저전력 디지털 신호처리 가속기 설계 기술

전동석
서울대학교

Ⅰ. 서 론

인류의 역사, 특히 기술 발전의 역사를 해석하기 위한 하나의 접근 방법은 인간과 주변 환경 사이의 지속적인 상호 관계 속에서 인간이 관찰한 것을 더 잘 이해하고자 하는 욕망에서 발현된 일련의 지적 탐구의 과정으로 이해하는 것이다. 비근한 예로 전자공학의 주요한 축을 담당하는 컴퓨터 또는 컴퓨팅 시스템의 역사를 들 수 있다. 최초의 기계식 컴퓨터로 일컬어지는 Charles Babbage의 Difference Engine은 복잡한 천문학적 연산을 빠르고 실수 없이 실행하기 위한 목적으로 설계되었고, 현대적인 형태를 갖춘 컴퓨터의 시초인 ENIAC 역시 포탄의 궤적을 빠른 계산을 통해 이를 미리 예측하고자 하는 인간의 욕구로 인해 탄생할 수 있었다. 같은 맥락에서 시야를 더 넓혀서 생각해 보면 인류의 기술 발전은 시각을 비롯한 다양한 형태로 얻어진 정보를 가공, 처리하여 각 개체의 주위 환경을 더 잘 이해하고 이를 적극적으로 활용하고자 하는 인간 본성의 발로라고 할 수 있을 것이다.

지난 수십 년간 지속적이면서도 빠르게 발전해 온 반도체 공정 및 회로 설계 기술은 인간 사회에 유례 없는 혁신을 가져다 주었고, 이제는 기술의 목적이 단순한 자연 현상의 이해에서 벗어나 사람 간의 새로운 형태의 의사 소통과 문화 창출의 도구로 이동하고 있음을 우리는 목도하고 있다. 이제는 너무나 익숙해진 도구인 PC와 스마트폰 뿐만 아니라 IoT 디바이스의 범주에 속하는 형태에 구애받지 않는 다양한 컴퓨팅 디바이스가 끊임없이 개발되어 우리 주변을 채우고 있다. 이러한 기술의 번영, 특히 모바일 디바이스의 소형화, 고성능화, 다각화는 끊임없이 발전하는 반도체 관련 기술이 가능케 한 저전력 컴퓨팅에 많은 빚을 지고 있다.

스마트폰 등 모바일 플랫폼은 연산 성능의 비약적 증가에 힘입어 기

〈그림 1〉 Iphone 7의 내부 구조[1]

존에는 상상할 수 없을 정도로 많은 활용 영역에서 사용되고 있으며 현재에도 그 영역을 계속적으로 넓혀가고 있다. 그래픽 처리 능력이 PC와 유사한 수준으로 빠르게 발전하면서 모바일 게임은 물론 본격적인 무선 VR의 시대로 다가서고 있다.

10nm 이하의 작은 반도체 소자가 상용화된 지금, 각 반도체 소자의 효율성과 성능은 크게 개선되었지만 반도체 관련 기술에 비해 현저히 느리게 발전하고 있는 배터리의 부피 및 무게당 저장 용량으로 인해 시스템의 전력 자원이 심각하게 제한되는 상황은 그대로 유지되고 있다. 그 결과 사용 시간의 극대화를 위해 대부분의 모바일 시스템에서는 배터리가 가장 큰 비중을 차지하고 있다 (〈그림 1〉). 따라서 제한된 전력 자원으로 최적화된 사용자 경험을 제공하고 사용 시간을 극대화 할 수 있는 저전력 컴퓨팅 기술이 모바일 시스템 설계의 핵심 기술이라고 할 수 있다.

본 글에서는 모바일 컴퓨팅 플랫폼을 위한 저전력 하드웨어 설계 기술의 다양한 측면에 대해 살펴볼 예정이다. CPU를 비롯한 범용 프로세서와는 달리 특정 연산을 고효율로 처리할 수 있도록 설계된 디지털 신호처리 가속기를 중심으로, 본 글에서는 지난 몇 년간 발표된 디지털 신호처리 하드웨어를 소개하고 이를 통해 다양한 전력 저감 기술을 소개할 것이다.

Ⅱ. 디지털 시스템의 전력 효율성

디지털 하드웨어의 소모 전력 감소와 효율성 제고를 위해서는 먼저 시스템의 각 구성 요소가 전력 소모에 미치는 영향을 알아야 한다. 동작 상태에서의 지속적인 전력 소모가 효율성의 주요 척도인 아날로그 회로와는 달리 디지털 회로는 각각의 연산에 소모되는 에너지가 시스템의 전력 소모와 직결된다. 디지털 회로의 전력 소모는 크게 동적 전력(Dynamic Power)과 정적 전력(Static Power)으로 구분하며 아래의 식으로 나타낼 수 있다.

$$P_{dyn} = \alpha C V_{DD}^2 f + I_{SC} V_{DD} \quad (1)$$

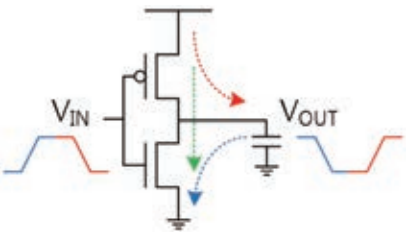
$$P_{static} = (I_{leak} + I_{contention}) V_{DD} \quad (2)$$

동적 전력(Eq. 1)은 각 디지털 회로가 다음 단의 input capacitance 및 parasitic capacitance(C)를 충전 및 방전하는데 필요한 switching power와 회로의 입출력 신호가 바뀌는 과정에서 짧은 시간 동안 NMOS와 PMOS가 동시에 약하게 켜지면서 발생하는 short circuit current(I_{SC})로 구성되어 있으며(〈그림 2〉) 일반적으로 전자가 80~90%를 차지한다.

정적 전력(Eq. 2)은 CMOS 소자의 gate, diffusion 영역 등에서 발생하는 누설 전류(I_{leak})와 특정 회로 구조에서 pull-up network와 pull-down network가 동시에 켜지면서 발생하는 contention에 기인한 전력 소모로 구분할 수 있다. 일반적인 static complementary CMOS 회로에서는 contention 부분은 무시 가능한 수준이다.

65nm와 그 이전의 CMOS 공정에서는 상대적으로 큰 소자의 크기와 높은 동작 전압으로 인해 동적 전력이 시스템의 전력 소모에서 대부분을 차지하였다. 따라서 저전력 설계 기술은 같은 연산을 수행할 때 소모되는 동적 전력 소모를 줄이는 것에 집중되었다. 하지만 공정 소형화가 계속 되면서 전체 시스템 전력 소모에서 누설 전류가 차지하는 부분이 증가하여 최신 설계에서는 정적 전력 소모 역시 반드시 고려해야 하는 중요한 요소가 되었다[2].

위에서 살펴본 동적 및 정적 전력 소모는 저전력 프로세서 또는 가속기 설계에서 폭넓게 적용되어 각 시스템의



〈그림 2〉 인버터의 동적 전력

효율성 비교를 위한 하나의 측정 기준(metric)으로 사용되고 있다. 하지만 Eq. 1과 2에서 볼 수 있듯이 전력 소모는 동작 주파수, 전압 등 여러 동작 조건에 직접적으로 영향을 받기에 서로 다른 시스템의 효율성 비교에는 적합하지 않다. 그 대신 시스템이 주어진 작업을 완료하는데 소모한 총 에너지를 환산하여 이를 측정 기준으로 삼는 것이 바람직하다. 이 때 한 번의 연산(Operation)당 소모되는 에너지와 주어진 작업에 필요한 총 연산의 수를 알면 필요한 전체 에너지를 쉽게 알 수 있으며, 이를 식으로 나타내면 아래와 같이 표현할 수 있다[2-4].

$$Energy / Op * \#Op = Total\ Energy \quad (3)$$

위 방식은 특히 한정된 전력 자원을 갖는 시스템에서 유용한데, 배터리가 한 번 충전된 이후 저장된 에너지로 얼마나 많은 연산을 수행할 수 있는지, 그리고 가능한 사용 시간이 얼마인지를 쉽게 알 수 있다는 장점이 있다.

본 글에서는 위의 Eq. 3에서 전력 소모를 결정하는 연산당 소모 에너지(E/Op)와 주어진 작업 또는 알고리즘을 완료하기 위해 필요한 연산의 수(#Op)를 줄임으로써 전력 효율성을 개선할 수 있는 여러 기법에 대해 소개할 것이다.

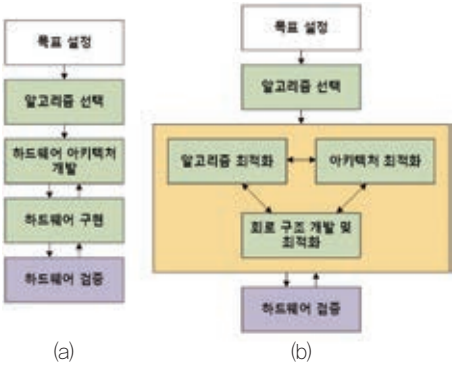
Ⅲ. 저전력 신호처리 가속기 설계 기술

신호처리 가속기는 특정 연산 또는 알고리즘을 고효율로 처리하기 위해 주로 사용된다. 즉, 어떠한 작업을 수행하고 어떤 결과를 내야 하는지가 명확하게 주어져 있으며, 단위 시간당 처리해야 할 데이터의 양도 미리 알 수 있다는 설계상의 장점이 있다. 고효율 및 저전력 가속기 제작을 위해서는 이와 같은 하드웨어 가속기의 특성을 잘 이해하고 설계 과정에서 적극적으로 이용하는 것이 핵심이라고 할 수 있다.

디지털 신호처리 가속기 설계는 크게 보면 총 세 개의 축으로 구성되어 있음을 알 수 있다. 하드웨어가 입력 데이터를 처리하여 원하는 형태의 출력을 내기 위한 적절한 알고리즘이 필요하고, 이를 하드웨어에서 효율적으로 수행하기 위해 적합한 하드웨어 아키텍처가 필요하다. 마지막으로 하드웨어의 각 모듈을 실제로 반도체 소자로 구현하려면 회로 구조의 개발 역시 이루어져야 한다.

일반적인 경우 위의 과정이 순차적으로 수행되며 최적화 과정은 각 단계별로 이루어진다(〈그림 3(a)〉). 즉, 시스템의 사용 환경과 성능 목표가 설정되면 이를 달성하기 위한 여러 알고리즘을 찾아 시뮬레이션을 통해 검증하고, 최고의 성능을 내는 알고리즘 중에서 하드웨어로 구현하기에 적합한 것을 최종적으로 선택한다. 이어서 선택된 알고리즘을 가장 낮은 비용(전력, die area 등)으로 구현할 수 있는 하드웨어 아키텍처를 개발하고, 이를 실제 standard cell library와 memory compiler 등을 이용한 합성 과정 및 회로 구조 설계를 통해 최종적인 하드웨어 설계를 얻는다. 마지막으로 다양한 CAD tool을 이용해서 하드웨어를 검증하고, 만약 성능, 효율성 등의 목표를 달성하지 못하면 하드웨어 아키텍처와 회로 구조 등을 개선하는 작업을 추가로 수행한다.

하지만 위에서 설명한 순차적인 설계 방법은 이전 단계에서 확정된 설계에 따라 각 단계에서 추가적으로 제한적인 최적화만 가능케 한다는 단점을 지닌다. 이러한 문제를 해결하기 위해 설계 과정에서 여러 요소를 동시에 고려하여 시스템의 성능을 그대로 유지하면서 효율성을 극대화 할 수 있는 합동 최적화(Co-optimization) 설계 기법이 제안되었다(〈그림 3(b)〉)[10]. 이는 시스템 설계의 구성 요소 중 두 개 이상을 동시에 최적화 공간(Optimization Space)에 대입하여 그 공간 내에서 최적점을 찾는 설계 방법이다. 일례로 알고리즘 설계 및 최적화 단계에서 효율적인 하드웨어 아키텍처 구현 가능



〈그림 3〉 (a)순차적 설계와 (b)합동 최적화 설계의 흐름도

성을 고려할 수 있다. 즉, 단순히 알고리즘의 성능이나 총 연산 수 등의 일차원적인 비용 함수(Cost Function) 대신에, 하드웨어 아키텍처로 환원했을 경우에 메모리 접근 용이성, 메모리 대역폭의 최대치, 연산 병렬화(Parallelization)의 가능성, 고정 소수점 연산으로 변환 시 성능 저하 정도 등을 종합적으로 고려하여 알고리즘 자체를 하드웨어 구현에 용이하게 재설계하는 방법이 가능할 것이다.

위에서 제안한 합동 최적화 설계 기법을 중심으로, 본 글에서는 하드웨어 설계의 세 축을 어떻게 결합하여 기존 연구와 다른 새로운 형태의 하드웨어 설계안을 도출할 수 있을지 최근 발표된 연구들을 통해 확인해본다.

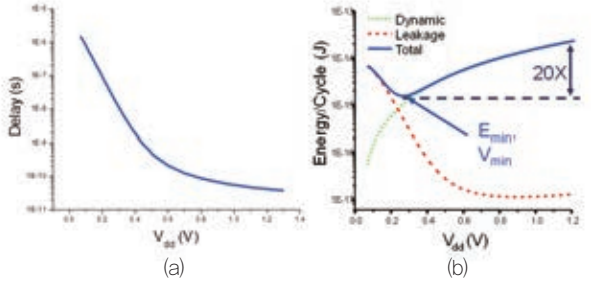
1. 아키텍처–회로 설계 최적화

Eq. 1에서 살펴봤듯이 디지털 회로의 동적 전력은 동작 전압의 제곱에 비례하고, 따라서 전압 감소 또는 조정(Voltage Scaling)을 통해 전력 소모를 극적으로 감소시킬 수 있다. 일반적으로 사용되는 Static Complementary CMOS 회로 구조는 큰 Noise Margin을 갖고 있기 때문에 상당히 낮은 전압에서도 문제 없이 동작 가능하다. 하지만 전압이 낮아질수록 공정–전압–온도 변이(PVT Variation)와 영향이 더 크게 나타나는 문제가 생기며, 회로의 동작 속도가 급격히 느려지는 현상이 발생한다. 그 결과 전압이 문턱 전압(Threshold Voltage) 근처에 이르게 되면 궁극적으로 정적 전력이 동적 전력을 능가하게 되는 결과로 이어진다.

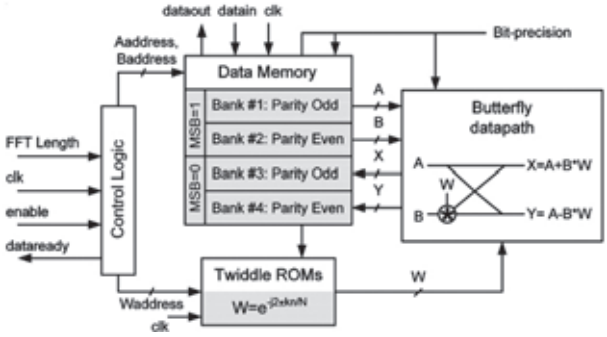
$$E = \alpha C V_{DD}^2 + I_{leak} V_{DD} T_{CLK} \quad (4)$$

시스템의 전력 또는 에너지 효율성을 계산하기 위해서는 각 연산당, 또는 한 clock cycle 마다 소모되는 에너지를 구해야 하는데, 이는 Eq. 4로 표현될 수 있다. 동작 전압이 낮아지면 V_{DD}와 I_{leak}이 모두 줄어들지만, 동시에 동작 속도가 느려지면서 한 clock의 주기(T_{CLK})는 더 빠른 속도로 증가하기 때문에, 정적 전력을 나타내는 두 번째 항은 오히려 증가하게 된다.

〈그림 4〉는 inverter chain을 동작 전압을 낮춰가면서 구동시켰을 때의 delay와 clock cycle당 에너지 소모를 나타낸 것이다. 동적 전력과 정적 전력의 합인 총 에너지 소모는 계속 줄어들다가 일정 전압(V_{min})에서 최소값(E_{min})을 가지고, 이후 정전 전력에 의해 다시 증가하게 된다. 즉, voltage scaling을 적용한다 하더라도 계속 효율성이 개선되는 것이 아니라 이론적으로 얻을 수 있는 최대의 효율성이 존재한다는 것이 이론적으로 증명되었다[4].



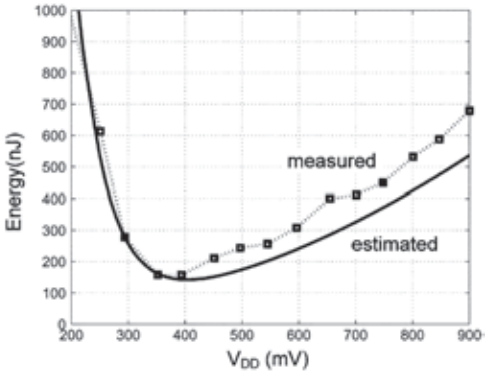
〈그림 4〉 Voltage scaling에 따른 inverter chain의 (a) delay와 (b) 에너지 추이



〈그림 5〉 메모리 기반 아키텍처를 차용한 FFT 가속기[3]

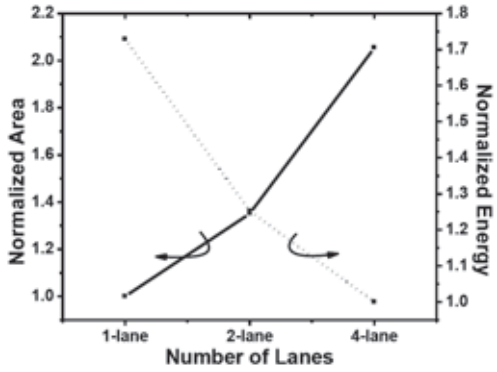
이러한 현상은 실제로 신호처리 가속기 IC를 통해서도 실험적으로 증명되었다. 메모리 기반의 아키텍처(〈그림 5〉)로 구현된 FFT(Fast Fourier Transform) 가속기 설계에서 실제도 에너지 소모의 최저점이 존재한다는 것이 밝혀졌고, 해당 지점에서 시스템을 구동함으로써 최대의 에너지 효율성을 얻을 수 있음을 보였다(〈그림 6〉)[3]. 이 결과는 회로 설계의 관점에서 이론적으로 가능한 최대 에너지 효율성을 실제로 달성하였다는 것에 의의가 있으며, 실제로 이후 오랜 기간 가장 높은 효율을 갖는 FFT 가속기의 기록을 보유하고 있었다.

위의 설계에서는 디지털 회로 자체의 이론적인 한계를 달성한 것이었고, 따라서 가속기의 에너지 효율성을 이 이상으로 높이기 위해서는 다른 접근 방법이 필수적임

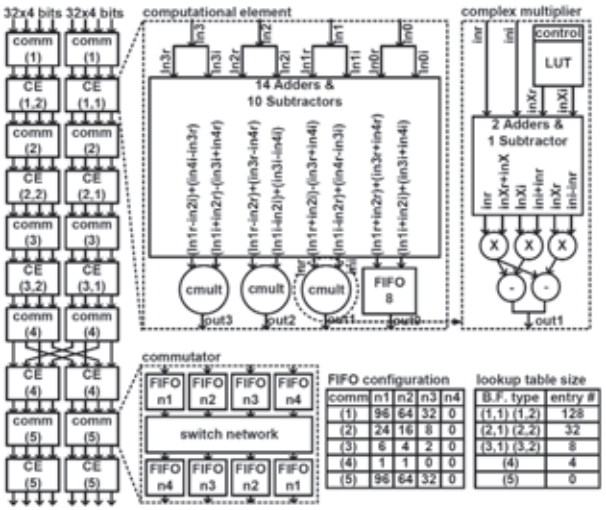


〈그림 6〉 Voltage scaling에 따른 에너지 소모의 예측 및 측정값[3]

을 알 수 있다. 이 문제에 대해 하드웨어 아키텍처와 결합하여 합동 최적화 과정을 통해 진일보된 해결책을 제시한 가속기 설계가 [5]에서 발표되었다. 단순한 voltage scaling의 적용에서 탈피하여, 기존에 존재하던 여러 형



〈그림 7〉 FFT 가속기의 병렬화에 따른 die area와 에너지 소모량[5]



〈그림 8〉 2개의 processing lane을 갖는 병렬 FFT 가속기의 구조[5]

태의 FFT 가속기 아키텍처(memory-based, pipelined, hybrid, many-core 등)를 에너지 효율성 관점에서 분석하였고, 그 결과 pipelined architecture가 메모리의 누설 전력을 억제함으로써 최적 전압(V_{min})과 최저 에너지(E_{min})를 동시에 줄일 수 있음을 확인하였다. 또한, pipelined architecture를 더 많이 병렬화하여 동시 처리량(throughput)을 늘림으로써 에너지 효율성을 추가적으로 개선할 수 있음을 확인하였다(〈그림 7〉).

위에 설명된 에너지 소모량 관점에서 수행된 아키텍처의 최적화 과정을 통해 〈그림 8〉에 보여지는 형태의 병렬 FFT 가속기 구조가 제안되었다. 65nm에서 제작된 IC의 측정 결과 1024-point FFT 연산당 15.8nJ의 에너지만을 소모함을 확인하였으며, 기존에 발표된 결과에 비해 2.4배 개선된 에너지 효율성을 보였다. FFT 가속기 관련 연구가 수십 년간 지속된 점을 고려했을 때, 이는 매우 큰 폭의 효율성 개선임을 알 수 있다.

2. 알고리즘-아키텍처 최적화

FFT 알고리즘은 상대적으로 연산이 단순하며 관련 분야에서 하나의 기준점으로 오랜 기간 사용되었기 때문에 알고리즘의 추가적인 최적화를 기대하기 어렵다. 그러나 최근 머신 러닝의 급격한 발전으로 인해 기존의 알고리즘에 비해 현저히 개선된 다양한 종류의 알고리즘이 존재하고 있으며, 이는 성능 저하를 최소화 하면서도 더 효율적인 하드웨어 구현을 위해 기존의 알고리즘을 최적화 할 수 있는 가능성과 자유도를 제공한다.

특성 추출(Feature Extraction)은 주어진 입력 데이터에서 원하는 종류의 정보 또는 특성값을 추출하는 알고리즘으로, 머신 러닝 분야 전반에서 폭넓게 사용되며 특히 이미지 또는 비디오를 다루는 컴퓨터 비전에서 많이 사용된다. 이미지의 경우 해상도에 따라 처리해야 할 픽셀의 수, 즉 차원(Dimensionality)이 너무 많아 이를 특성 추출 과정을 통해 처리함으로써 이후 인식(Recognition) 또는 분류(Classification) 과정에서 처리해야 할 데이터의 양을 크게 줄이는 역할을 한다.

특성 추출을 위해 가장 많이 사용되던 알고리즘 중 하나는 SIFT(Scale-Invariant Feature Transform)이다[6].

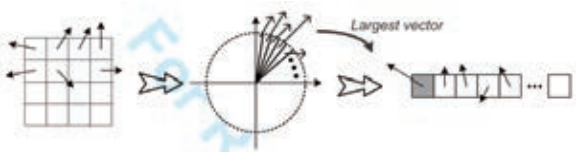
여러 scale의 Gaussian filter에 이미지를 통과시켜 그 차이에서 극점을 찾아내는 방식으로, 회전과 크기 변화를 포함한 이미지의 다양한 변형 하에서도 같은 특성을 추출할 수 있기에 알고리즘의 신뢰성을 담보할 수 있다. 하지만 이미지 처리 과정에서 요구 연산량이 많아서 고속 처리가 어렵다는 단점이 있었고, 이를 개선하기 위해 보다 간소화된 SURF(Speeded-Up Robust Features) 알고리즘이 제안되었다[7]. 필터 형태의 최적화 등의 기법으로 기존 SIFT 알고리즘에 비해 3배 내외의 속도 향상을 달성하면서도 동등한 수준의 정확도를 얻을 수 있었다.

SURF 알고리즘은 적은 연산량으로 전력 자원이 제한된 시스템에 적합하다. 실제로 소형 드론인 MAV(Micro Air Vehicle)에 탑재되어 밀폐된 실내 공간에서 SLAM(Simultaneous Localization and Mapping)을 통해 자율 비행을 하는데 사용되었다[8-9]. 그러나 여전히 높은 알고리즘의 연산 요구량과 마이크로 프로세서의 제한된 컴퓨팅 자원으로 인해 초당 최대 1개의 프레임만을 처리할 수 있었으며 소모 전력도 1W에 달하였다. 충분한 비행 시간을 확보하기 위해서는 전력 효율성을 획기적으로 개선하는 것이 필요하여 ASIC(Application-Specific IC)의 형태로 하드웨어 가속기의 개발이 진행되었다.

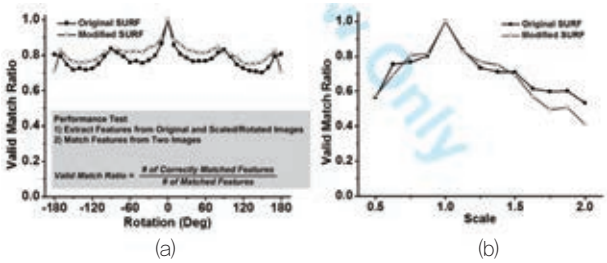
하지만 SURF 알고리즘에서는 이미지의 각 부분에서 sampling을 수행하여 각 지점의 대표각(Orientation)을 찾아내고, 이 각도에 따라 sampling window를 새로 정의하여 다시 sampling 및 이미지 처리를 해야 한다. 이로 인해 높은 메모리 대역폭이 필요하며 병렬 처리를 구현하기도 어렵다. 따라서 하드웨어 가속기를 개발한다 하더라도 범용 프로세서에 비해 크게 개선된 효율성을 얻기 어렵다는 한계가 존재한다.

[10]에서는 이러한 문제를 알고리즘과 하드웨어 아키텍처의 합동 최적화로 해결할 수 있음을 보여준다. 각 sampling 지점에서 sampling window를 중심각의 크기가 11.25도인 부채꼴의 형태로 세부 영역을 나누어 추출을 수행하고, 대표각을 찾은 뒤에 기존의 sampling 결과를 단순히 재배열 함으로써 두 단계의 특성 추출 알고리즘을 하나로 통합하였다. 그 결과 메모리 요구량이 89% 감소하였고, 시뮬레이션 결과 제안된 알고리즘이 기존 SURF와 비교했을 때 동등한 수준의 불변성(Invariance)을 갖는 것을 확인하였다(〈그림 10〉).

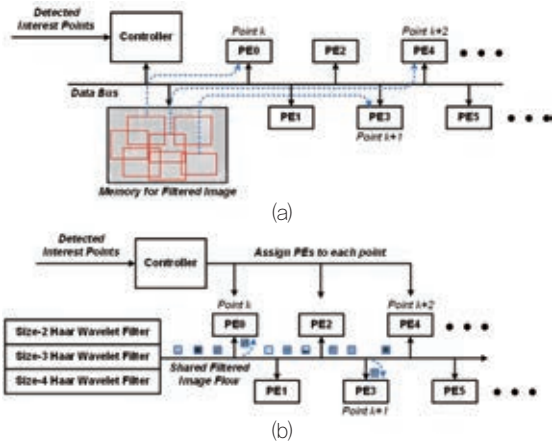
또한 위의 알고리즘 최적화는 각 추출 지점 주위의 sampling region을 임시로 저장해야 한다는 제한 조건을 완전히 제거하여 새로운 가속기 하드웨어 아키텍처 개발을 가능케 하였다. 〈그림 11(a)〉에 그려진 기존의 멀티코어 아키텍처에서는 하나의 data bus를 여러 개의 코어 또는 PE(Processing Element)가 공유하고 있기 때문에 데이터 병목 현상을 피하기 위해 data bus가 고속도로 동작해야 한다. 하지만 변형된 알고리즘을 적용하면



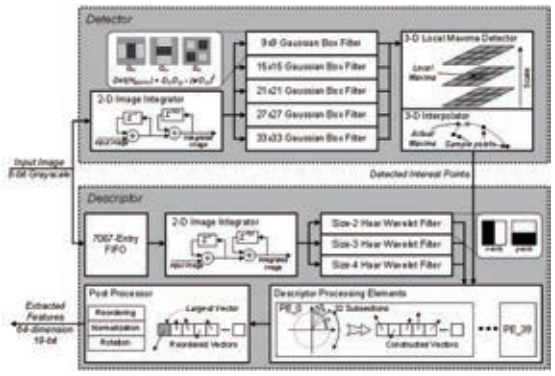
〈그림 9〉 제안된 특성 추출 알고리즘[10]



〈그림 10〉 변형된 SURF 알고리즘의 이미지 크기 조정 및 회전에 대한 불변성 특성[10]



〈그림 11〉 (a) 기존 멀티코어 아키텍처와 (b) 제안된 single-stream 아키텍처[10]



〈그림 12〉 변형된 SURF 기반 하드웨어 가속기 구조[10]

〈그림 13〉 가속기 IC의 측정 환경[10]

〈그림 11(b)〉의 single-stream 아키텍처를 사용하는 것이 가능하다. Data bus를 통해 전체 이미지가 일정한 속도로 전송되면, 각각의 PE는 필요한 영역의 데이터만을 선택하여 실시간으로 처리(on-the-fly processing)하고 입력 데이터를 별도로 메모리에 저장하지 않는다. 따라서 data bus가 저속도로 동작 가능하고, 각 PE의 크기 및 전력 소모도 크게 줄일 수 있게 된다.

위에서 설명한 알고리즘과 아키텍처의 최적화 과정을 종합하여 〈그림 12〉의 하드웨어 가속기 구조가 제안되었다. 28nm에서 제작된 가속기 IC는 30fps의 속도로 VGA 입력 비디오를 처리하면서 2.8mW의 전력만을 소모하였다(〈그림 13〉). 이는 기존에 보고된 연구 결과에 비해 3.5배 개선된 전력 효율성을 보이는 것이다.

3. 알고리즘-회로 설계 최적화

일반적으로 신호 처리 가속기 설계 과정에서는 사용하는 알고리즘과 이를 구현하는 하드웨어 아키텍처가 가장 중요하게 다뤄진다. 이는 상위 계층으로 갈수록 설계 변경이 용이하고, 또 최적화를 통해 더 많은 전력 효율성 개선을 달성할 수 있기 때문이다. 그리고 현실적으로 IC 제작 과정에서 foundry에서 제공한 standard cell library와 memory compiler에 의존하여 하드웨어 설계를 진행하기 때문에, 회로 구조의 변경이 쉽지 않고 재설계에 많은 비용이 투입되어야 하는 문제점이 존재한다.

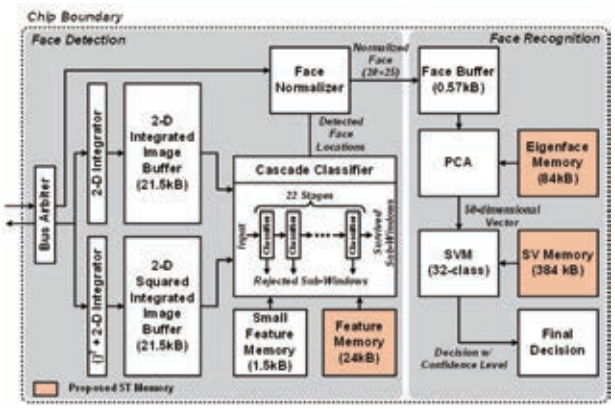
그러나 신호처리 하드웨어의 설계 기술이 성숙되고, 특히 딥러닝을 비롯한 최신 알고리즘의 성능이 충분히 발전하여 일종의 표준이 존재하는 현 상황에서는 효율성의 지속적인 개선을 위해서 새로운 돌파구가 필요한 상황이다. 따라서 지금까지는 많이 연구되지 않았던 가장 높은 층위인 알고리즘과 가장 낮은 층위인 회로 설계를 결합하여 최적의 설계를 탐색하는 새로운 접근 방법이 대안이 될 수 있을 것이다.

이러한 알고리즘-회로 설계 합동 최적화의 예는 [11]에서 찾아볼 수 있다. 사물 인식(Object Recognition)은 컴퓨터 비전 분야에서 지난 수십 년간 주요하게 다뤄진 문제이다[12-13]. 또한 얼굴 인식(Face Recognition)은 사물 인식의 일부로 출발하여 현재는 독립된 분야로 활발히 연구되고 있다[14]. 최근 급격히 발전한 딥러닝을 포함하여 대부분의 얼굴 인식 알고리즘에서는 입력 이미지에서 적절한 특성(Feature)을 추출하고, 이를 이용해 분류기(Classifier)를 학습시켜 얼굴 인식 기능을 구현한다. [11]에서는 인식 알고리즘에서 많이 사용되는 Haar-like feature 기반의 cascaded classifier[15]로 얼굴 검출을 구현하고, 인식된 얼굴에 PCA(Principal Component Analysis)를 적용하여 Eigenface를 추출[16]하였다. 이후 SVM(Support Vector Machine) 기반 인식기에서 데이터베이스에서 해당 인물을 찾아내는 방식으로 진행된다(〈그림 14〉).

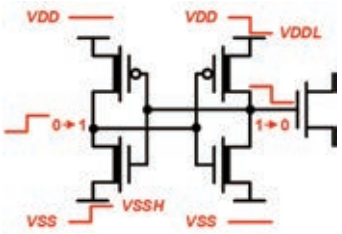
〈그림 14〉 얼굴 인식 알고리즘의 흐름도[11]

설계 단계에서 먼저 위에서 언급된 알고리즘과 아키텍처 최적화 기법을 얼굴 인식 가속기에도 동일한 방식으로 적용하여 〈그림 15〉와 같은 얼굴 인식 가속기 아키텍처를 개발하였다. 에너지 효율성의 극대화를 위해서 알고리즘의 학습 데이터를 IC 내부 메모리에 저장해야 했고, 이는 알고리즘의 최적화 과정에서 비용 함수에 포함되어 학습 데이터의 저장량을 큰 폭으로 줄이는 성과를 거두었다. 그러나 이러한 최적화 과정 이후에도 총 500kB 이상의 학습 데이터가 on-chip memory 내에 저장되어야 할 것으로 예측되었고, 일반 CMOS 공정에서 SRAM으로 구현하기에는 매우 큰 die area가 필요할 뿐만 아니라 허용 수준을 벗어나는 누설 전력이 발생할 것이 자명하였다.

따라서 메모리 모듈의 전력 소모를 줄이기 위해 알고리즘을 고려한 회로 구조의 최적화 기법을 시도하였고, 제안된 얼굴 인식 알고리즘의 특성을 면밀히 살펴본 결과 on-chip memory에 저장되는 학습 데이터는 업데이트가 매우 드물게 필요하다는 점을 확인하였다. 즉, 가속기 초기화 과정에서 학습 데이터를 칩 내부에 저장하면 찾고자 하는 인물 목록을 바꾸지 않는 한 이를 업데이트 할 필요가 없다. 따라서 얼굴 인식 연산 중에는 단순히 학습 데이터를 불러오기만 하면 되며, 이를 이용하면 메모리를 읽기 동작에 최적화된 형태로 구현하여 전력 소모를 낮출 수 있을 것으로 예상되었다.



〈그림 15〉 얼굴 인식 가속기의 구조[11]

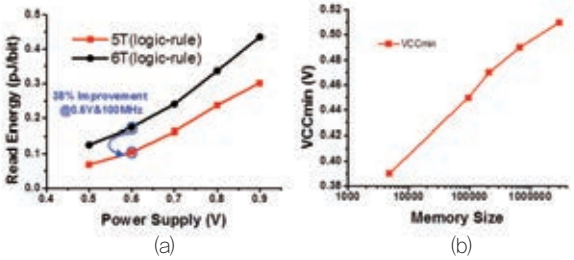


〈그림 16〉 제안된 5T 메모리 bit cell과 쓰기 동작의 개념도[11]

위의 특성을 기반으로 읽기 연산에 최적화된(Mostly Read-Only) 5T bit cell 기반 메모리를 제안하였다(〈그림 16〉). 기존의 5T 또는 6T 구조와는 달리 access transistor는 읽기 동작에만 사용되며, 쓰기 동작에는 V_{DD}, V_{SS} supply rail을 사용한다. 8T 및 10T 구조와 유사하게 access transistor의 gate에 bit cell의 internal node가 연결되어 한 개의 bitline으로도 신뢰성 높은 읽기 동작을 수행할 수 있다. 데이터를 bit cell에 쓰기 위해서는 power supply rail의 전압을 바꿔줘야 하기 때문에 기존 bit cell에 비해서 많은 에너지를 소모하나, 위에서 언급한 바와 같이 가속기의 일반적인 사용 환경에서는 쓰기 연산이 거의 일어나지 않기 때문에 전체적인 에너지 효율성에는 큰 영향을 미치지 않는다.

시뮬레이션 결과 제안된 5T 구조는 기존 6T에 비해 읽기 동작당 38% 낮은 에너지가 필요했으며, 동시에 bit cell 면적도 7.2% 감소함을 확인하였다. 이는 기존에 발표된 5T, 7T, 8T 등의 구조와 비교해서도 현저히 낮은 수치이다.

제안된 가속기 설계는 40nm CMOS 공정에서 제작되어 HD 해상도의 비디오를 5.5fps의 속도로 처리하면서 23mW의 전력만을 소모하는 것으로 측정되었다. 또한 제안된 5T의 높은 read margin으로 인해 모든 메모리 모듈이 600mV에서도 오류 없이 동작하였다.



〈그림 17〉 시뮬레이션에서 (a) 기존 6T 구조와의 읽기 에너지 비교 및 (b) 메모리 크기에 따른 최소 동작 가능 전압

Ⅳ. 향후 전망 및 결론

하드웨어는 한 번 제작된 이후에는 수정이 거의 불가능하다는 특성이 있고, 따라서 실험적인 최신 신호처리 알고리즘을 즉각적으로 반영하기에는 현실적인 어려움이 있다. 신호처리 가속기는 최신의 알고리즘 보다는 안정화, 최적화가 완료되어 널리 사용되는 알고리즘을 기반으로 제작되는 것이 일반적이다.

같은 맥락에서, 신호처리 가속기 관련 연구의 미래는 알고리즘 분야에서 현재 이루어지는 연구를 보면 예측할 수 있다. 지난 몇 년간 딥러닝의 성능이 발전하고 그 활용 영역이 급격히 확대되면서 인공 지능에 대한 관심은 이미 그 정점에 다다랐다고 해도 과언이 아니다. 보다 더 다양한 형태의 딥러닝 알고리즘이 하드웨어 가속기의 형태로 구현될 것이며, 특히 IoT 등 모바일 플랫폼에 대한 시대적 요구와 맞물려서 각 클라이언트에서 획득한 데이터를 실시간으로 정확히 처리하는데 필수적인 저전력 딥러닝 가속기에 대한 수요는 급증할 것이다.

현 시점에서 가장 기본적이며 널리 사용되는 CNN (Convolutional Neural Network)과 RNN (Recurrent Neural Network)의 하드웨어 구현 연구는 이미 상당 부분 진척되었다. 위에서 살펴본 설계 기법들이 일정 부분 하드웨어 아키텍처 최적화[17-18], 알고리즘 최적화[19-20], 회로 설계를 통한 최적화[21-22]의 형태로 제안되어 유의미한 효율성 개선을 이루었다. 그러나 여전히 스마트폰과 같은 소형 모바일 시스템에서 고성능 알고리즘의 실시간 처리를 always-on 형태로 달성하기까지는 많은 장벽이 남아 있으며, 본 글에서 소개한 다양한 설계 기법이 적극적으로 결합, 적용되어 새로운 돌파구가 마련되기를 기대해 본다.

참고문헌

[1] http://ifixit.org/blog/8409/iphone-7-and-7-plus-internals-wallpapers/

[2] G. Chen et al., "Millimeter-scale nearly perpetual sensor system with stacked battery and solar cells," in IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers, Feb. 2010, pp. 288-289.

[3] A. Wang and A. Chandrakasan, "A 180-mV subthreshold FFT processor using a minimum energy design methodology," IEEE J. Solid-State Circuits, vol. 40, no. 1, pp. 310-319, Jan. 2005.

[4] B. Zhai et al., "Theoretical and Practical Limits of Dynamic Voltage Scaling," in Proc. Design Automation Conf., May 2005, pp. 868-873.

[5] D. Jeon et al., "A Super-Pipelined Energy Efficient Subthreshold 240 MS/s FFT Core in 65 nm CMOS," IEEE J. Solid-State Circuits, vol. 47, no. 1, pp. 23-34, Jan. 2012.

[6] D. G. Lowe, "Object recognition from local scale-invariant features," in Proc. IEEE Int. Conf. on Computer Vision, Sep. 1999, pp. 1150-1157.

[7] H. Bay et al., "SURF: Speeded Up Robust Features," Computer Vision and Image Understanding, vol. 110, no. 3, pp. 346-359, 2008.

[8] S. Shen et al., "Autonomous Multi-Floor Indoor Navigation with a Computationally Constrained MAV," in Proc. IEEE Conf. on Robotics and Automation, May 2011, pp. 20-25.

[9] G. Grisetti et al., "Improving Grid-based SLAM with Rao-Blackwellized Particle Filters by Adaptive Proposals and Selective Resampling," in Proc. IEEE Conf. on Robotics and Automation, Apr. 2005, pp. 2432-2437.

[10] D. Jeon et al., "An Energy Efficient Full-Frame Feature Extraction Accelerator With Shift-Latch FIFO in 28 nm CMOS," IEEE J. Solid-State Circuits, vol. 49, no. 5, pp. 1271-1284, May. 2014.

[11] D. Jeon et al., "A 23-mW Face Recognition Processor with Mostly-Read 5T Memory in 40-nm CMOS," IEEE J. Solid-State Circuits, to appear.

[12] M. Schwarz, H. Schulz, and S. Behnke, "RGB-D object recognition and pose estimation based on pre-trained convolutional neural network features," in Proc. IEEE International Conference on Robotics and Automation, May. 2015, pp. 1329-1335.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet

classification with deep convolutional neural networks," in Proc. Advances in Neural Information Processing Systems, Dec. 2012, pp. 1–9.

[14] N. Sumi, A. Baba, and V. G. Moshnyaga, "Effect of computation offload on performance and energy consumption of mobile face recognition," in Proc. IEEE Workshop on Signal Processing Systems, Oct. 2014, pp. 1–7.

[15] P. Viola and M. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features," in Proc. IEEE Conf. on Computer Vision and Pattern Recognition, Dec. 2001, pp. 511–518.

[16] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in Proc. IEEE Conf. on Computer Vision and Pattern Recognition, Jun. 1991, pp. 511–518.

[17] Y.-H. Chen et al., "Eyeriss: An Energy-Efficient Reconfigurable Accelerator for Deep Convolutional Neural Networks," in IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers, Feb. 2016, pp. 262–263.

[18] D. Shin et al., "DNPU: An 8.1TOPS/W Reconfigurable CNN-RNN Processor for General-Purpose Deep Neural Networks," in IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers, Feb. 2017, pp. 240–241.

[19] J. Sim et al., "A 1.42TOPS/W Deep Convolutional Neural Network Recognition Processor for Intelligent IoE Systems," in IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers, Feb. 2016, pp. 264–265.

[20] J. Chung and T. Shin, "Simplifying Deep Neural Networks for Neuromorphic Architectures", in Proc. IEEE/ACM Design Automation Conference (DAC), June 2016, pp. 1–6.

[21] P. A. Whatmough et al., "A 28nm SoC with a 1.2GHz 568nJ/Prediction Sparse Deep-Neural-Network Engine with >0.1 Timing Error Rate Tolerance for IoT Applications," in IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers, Feb. 2017, pp. 242–243.

[22] S. Bang et al., "A 288μW Programmable Deep-Learning Processor with 270KB On-Chip Weight Storage Using Non-Uniform Memory Hierarchy for Mobile Intelligence," in IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers, Feb. 2017, pp. 250–251.

전동석

• 2009년 9월 서울대학교, 학사 (전자공학)
• 2014년 12월 University of Michigan, PhD (Electrical Engineering)
• 2014년 10월~2015년 12월 MIT, Postdoctoral Associate
• 2016년 3월~현재 서울대학교 융합과학기술대학원, 조교수

〈관심분야〉
SoC/회로 설계, 신호 처리, 머신 러닝