

웹 표준 기반 고성능 이미지 프로세싱 기술 연구

남현우¹, 이명호², 박능수³

¹건국대학교 컴퓨터공학과 박사

²명지대학교 컴퓨터공학과 교수

³건국대학교 컴퓨터공학과 교수

namhw@konkuk.ac.kr, myunghol@mju.ac.kr, neungsoo@konkuk.ac.kr

Research on high-performance image processing technology based on web standards

Hyunwoo Nam¹, Myungho Lee², Neungsoo Park³

^{1,3}Dept. of Computer Science and Engineering, Konkuk University

²Dept. of Computer Science and Engineering, Myongji University

요 약

웹 브라우저에서 고성능 이미지 프로세싱을 하기 위해 비표준 플러그인 기술들을 활용하거나 서버 기반 처리 방식을 적용하였다. 하지만 최신 웹 브라우저에서는 비표준 기술 지원을 중단하고 있으며, 서버 방식의 경우 운영 비용이 증가한다는 문제가 발생한다. 따라서 본 논문에서는 고성능 프로세싱을 위한 최신 웹 표준 기술들에 대하여 성능 검증 및 적용 가능성을 검토하였다. 최종 실험 결과 웹 표준 기술만을 사용하여 고성능 웹 이미지 프로세싱을 위한 기능 구현이 가능함을 검증하였다.

1. 서론

과거의 웹 브라우저는 웹 서버와의 인터페이스 및 HTML 문서 렌더링을 위한 뷰어 정도의 역할만을 담당하였다. 하지만 웹 3.0 시대가 도래하면서 AR/VR, 인공지능 및 이미지/비디오 프로세싱과 같은 서비스들이 웹 기반 형태로 개발되고 있다.[1] 이를 위해 과거에는 다양한 플러그인 기술들을 활용하거나 서버 기반으로 서비스를 제공하였다. 하지만 최근에는 여러 웹 표준 기술들을 활용하여 서비스 개발이 가능해졌으며, 특히 고성능 이미지 프로세싱을 위해 적용 가능한 최신 표준 기술들의 성능 분석 및 적용 가능성을 검증하고자 한다.

2. 웹 고속화 관련 기술 분석

웹 브라우저에서 고성능 이미지 프로세싱을 위하여 다양한 기술들과 표준들이 발전해오고 있다. 기본적으로 웹 브라우저와 인터프리터 방식의 Javascript 언어는 속도가 느려 고성능 처리나 다양한 기능 구현이 어려웠며, 이와 같은 한계점들을 해결하고자 네이티브 코드를 사용할 수 있는 Active-X, Flash, Java Applet과 같은 다양한 플러그인 기술들이 사용되었다. 하지만 현재 대부분의 플러그인 기술들은 보안 문제와 기술 과편화로 인하여 지원이 중단되고 있다.

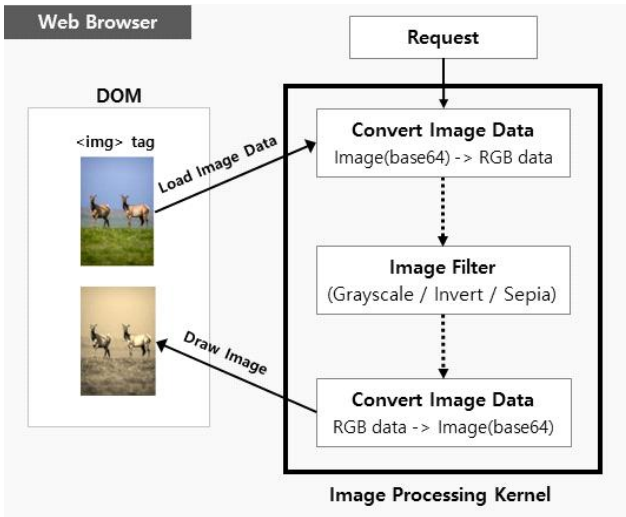
웹 플러그인 기술 외에도 웹 이중 병렬 컴퓨팅을 위한 OpenCL 기반의 WebCL 표준 초안이 발표되었다. 하지만 WebCL도 현재 각 기업들 간의 상이한 입장 차이로 인해 현재는 모든 범용 웹 브라우저에서 지원이 중단된 상태이다.

최근 웹 표준 기반의 고속화 기술로서 asm.js를 기반으로 한 고성능 목적의 웹 브라우저 실행 바이너리 포맷인 WASM(Web Assembly) 웹 표준 기술을 2017년 발표하며 최근까지 발전을 지속해오고 있다.[2][3] 또한 웹 환경에서 GPU를 사용하기 위해 WebGL 표준이 존재하나 이는 3D 렌더링을 위한 목적이 강하며, GPGPU 목적으로 최근 크롬 웹 브라우저에 공식 탑재된 WebGPU의 확장 가능성이 기대되고 있다.[4][5]

또한 WASM 및 WebGPU를 이용한 웹 기반 이미지 프로세싱 라이브러리인 opencv.js, Photon, wimage 등이 현재 오픈소스로 개발되어 활용이 가능한 상황이다.[6][7][8]

3. 웹 기반 고성능 이미지 프로세싱을 위한 방안

웹 브라우저에서 실행하기 위한 목적으로 고성능 이미지 프로세싱 알고리즘을 신규로 개발하는 경우 목표하는 실행 환경에 따라 WASM 코드와 WebGPU 코드로 개발하는 방법을 적용할 수 있다.



(그림 1) 웹 기반 이미지 프로세싱 처리 단계

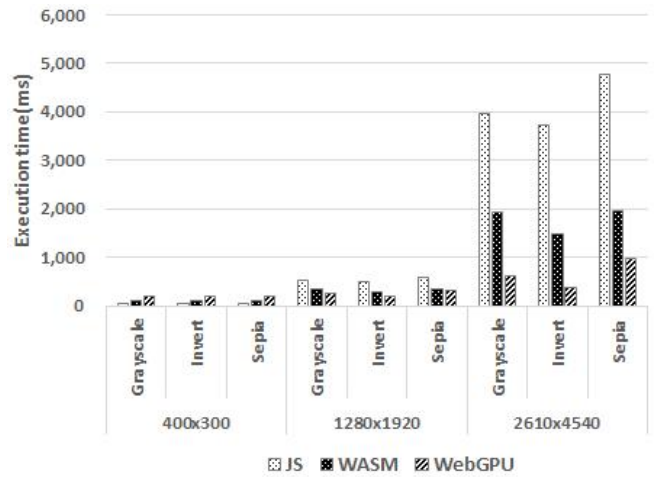
실행 환경이 최신 크롬 웹 브라우저를 사용하고 적정 수준의 GPU가 탑재된 컴퓨팅 환경인 경우 핵심 커널 코드를 WebGPU 셰이더 코드로 작성하는 것이 성능상 가장 큰 이점을 가진다.

하지만 그렇지 않은 컴퓨팅 환경의 경우 CPU 실행에 최적화된 WASM 코드로 개발이 필요하다. 이때 WASM 코드를 생성하기 위한 프로그래밍 언어로서 C, C++, Rust 등의 언어가 사용 가능하다. 또한 WASM 파일의 경우 SIMD 명령어를 지원하고 있어 벡터화된 이미지 데이터 처리에 효과적이며, 이와 같은 기능들을 충분히 활용하여 고성능 이미지 프로세싱 코드를 작성해야 할 것이다.

4. 성능 평가

제안 방안을 검증하기 위해 이미지 프로세싱 알고리즘인 GrayScale, Invert, Sepia 연산을 JS, WASM, WebGPU 코드로 각각 개발하여 실행 시간을 측정 하였으며 세부 처리 단계는 (그림1)과 같다.

실험 방법은 이미지의 크기를 달리해가며 이미지 필터 알고리즘 별로 실행 시간을 측정하였으며 최종 결과는 (그림2)와 같다. 먼저 작은 이미지 크기인 400x300 해상도에서는 JS 코드가 가장 속도가 빨랐으며 오히려 WebGPU 코드가 가장 느렸다. 이는 WebGPU 코드의 실행 초기 코드 빌드 시간 그리고 데이터 송/수신 작업으로 인한 오버헤드 때문이다. 따라서 작은 크기에서는 GPU 코드가 비효율적이지만 이미지 크기가 1280x1920 및 2610x4540 정도로 커지는 경우 WASM 및 WebGPU 코드의 실행 속도가 월등하게 향상되는 것을 확인할 수 있었다.



(그림 2) 이미지 프로세싱 실행 시간 측정 결과

5. 결론

본 논문의 연구 결과를 통해 최신 웹 표준 기술을 기반으로 CPU, GPU를 모두 활용하여 웹 브라우저에서 직접 고성능 이미지 프로세싱 기능 구현이 가능함을 확인 하였다. 이를 통해 서버 기반의 프로세싱 부하를 줄여주며 네트워크 트래픽 및 실행 시간의 단축 그리고 실시간 처리가 가능해질 것이다.

감사의 글

이 성과는 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (RS-2023-00321688)

참고문헌

[1] 박진태 and 문일영, "웹 융합 서비스 구현 기술의 발전 동향," 한국통신학회지(정보와통신), Vol.38, No.4, pp.3-9, 2021.

[2] 김현국, "웹 어셈블리 기술 동향 연구," 한국통신학회지(정보와통신), Vol.40, No.2, pp.3-9, 2023.

[3] 신창현, 여지환, and 문수목, "웹어셈블리의 수행 과정 및 성능 분석," in 한국정보과학회 학술발표논문집, 개척지, pp.1546-1548, 2018.

[4] 강진규, "WebGPU 기반의 3D 렌더링 - 디지털 트윈 및 증강 현실 서비스 적용 중심으로 -," 방송과 미디어, Vol.28, No.3, pp.67-74, 2023.

[6] 남현우, 박능수, "WebGPU를 이용한 AES 알고리즘 가속화," 전기학회논문지, Vol.71, No.7, pp.1008-1014, 2022.

[6] opencv.js, <https://github.com/TechStark/opencv-js>

[7] Photon, <https://silvia-odwyer.github.io/photon/>

[8] wgpuimage, <https://github.com/neka-nat/wgpuimage>