낼 수 있는 장점이 있다. 하지만 배경영역의 픽셀값과 이동물체의 픽셀값의 차이가 거의 없으면 배경영상과의 차연산 방법으로는 충분한 움직임 에너지를 얻을 수 없어 이동물체를 검출할 수 없거나 움직임 물체의 검출이 불가능하다. 따라서 움직임 영역을 정확히 추출하기 위한 중요한 요소 중 하나는 환경에 맞는 임계값의 설정이다.

움직임 에너지를 추출하기 위하여 식 1과 같이 시간에 대한 미분을 연속된 영상 내 픽셀들 사이의 차를 구하고 절대값을 계산한다.

$$\frac{dI(x,y,t)}{dt} \cong |I(x,y,t) - B(x,y,t)| \quad (1)$$

위 식에서 $I(x,y,t)$ 은 시간 t 에 입력된 영상 F 의 (x,y) 에 위치한 픽셀의 명암값을 나타내며, 이때 픽셀의 위치를 n

타내는 (x,y) 는 $x,y \in F$ 을 만족한다. 이 차영상을 이진화하여 움직임 에너지를 추출함과 동시에 배경영역으로부터 움직임 영역을 분리할 수 있는 이진 차영상을 얻는다. 이진화를 위한 임계값은 영상의 평균값이 낮기 때문에 이진화 영상 추출이 용이하도록 실험을 통해 평균값에 $0x30$ 을 더한 값을 사용하였다.

2.2 잡음제거

이동물체 검출의 정확성 유무는 알고리즘의 초기단계인 배경영역과 움직임 영역의 성공적인 분리에 달려있다. 그러나 외부의 조도 변화 등의 여러 원인에 따른 배경영역 내에 변화에 의해 작은 수의 픽셀들로 이루어진 임펄스 잡음이 존재하여 실제 이동물체외의 영역을 움직임 영역으로 잘못 인식할 수 있는 경우가 발생한다. 따라서 이진 차영상에 대해 미디언 필터링을 취함으로써 이러한 임펄스 잡음을 제거한다. 미디언 필터링은 마스크의 크기를 지정하여 마스크 내 영상의 화소값들을 추출한 후, 크기 순서대로 정렬한 다음에 중간값을 찾아서 이것을 해당 픽셀의 화소값으로 대체하는 것이다.

미디언 필터링은 임펄스 잡음을 갖는 영상에 대해서는 확실히 잡음을 제거하며 경계선들을 보존할 수 있으나 균질 잡음(Uniform noise) 영상에 대해서는 잡음 제거에 한계가 있다. 블록 형태의 마스크는 효율이 좋은 편이며, 교차형(Cross) 마스크는 아주 단순하게 처리할 때 사용하며, 다이아몬드 형은 가장 많이 사용하는 것으로 알려져 있다. 본 연구에서는 $3x3$ 크기의 블록마스크를 적용하였다.

2.3 영상투영

움직임 영역의 정확한 위치를 찾기 위하여 이진 차영상에 대하여 그림 2와 같이 수평과 수직 방향으로 투영(Projection)하여 움직임 영역을 1차원 위치정보로 변환한 후, 적당한 임계값 이상의 투영 결과만을 최소 경계영역(MBB)으로 결정하고 그 좌표를 추출한다. 그림의 수평 및 수직 방향으로의 투영은 움직임 영역에서 픽셀값이 255인 픽셀수를 수평과 수직 방향으로 누산하여 구한 값에서 일정값 이상의 영역을 움직임 영역으로 판정한다.

그러나 영상 전체 영역에 대하여 수평/수직 방향으로 투영하면 그림 2와 같이 두 개의 분리된 물체가 하나의 물체로 인식된다. 따라서 먼저 이진 차영상을 $M \times N$ 개의 다중 영역으로 분할한 후, 각 다중 영역별로 수평/수직 방향으로 투영한다. 따라서 투영 결과로 움직임 영역내에 위치한 다중영역에서 한 개씩의 작은 MBB를 얻는다. 그림 3에서 확인할 수 있듯이 투영 결과로 얻은 작은 사각형들이 이동물체의 영역위로 채워지게 되어 다중 영역의 MBB들이 이동물체의 형태대로 분포된다. 이와 같이 다중 영역 분할을 이용하여 작은 MBB들을 얻고 이 MBB들을 합병하여 독립된 이동 물체를 얻게 되는데, 그림 3과 같이 직사각형과 타원이 각각 서로 다른 이동물체로 인식된다.

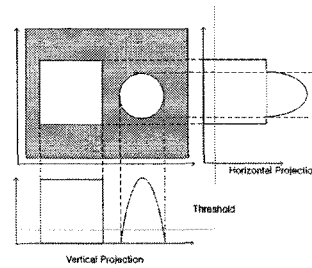


그림 2. 수평/수직 방향 투영

Fig. 2. Projection of horizontal/vertical directions

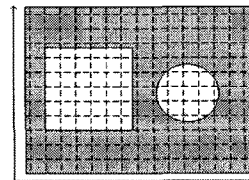


그림 3. 다중영역에서의 투영결과

Fig. 3. Projection result in multiple areas

2.4 합병에 의한 이동물체 추출

이동물체의 경계영역을 추출하기 위해서는 다중 영역의 MBB를 인접한 것들로 서로 합병하는 과정이 필요하다. 다중 영역의 MBB 합병은 그림 4와 같이 세 가지 조건을 적용하였다. 그림 4에서 점선으로 그려진 2개의 작은 사각형들이 합병되어야 할 다중 영역의 Small MBB들이다. 다중 영역에서의 수평/수직 방향 투영을 통하여 얻은 Small MBB들의 좌표는 식 2와 같이 표현된다.

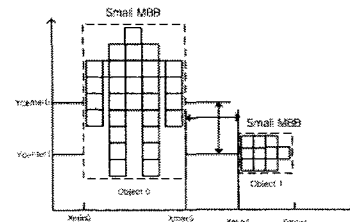


그림 4. 다중 영역의 MBB 합병 조건

Fig. 4. MBB merging conditions of multiple areas

$$Small\ MBB\ Border = [X_{min}, X_{max}, Y_{min}, Y_{max}] \quad (2)$$

다중 영역의 MBB 합병과정은 영상의 좌측 상단 MBB로부터 우측 하단에 위치한 MBB의 순으로 수행한다. 영상 크기가 320×240 이고 다중 영역 분할이 $M \times N$ 에서 다중 영역의 MBB가 x-축상에서 인접해 있는지를 우선 조사한다. x-축상에서의 거리 측정은 그림 4에서 확인할 수 있듯이 각 MBB의 x-축 최소점과 최대점을 사용하는데, x-축상에서 다중 영역의 MBB 합병여부는 식 3에 따라서 판별한다. 일단 x-축상에서의 거리가 충분히 짧다면 y-축상에서의 거리를 측정하는데, y-축상의 거리 측정은 그림에서 각 MBB의 y-축상의 중심점을 사용하며, 합병여부는 식 5의 조건으로 결정한다. 3.2절의 실험에서 이동물체인 사람을 합병하기 위한 값으로서 많은 실험을 통해 식 3과 식 5의 합병조건이 결정된 것이다.

$$\left| X_{\max 0} - X_{\min 0} \leq \frac{320}{2N} \right| \quad \text{혹은} \quad \left| X_{\max 1} - X_{\min 0} \leq \frac{320}{2N} \right| \quad (3)$$

$$Y_{\text{center}0} = \frac{Y_{\min 0} + Y_{\max 0}}{2}, Y_{\text{center}1} = \frac{Y_{\min 1} + Y_{\max 1}}{2} \quad (4)$$

$$\left| Y_{\text{center}0} - Y_{\text{center}1} \right| \leq \frac{240}{M} \times 3 \quad (5)$$

III. DSP 시스템 구현 및 실험

3.1 DSP 기반 시스템 설계 및 제작

이동물체 검출 및 추적 시스템은 영상입출력, 영상처리, 그리고 이동물체 추적표시 기능으로 구분할 수 있으며, 하드웨어로는 그림 5와 같이 카메라, FPGA 보드, DSP 보드와 모니터로 구성된다.

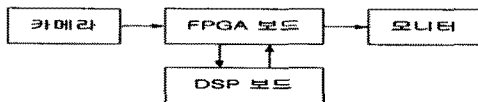


그림 5. 시스템 하드웨어 구성도
Fig. 5. System hardware configuration

다중 이동물체 검출 및 추적 알고리즘 구현은 CCD 카메라로부터 영상을 획득하고 난 후 영상처리를 통해 이동물체를 인식하고 연속적인 입력 영상에 따라 이동물체의 위치를 지속적으로 검출하여 모니터에 이동물체를 포함하는 박스 형태로 추적결과를 나타내도록 구현하였다. 영상 데이터 입력과 이동물체 표시를 위해 처리된 영상 데이터 입출력은 FPGA 보드에서 수행되며, 영상처리는 DSP 보드에서 수행된다.

3.1.1 FPGA 보드 설계

영상입력과 출력을 위한 FPGA 보드는 직접 설계하고 제작하였으며, 그림 6은 FPGA 보드의 블록도와 외부와의 인터페이스이며, 그림 7 제작된 FPGA 보드를 보여주고 있다. FPGA 보드에는 Altera 사의 CycloneIII EP3C25Q240C8NES 칩, 비디오 디코더, VGA 인코더가 내장되어 영상의 입력과 출력을 수행하도록 설계하였다.

비디오 디코더는 Analog Devices사의 ADV7180을 사용하여 카메라로부터 영상신호를 입력받는다. ADV7180은 NTSC/PAL/SECAM 등의 영상신호를 지원하는 10 비트 ADC를 내장하고 있고 3개의 입력 채널에는 Anti-aliasing 필터를 내장하여 CVBS(Composite), Y/C(S-video), YPrPb(Component)등의 비디오 입력을 지원한다.

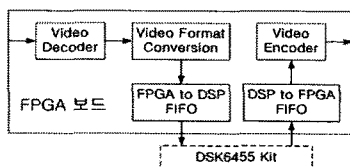


그림 6 FPGA 보드 블록 다이어그램
Fig. 6 Block diagram of FPGA board

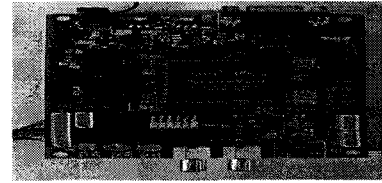


그림 7 FPGA 보드
Fig. 7 FPGA board

카메라로부터 들어오는 아날로그 영상신호를 비디오 디코더를 이용하여 10-Bit 디지털 신호로 변환하여 FPGA로 보낸다. FPGA 칩에서는 그림 8과 같이 입력된 ITU646 YCbCr 4:2:2포맷 데이터를 비디오 디코더에 동기를 맞추어 YCbCr 4:4:4 포맷으로 변환한 후, DSP가 영상처리를 효율적으로 수행할 수 있도록 8-bit RGB신호로 변환하여 내부 FIFO에 저장한다. FPGA 내부 FIFO는 고속 데이터 전송을 위해 100MHz 동기식 EMIF(External Memory Interface)로 구현하였으며 효율적인 DSP와의 인터페이스를 위해 Double buffer를 사용하였다.

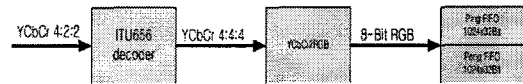


그림 8. 데이터 변환 과정
Fig. 8. Data conversion process

DSP 보드에서 처리한 영상 데이터는 고속 EMIF를 통하여 FPGA 내부 메모리에 저장되고 구현된 VGA Timing Core에 의해서 비디오 DAC로 전달되어 모니터로 처리된 영상을 출력한다.

3.1.2 DSP 보드

영상처리는 DSP 보드에서 수행되는데 알고리즘의 DSP 구현을 위해서 Spectrum Digital사의 DSK6455 키트를 사용하여 코딩하였다. 이 키트는 TI사의 고정 소수점 TMS 320C6455 processor를 이용하여 개발한 것으로 본 영상처리에 적합하며 DSP 보드 블록도와 DSP 보드는 각각 그림 9와 그림 10과 같다.

DSP 보드는 1.2Ghz의 연산능력을 갖는 고성능 DSP를 중심으로 128Mbyte의 250Mhz DDR2 RAM, 4Mbytes의 non-volatile Flash memory, 외부 메모리와 통신할 수 있는 64bits EMIF 연결 커넥터 등을 내장하고 있다. 또한, 빠른 접근을 위한 2 Level cache와 효율적인 메모리 데이터 전송을 위한 EDMA3(Enhanced Direct Memory Access Ver.3)을 지원한다.

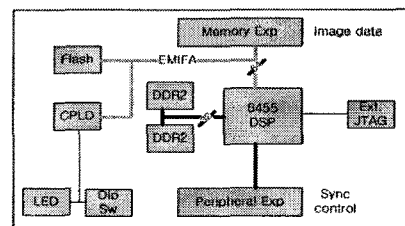


그림 19. DSP 보드 블록도
Fig. 9. Block diagram of DSP board

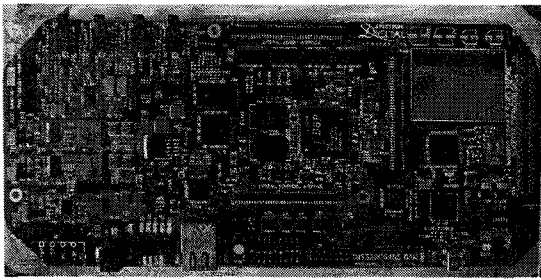


그림 10. DSP 보드
Fig. 10. DSP board

DSP 보드는 FPGA 보드로부터 영상 데이터를 입력받아 처리하고 다시 FPGA 보드로 처리된 영상 데이터를 출력한다. 영상의 모든 이동과정은 EDMA3 방식의 메모리 전송방식을 이용한다. EDMA3는 DMA 메모리 전송 전송방식에서 메모리 전송을 더 원활하게 하기 위하여 TI 사에서 사용하고 있는 전송방식으로서 DSP 코어는 메모리 전송에 시간을 소비하지 않고 영상처리에 많은 시간을 사용하게 함으로써 처리효율을 극대화할 수 있다.

영상입력으로는 DSP의 외부 DDR2 메모리에 두 개의 영상 입력 버퍼를 선언하여 하나의 버퍼가 입력을 받을 때 다른 입력 버퍼가 이전에 입력받은 영상을 보관하고 있다. 영상처리는 영상 입력과 동시에 이루어지는데 하나의 버퍼에 대한 영상처리가 수행되는 동안에 다른 버퍼는 다음의 영상을 입력하게 된다. 그림 11과 같이 영상입출력을 Ping-pong 더블 버퍼를 이용하여 교대로 사용함으로써 영상처리와 영상입출력을 효율적으로 동작시킬 수 있다.

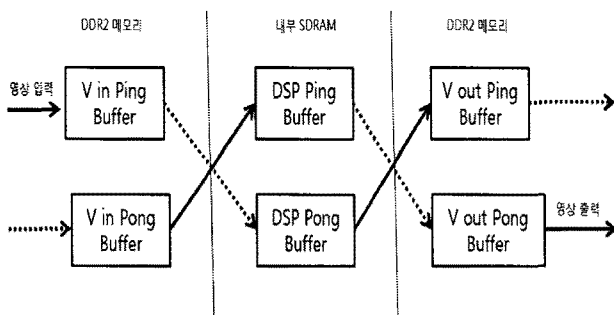


그림 11. 메모리 구조 및 데이터 흐름도
Fig. 11. Memory structure and data flow

구체적으로 살펴보면 카메라로부터 획득한 영상데이터를 DSP 내부의 DDR2 메모리의 V in Ping 버퍼에 한 라인씩 EDMA3 방식으로 저장하는데, V in Ping 버퍼에 영상을 저장하는 동안 이전 영상이 저장되어 있는 V in Pong 버퍼의 한 프레임에 해당하는 영상을 DSP Ping 버퍼라 명명된 DSP 내부 메모리에 전송하여 영상처리 알고리즘을 수행한다. 그리고 영상처리된 결과를 V out Ping 버퍼라 명명된 DDR2 메모리로 전송함과 동시에 V out Pong 버퍼에 저장되어 있던 이전 프레임의 영상처리 결과가 한 라인씩 FPGA로 보내 모니터로 출력된다. 이와 같은 과정이 반복

수행 되면서 영상입력, 영상처리, 영상출력의 동작이 교대로 수행된다. 카메라에서 FPGA에 입력되는 영상 해상도는 640x480이나 영상처리 시간을 단축시키기 위해서 외부의 DDR2 메모리에서 영상을 320x240 해상도로 축소하여 내부 메모리로 옮겨서 영상처리를 수행함으로써 영상처리 속도를 향상시켰다.

3.2 이동물체 검출 및 추적 실험

실험은 실내(학교건물 내)의 실외(야외 도로변과 잔디밭)에서 수행되었으며, 이진화를 위한 임계값은 0x15로, 투영할 영상 분할은 10x10 픽셀크기로 설정하였다. 그리고 합병조건을 위해서 이동물체는 움직이는 사람을 기준으로 하였으며 이동속도는 달리기 이하로 가정하였다.

그림 12는 단일 이동물체와 다중 이동물체의 검출 및 추적결과를 보여주고 있는데 그림 (a)-(c) 야외 도로변의 이동물체 추적으로서 영상내에서 이동중인 사람이 검출됨을 알 수 있다. 그림 (d)는 야외 잔디밭에서 이동물체로서 이 경우에는 사람 영상과 배경영상이 도로변의 영상보다는 뚜렷하지 않은 경우이다. 그림 (e)는 실내에서 이동물체 검출을 보여주고 있는데, 특히 10x10으로 분할된 각 영역에서 검출된 MBB들이 사각형으로 표시되어 있으며 합병조건에 의해 합병된 물체 전체의 외곽선도 더 자세하게 확인할 수 있다. 구현된 다중 이동물체 검출 및 추적기능이 실내와 야외 환경에서 성공적으로 수행됨을 확인하였다.



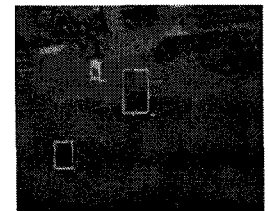
(a) 한 사람 이동물체
(a) 1-man moving object



(b) 2 사람 이동물체
(b) 2-men moving objects



(c) 3 사람 이동물체
(c) 3-men moving objects

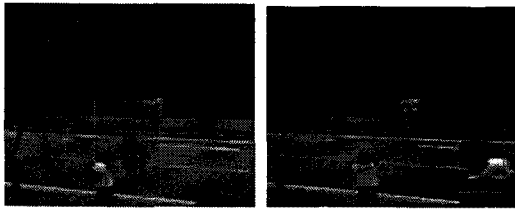


(d) 3 사람 이동물체
(d) 3-men moving objects



(e) 2 사람 이동물체
(e) 2-men moving objects

그림 12. 이동물체 검출 및 추적 결과
Fig. 12. Detection and tracking results of moving objects



(a) 한 개의 물체로 인식
(a) Recognition of 1 object
(b) 3 개의 물체로 인식
(b) Recognition of 3 objects

그림 13. 인접한 다중 이동물체 추적 결과

Fig. 13. Tracking results of closer moving objects

그림 12에서 다중 이동물체인 경우 이동물체의 MBB 사이의 거리가 40픽셀 이하인 경우 하나의 물체로 인식하도록 설정하여 실험하였으며, 이동물체 크기에 따라서 조절할 수 있다. 그림 13은 이동물체간의 거리가 상대적으로 짧은 경우로서 그림 (a)는 3개의 이동물체가 하나로 인식하는 경우이고, 그림 (b)는 4개의 이동물체가 3개의 이동물체로 인식되는 경우이다. 물론 시간이 지남에 따라서 이동물체간의 거리가 멀어지면 독립적인 이동물체로 인식하게 된다.



그림 14. 이동물체 추적 모니터링

Fig. 14. Tracking monitoring of moving objects

그림 14는 VGA 모니터에 나타나는 추적결과 화면으로서 좌측 상단의 영상은 현재 획득한 영상을, 우측 상단의 영상은 배경영상을, 그리고 좌측 하단 영상은 상단 좌우측 영상에 대한 차영상으로서 이동물체를 확인할 수 있는 기능도 구현하였다.

실험결과를 요약하면 이동물체의 추적속도는 320x240 영상 해상도에서 30 frames/sec로서 이동물체의 실시간 검출 및 추적이 만족스럽게 동작되었으며, 배경영상과 이동물체의 Graylevel이 유사한 경에는 이동물체 인식방법의 개선이 요구된다.

제작된 시스템에서 사용한 카메라의 영상획득 최대속도가 30 frames/sec로서 영상을 읽거나 출력할 때 카메라의 싱크신호를 사용하기 때문에 전체 속도는 카메라의 영상획득 속도에 제한된다. 표 1은 DSP에서 수행된 시간으로서 한 프레임의 영상데이터를 DSP 외부와 내부로의 읽기와 쓰기, 영상처리 알고리즘 구현 시간을 오실로스코프로 직접 측정된 결과이다. 영상읽기는 320x240 크기로 읽는 반면 영상쓰기에는 그림 14와 같이 640x480 크기로 보내기 때문에 영상쓰기 시간이 더 길다. 표에서와 같이 한 프레임의 영상을 처리하는 시간을 모두 더하면 실제 소요시간은 약 10msec로서, 카메라의 영상획득 속도를 고려하지 않으면 실제 추적속도는 초당 100 프레임까지 동작이 가능하다.

표 1. 실행 시간

Table 1. Processing time

처리 내용	소요 시간
영상 읽기(DDR2 → DSP RAM)	0.1146 μ sec
영상처리	9.8 msec
영상 쓰기(DSP RAM → DDR2)	0.6506 μ sec

IV. 결론

본 논문에서는 무인 영상감시를 위해 기존의 DSP 키트와 영상 입출력을 위해 설계 및 제작된 FPGA 보드를 사용하여 다중 이동물체의 실시간 검출 및 추적에 대한 연구를 수행하였다. 카메라를 통해 연속적으로 획득한 영상에서 움직임 에너지를 이용한 이동물체 검출로서 기존 방법에서 영상처리 시간을 단축하기 위해 알고리즘을 간략화하였으며 DSP 기반 하드웨어에 구현하였다. 개발된 시스템을 사용하여 수행한 실험 결과에 의하면 다중 이동물체의 검출과 추적은 실시간으로 동작함으로써 DSP 기반의 임베디드 시스템 구현 가능성을 보여주었다.

이동물체가 카메라에서 획득한 영상영역의 바깥쪽으로 이동할 경우에는 더 이상 추적이 불가능하다. 이 경우 지속적으로 이동물체를 추적하기 위해서는 카메라 방향을 이동물체에 따라서 움직여 획득한 영상내에 항상 존재하도록 카메라의 연속적인 제어가 요구된다. 이를 위해 현 시스템을 사용하여 특정 이동물체를 따라서 카메라를 움직이면서 이동물체를 추적할 수 있는 연구를 수행할 예정이다.

참고 문헌

- [1] G. L. Foresti, "Real-Time Detection of Multiple Moving Objects in Complex Image Sequences", Int. Journal of Imaging System and Tech., vol. 10, issue 4, pp. 305-317, 1999.
- [2] Young Ho Kim, Kyu Won Lee, Jun Geun Jeon and Kyu Tae Park, "Moving object discrimination and tracking for unmanned surveillance system", SPIE, vol. 2727, pp. 175-185, 1996.
- [3] Jakub Segen, Sarma Pingali, "A Camera-Based System for Tracking People in Real Time", IEEE Proceedings of ICPR '96, vol. 3, pp. 63-67, 1996.
- [4] Johnson I Agbinya, David Rees, "Multi-Object Tracking in Video", Real-Time Image, vol. 5, pp. 295-304, 1999.
- [5] Holger Lange "Real Time Motion Detection for Target Acquisition "on the move" based on a Nonlinear Filter using Short Time and Medium Time Image Differences", SPIE Conference on Real-Time Image IV, vol. 3645, pp. 12-23, 1999.
- [6] Kyu Won Lee, Jin Woong Kim, "Moving Object Segmentation Based on Statistical Motion Model", Electronics Letters, vol. 35 Issue: 20, pp. 1719-1720,

30 Sept. 1999.

[7] G. Rigoll, B. Winterstein, S. Muller, "Robust Person Tracking in Real Scenarios with Non-Stationary Background Using a Statistical Computer Vision Approach", Visual Surveillance, 1999. Second IEEE Workshop on, (VS '99), pp. 41-47, 1999.

[8] Don Murray, Anup Basu, "Motion Tracking with an Active Camera", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 16, no. 5, pp. 524-530, May 1994.

[9] Q. Cai, A. Mitiche, and J.K Aggarwal, "Tracking Human Motion in an Indoor Environment", IEEE, pp. 70-76, 1995.

[10] 유성희, "무인 영상 감시를 위한 이동물체 검출 및 추적 시스템의 실시간 구현에 관한 연구", 수원대학교 대학원, 석사 학위 논문, 2001.

[11] Anil K. Jain, "Fundamentals of Digital Image Processing", Prentice Hall, 1989.

[12] 김상기, "무인 영상 감시를 위한 이동물체 검출 및 추적에 관한 연구", 금오공과대학교 대학원, 석사 학위 논문, 2004.



이 옥재(Uk-jae Lee)

2008년 금오공과대학교 전자제어공학(공학사)
 2010년 금오공과대학교 전자제어공학(공학석사)
 2010년 ~ 현재 박사과정 중
 ※ 관심분야: 디지털 신호처리, 영상처리



이 상락(Sang-rak Lee)

1997년 계명대학교 전자공학과 (공학사)
 2006년 금오공과대학교 전자제어공학(공학석사)
 2008년 ~ 현재 박사과정
 2008년 ~ 현재 삼진(주) 상무이사
 ※ 관심분야: 영상압축 및 전송, 실시간 원격 모니터링



최 한고(Han-go Choi)

1979년 경북대학교 전자공학과 (공학사)
 1988년 Univ. of Florida, USA
 전기 및 전자공학과 (공학석사)
 1992년 Univ. of Florida, USA
 전기 및 전자공학과 (공학박사)

2001년 MIT, USA (Research Scientist) 1979년 ~
 1986년 국방과학연구소, 연구원
 1993년 ~ 현재 금오공과대학교 전자공학부 교수
 ※ 관심분야: 실시간 DSP, 컴퓨터비전, 생체 신호처리



김 양수(Yang-soo Kim)

1996년 경일대학교 전자공학과 (공학사)
 2001년 금오공과대학교 전자통신공학(공학석사)
 2008년 ~ 현재 박사과정 중
 2001년 ~ 현재 3B시스템 H/W개발팀장
 ※ 관심분야: DSP H/W 설계, DSP 영상 신호처리