

GPU기반 Depth Map 획득을 위한 고속 병렬처리 기법

*진문섭 **최지윤 ***추현곤 ****김진웅 *박종일¹⁾

한양대학교 전자컴퓨터통신공학과, 한국전자통신연구원

*jinms@mr.hanyang.ac.kr, **jeromechoi@mr.hanyang.ac.kr, ***hyongonchoo@etri.re.kr
****jwkim@etri.re.kr, *jipark@hanyang.ac.kr

Fast Depth Map Estimation using Parallel Processing based on GPU

*Jin, Moon-Sub **Choi, Ji-Yoon ***Choo, Hyon-Gon ****Kim, Jin-Woong *Park, Jong-II

Dept. Electronics and Computer Engineering, Hanyang University,

Electronics and Telecommunications Research Institute

요약

본 논문은 두 대의 카메라와 한 대의 프로젝터로 구성된 Pro-cam시스템을 이용하여, 출력된 패턴 영상을 카메라로 촬영하고 이를 기반으로 Depth Map을 계산하는 모듈의 실시간 처리를 위한 GPU기반 병렬처리 기법을 제안한다. 입력받은 영상으로부터 구조광의 패턴을 해석하고, Depth Map을 계산하기 위해서, Dynamic pattern decoding하는 과정은 프로젝터의 패턴영상과 촬영된 카메라 패턴영상 간의 관계를 반복적으로 비교하므로, 이를 GPU 프로그래밍을 이용하여 병렬 처리를 통해 고속화하였다. 결과적으로 본 논문에서는 기존 CPU에서 수행했던 속도에 비해 약 18배정도 속도를 개선 할 수 있었다.

1. 서론

3D 방송을 위해서 실시간으로 Depth Map을 획득하고 적용하는 기술은 필수이다. 하지만 카메라로 획득한 영상들의 방대한 데이터 처리의 문제로 인해, Depth Map 추출을 위해 정확도와 속도를 높이는 연구가 활발히 수행되고 있다. 구조광을 이용한 Depth Map 획득 방법은 가장 보편적으로 사용되었고, 많이 연구되고 있는 방법 중에 하나이다. 구조광은 Black&White나 Gray scale의 stripe 패턴을 이용한 방법 [1][2]과 Color stripe pattern을 이용한 방법 [3]이 있다.

본 논문에서는 de Bruijn sequence 와 stripe 를 기반으로 한 Color stripe pattern을 이용하였다. Dynamic decoding programming을 이용하여 Depth를 추출하는 방법 [3]을 GPU를 이용하여 고속화하였다. 알고리즘의 개선으로는 한계가 있으며, 영상의 크기가 크고, 실시간으로 처리하기 위해서는 계산 시간이 증가 할 수 밖에 없다. 높은 해상도의 영상에서의 고속 depth map 추출하는 단계에서 Pattern Decoding 하는 과정은 동일한 연산을 반복 비교 수행으로 이루어진다. 이 점을 착안 하여 반복적인 연산부분을 병렬처리 한다면, 높은 화질의 영상에 대해서 실시간으로 계산 할 수 있게, 고속화를 할 수 있다.

본 논문은 다음과 같이 구성된다. 2장에서는 Depth map 획득 시스템에 대해 설명하고, 3장에서는 병렬처리를 이용한 Pattern Identification, Segmentation, Dynamic decoding 방법을 제시하고, 4장에서는 이를 구현한 실험을 통하여, GPU를 이용한 병렬처리의 유용

함을 증명한다. 마지막으로 5장에서는 본 논문의 결론과 앞으로의 연구 방향을 정리한다.

2. Depth map 획득 시스템

그림 1은 본 논문에서 사용한 Structured light 를 이용한 depth map 획득 시스템 구성도이다. 두 대의 카메라와 한 대의 프로젝터를 통해 패턴 영상을 대상 물체에 조사하고, 이를 촬영한 영상을 통해 Depth map을 추출한다. 입력되는 영상은 각 프레임 당 1280x800크기의 Stereo 영상 이고, 최대 120fps를 가진다.

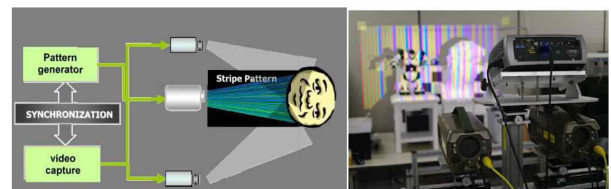


그림 1. Structure light를 이용한 Depth map 획득 시스템 구성도

본 논문에서는 하나의 depth map을 계산하기 위해서 패턴이 다른 좌우 각각 3장의 이미지와 좌, 우 시점 영상 이미지 2장이 필요하다. Depth map을 계산 하는 과정은 그림 2와 같다. 이때 연산해야하는 데이터 양이 많기 때문에, 실시간 처리를 위해서 GPU를 이용한 고속화

1) 교신저자

작업이 필요하다. 그림 2의 과정에서 GPU를 이용하여 고속화 하는 부분은 Pattern Identification, Pattern Segmentation, Depth Estimation 부분이다.

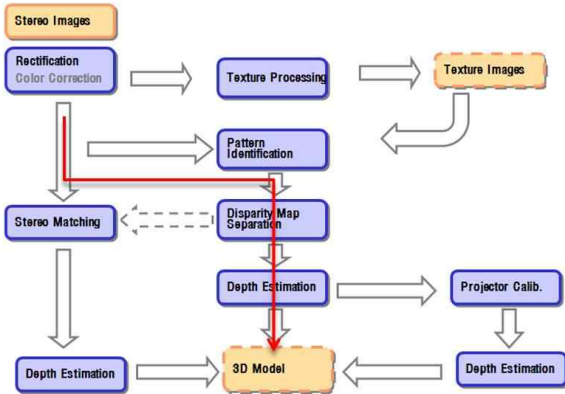


그림 2. Depth map 획득을 위한 수행 과정

3. 영상 데이터 처리

3.1 Pattern Identification

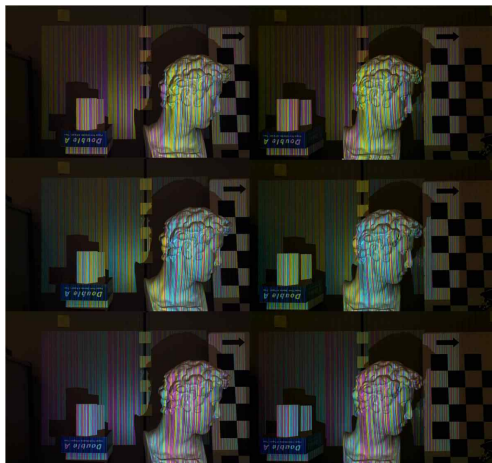


그림 3. 세 가지 다른 컬러 패턴 이미지(좌, 우)

그림 3은 Input 이미지이다. 주어진 6장(좌, 우 각 3장의 패턴이 다른 영상)의 영상을 2장의 시점 영상으로 합쳐 그림4(a)와 같은 Textured 이미지를 만든다. 합쳐진 Textured 이미지로부터 그림자를 제외한 관심영역을 구한다. 최종 관심 영역은 그림4(b)와 같다.

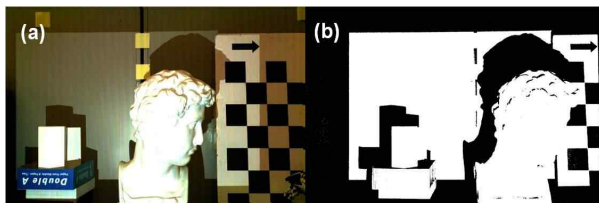


그림 4.(a) Textured 이미지, (b) ROI 이미지

3.2 Pattern Segmentation

Color Stripe Pattern은 기존 연구에서 제시한 방법을 이용한다.[3] 식 1은 N개의 다른 color에서 길이 k를 가지는 stripe sequence를 만드는 전체 패턴 개수를 표현하였다.

$$n(N, k) = N(N-1)^{k-1} \quad (1)$$

3.1에서 구한 결과 이미지로부터 패턴을 구분한다. 해당 영상에서 각 화소별 정확한 Stripe 색상을 검출하기 위해 CIE Lab색 공간으로 변환하여 그림4(a)와 화소 별 색 성분을 구한다. 그림4(b) ROI 이미지와 비교하여 정확한 패턴 비트를 검출해 낸다. 검출된 색에 따라 0-red, 1-green, 2-blue, 4-unknown or black 영역으로 인덱스를 부여한다. Segmentation 단계를 수행한 이미지는 그림5와 같다.

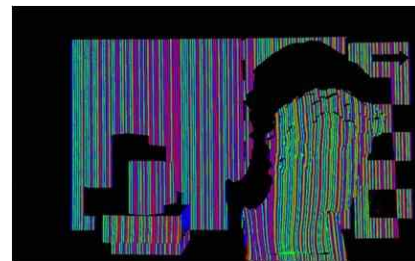


그림 5. Segmented 이미지

3.3 Dynamic Pattern Decoding 병렬화 기법

Structured light 의 Pattern Decoding하는 방법은 크게 LUT를 이용한 Normal Decoding 방법과 Dynamic Programming을 이용하는 방법 두 가지로 나뉜다. 본 논문에서는 Dynamic Programming 방법을 이용하여 Pattern Decoding을 구현하였다[3]. Dynamic Decoding은 두 개의 1차원 Sequence 간의 cost function을 계산한다. 원래의 sequence와 찾아진 sequence를 비교하여 같은 경우, 이전 cost를 증가시켜, 해당 열에서 최고 cost가 후보로 판정한다.

그림 6은 Pattern Decoding과정에서 병렬처리를 이용한 고속기법 개요도이다. 패턴의 길이는 실험 환경에 따라 달라지지만, 병렬처리를 하기 위해서 고정된 어레이가 필요하다. 그러므로 본 논문에서는 패턴 길이를 776으로 고정시켜, GPU를 이용하여 Decoding을 수행한다. GPU에서 처리할 수 있는 블록은 최대 1024개 이다. 영상의 가로(1280)x세로(800)x패턴길이(776)을 가지는 데이터 구조이기 때문에, 병렬처리를 640x800x776의 크기로 두 번 반복하여 수행한다.

위의 데이터 구조를 통하여, Decoding된 패턴 정보는 위상 값을 가진다. 위상 값을 구하기 위해 매트릭스의 최대값을 취하게 된다. 기존의 방법은 위상 값을 일정한 조건에 의해서 최대값을 가지는 화소의 위치를 저장할 것인지, 매트릭스의 최대값을 저장할 것인지 결정하게 된다. 하지만 CUDA는 SIMT(Single-Instruction, Multi-Thread) 구조이기 때문에, 분기 문에 취약하다. 그래서 본 논문에서는 속도 개선을 위해, 매트릭스의 최대값을 가지는 화소위치를 우선 구하고, 조건에 따라서 값을 업데이트하는 방식으로 구조를 변경하였다.

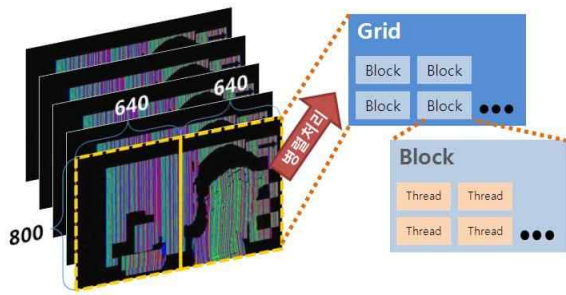


그림 6. 병렬처리를 이용한 Pattern decoding 기법

3.3 Depth Estimation

Dynamic decoding 을 통하여 얻어진 정보를 토대로 최종적인 Depth정보를 획득하는 과정이다. 속도향상을 위해 decoding 된 소수점(float) 정보를 char형으로 GPU를 이용하여 Normalize 하고 다시 float형태의 정보로 복구하는 Post-normalize를 한다. Normalization 영상은 그림 7(a)과 같고 최종 Depth map은 그림7(b)와 같다.



그림 7. (a) Normalization 이미지, (b) 최종 결과 Depth 이미지

4. 실험 및 구현 결과

본 장에서는 구현 결과에 대하여 설명한다. CUDA를 이용하여 제안한 알고리즘을 구현하였으며, 시스템에서 촬영한 1280x800 해상도의 실제 영상을 샘플 영상 데이터로 사용하였으며 실험한 시스템의 환경은 표 1과 같다.

표1. 실험환경

항목	세부사항
CPU	Intel Core2 Quad CPU 2.67Ghz
MEMORY	3.25GB
GPU	Nvidia GeForce GTX480
OS	Window 7

최종적으로 출력 된 영상은 그림 3과 같다. 표2는 그림 3을 가지고 CPU환경과 GPU 환경에서 Depth map 획득에 소요되는 시간을 각 단계 별로 비교한 결과 이다.

표2. 단계별 수행 시간 비교

단계	CPU[ms]	GPU[ms]
Texture mapping		
ROI Segmentation	5585	15.43
Pattern Decoding	3369	611.70
Depth Estimation	7129	238.43
Total Time	16083	879.08

3장의 영상을 한 장으로 만드는 Textured Time 및 ROI영역 검출, Pattern Segmentation에 걸리는 시간은 약 15ms로서 CPU에서 연산한 결과 대비 약 370배 정도의 속도 향상이 있었다. Pattern decoding 을 GPU를 이용하였을 경우, 약 611ms 의 시간이 소요되었으며, CPU에 비해 약 5.5 배의 속도향상이 있었으며, Depth Estimation의 경우 약 238ms의 시간이 소요되었으며, CPU에 비해 약 30배정도 향상이 있었다. 총 수행된 시간은 CPU환경에서는 16083ms, GPU환경에서는 약879ms이 소요되었다. 약 18배정도 CPU에 비해 GPU환경에서 성능 향상이 있었다.

5. 결론

본 논문에서는 두 대의 카메라와 한 대의 프로젝터를 통해 패턴 영상을 대상 물체에 조사하고, 이를 촬영한 영상을 통해 Depth Map을 고속으로 계산하기 위한 GPU 프로그래밍에 기반한 병렬처리 기법을 제안하고 구현하였다.

제안한 기법은 depth map정보 획득을 위하여, 구조광의 패턴정보를 Dynamic decoding방법을 이용하고, 각 단계 별로 GPU로 병렬처리 연산을 수행하게 함으로써, CPU로 수행했을 때보다, 연산시간을 줄일 수 있는 기법이다. GPU를 이용하여 기존의 CPU환경에 비해서 약 18배정도의 속도 향상이 있었으며, 초당 1frame 이상의 속도로 depth 정보 획득이 가능하였다. Dynamic Pattern Decoding에서 현재 알고리즘에 고려되지 않은 병렬적인 구조를 고려하여 최적화 하는 연구가 필요할 것 이다.

감사의 글

본 연구는 지식경제부와 ETRI의 IT R&D 프로그램(11Z11130, 3DTV 방송을 위한 Depth Map 획득 및 적용기술 개발)의 지원으로 진행되었음.

참 고 문 헌

[1] K. Sato and S. Inokuchi, "Three-dimensional surface measurement by space encoding range imaging," Journal of Robotic System, 2:27 - 39, 1985

[2] E. Horn and N. Kiryati, "Toward optimal structured light patterns," Image and Vision Computing, 17, 1999

[3] L. Zhang, B. Curless, and S. M. Seitz, "Rapid Shape Acquisition Using Color Structured Light and Multi-pass Dynamic Programming," 1st international symposium on 3D data processing, visualization, and transmission, Padova, Italy, June 19-21, 2002