

OpenCL을 이용한 모바일 GPU에서의 초해상도 영상 생성 가속

이충현⁰, 신일섭, 강승현, 박인규
인하대학교 정보통신공학부

{pillarofcode@gmail.com, shinilseob@naver.com, ha00ha2@gmail.com, pik@inha.ac.kr}

요약

본 논문에서는 OpenCL을 이용한 모바일 GPU에서의 영상처리 프레임워크를 제안한다. 이 프레임워크를 이용한 병렬 컴퓨팅을 통해 초해상도(superresolution) 영상 생성 기법을 가속하였다. 이 프레임워크는 OpenCL을 지원하는 GPU를 장착한 스마트폰과 같은 최신 모바일 기기에 적용할 수 있고, 다른 영상처리 알고리즘을 간단히 추가할 수 있다. 본 논문에서는 초해상도 영상 생성 기법을 CPU에서 구현한 방법과 속도를 비교하였으며, 그 결과 CPU에 비해 16배 이상의 속도 개선을 보였다.

1. 서론

스마트폰의 보급과 디스플레이의 화질 향상, 모바일 네트워크의 속도 향상 등으로 인해 고화질 동영상에 대한 사용자의 요구가 급증하고 있다. 또한, 네트워크의 대역폭과 사용량 제한으로 인해 적은 데이터를 이용해 고화질 동영상을 감상할 수 있는 방법이 요구되고 있다.

최신 모바일 GPU의 발전에 따라 많은 어플리케이션에서 GPU를 사용한 병렬화 방법이 진행되고 있다. 또한 이전의 OpenGL만 지원하던 모바일 GPU들이 OpenCL을 지원하게 되어 모바일 병렬화의 새로운 패러다임을 열고 있다. 본 논문에서는 모바일 GPU에서 OpenCL을 이용한 영상처리 병렬화 프레임워크를 제안하고, 이 기법을 통해 기존의 초해상도 영상 생성 알고리즘을 병렬화하여 수행 속도를 가속하였다. 제안하는 병렬화 기법은 상용 스마트폰에서 구현되었으며, 기존의 초해상도 영상 생성 알고리즘을 동영상에 적용할 수 있을 만한 속도로 수행할 수 있다.

2. 모바일 GPU에서의 OpenCL 기반 영상처리

OpenCL을 이용하여 모바일 GPU에서 병렬 컴퓨팅을 수행하기 위해서는 여러 준비 과정이 필요하다. OpenCL을 이용해 구동할 알고리즘이 C 코드로 제공되어야 하고 Context과 작업 큐 등이 필요하며 각각 초기화되어 있어야 한다[1]. 따라서, 이러한 공통 작업을 진행하는 프로그램을 구현하여야 해당 프로그램 위에서 여러 알고리즘을 실험할 수 있다. 본 논문에서는 이를 모바일 OpenCL 영상처리 프레임워크라 명명하였다.

2.1 구조

이 프레임워크는 스마트폰에서 OpenCL을 GPU에서 구동하여 동영상을 실시간으로 확대하기 위한 환경을 제공한다. 이 프레임워크의 구조는 그림 1과 같다.

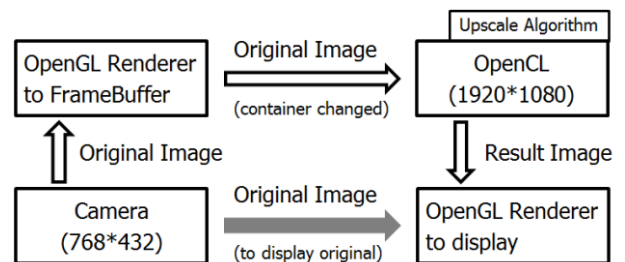


그림 1. 모바일 GPU에서의 OpenCL 기반 영상처리 프레임워크의 구조

Android 운영체제는 카메라의 preview 영상을 OpenGL의 외부 텍스처로 제공하는 기능을 포함하므로, 해당 기능을 이용해서 입력 영상을 얻을 수 있다. 이 영상을 OpenCL을 통해 처리하기 위해서는 이 외부 텍스처를 일반적인 2D 텍스처로 변환하고, 이 변환된 텍스처를 다시 Image2d_t 형태로 변환해야 한다. 이 영상을 OpenCL로 처리한 후, 다시 OpenGL의 텍스처로 변환시키고 이를 출력하는 것을 반복하는 것으로 동영상을 출력할 수 있다.

3. 초해상도 영상 생성 알고리즘

본 논문에서 구현한 초해상도 알고리즘은 Bilinear, Bicubic, Sinc, Lanczos3의 네 가지 기초 보간 알고리즘이다[2]. 이 알고리즘들은 하나의

화소값을 얻기 위하여 해당 위치 주변 화소값을 연산에 이용한다. 이는 각각의 화소를 기준으로 알고리즘을 정의할 수 있다는 것을 의미하고, 이 알고리즘을 모든 결과 화소에 대하여 독립적으로 병렬 수행함으로써 초해상도 영상을 얻을 수 있다. 따라서 알고리즘을 결과 화소에 대해 정의하는 것이 알고리즘의 병렬화의 핵심이 된다.

수행 속도의 비교를 위해 동일한 초해상도 영상 생성 알고리즘을 스마트폰의 CPU에서 구현하였다. 이때 OpenCL 대신 다중 스레드의 CPU 코드를 이용한다는 점을 제외하고는 GPU에서의 OpenCL 기반 영상처리 프레임워크와 일치하는 구조를 가진다. 본 논문에서는 16개의 스레드를 사용하였다. 이는 현재 실험 기기에서 최적의 성능을 보이는 개수로서 실험적으로 찾아내었다.

또한 CPU에서의 구현 역시 최적화를 하기 위하여, 연산 수행속도를 향상시키기 위하여 모든 부동소수점 연산을 고정 소수점 연산으로 변경하였다.

4. 실험 결과

본 논문에서는 앞서 초해상도 영상 생성 알고리즘의 구현을 위하여 Android 4.2기반의 LG G2 스마트폰을 사용하여 실험을 진행하였다. 이 기기에는 1920×1080의 해상도의 디스플레이가 내장되어 있으며, 응용프로세서는 Qualcomm의 Snapdragon 800이다[3]. 이 AP는 2.26GHz의 쿼드코어 Krait 400 CPU와 Adreno 330 GPU를 내장하고 있다[4].

상기의 알고리즘을 구동하기 위하여 GPU에서 OpenCL 기반의 병렬처리 구현과 CPU에서의 고정 소수점 구현에 대한 네 가지 초해상도 알고리즘의 수행 속도를 비교하였다. 카메라의 preview 기능으로 실시간으로 취득한 원본 영상은 768×432의 해상도를 가지고, 결과 영상은 1920×1080의 해상도로 확대된다 표 1에 수행 속도 비교를 제시하였다.

표 1. 모바일 GPU와 CPU 간 초해상도 알고리즘의 수행 속도 비교 (milliseconds).

알고리즘	GPU	CPU
Bilinear	12.14	230.31
Bicubic	22.13	357.71
Sinc	30.87	497.35
Lanczos	37.85	755.29

표 1에서 보는 바와 같이, OpenCL의 적용으로 CPU만을 사용한 경우에 비해 평균 17.8배의 성능 향상을 볼 수 있었다. 이 결과는 최종 영상을 얻기 위한 시간만을 측정한 것으로, 메모리 복사, 화면 출력 등의 시간은 포함하지 않는다. 그림 2는 생성된 결과 영상이다.

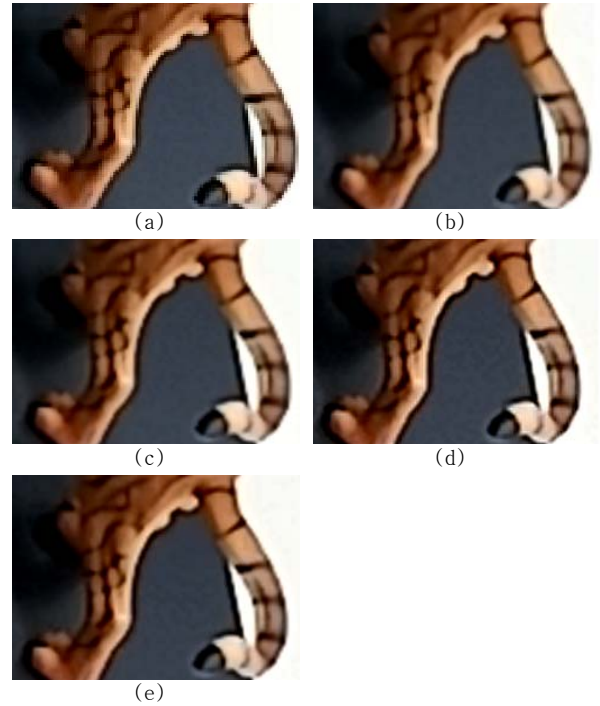


그림 2. 실험 결과. (a) 원본 영상 (단순 확대). (b) Bilinear 보간. (c) Bicubic 보간. (d) Sinc 보간 (e) Lanczos 보간.

5. 결론

본 논문에서는 모바일 GPU에서 OpenCL을 이용한 영상처리 병렬화 프레임워크를 제안하였다. 또한 초해상도 영상 생성 알고리즘의 병렬화를 통해 최대 수십배의 수행 속도 향상을 달성하였다. 이 구현 사례를 통해 모바일 GPU에서 OpenCL을 이용한 고속 영상 처리의 가능성을 확인할 수 있다.

감사의 글

본 논문은 산업통상자원부 산업융합원천기술개발사업으로 지원된 연구결과입니다. [10041664, 멀티 Shader GPU 통합형 멀티 코어 퓨전 프로세서 원천 기술 개발]

참고문헌

[1] A. Munshi and others, OpenCL 프로그래밍 가이드, (이영민, 율김), BJ퍼블릭, 서울, 2012.
 [2] P. Getreuer, "Linear methods for image interpolation," Image Processing On Line, vol. 2011. http://dx.doi.org/10.5201/ipol.2011.g_lmii
 [3] LG G2 스마트폰 (LG-F320L), <http://www.lgmobile.co.kr/mobile-phone/F320/LG-F320L/>
 [4] Qualcomm Inc., Snapdragon 800시리즈 프로세서, <http://www.qualcomm.co.kr/snapdragon/processors/800>