

그래픽 처리 장치를 사용한 실시간 배경 모델링

이선주^o 정창성

고려대학교 전자컴퓨터 공학과

sunju0807^o@koea.ac.kr, csjeong@charlie.korea.ac.kr

Real-Time Background Modeling Using the Graphic Processing Units

Sun-Ju Lee^o, Chang-Sung Jeong

Dept. of Electronics and Computer, Korea University

요 약

움직이는 오브젝트(Object)를 추출하기 위한 배경 제거(Background Subtraction) 단계는 실시간 감시 시스템(Real-time Surveillance System)에서 중요한 과정 중에 하나이다. 배경 제거를 효과적으로 진행하기 위한 배경을 모델링, 배경 유지 보수 방법이 존재하는데, 효율성이 높은 방법으로 적응적 가우시안 혼합 배경 모델링(Adaptive Gaussian Mixture Background Modeling)이 제시되고 있다. 본 논문에서는 이 기법을 바탕으로 하여 이러한 실시간 배경 모델링 시스템을 구현하려 하고, 중앙 처리 장치(CPU)가 아닌 그래픽 처리 장치(Graphic Processing Units : GPU)를 사용하여 보다 향상된 방법을 구현함으로써 관련 사항을 제안하려 한다.

1. 서 론

실시간 감시 시스템은 컴퓨터 비전(Computer Vision)에서 흥미롭게 다뤄지는 부분으로 현재 많은 곳에서 연구가 수행 되어 지고 있다. 이러한 실시간 감시 시스템에서 배경 모델링과 배경을 유지 보수하기 위한 업데이트는 매우 중요한 부분이고, 시스템의 다음 단계에서 좋은 결과를 얻기 위해서 더욱 정확하게 다루어져야 한다.

배경을 모델링 하는 방법에는 여러 가지 기법들이 연구 되었다. 그 중 [1]에서 제시한 부분은 장면에서 나타나는 빛에 변화나 잡음들에 대한 문제들을 가우시안 혼합 모델링을 이용하여 해결하였다. 시간이 지나면서 발생하는 점에 대하여 점층적으로 배경을 업데이트 하고 가우시안 분포를 하나가 아닌 다수를 사용함으로써 효과적으로 배경 모델링을 수행한다. 이 방법은 최근에 가장 많이 사용되고 [1,2,4,5], 배경 모델링의 더욱 향상된 결과를 얻기 위해 연구가 계속 진행되고 있다. [2]에서는 [1]에서 구현된 부분을 RGB 채널이 아닌 YUV 채널을 이용하고 더 정확한 결과를 얻기 위해서 깊이(Depth) 정보를 더 하여 시스템을 구현하였다.

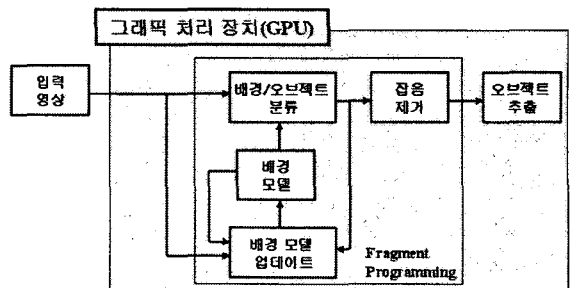
그래픽카드(Graphic Card)는 컴퓨터 영상정보의 처리, 가속화, 신호전환 등 CPU의 그래픽 작업으로 생기는 병목 현상을 해결하기 위해서 특수한 목적으로 만들어졌다. 그래픽 처리 장치(Graphic Processing Unit : GPU)는 3D 그래픽을 구현하는 데 있어 애플리케이션을 위해 사용했던 3D 그래픽 가속 칩을 개량한 것으로 그래픽 연산처리를 하는 코어이다. 이 그래픽 처리 장치는 수학적 연산이나 안티 앨리어싱(Anti Aliasing) 등과 같은 영상처리 관련 된 부분을 매우 빠르게 처리한다. 그래서 현재 3D 영상을 다루는 부분과 컴퓨터 비전과 같이 많은 분야에서 관련 연구를 진행하고 있다.

만약 실시간 감시 시스템을 그래픽 처리 장치를 이용하여 구현한다면 그 효율성은 높아질 것이 분명하다. 게다가 우리는 [1]에서 사용한 가우시안 혼합 배경 모델링 기법을 참조하고, 그래픽 처리 장치를 기반으로 실내 환경에서 움직이는 오브젝트 추출, 배경 모델링, 그리고 배경 유지 보수 등을 적용해 보기로 한다.

앞으로 이 논문에서 설명할 부분으로는 2장에서는 배경을 모델링을 하는 가우시안 혼합 모델링, 오브젝트 추출 그리고 배경 모델링에 적용된 그래픽 처리 장치에 대해 시스템 구조와 함께 전반적인 연을 논할 것이다. 3장에서는 구현된 시스템에 대한 결과 및 분석을 보여주고, 4 장에는 결론을 제시 하도록 한다.

2. 그래픽 처리 장치를 이용한 시스템

그래픽 처리 장치를 이용한 실시간 배경 모델링의 시스템의 전체적인 구조는 <그림 1>에서 보이는 바와 같다.



<그림 1> 구조

CPU에서 실시간으로 영상을 입력받아 그래픽 처리 장

치를 이용하여 가우시안 혼합 모델을 바탕으로 배경과 오브젝트를 분류하고, 배경모델을 업데이트 한다. 그리고 오브젝트로 분류된 영상에서 나타나게 된 잡음을 제거하여 최적화된 결과를 도출한다. 영상입력을 제외한 모든 과정들은 그래픽 처리장치를 사용하여 한층 더 좋은 결과를 얻을 수 있도록 구현한다.

2.1 가우시안 혼합 배경 모델링

먼저 움직이는 오브젝트를 추출하기 위해서 배경을 모델링 한다. 이 시스템에서 사용한 배경 모델링의 각 픽셀은 K개의 가우시안 분포를 가지고 있다. 가우시안 분포는 식1로 $\mu_{i,t}$ 평균, $\sum_{i,t}$ 분산, $w_{i,t}$ 가중치로 이루어져 있고, K개의 가우시안 분포를 사용함으로써 조명의 변화나 카메라의 미세한 움직임 등에 대한 부분을 보정한다. 가중치들의 합은 1이며, K는 보통 실내의 경우 3을 사용하고, 배경의 변화가 많은 실외의 경우 실내보다 좀 더 많은 5~7을 사용한다.

$$f(X_t) = \sum_{i=1}^K w_{i,t} * \gamma(x_t, \mu_{i,t}, \sum_{i,t})$$

$$\gamma(X_t, \mu, \sum) = \frac{1}{(2\pi)^{\frac{n}{2}} |\sum|} e^{-\frac{1}{2}(X_t - \mu)^T \sum^{-1} (X_t - \mu)} \quad (1)$$

만약 현재 입력되는 영상의 픽셀의 분포가 모델링 되어 있는 3개의 가우시안 분포 중에서 하나의 분포와 일치하게 되면, 해당되는 배경 모델의 구성인 평균, 분산, 가중치를 업데이트한다. 그러나 움직이는 오브젝트가 발생하여 가우시안 분포가 적합하지 않게 되면, 배경 모델은 업데이트 하지 않고 그대로 유지한다. 업데이트는 배경 모델링과 현재 입력된 영상을 시간에 대하여 점층적인 영향을 주도록 한다. 업데이트 관련 식2는 [1,3]을 참고한다. α 는 학습률(Learning Rate)로 시간에 따른 값이다. 이 값은 시간이 지속 될수록 점점 감소한다. 가중치의 경우, 아래의 식에서와 같이 일치하는 가우시안 분포의 가중치는 증가하고($M = 1$), 일치하지 않는 분포의 가중치에 대해서는 값은 감소한다. ($M = 0$). 또한 만약에 입력 영상의 분포가 여러 가지의 가우시안 분포와 일치하게 되는 경우, 그 중 가중치가 가장 큰 값의 가우시안 분포를 업데이트 한다.

$$w_{i,t} = (1 - \alpha)w_{k,t} + \alpha M \begin{cases} match & 1 \\ unmatch & 0 \end{cases}$$

$$\mu_i = (1 - \rho)\mu_{i-1} + \rho X_t$$

$$\sigma_i^2 = (1 - \rho)\sigma_{i-1}^2 + \rho(X_t - \mu_i)^T (X_t - \mu_i)$$

$$\rho = \alpha \gamma(X_t | \mu_k, \sum_k) \quad (2)$$

2.2 오브젝트 추출

입력된 영상의 분산이 K개의 가우시안 분포로 이루어진 배경 모델에서 표준편차의 2.5배 보다 적은 값을 가지게 되면 이 시스템은 움직이는 오브젝트로 인식한다. 마치

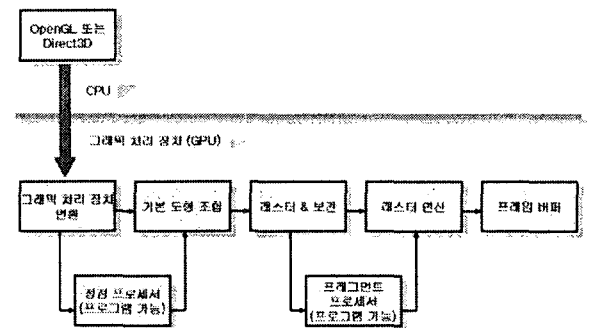
막으로 움직이는 오브젝트만으로 이루어진 영상에서 생길 수 있는 잡음을 제거하여 최종 적인 결과를 얻어 낸다.

2.3 그래픽 처리 장치

앞에서 논한 가우시안 혼합 배경 모델링을 CPU가 아닌 그래픽 처리 장치를 사용하여 프로그램을 시도한다는 점이 이 논문에서 제시한 부분이다. 그래픽 처리 장치(Graphic Processing Unit : GPU)는 3D장면을 그리고, 그래픽 처리 연산을 하는 등 특수한 목적에 의해 설계되어 CPU에 비해서 그래픽 관련 작업에서는 매우 유용한 장점을 가지고 있다. 최신 그래픽 하드웨어의 그래픽 처리 장치는 1초에 몇 천만 개의 정점(Vertex)들을 변환하고, 수억 혹은 몇 수십억 개의 프래그먼트(fragment)들을 래스터화(Rasterization)한다. 앞으로 그래픽 처리 장치는 더욱더 많이 발전되면 훨씬 더 빨라질 것이다. 하지만 그래픽 처리 장치는 CPU에서 처리하는 범용적인 부분은 처리하지 못한다는 점을 고려해야 한다. [7,8]

고성능의 그래픽 처리장치를 위해서 Cg(C for Graphic)가 존재 하는데 이러한 Cg는 정점과 프래그먼트 처리의 특수한 작업을 수행하기에 가장 적합하게 작성되어 있다.

<그림 2>는 오늘날 사용하는 그래픽 처리 장치 파이프라인이다. 애플리케이션은 일반적으로 다각형, 선분, 점등과 같은 기하 도형들로 조합될 정점들을 그래픽 처리 장치로 보낸다. 모든 정점은 위치정보, 색상, 텍스처 좌표 그리고 법선 벡터와 같이 여러 가지 속성 정보를 가지고 있다. 변환된 정점으로 기본도형을 조합하고 래스터화 한다. 또한 프래그먼트 매개변수를 보간하고, 텍스처를 입히고, 수학 연산들을 수행하고 최종 색상을 정한다. 래스터 연산은 프래그먼트마다 최종적인 연산들을 즉각 수행한다.



<그림 2> 그래픽 파이프라인

그래픽 파이프라인의 프로그래밍 가능한 부분이 정점 프로세서와 프래그먼트 프로세서이다. 최근의 그래픽 처리 장치는 계속적으로 발전되고 있으며, 정점과 프래그먼트 프로세서들에서 단순한 설정뿐만 아니라 프로그래밍도 가능하다.

이 논문에서는 정점프로세서 프로그래밍은 사용하지 않

고, 프래그먼트 프로세서만 사용하여 시스템을 프로그래밍한다. 입력된 영상을 프래그먼트 프로그램을 이용하여 배경을 모델링, 움직이는 오브젝트 추출, 배경 모델 업데이트 그리고 마지막으로 잡음 제거를 한다. 이로 인한 결과는 프레임 버퍼(Frame Buffer)에 저장한다. 프래그먼트 프로그램은 Cg로 구현한다.

3. 실험 결과

가우시안 혼합 배경모델링을 그래픽 처리 장치를 이용해서 구현하기 위해서 이 시스템은 OpenGL과 Cg를 이용하였다. 그래픽 하드웨어에 렌더링 명령을 알려주기 위해서 OpenGL을 사용하고, 프래그먼트 프로그램을 Cg로 사용하였다. 그래픽 처리 장치는 Nvidia Geforce 6 시리즈를 이용하고, 입력 영상으로 사용되는 프레임의 크기는 320×240 이다.

<그림 3> 이러한 실험의 영상을 보여주고 있다. 왼쪽 영상의 경우 실시간으로 입력되고 있는 영상이고, 오른쪽의 영상은 가우시안 혼합 모델링 기법을 이용하여 추출된 움직이는 오브젝트의 영상이다.

그래픽 처리 장치를 기반으로 실험한 경우, 초당 약 83 프레임 정도의 성능을 보여준다. 기존에 나왔던 많은 시스템들의 프레임에 비해 3~4배 정도 빠른 속도이다. <표1>은 CPU를 사용한 논문들과의 결과를 비교한다. 표에서 나타내는 바와 같이 그래픽 처리 장치를 사용한 본 시스템은 기존에 비해 뛰어난 결과를 보여준다. 만약 그래픽 처리 장치의 특성에 맞게 최적의 프로그램을 구현한다면 더욱 좋은 결과를 얻을 수 있을 것이다.

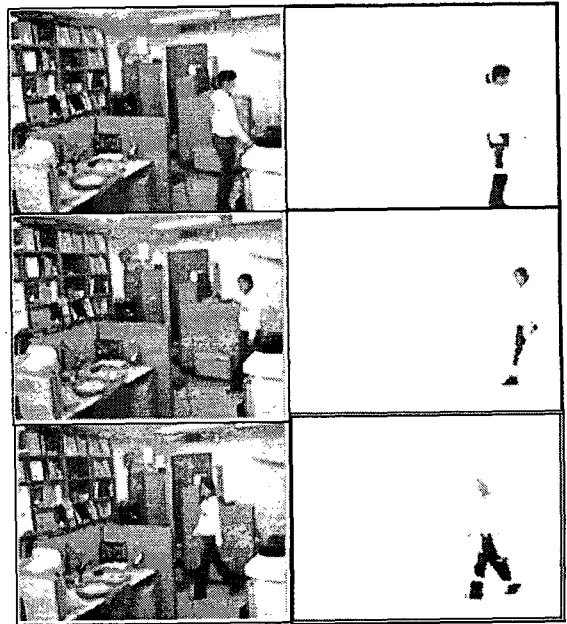
method	Frame size	speed
Adaptive Background Mixture Modeling[1]	160×120	11~13 frame/sec
Modeling Using Color and Depth[2]	320×240	15 frame/sec
Effective Gaussian Mixture Learning[6]	160×120	15 frame/sec
Our System	320×240	83 frame/sec

<표 1> 비교

4. 결론

그래픽 처리 장치는 그래픽 처리를 위한 특수 목적을 가지고 만들어진 하드웨어이다. 이것은 영상을 이용하여 처리하는 과정에 생기는 대부분의 과정들을 하드웨어로 이용하여 처리할 수 있기 때문에 CPU기반의 시스템 보다 3~4배 정도 좋은 결과를 얻을 수 있다. 기존의 시스템뿐만 아니라 영상을 이용하여 구성하는 시스템을 그래픽 처리 장치를 사용하여 현재 많이 시도 되고 있고, 그에 따라 이번 시험 역시 앞으로 발전 가능성에 맞추어 그 성능을 알아보았다. 이번 실험을 토대로 앞으로 진행되는 시스템의 구성을 그래픽 하드웨어로 구현하고 또한 그 특성에 맞게 최적의 프로그램을 한다면 훨씬 좋은 결

과를 얻을 수 있다는 것을 확인할 수 있게 되었다.



<그림 3> 입력된 RGB영상에 가우시안 혼합 배경 모델링을 적용하여 움직이는 오브젝트를 추출한 결과 영상(왼쪽: 입력 영상, 오른쪽: 결과 영상)

참고문헌

- [1]C.Stauffer, W.E.L. Grimson, "Adaptive Background Mixture Models for Real-Time Tracking," Proc. CVPR, Vol.2, 1999, 246-252.
- [2]M.Harville, G.Gordon, and J. Woodfill "Adaptive Video Background Modeling Using Color and Depth," Proc. IEEE. Published in the 2001 International Conference on Image Processing (ICIP-2001), October 7-10, 2001
- [3]A.M.Mclvor, "Background subtraction techniques", In Prof. of Image and Vision Computing, Auckland, New Zealand, Oct. 2000.
- [4]P.W.Power and J.A.Schoonees, "Understanding background mixture models for foreground segmentation", Proc. of IVCNZ 2002, Nov. 2002.
- [5]P.K.T.K. Pong, and R.Bowden: An improved adaptive background mixture model for real-time tracking with shadow detection. In Proc. 2nd European Workshop Advanced Video Based Surveillance System, Sept 2001.
- [6]D.S.Lee, "Effective Gaussian Mixture Learning for Video Background Subtraction", IEEE Transactions on PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 27, NO. 5, MAY 2005
- [7]R.Fernando Series Editor GPU Gems 2
- [8]R.Fernando The CG Tutorial: the Definitive Guide to Pogramable Real-Time Graphics