저전력 온디바이스 비전 SW 프레임워크 기술 동향

Trends in Low-Power On-Device Vision SW Framework Technology

이문수 (M.S. Lee, mslee@etri.re.kr) 고성능디바이스SW연구실 책임연구원
배수영 (S.Y. Bae, manim75@etri.re.kr) 고성능디바이스SW연구실 책임연구원
김정시 (J.S. Kim, sikim00@etri.re.kr) 고성능디바이스SW연구실 책임연구원
석종수 (J.S. Seok, jsseok@etri.re.kr) 고성능디바이스SW연구실 연구원

ABSTRACT

Many computer vision algorithms are computationally expensive and require a lot of computing resources. Recently, owing to machine learning technology and high-performance embedded systems, vision processing applications, such as object detection, face recognition, and visual inspection, are widely used. However, on-devices need to use their resources to handle powerful vision works with low power consumption in heterogeneous environments. Consequently, global manufacturers are trying to lock many developers into their ecosystem, providing integrated low-power chips and dedicated vision libraries. Khronos Group—an international standard organization—has released the OpenVX standard for high-performance/low-power vision processing in heterogeneous on-device systems. This paper describes vision libraries for the embedded systems and presents the OpenVX standard along with related trends for on-device vision system.

KEYWORDS 온디바이스, 고성능, 저전력, 비전처리

Ⅰ. 서론

과거의 디지털 영상 처리는 방송, 의료, 로봇 등 특수 목적을 수행하기 위해서 전문가를 대상으로 한 고비용, 고사양 시스템 위주로 개발하였다. 이러한 시스템은 제한된 공간에서 설치 운용되어, 성능과 전력에 대한 문제보다 사용 목적에 따른 결과의 정확도에 더 많은 중요도를 두었다. 하지만 오늘날 급속한 기술 발전으로 인해 고사양 카메라의 보편화와 영상 유튜버와 같은 실시간 스트리밍 서비스가 활성화되면서 영상 합성, 특수 효과 등의 복잡한 이미지 처리에 대한 수요가 스마트폰과 같은 개인 사용자를 위한 스마트기기상에서도 점점 높아지고 있다.

최근 스마트기기에서 증가되고 있는 다양한 객체 인식에 대한 영상 처리는 연산 중심(Compu-

* DOI: https://doi.org/10.22648/ETRI.2021.J.360206
* 이 논문은 2021년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임[No. 2017-0-00142, 스마트기기를 위한 온디바이스 지능형 정보처리 가속화 SW플랫폼 기술 개발].

tation intensive)의 작업이 필요한 분야로서 CPU, 메모리, 전력 등 많은 컴퓨팅 자원을 요구한다. 특히, 배터리를 장착하는 임베디드 디바이스는 전력 문제가 매우 중요한 이슈가 된다. 일반적으로 고성능과 저전력은 서로 상관관계(Trade-off)를 가지므로, 전력 소모를 최대한 낮추면서 얼마나 높은 성능을 낼 수 있을 것인지, 높은 성능을 내면서 전력 소모를 얼마나 줄일 수 있는지, 전력과 성능 비에 관한 연구는 항상 중요한 화두가 되고 있다.

이러한 문제를 해결하기 위해 특성화된 하드웨어 모듈들이 개발되고 있으며, 글로벌 제조업체들은 스마트기기 내에 다양한 저전력 전용 하드웨어 모듈을 탑재하여 이기종 시스템(Heterogenous system) 환경을 구축해 나가고 있다. 시스템 내에 서로 이질적인 하드웨어 모듈이 설치되어 함께 운용되는 이기종 시스템 환경은 한 시스템 내에 서로 이질적인 하드웨어 모듈이 설치되어 함께 운용되는 것으로, 실행되는 클럭 주파수, 실행에 필요한 환경, 사전 조건 등 상이한 환경을 가진다. 따라서 이러한 시스템 응용을 개발하기 위해서는 각 프로세서 구조에 특화된 코드를 적용해야 하고, 응용 실행 시에도 어느 시점에 어떤 가속 프로세서를 사용할지 잘 결정해야 한다.

본 고에서는 다양한 하드웨어 모듈을 기반으로 하는 이기종 온디바이스상에서 저전력 소프트웨어 연구 개발에 요구되는 고려 사항을 살펴본다. 그리고 저사양 디바이스의 영상처리 응용 개발에 사용하고 있는 공개/비공개 비전 라이브러리에 대해 알아본다. 마지막으로 저전력 온디바이스 비전 SW 프레임워크인 OpenVX 국제 표준과 이와 관련된 제품 동향을 알아보고 결론을 맺는다.

Ⅱ. 이기종 온디바이스 시스템

일반적으로 IoT 디바이스나 차량의 센서 스트림 데이터는 CPU 사용이 유리한 반면, 대용량 이미지 데이터나 배치 데이터는 병렬 처리에 특화된 GPU가 더 효율적이다. 또한, 모바일기기의 동영상 전용 인코더/디코더는 전력 소모를 줄이기 위해 전력 대비 연산 효율이 높은 DSP나 전용 FPGA를 사용하게 된다. 따라서 칩 제조사들은 처리 대상 데이터에 따라 효과적인 성능을 확보하기 위해 필요한 전용 HW 모듈을 자체 생산 칩에 통합한 원칩 솔루션 형태로 생산하고 있다. 대표적인 구성 사례로 퀄컴이 Snapdragon 제품에 CPU, GPU, DSP, 모뎀 등 여러 종류의 이기종 프로세서를 탑재시키는 것을 들 수 있다.

온디바이스 시스템은 외부 클라우드나 서버의 도움을 받지 않고 자체 기능만으로 주어진 미션을 수행하는 기기로서, 최근에 주목받는 AI 연구와 병행하여 “온디바이스 AI”라는 용어로 많이 사용된다. 특히, 이기종 온디바이스는 상이한 big. LITTLE 아키텍처를 갖는 이기종 멀티코어를 포함하여, GPU, DSP, NPU 등의 서로 다른 가속 프로세서를 갖춘 임베디드 시스템을 의미한다.

온디바이스 내에 다양한 프로세서들이 장착되면 응용 개발에 있어서는 각각의 장점을 취할 수 있지만, 반대로 시스템 아키텍처는 복잡해지고, 프로세서 간의 동일 메모리 접근으로 인한 충돌 가능성이 높아진다. 따라서 응용 구현 시, 개발자는 디바이스가 가진 리소스로 최대한 성능을 발휘할 수 있도록 응용 프로그램 코드를 여러 개로 분할하고, 어느 부분을 어떤 프로세서에게 할당할지를 결정해야 한다. 또한, 각 프로세스의 유휴(Idle) 시간을 최소화하고, 이들의 실행을 관리하기 위한 스케

줄러를 개발해야 한다[1].

이기종 프로세서는 메모리 사용에 있어서도 각각의 별도 메모리 공간이 할당된다. 이로 인해 많은 메모리를 사용하게 되므로 메모리 재사용이나 내부 모듈 간 데이터 전달을 최소화하는 효율적인 메모리 관리 기법도 필요하게 된다[2,3].

이와 같이 이기종 온디바이스 환경에서 저전력 비전 처리를 위해서는, 하드웨어 측면에서 기기에 장착된 병렬 가속 프로세서를 최대한 활용하고, 소프트웨어 측면에서 저전력 또는 고성능 목적에 따라 적절한 모듈 분리, 프로세서 선정과 실행 경로 최적화, 메모리 관리 기술 등이 요구된다.

III. 저전력 온디바이스 비전 SW 프레임워크

일반적으로 비전처리는 단순한 이미지 처리를 위한 고전적 영상처리와 딥러닝을 이용한 머신러닝 영상처리로 크게 구분된다. 이 장에서는 고전적 영상처리 관점에서 현재 많이 사용되고 있는 비전 라이브러리들에 대해 알아본다.

1. OpenCV

OpenCV는 실시간 컴퓨터 비전처리를 위해 인텔이 초기 개발을 주도적으로 진행하였으나, 현재는 사실상(De-facto) 표준으로 정착해 가장 많이 사용되는 비전 라이브러리가 되었다. 오픈 소스로 공개되어 커뮤니티를 중심으로 지속적으로 확장 개발되고 있으며, 버전 4.4 이전 버전은 BSD 라이선스 적용을 받는다. 최신 버전 4.5부터는 Apache2 라이선스로 변경되었으나, 상업용으로 사용하기에는 크게 문제가 없을 것으로 판단된다. 크로스 플랫폼 지원으로 데스크톱뿐만 아니라 다양한 임베디드 기기와 안드로이드 기기에서도 사용할 수 있다.

현재, OpenCV는 500개 이상의 함수들을 지원한다. 이들 함수는 크게 이미지 처리와 비전 알고리즘이 포함된 CV 모듈, 공통 데이터 포맷과 수학 연산 수행을 위한 CXCORE 모듈, 영상 통계 분석과 머신러닝을 위한 MLL(Machine Learning Library), 영상 입 · 출력을 위한 HighGUI로 크게 4가지 모듈로 나뉜다. 각 모듈은 데스크톱 환경의 고성능 메모리 아키텍처를 기반으로 개발되어 있어서, 온디바이스 시스템에 바로 적용 시, 다른 라이브러리에 비해 많은 리소스와 전력 소모가 발생된다. 온디바이스용 비전 응용 개발은 함수 레벨에서의 적절한 저전력 프로세서 적용이 필수적이지만, OpenCV에는 이러한 미세 제어가 쉽지 않아 주로 데스크톱용 응용이나 임베디드 시스템을 위한 프로토타입 개발용으로 많이 활용되고 있다.

2. ARM ACL

ARM ACL(ARM Compute Library)은 ARM사가 보유하고 있는 CPU/GPU 아키텍처 설계 능력을 바탕으로 SIMD 기술을 통해 ARM 하드웨어에 최적화된 비전 및 머신러닝 처리 라이브러리를 제공하고 있다. 리눅스와 안드로이드 등 멀티 플랫폼을 지원하며, CPU의 NEON과 GPU의 OpenCL/GLES를 기반으로 성능 가속화 기능을 제공한다[4,5]. 또한, 사용자가 비교적 쉽게 성능 향상할 수 있도록 알고리즘의 기본 구현인 코어 라이브러리와 장치별 가속이 적용된 런타임 라이브러리로 구분하여 오픈소스로 공개하고 있다.

ACL은 저수준 이미지 처리 함수뿐만 아니라 머신러닝에 많이 사용되는 다양한 네트워크 레이어도 지원한다. 따라서 프레임워크 내에서 영상 전 · 후

처리와 머신러닝 처리까지 모두 포함한 통합 개발 환경을 갖추고 있으며, 이러한 모든 구성들을 통해 고성능과 저전력 처리에서 커다란 이점을 가지게 된다[6]. 또한, ACL은 세분화된 기능 구현을 통해 빌드 옵션에 따라 마이크로 컨트롤러 유닛(MCU: Micro-Controller Unit)과 같은 낮은 사양에서도 충분히 동작 가능한 유연성을 가진다.

3. Qualcomm FastCV

퀄컴사는 자사 칩셋을 탑재한 모바일기기에서 사용 빈도가 높은 기능 위주로 최적화된 FastCV 비전 라이브러리를 개발하였다. 이 라이브러리는 그림 1과 같이 범용 ARM 프로세서를 대상으로 하는 "FastCV for ARM"과 퀄컴 SoC를 대상으로 하는 "FastCV for Snapdragon", 두 개의 제품군으로 구성된다. 이들은 통합 바이너리(Unified Binary) 형태로 제공되며, 프로세스 독립(Processor-agnostic)된 가속 API를 제공하여 개발자가 하위 하드웨어에 대한 전문적 지식이 없어도 응용 개발할 수 있다.

특히, "FastCV for Snapdragon" 제품은 CPU와 GPU뿐만 아니라 DSP 기능을 포함한 이기종 프로세스를 지원한다. 특히, 퀄컴의 Hexagon DSP는 CPU보다 3배 느리지만 전력 소모가 적어 저전력 장치에 적합하다. 주로 카메라로부터 영상과 음성을 취득하고, 데이터 변환 및 필터링과 같은 비전 처리 전 단계를 담당하는데, 이를 통해 CPU 부하를 약 50%까지 줄일 수 있다[7].

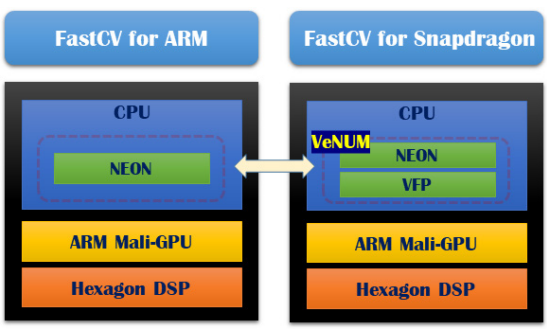

그림 1 퀄컴 FastCV 제품군[7]

퀄컴은 또한 "VeNum"이라고 하는 멀티미디어 처리 엔진을 통해 기존 ARM NEON 명령어에 VFP(Vector Floating-Point) 명령어를 확장하여 고성능 실수 연산을 수행할 수 있도록 하였다.

Ⅳ. 고성능/저전력 OpenVX 프레임워크

OpenVX는 이기종 플랫폼에서 컴퓨터 비전 처리를 수행할 수 있도록 정의한 가속 표준으로, 비영리 표준 단체인 크로노스 그룹에서 별도의 로열티가 없도록 제정한 개방형 표준이다. 크로노스 그룹은 인텔, 구글, 엔비디아, 삼성, 퀄컴, AMD, 애플 등 전 세계 많은 소프트웨어 및 하드웨어 기업 컨소시엄으로 구성되어, 그래픽스와 병렬 컴퓨팅, 비전, 신경망 가속 등 여러 분야에 사용되는 표준들을 진행하고 있다. 현재 OpenVX를 포함한 OpenCL, OpenGL, Vulkan, NNEF, OpenXR 등 다양한 표준 제정 활동이 활발히 진행되고 있다[8].

1. 개요

임베디드 비전 응용 개발은 주어진 하드웨어와 소프트웨어상에서 성능은 최대로, 에너지는 최소한 소비하는 것을 목표로 한다. 그러나 일반적인 임베디드 비전 응용 개발의 가속화 작업은 알고리즘별 성능 향상 위주의 비교적 접근하기 쉬운 방식으로 이루어지고, 전력 소비와 대역폭 부하, 저지연, 프로세서 간의 통신 오버 헤드 등 전체 시스템 관점에서 비효율성과 병목 현상은 해결이 어려운 문제로 남게 된다.

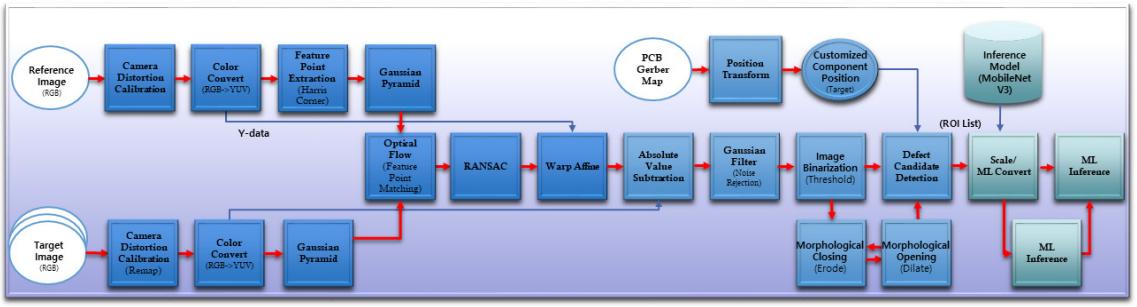

그림 2 OpenVX 응용 그래프(PCB Visual Inspection)

OpenVX은 이러한 문제를 해결하기 위해서, 임베디드 시스템 개발자가 시스템 수준의 최적화와 하드웨어, 소프트웨어 가속기를 통한 커널 수준의 최적화를 동시에 고려할 수 있는 표준 프레임워크 API를 제시한다[9].

OpenVX는 모바일이나 임베디드 시스템과 같이 저전력을 요구하는 도메인에서 소프트웨어 재사용 및 타 플랫폼 이식이 용이하다. OpenVX 표준은 ISP, DSP, GPU, 멀티코어 CPU를 포함한 이기종 플랫폼을 지원하고 엄격하게 정의된 적합성 테스트를 통해 다양한 하드웨어에서 안정적인 실행을 보장한다. 2014년 첫 1.0버전을 발표한 이래 꾸준한 업데이트하고 있으며, 2019년 8월 1.3버전을 발표하고, 머신러닝 및 다양한 분야와 연계할 수 있도록 지속적으로 확장하고 있다[10].

2. 주요구성

OpenVX는 비전 응용에 대한 이식성과 최적화, 전력 효율적인 응용을 개발하기 위해서 객체 기반으로 표준을 정의한다. 이 방식은 OpenGL과 유사하게 소프트웨어와 하드웨어, 이를 사용하는 응용 개발 계층을 구조적으로 분리함으로써 각 계층별로 독립적으로 구현, 발전될 수 있도록 추상화를 제공된다. 즉 개발자가 직접 사용할 수 있는 가속 레이어를 API로 정의하고, 하드웨어 및 소프트웨어 벤더는 자신의 칩에 특화된 방법으로 각 레이어에 속한 가속 라이브러리를 구현하도록 한다.

그림 2와 같이 OpenVX 응용은 하나의 그래프로 표현된다. 영상 처리 기능을 수행하는 커널과 이들 커널 입 · 출력에 필요한 메모리들을 추상화한 노드 객체가 정의되고, 이를 서로 연결하여 하나의 그래프 형태로 표현한다. OpenVX 그래프는 단방향 그래프로서 상호 참조하는 데이터 간 의존성을 고려하여 실행의 순서를 정하게 된다. 그리고 그래프가 실행되기 전에 검증 과정을 거쳐, 그래프 내의 각 노드별 데이터 연결 정확성 및 일관성을 확인하고, 실행 시 발생될 수 있는 비효율적인 병목구간을 확인하여 전체 실행 경로를 최적화한다.

OpenVX 표준은 크게 프레임워크 객체와 공통 데이터 객체로 정의로 구성된다(그림 3). 프레임워크 객체는 OpenVX 응용 그래프를 구성하고 런타임 실행을 위해 필요한 객체들로서, 하나의 응용을 관리하는 컨텍스트(Context), 응용 전체의 구조를 나타내는 그래프(Graph), 그래프를 구성하는 노드(Node), 노드 내에 포함되어 비전 처리 기능을 수행하는 커널(Kernel), 커널의 입 · 출력을 나타내는 파라미터(Parameter) 객체가 포함된다. 공통 데이터 객체는 물리적 메모리를 추상화하여 그래프의 커널 내부에서 사용되는 객체들로서, Scalar, Array,

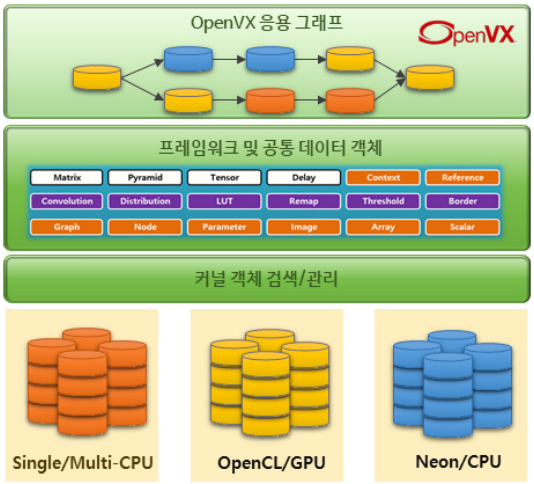

그림 3 OpenVX 표준 구성

Matrix와 같은 기본 데이터 형식부터 Image, Pyramid 등 영상 데이터 형식까지 다양한 형태가 존재한다.

노드는 응용 그래프의 최소 구성단위이고, 실제 영상 처리 기능을 수행하게 되는 커널 객체는 이기종 프로세서별로 CPU-only, OpenCL, NEON 등으로 하위 시스템 하드웨어 환경에 따라 여러 가속 라이브러리로 구현될 수 있다. 따라서 OpenVX 런타임 엔진은 시스템 가용 자원이나 실시간 사용량을 모니터링하여 경량 메모리, 고성능, 저전력 등 목적에 따라 노드 내 커널 객체를 변경하여 실행할 수 있다.

3. OpenVX 관련 제품 동향

가. VisionWorks

Nvidia는 글로벌 시장의 최고의 GPU 개발 회사로서, 서버에서부터 일반 PC, 임베디드, 자동차 모빌리티, IoT에 이르기까지 다양한 환경에서 사용 가능한 하드웨어와 이를 지원하기 위한 SDK를 제공하고 있다. Nvidia의 대표적인 임베디드 비전 처리를 위한 VisionWorks 툴킷은 크로노스 OpenVX 비전 처리 프레임워크를 포함하고 있다[11].

Nvidia의 GPU는 다른 기업보다 월등히 많은 병렬 처리 유닛을 포함하고 있으며, 이를 쉽게 사용할 수 있는 CUDA 병렬화 라이브러리와 API를 함께 제공한다. CUDA는 GPGPU 기술을 기반으로 병렬 알고리즘 구현이 용이하도록 설계되었고, 최근 대용량 데이터를 활용한 머신러닝 학습과 추론에 널리 사용되고 있다.

VisionWorks는 OpenVX 1.2 API를 기반으로 구현하고, 커널들은 CUDA를 통해 가속화가 적용되었다. 이러한 가속 커널들은 소모 전력이 많지만, 반면 수많은 GPU 유닛에서 병렬화 처리되어 처리 속도 관점에서는 다른 비전 처리 프레임워크에 비해 우수한 성능을 보장한다[12].

또한, 실제 산업 환경에서 자주 사용되지만 OpenVX 표준 내에 정의되지 않은 함수들을 추가 확장하여 다양한 환경에서 활용할 수 있도록 구성되어 있다. 예를 들면, 카메라의 깊이(Depth) 영상 처리와 특징점 추출(Feature extraction) 관련 함수들이 추가되어, 자율주행자동차에서 카메라를 통한 데이터 취득 및 가공에 활용될 수 있다.

나. OpenVINO

OpenVINO(Open Visual Insurance & Neural Network Optimization)는 딥러닝 모델을 보다 빠르게 추론할 수 있도록 인텔이 제공하는 툴킷이다. 개발자들이 비용 측면에서 효율적이고, 강력한 컴퓨터 비전 응용을 만들 수 있도록 돕고 있다. 엣지 시스템 상에서도 원활한 딥러닝 추론이 가능하도록, 인텔이 개발한 여러 이기종 컴퓨터 비전 가속기(CPU, GPU, Intel®, Movidius™, Neural Compute Stick 및 FPGA)들을 지원하여 하나의 인텔 기술 생태계를 만들어가고 있다.

OpenVINO 툴킷에는 기본적 비전 시스템으로 자사가 최초 개발한 OpenCV뿐만 아니라 Intel® 하드웨어(CPU, GPU)에서 실행이 최적화된 OpenVX* 라이브러리인 Intel® Distribution of OpenVX* Implementation을 포함하고 있다.

Intel® Distribution of OpenVX는 Intel® 스레딩 빌딩 블록을 포함한 다중 스레드 지원과 벡터화된 CPU 커널을 제공하고, OpenCL을 통해 GPU를 활용할 수 있도록 확장하여 실행 성능을 향상시켰다. 또한, 노드 간 입력 및 출력 데이터에 대한 자동 데이터 타일링(Automatic data tiling) 기법을 지원하여 데이터 읽기/쓰기 작업에 대한 성능 향상도 지원한다.

Intel의 OpenVX* Primitives를 통해서 OpenVX 표준 커널을 포함한 다양한 비전 처리 커널을 확장하여 제공하고, 프로세싱 파이프라인에 개발자 자신의 고유한 알고리즘을 적용하여 효율적인 성능을 낼 수 있도록 CPU와 GPU 각각에 대해 데이터 타일링과 OpenCL 커스텀 커널 파이프라인 확장 기법을 제공한다. 또한, 컴퓨팅 리소스를 최대한 활용하기 위해 태스크 및 데이터 병렬 처리를 모두 지원한다.

다. MiVisionX

AMD사는 초기에 AMDOVX 오픈 소스를 통해 코드를 공개하였고, 추후 MiVisionX로 프로젝트명을 변경하여 AMD의 ROCm 생태계에서 컴퓨터 비전과 머신러닝 응용 개발을 지원하고 있다. MiVisionX 툴킷은 주로 데스크톱 환경인 x86 CPU와 OpenCL 기반 GPU를 적용하여 최적화하였고, OpenVX의 Neural-Network Extension과 NNEF 표준을 지원하여 하나의 OpenVX 그래프에서 일반 비전 처리와 머신러닝 기능을 함께 사용할 수 있게 하였다. 또한, AMD는 OpenVX 응용 그래프 내 노드, 입출력 데이터와 그들 간의 연관성을 직관적으로 기술할 수 있도록 GDF(Graph Description Format) 규격을 별도로 제공하고 있다.

라. TIOVX

TI사는 자사의 TDA2x/3x SoC상에서 OpenVX 1.1 표준을 지원하기 위한 TIOVX 소프트웨어 스택을 제공한다. 스택에는 OpenVX 표준 커널뿐만 아니라 OpenVX 그래프 실행과 저수준의 원시(Native) 기능을 연동하기 위한 커널 래퍼, OpenVX 프레임워크를 운용하기 위한 하부 리눅스 또는 실시간 OS 플랫폼이 포함되어 있다.

TIOVX는 여러 개의 C66x DSP와 Vision AccelerationPac(EVE)를 활용할 수 있도록 개발되었고, EDMA(Enhanced-DMA)를 이용하여 대용량의 2D 이미지 메모리 데이터를 CPU L2 캐시로 바로 전송할 수 있도록 하여 2배에 가까운 성능 향상을 보여주고 있다. 또한, 파이선 언어를 통해 OpenVX 응용을 쉽게 개발할 수 있도록 PyTIOVX를 제공하고 있고, 이를 통해 응용 코드를 자동 생성하거나 시뮬레이션 기능도 제공한다.

특히, Vision AccelerationPac은 전력 소모가 적고, 저지연이 요구되는 차량 및 실시간 응용을 위해 특별히 고안된 프로그램 가능한 비전 가속기이다. 일반 모바일 CPU 메모리 대역폭인 128bits에 비해 6배 넓은 768bits 병렬 처리 인터페이스를 제공하여 8배 이상의 전력 대비 연산 능력을 나타낸다.

마 CitiusVision

ETRI는 모바일 GPU상에서 OpenCL 기반으로 고성능, 저전력 영상 처리를 지원하기 위해 OpenVX 1.2 표준을 지원하는 온디바이스 비전 처리 프레임워크인 CitiusVision을 개발하였다(그림 4). ARM 멀티코어 프로세서의 NEON 및 OpenMP 기반 커널도 함께 제공하여 하이브리드(CPU-GPU)

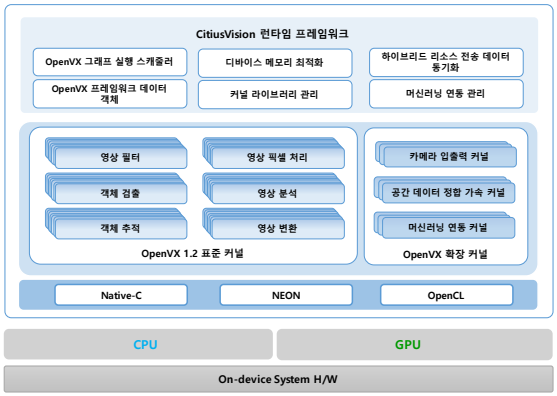

그림 4 ETRI의 CitiusVision

커널 연동 기능을 제공한다.

CitiusVision은 RGB/IR뿐만 아니라 스테레오 카메라를 통한 공간정보 처리 기능을 추가 확장하고, 일반적으로 많이 사용되는 SVM(Support Vector Machine) 및 DNN(Deep Neural Network)의 머신러닝 기능도 커널 형태로 지원하고 있다. 또한 이미지/동영상 캡처, 원격 웹 콘솔 등 입 · 출력 유틸리티를 자체 제공하여 외부 라이브러리 없이 응용 개발을 용이하게 하였다.

V. 결론

CCTV, 자율차량, 로봇 등 영상 전문가 위주로 개발되고 사용되었던 비전 처리는 딥러닝 기술 발전으로 일반인도 응용 서비스를 만들 수 있을 만큼 보편화되고 있다. 또한, 사람과 거의 유사한 성능의 인식 정확도를 보이면서 응용 개발에 있어 필수적인 기능이 되어가고 있다.

이러한 서비스 요구 상황에서 네트워크 연결 없이 즉각적으로 대응해야 하는 온디바이스는 자체적으로 빠른 응답을 내기 위해 성능을 높여야 하는 반면, 소모 전력은 줄여야 하는 상황에 놓여 있다. 따라서 고성능과 저전력을 동시에 만족하기 위해서 CPU외 GPU, DSP, FPGA, NPU 등과 같이 특정 데이터 처리에 효율적인 별도의 하드웨어를 탑재하게 되고, 글로벌 하드웨어 벤더들은 자사 칩에 별도의 저전력 코어를 추가하여 자신들의 기술 생태계에 고착(Lock-in)될 수 있도록 전용 비전 처리 라이브러리를 제공한다.

국제 크로노스 그룹에서는 개방형 OpenVX 표준을 통해 하드웨어 종속성을 줄이면서 이식성은 높일 수 있는 프레임워크를 제공하고, 이기종 온디바이스 환경에서 비전 응용 개발과 최적화가 가능하도록 지원하고 있다. 즉 하드웨어 벤더는 자사 제품을 표준 API별로 최적화된 라이브러리를 제공하고, 응용 개발자는 개발 플랫폼상에서 파이프라인 형태로 적합한 프로세서와 그에 해당하는 API를 매핑하여 응용을 개발할 수 있도록 한다.

파이프라인 기반 그래프 응용 실행 기법은 응용 개발의 편의성뿐만 아니라 시스템 수준의 최적화를 자동 수행해 줌으로써 소프트웨어 성능 및 품질 향상을 동시에 이룰 수 있게 한다. 향후 이러한 개발 방향은 이기종 플랫폼을 가진 온디바이스 응용 개발에 필수적인 요소로 자리 잡게 될 것이고, WYSIWYG 편집기를 통해 Drag-and-drop으로 코딩 없이 응용 개발이 가능한 No-code 개발 플랫폼으로 진화해 나갈 것으로 예상된다.

용어해설

임베디드 데스크톱이나 서버와 다르게 특정한 목적을 위해 구성된 매우 적은 사양의 소형 하드웨어 시스템

비전처리 이미지에 대해 획득, 변환 등의 저수준 처리부터 분할, 인식 등의 고수준 처리까지의 전반적인 데이터 처리와 가공 과정을 의미함

OpenCL 크로노스 국제 표준 그룹이 CPU, GPU, DSP 등 이종 디바이스들을 일반 연산 처리 프로그램에 활용할 수 있도록 한 개방형 범용 병렬 컴퓨팅 프레임워크

NEON ARM에서 지원하는 CPU 레지스터 기반의 SIMD 기술로써 GPU에서 동작하는 OpenCL이나 CUDA에 대응되는 병렬화 기법

약어 정리

ACL	ARM Compute Library
DSP	Digital Signal Processing
EDMA	Enhanced Direct Memory Access
EVE	Embedded Vision Engine
FPGA	Field Programmable Gate Array
GPGPU	General-Purpose computing on Graphics Processing Units
GPU	Graphics Processing Unit
IoT	Internet of Things
NPU	Neural Processing Unit
ROCm	Radeon Open Compute platforM
SIMD	Single Instruction Multiple data
WYSIWYG	What You See Is What You Get

참고문헌

[1] H. Andrade et al., "Software deployment on heterogeneous platforms: A systematic mapping study," IEEE Trans. Softw. Eng. Aug. 2019.

[2] Z. Zheng et al., "HiWayLib: A software framework for enabling high performance communications for heterogeneous pipeline computations," in Proc. ASPLOS, New York, NY, USA, Apr. 2019, pp. 153-166.

[3] S. Aldegheri et al., "Rapid prototyping of embedded vision systems: Embedding computer vision applications into low-power heterogeneous architectures," in Proc. Int. Symp. Rapid Syst. Prototyp. Turin, Italy, Oct. 2018. pp. 63-69.

[4] ML Group, "Speed up your AI designs with dedicated ARM machine learning hardware," ARM Tech Symposia, 2018.

[5] G. Jo et al., "OpenCL framework for ARM processors with NEON support," in Proc. Workshop Program. Models SIMD/Vector Process. Orlando, FL, USA, Feb. 2014. pp. 33-40.

[6] D. Sun et al., "Enabling embedded inference engine with ARM compute library: A case study," arXiv preprint, CoRR, 2017, arXiv:1704.03751

[7] Qualcomm Technologies, Inc, "Qualcomm hexagon DSP: An architecture optimized for mobile multimedia," 2013.

[8] Khronos Group, https://www.khronos.org/

[9] E. Rainey et al., "Addressing system-level optimization with OpenVX graphs," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops, Columbus, OH, USA, Jun. 2014. pp. 644-649.

[10] Khronos Group, OpenVX Specification ver. 1.3, Aug. 2019, https://www.khronos.org/registry/OpenVX/

[11] Nvidia VisionWorks, https://developer.nvidia.com/embedded/visionworks

[12] M. Qasaimeh et al., "Comparing energy efficiency of CPU, GPU and FPGA implementations for vision kernels," in Proc. Int. Conf. Embed. Softw. Syst. Las Vegas, NV, USA, June 2019.