



대한임베디드공학회논문지

IEMEK Journal of Embedded Systems and Applications

논문

특집호 : 인공지능 기반 영상/신호 처리 응용분야와 시스템 최적화

- | | |
|-----|---|
| 309 | 37. 웨이블릿 영역에서 회전 불변 에너지 특징을 이용한 이중 브랜치 복사-이동 조작 검출 네트워크
박준영, 이상인, 엄일규* |
| 319 | 38. 객체 탐지 과업에서의 트랜스포머 기반 모델의 특징점 분석 연구
하정민, 이현중, 엄정민, 이재구* |
| 329 | 39. 잡음 제거 기술 기반의 비디오 인페인팅 성능 연구
서정윤, 백한결, 박상효* |
| 337 | 40. 마스크 생산 라인에서 다중 영상 기반 마스크 이어링 검사 방법
우지명, 이상현, 이현철* |
| 347 | 41. 텍스트-비디오 검색 모델에서의 캡션을 활용한 비디오 특성 대체 방안 연구
이동훈, 허 찬, 박혜영*, 박상효* |
| 355 | 42. 실시간 탄도 궤적 목표물 추적을 위한 GPU 기반 병렬적 입자군집최적화 기법
한운호, 이현철*, 권혁훈, 최원석, 정보라 |
| 367 | 43. FPGA기반 뉴럴네트워크 가속기에서 2차 타일링 기반 행렬 곱셈 최적화
권진세, 이제민, 권용인, 박제만, 유미선, 김태호, 김형신* |
| 375 | 44. 차량용 임베디드 프로세서에서 저전력 반응적 제어를 위한 이기종 멀티코어 협력적 스트리밍 온-칩 소프트웨어 구조
권지수, 박대진* |

Editorial Committee

Editor-in-Chief :

Ju Hyun Park (Yeungnam University)

Email: jessie@ynu.ac.kr

Associate Editors :

Sangmoon Lee (Kyungpook National University)

Email: moony@knu.ac.kr

Joon Hyuk Yoo (Daegu University)

Email: joonhyuk@daegu.ac.kr

Managing Editor :

Jong Pil Yun (KITECH)

Email: rebirth@kitech.re.kr

Editorial Board :

Sang Seon Byun (Catholic University of Pusan)

Email: ssbyun@cup.ac.kr

Hyeonjoong Cho (Korea University)

Email: raycho@korea.ac.kr

Gyu Sang Choi (Yeungnam University)

Email: castchoi@ynu.ac.kr

Seokin Hong (Sungkyunkwan University)

Email: seokin@skku.edu

Hyunwoo Joe (ETRI)

Email: hwjoe@etri.re.kr

Moon Gab Joo (Pukyong National University)

Email: gabi@pknu.ac.kr

Heechul Jung (Kyungpook National University)

Email: heechul@knu.ac.kr

Myoungsoo Jung (KAIST)

Email: mj@camelab.org

Suk Ju Kang (Sogang University)

Email: sjkang@sogang.ac.kr

Hyunjin Kim (Dankook University)

Email: hyunjin2@dankook.ac.kr

Soontae Kim (KAIST)

Email: kims@kaist.ac.kr

Baalint Kiss (Budapest University)

Email: bkiss@iit.bme.hu

Young Woong Ko (Hallym University)

Email: yuko@hallym.ac.kr

Donghwa Lee (Daegu University)

Email: leedonghwa@daegu.ac.kr

Hyungkyu Lee (Duksung Women's University)

Email: hglee@duksung.ac.kr

Ingyu Lee (Yeungnam University)

Email: inleeatyu@yu.ac.kr

Jong Hun Lee (DGIST)

Email: jhlee@dgist.ac.kr

Jung Hoon Lee (Gyeongsang National University)

Email: leejh@gsnu.ac.kr

Sang Jun Lee (Jeonbuk National University)

Email: sj.lee@jbnu.ac.kr

Sungjin Lee (DGIST)

Email: sungjin.lee@dgist.ac.kr

Tae Hee Lee (Chonbuk National University)

Email: thlee@jbnu.ac.kr

Sung Kyu Lim (Georgia Institute of Technology)

Email: limsk@ece.gatech.edu

Sang Hong Park (Pukyong National University)

Email: radar@pknu.ac.kr

Donghwa Shin (Soongsil University)

Email: donghwashin81@gmail.com

Hoe Seok Yang (Santa Clara University)

Email: hyang8@scu.edu

Honorary Editor :

Chuck Yoo (Korea University)

Email: hxy@os.korea.ac.kr

Institute of Embedded Engineering of Korea

IEMEK OFFICERS

President :

Yongwan Park (Yeungnam University)

Vice-Presidents :

Byung Jae Choi (Daegu University)

Chae Deok Lim (ETRI)

Wooyong Jung (DGIST)

Sam Hyuk Noh (UNIST)

Shik Kim (Semyung University)

Juchan Son (ETRI)

Dong Ha Lee (DGIST)

Chuck Yoo (Korea University)

Auditors :

Byung Hyun Moon (Daegu University)

Young Seop Kim (Dankook University)

Executive Directors :

Won Kee Hong (Daegu University)

Jong Hun Lee (DGIST)

Moon Gab Joo (Pukyong National University)

Sunghoon Lee (DGIST)

Ho Youl Jung (Yeungnam University)

Ju Hyun Park (Yeungnam University)

Do Hyeon Kim (Jeju National University)

Jeong Tak Ryu (Daegu University)

Hyungshin Kim (Chungnam National University)

Sooyoung Seok (GITC)

Taeho Kim (ETRI)

Honorary President :

Kyu Suk Chung

Sang Hyuk Son

Yong Doo Lee

Yong Kee Jun

Heung Nam Kim

인공지능 기반 영상/신호 처리 응용분야와 시스템 최적화
(System Optimizations and Applications of Artificial Intelligence-based Image and Signal Processing)

특집호 서문

인공지능 기술이 자율주행, 정보통신, 의료, 제조 등 다양한 분야에 적용되면서 우리 주변의 사물과 시스템이 빠르게 지능화되어가고 있습니다. 최근, 인공지능에 기반을 둔 영상/신호 처리 기술이 크게 발전하면서 다양한 임베디드 시스템 응용분야에 적용되는 사례가 증가하고 있고, 이에 따라 컴퓨팅 자원이 제한적인 임베디드 시스템에 인공지능 기술을 효과적으로 적용하기 위한 하드웨어 및 소프트웨어 기술 개발의 필요성이 대두되고 있습니다.

이번 특집호는 인공지능에 기반을 둔 영상/신호 처리 기술의 응용분야와 이를 위한 하드웨어 및 소프트웨어 최적화 기술에 대한 8편의 논문으로 구성하였습니다.

첫 번째 논문 “잡음 제거 기술 기반의 비디오 인페인팅 성능 연구”는 딥러닝을 활용한 영상 잡음 제거와 인페인팅 성능의 상관관계를 분석하였고, 두 번째 논문 “텍스트-비디오 검색 모델에서의 캡션을 활용한 비디오 특성 대체 방안 연구”는 딥러닝을 활용하여 비디오 영상 정보를 담아낼 수 있는 캡션을 생성하고 이를 통해 효과적으로 텍스트-비디오 검색 문제를 해결하는 기술을 제안하고 있습니다. 세 번째 논문 “웨이블릿 영역에서 회전 불변 에너지 특징을 이용한 이중 브랜치 복사-이동 조작 검출 네트워크”는 딥러닝을 활용한 영상의 복사/이동 조작 검출 방법을 소개하고 있고, 네 번째 논문 “마스크 생산 라인에서 다중 영상 기반 마스크 이어링 검사 방법”는 딥러닝을 적용하여 마스크 생산 라인에서 불량품을 탐지하는 시스템을 소개하고 있습니다. 다섯 번째 논문 “객체 탐지 과업에서의 트랜스포머 기반 모델의 특징점 분석 연구”는 트랜스포머에 기반을 둔 객체 탐지 모델들을 비교 평가하여 임베디드 시스템 환경에서 해당 모델을 적용할 수 있는지를 논하고 있습니다. 여섯 번째 논문 “실시간 탄도 궤적 목표물 추적을 위한 GPU 기반 병렬적 입자군집최적화 기법”은 입자군집최적화 알고리즘을 임베디드 시스템에서 고속 실행하기 위한 GPU기반의 병렬처리 기술에 대해 설명하고 있고, 일곱 번째 논문 “차량용 임베디드 프로세서에서 저전력 반응적 제어를 위한 이기종 멀티코어 협력적 스트리밍 온-칩 소프트웨어 구조”는 차량용 임베디드 프로세서의 이기종 코어에서 소프트웨어를 효과적으로 분담 처리하고 이를 통해 실행 시간과 전력 소모를 개선할 수 있는 기술을 소개하고 있습니다. 마지막 논문 “FPGA기반 뉴럴 네트워크 가속기에서 2차 타일링 기반 행렬 곱셈 최적화”에서는 신경망의 핵심 연산인 행렬 곱셈을 가속 실행하기 위해 신경망 가속 프로세서의 하드웨어의 특성을 고려한 2-stage 타일링 기반의 행렬 곱셈 최적화 방법을 제안하고 있습니다.

위와 같이 구성된 특집호의 발간을 통해 인공지능 기반 영상/신호 처리 기술의 다양한 응용 분야 적용을 촉진하고, 이를 임베디드 시스템에 적용하기 위한 최적화 연구의 필요성을 환기시키는 계기가 되기를 바랍니다.

마지막으로 본 특집호를 위해 소중한 연구의 결과물인 논문을 본 호에 투고해주신 많은 저자분들께 진심으로 감사드리며, 특집호의 질적 향상을 위해 애써주신 편집위원장 및 편집부위원장과 심사위원님들의 협조에 매우 깊은 감사의 마음을 전합니다.

2022년 12월

인공지능 기반 영상/신호 처리 응용분야와 시스템 최적화 특집호 편집위원
성균관대학교 교수 홍 석 인

인공지능 기반 영상/신호 처리 응용분야와 시스템 최적화 특집호 부편집위원
경북대학교 교수 김 용 태

논문 2022-17-37

웨이블릿 영역에서 회전 불변 에너지를 이용한 이중 브랜치 복사-이동 조작 검출 네트워크

(Dual Branched Copy-Move Forgery Detection Network Using Rotation Invariant Energy in Wavelet Domain)

박 준 영, 이 상 인, 엄 일 규*

(Jun Young Park, Sang In Lee, Il Kyu Eom)

Abstract : In this paper, we propose a machine learning-based copy-move forgery detection network with dual branches. Because the rotation or scaling operation is frequently involved in copy-move forger, the conventional convolutional neural network is not effectively applied in detecting copy-move tampering. Therefore, we divide the input into rotation-invariant and scaling-invariant features based on the wavelet coefficients. Each of the features is input to different branches having the same structure, and is fused in the combination module. Each branch comprises feature extraction, correlation, and mask decoder modules. In the proposed network, VGG16 is used for the feature extraction module. To check similarity of features generated by the feature extraction module, the conventional correlation module used. Finally, the mask decoder model is applied to develop a pixel-level localization map. We perform experiments on test dataset and compare the proposed method with state-of-the-art tampering localization methods. The results demonstrate that the proposed scheme outperforms the existing approaches.

Keywords : Copy-move forgery, Copy-move forgery localization, Dual branched network, Convolutional neural network, Rotation-invariant, Wavelet transform.

1. 서 론

영상은 때때로 어떤 사건을 명확히 하거나 검증하기 위한 중요한 정보로 사용된다. 그러나 다양한 영상 편집 툴의 발달로 영상의 조작이 쉬워졌다. 또한, 악의적 목적으로 조작된 영상을 사용하는 것은 사회에 부정적인 영향을 미칠 수 있다. 인간의 눈은 조작된 영상을 쉽게 감지하기가 어렵기 때문에, 신뢰할 수 있는 영상 조작 감지 방법이 필요하다. 이를 위해, 다양한 영상 위조 검출 방법에 관한 광범위한 연구가 수행되고 있다 [1, 2].

영상 조작에는 여러 가지 방법이 존재하지만, 하나의 영상 안에서 일정 영역을 복사하여 같은 영상의 다른 영역으로 붙이는 복사-이동 조작이 빈번히 사용된다. 배경을 복사하여 특정한 물체를 숨기거나, 물체를 여러 개로 보이기 위해 복사하는 방법이 흔히 사용된다. 그림 1은 복사-이동 조작 영상의 예를 나타내고 있다. 복사-이동 조작을 통하여, 영상이 가지는 정보나 의미가 숨겨지거나 바뀔 수 있으므로, 영상의 진위를 확인하고 복사-이동 조작영역을 찾아내는 것이 중요하다.

복사-이동 조작검출의 방법은 크게 블록, 키포인트, 기계



그림 1. 복사-이동 조작 이미지의 예
Fig. 1. Example of copy-move forgery image

학습 기반 방법으로 나눌 수 있다. 블록 기반 방법은 주로 다양한 블록분할과 영역분할 알고리즘을 통하여 블록 내부에서 크기와 회전 변환에 불변하는 특징들을 추출한다. 그리고 이러한 특징들을 블록 사이의 매칭을 통하여 조작된 부분을 찾아낸다. 키포인트 방법은 SIFT (scale invariant feature transform)와 같이 크기와 회전 변환에 강한 포인트들을 추출하여 매칭을 통해 조작영역을 검출한다. 최근 CNN (convolutional neural network)의 객체 감지 분야에서 보여주는 좋은 성능을 바탕으로 다양한 기계학습 기반의 복사-이동 조작검출의 방법들이 도입되어 유망한 탐지 결과를 보여주었다.

블록 기반의 복사-이동 조작검출 방법은 의심되는 이미지를 블록분할, 특징 추출, 추출된 특징을 매칭, 위조영역을 국부화하는 4단계로 이루어진다. 첫 번째 단계는 다양한 블록을 나누는 방법과 영역분할 방법을 이용하는 전처리 단계이다. 정사각형 블록을 중첩시키는 방법 [3], 정사각형 블록

*Corresponding Author (ikeom@pusan.ac.kr)
Received: Oct. 13, 2022, Revised: Nov. 22, 2022, Accepted: Nov. 26, 2022.
J. Y. Park: Pusan National University (Ph.D. Student)
S. I. Lee: Pusan National University (M.S. Student)
I. K. Eom: Pusan National University (Prof.)
※ 본 논문은 부산대학교 기본연구지원사업 (2년)에 의하여 연구되었음.

을 증척시키지 않는 방법 [4] 또는 원형 블록 [5]의 방식을 사용할 수 있다. 특징 추출은 블록 기반의 방법에서 핵심적인 부분이다. 추출된 특징은 크기와 회전에 대하여 불변해야 하고, 잡음, 블러 등 여러 후처리에 대하여 견고해야 한다. 이 단계에서는 극좌표 코사인 변환 [6], 푸리에-멜린 변환 [7], 극복수 지수변환 [8], 푸리에, 이산코사인 [9], 웨이블릿 변환 [10] 등 여러 가지 변환들이 불필요한 정보 제거 및 특징 추출에 사용된다. 특징 매칭은 추출된 특징을 이용하여 원본 부분과 복사-이동 조작된 부분의 후보 쌍을 결정한다. 이 단계에서는 탐색과 유사성 측정 방법들을 이용한다. 탐색에는 다양한 정렬 방법 [10, 11] 및 해싱 프로세스 [12]가 사용된다. 복사-이동 조작의 마지막 단계는 조작영역의 국부화이다. 탐지 결과는 의심되는 영상에서 소스 부분과 조작된 영상 부분이 이진 맵으로 표현된다. 대부분의 블록 기반의 방법은 특징 수가 너무 많고 아핀 변환에 대해 약한 단점이 있다. 또한, 영상의 블록 부분의 값을 이용하기 때문에 블러와 뒤집는 변환에 대해 약한 단점이 있다.

복사-이동 조작에서 키포인트에 관련된 연구는 활발하게 진행되었다. SIFT는 크기 변환과 회전 변환에 강한 키포인트를 뽑아내기 때문에 복사-이동 조작영역 검출에 적합하다 [13-17]. SIFT 기반 복사-이동 조작영역 검출의 첫 번째 단계는 영상 피라미드에 의한 스케일 공간을 통하여 다양한 스케일에서 추출되는 스케일 불변의 키포인트를 생성한다. 두 번째 단계는 키포인트의 회전 불변성을 위하여 키포인트에 방향을 지정하고 주변 픽셀을 128차원의 설명자로 표현한다. 매칭 과정에서 키포인트의 설명자는 다른 모든 키포인트의 설명자와 비교되어 매칭 여부를 결정한다. 하지만 이러한 매칭에서 잘못된 매칭은 항상 발생하기 때문에 이러한 매칭을 제거해주기 위해 J-linkage [18], 거리 기반 [19] 및 계층적 클러스터링 [20]과 같은 다양한 클러스터링 알고리즘이 사용된다. 복사-이동 조작 부분이 작거나 배경 영역을 포함하는 경우 SIFT 기반의 알고리즘은 성능이 저하될 수 있다 [15, 17]. 또한, SIFT 기반의 방식은 키포인트 주변의 광역 정보를 얻는 것은 어렵다. 따라서 복사-이동 조작된 부분과 원본의 경계면에 키포인트가 존재하거나, 영상을 압축하는 경우 성능이 저하될 수 있다.

최근 기계학습은 영상 처리 및 컴퓨터 비전 작업에서 획기적인 성능을 보였다. 따라서 기계학습 기반의 복사-이동 조작 연구도 활발히 이루어졌다. 대표적으로 Wu 등은 Simi-Det 및 Mani-Det을 포함하는 이중 병렬 구조의 CNN 방식 (BusterNet) [21]을 제안했다. Simi-Det은 유사성 감지를 위해 설계되었으며 Mani-Det은 원본과 조작영역 구별을 위해 설계되었다. 그러나 각 BusterNet의 구조가 조작영역을 정확하게 찾지 못하면 BusterNet은 본연의 목적인 원본 지역과 대상 지역을 구분할 수 없다. Zhong 등은 다차원 밀집 특징 연결 네트워크인 Dense-InceptionNet을 사용하는 복사-이동 조작 검출 알고리즘 [22]을 제안했다. Dense-InceptionNet은 피라미드 특징 추출기, 특징 상관 매칭 및 계층적 후처리 모듈로 구성된다. 네트워크가 유사한 역할을 하는 여러 모듈을 사용하여 탐지 정확도를 향상했지

만, 훈련 매개변수의 수가 급격히 증가하여 계산적 비용이 크다는 단점이 있다. Chen 등은 유사성 및 조작 감지를 위해 직렬 구조의 복사-이동 조작 검출 방법 [23]을 제안했다. 이 방법은 VGG16 [24]의 최종 컨볼루션 계층에서 atrous convolution [25]을 사용하였고 계층적 특징 비교를 위해 상관 모듈에서 이중 상관계수를 사용했다. 또한, 조작영역의 위치를 정확하게 검출하기 위한 attention mechanism을 제안하였다. 마지막으로 영상 레벨 네트워크를 사용하여 유사성 탐지 구조에서 얻은 영역이 변조되었는지를 결정했다. 실험 결과는 제안한 알고리즘이 BusterNet보다 우수한 성능을 달성했음을 보여주었다. 그러나 이 알고리즘은 여전히 성능 면에서 개선될 필요가 있다. 최근에 Lee 등은 회전 불변 특성을 가지는 웨이블릿 고주파 계수의 에너지를 이용한 조작 검출 CNN 구조 [26]를 제안하였다. 이 방법은 특히 회전 변환이 추가된 복사-이동 검출 조작에 대해 우수한 성능을 보였다.

본 논문에서는 이중 브랜치 구조를 가지는 복사-이동 조작 검출 네트워크를 제안한다. 제안된 네트워크는 SWT (stationary wavelet transform) 기반의 특징 분리 모듈, CNN을 사용한 특징 추출 모듈, 유사성을 확인하는 상관분석 모듈, 이진 검출 맵의 크기를 증가시키는 디코딩 모듈, 이중 가지 구조의 결과 결합을 위한 결합 모듈 총 5가지 모듈로 구성된다. 제안된 네트워크의 구조는 입력으로 RGB 채널의 색 정보를 활용하는 것보다 웨이블릿 계수를 활용하였기 때문에 크기 조정 및 회전에 더 적합한 입력이라 할 수 있다. 실험 결과를 통하여 제안된 네트워크의 입력 및 구조가 기존 방법들에 비하여 우수한 복사-이동 조작 위치 검출 결과를 나타냄을 보이고자 한다.

논문의 나머지 부분은 다음과 같이 구성된다. 2장에서는 제안 방법의 구조에 대해 설명한다. 3장에서는 제안 방법의 결과를 다른 방법들과 비교하고 4장에서는 결론을 서술한다.

II. 제안 방법

본 논문에서 제안하는 복사-이동 검출 네트워크는 영상의 스케일 정보를 가진 웨이블릿 저주파 밴드의 계수와 회전-불변 정보를 가진 고주파 계수의 에너지 정보를 각각 학습시켜 결합하는 이중 가지 구조를 기반으로 한다. 네트워크 구조는 특징 분리 모듈, 특징 추출 모듈, 상관계수 모듈, 디코딩 모듈로 구성되고 최종적으로 각 구조에서 나온 결과를 합치는 결합 모듈로 구성된다.

그림 2는 제안된 복사-이동 검출 네트워크의 구조를 나타낸다. 특징 분리 모듈은 복사-이동 조작에서 발생할 수 있는 스케일과 회전 변환을 검출하기 위한 것이고, 두 개의 특징 추출 모듈에서 각각의 특징들을 추출한다. 상관분석 모듈은 추출된 특징들의 유사성을 판단하여 복사-이동된 영역의 후보자는 선택한다. 디코딩 모듈은 입력 영상과 같은 크기의 복사-이동 조작 영상의 상세한 정보를 복원하기 위해 사용되며, 마지막으로 결합 모듈에서 회전 변환과 스케

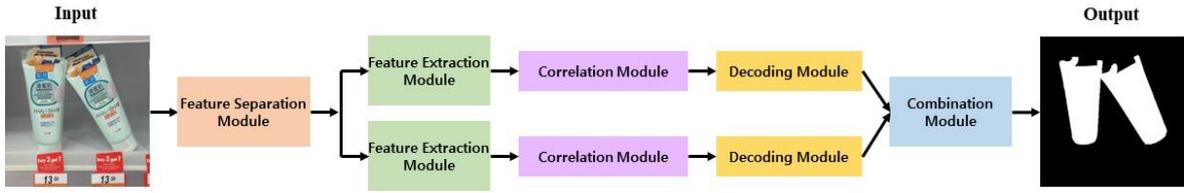


그림 2. 제안된 네트워크 구조
Fig. 2. Proposed network architecture

일 변환이 가해진 조작 영상에 대한 복사-이동 영역을 결합하여 최종적인 검출 결과를 얻는다.

1. 특징 분리 모듈

기존의 CNN은 다양한 크기의 필터를 사용하여 컨볼루션이 수행되기 때문에 다양한 스케일 특징을 추출할 수 있다. 그러나 회전-불변성은 기본 CNN 구조에서 과생될 수 없기 때문에, CNN 회전 불변성을 만들기 위한 다양한 시도가 있었다 [27]. 논문에서는 CNN에 회전 불변성을 제공하기 위해 웨이블릿 변환을 사용하여 스케일 특징을 추출하고, 웨이블릿 계수의 에너지를 이용하여 회전-불변 특징을 추출하는 방법을 제안한다.

어떤 스케일 $i(i=0,1,2,\dots)$ 에서 주어진 저주파 부밴드 영상 $W_{i,LL}$ 입력 영상을 웨이블릿 변환하면 다음과 같다.

$$W_{i+1,o} = SWT(W_{i,LL}). \quad (1)$$

여기서 $SWT(z)$ 는 부밴드 영상 z 의 웨이블릿 변환이고, $o(o=LL, LH, HL, HH)$ 는 부밴드의 방향을 나타낸다. LL 는 저주파 부대역을 나타내고, LH, HL 및 HH 는 각각 수평, 수직 및 대각 방향을 가지는 고주파 부밴드를 나타낸다. $i=0$ 는 입력 영상을 표현한다.

제안된 네트워크에서 사용되는 특징 분리 모듈은 그림 3과 같이 구성된다. 특징을 분리하기 전 먼저, 입력 영상을 256×256 크기로 변환한다. 입력 영상에 대하여 Haar 필터에 기반한 2단계 변환을 통해 두 개의 저주파 부밴드 $W_{1,LL}$ 와 $W_{2,LL}$ 을 생성한다. 저주파 부밴드와 함께 형성된 고주파 부밴드 계수를 이용하여 본 논문에서는 회전-불변 특성을 생성하기 위하여 에너지 특징을 사용한다. 제안된 에너지 특징은 고주파 부밴드 에너지의 평균 제공근으로 다음과 같이 정의된다. 즉,

$$E_i(x,y) = \sqrt{(W_{i,LH}^2(x,y) + W_{i,HL}^2(x,y) + W_{i,HH}^2(x,y))/3}. \quad (2)$$

여기서 $E_i(x,y)$ 는 i -단계 웨이블릿 부밴드의 (x,y) 위치에서의 평균 에너지의 제공근이다. 그리고 $W_{i,LH}^2(x,y)$, $W_{i,HL}^2(x,y)$, 및 $W_{i,HH}^2(x,y)$ 는 각각 (x,y) 위치에서 수평, 수직 및 대각 성분 에너지값을 나타낸다. 회전 변환이 생기더라도 웨이블릿 부밴드의 전체 에너지는 변하지 않으므로 식 (2)의 에너지 특징은 회전 변환에 강인하게 작용할 수 있다.

그림 3은 특징 분리 모듈을 나타낸다. 웨이블릿 통해 얻어

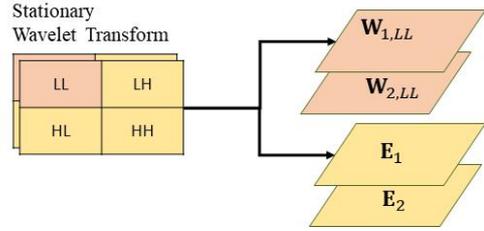


그림 3. 특징 분리 모듈
Fig. 3. Feature separation module

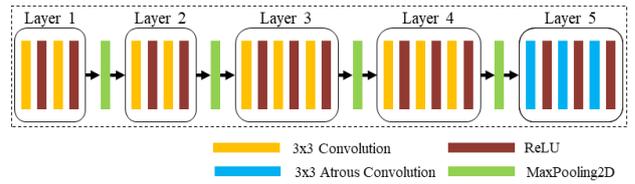


그림 4. 특징 추출 모듈
Fig. 4. Feature extraction module

낸 두 개의 저주파 부밴드 $W_{1,LL}$ 및 $W_{2,LL}$ 은 스케일이 다른 영상을 얻을 수 있어서, 복사-이동 조작에서 복사 영역의 스케일 변화의 특징을 추출할 수 있다. 또한, 고주파 부밴드의 에너지 특징은 회전 변환에 강인하다. 이러한 특징을 활용하여 본 논문에서는 이중 가지를 가지는 네트워크를 구성하여 더욱 강한 복사-이동 조작 검출 결과를 얻고자 한다.

2. 특징 추출 모듈

VGGNet [24]은 적은 학습 파라미터 수와 3×3 컨볼루션 필터만으로도 강력한 성능을 보여준다. 또한, 기존의 CNN 기반 복사-이동 조작 검출에서 VGG16을 이용한 방법들이 사용되었다 [21, 23, 26]. 본 논문 또한 기존 방법과의 성능 비교를 위해 다양한 백본 네트워크 중 VGG16을 특징 추출 모듈 사용하였다. 본 논문에서는 저주파 부밴드 및 웨이블릿 에너지를 입력으로 하는 두 개의 특징 추출 모듈을 그림 2와 같이 병렬로 사용한다. 즉, 하나의 VGG16 특징 추출 모듈에는 웨이블릿 저주파 부밴드 입력 $W_{1,LL}$ 와 $W_{2,LL}$ 가 사용되고, 또 다른 VGG16 특징 추출 모듈에는 E_1 및 E_2 가 입력으로 사용된다.

그림 4는 특징 추출 모듈의 구성을 나타낸다. 각 모듈은 다섯 개의 층으로 이루어져 있으며 3×3 크기의 필터, ReLU 활성화 함수를 교차로 사용하면서 영상의 특징을 넓

은 영역에서 찾아낸다. 또한, 특징 맵의 크기를 줄이고 파라미터의 수를 줄이면서 특징을 추출하기 위해 2×2 크기의 최대 풀링을 이용한다. 모든 층의 구성은 기존의 VGG16 구조와 비슷하나 마지막 층은 수용 영역의 증가를 위해 atrous convolution [25]을 이용해 특징 맵을 분석한다.

3. 상관분석 모듈

두 가지의 특징 추출 모듈에서 얻어낸 스케일 및 회전 변환에 강인한 특징 맵은 복사-이동된 영역의 유사도의 점수를 얻기 위해 상관분석 모듈에 입력된다. 특징 추출 모듈에서 추출된 두 개의 특징 맵을 각각 f_i 및 f_j 벡터로 표현하면, 두 개의 특징 벡터에 대한 상관계수 $\rho(i, j)$ 는 다음과 같다.

$$\rho(i, j) = \tilde{f}_i^T \tilde{f}_j / 512. \quad (3)$$

여기서 \tilde{f}_i 는 정규화된 특징 벡터로서 다음과 같이 계산된다. 즉,

$$\tilde{f}_i = \frac{f_i - \mu(f_i)}{\sigma(f_i)}. \quad (4)$$

여기서 $\mu(f_i)$ 및 $\sigma(f_i)$ 는 f_i 는 각각 f_i 의 평균 및 표준편차이다.

그림 5는 제안된 네트워크의 상관분석 모듈을 나타낸다. 상관분석 모듈은 BusterNet [21]의 구성과 유사하지만 각 채널의 집중하는 CAM (channel attention module) [28]을 사용하는 대신 조작된 화소 위치에 집중하기 위하여 특징 맵 내의 특정 화소에 중요한 정보가 있는지 집중하는 모듈인 SAM (spatial attention module) [28]을 이용하였다. 복사-이동 조작된 영역의 화소 간의 유사 점수를 파악하기 위해 BusterNet [21]에서 이용한 피어슨 상관관계수에 의한 자기 상관도 방법을 이용한다. 유사 점수를 내림차순으로 정렬하고 필요한 정보를 추출하는 percentile pooling을 이용하여 유사 점수가 높은 상위 100개의 특징 맵을 추출한다. 마지막으로 층이 깊어지면서 발생할 문제를 대비하기 위해 배치 정규화를 진행한다.

4. 디코딩 모듈

입력 영상이 특징 추출 모듈과 상관분석 모듈을 거치며 16×16 크기의 낮은 해상도를 가진 복사-이동 조작된 영역에 대한 대략적인 정보를 가진 특징 맵이 생성된다. 입력 영상과 같은 크기의 복사-이동 조작 영상의 상세한 정보를 가진 특징 맵으로 복원하기 위해 디코딩 모듈을 적용한다. 그림 6은 디코딩 모듈을 나타낸다.

영상은 다양한 보간법에 따라 확대될 수 있다. 본 논문에서는 쌍선형 보간법을 이용해 16배의 확대 과정을 거친다. 기존의 방법 [21]과 같이 BN-Inception을 모든 확대 과정에 사용하는 작은 적은 영역에 있는 오류를 확대시키는 경향이 있다. 따라서 본 논문에서는 전체의 확대 과정이 아니라 128×128 과 256×256 크기의 확대 과정에만 BN-Inception을



그림 5. 상관분석 모듈
Fig. 5. Correlation module

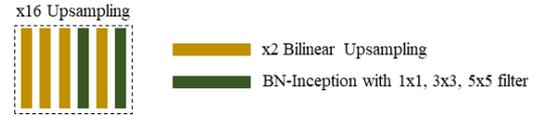


그림 6. 디코딩 모듈
Fig. 6. Decoding module

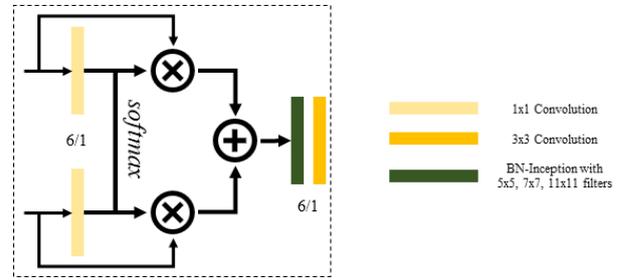


그림 7. 결합 모듈
Fig. 7. Combination module

적용하였다. 이러한 방법은 제대로 검출된 영역이 확대 과정에서 다소 부정확할 수 있지만, 마지막의 결합 모듈에서 개선될 수 있으므로 제안된 디코딩 모듈에서 쌍선형 보간법을 더 많이 사용하였다.

5. 결합 모듈

지금까지의 과정으로 저주파 부밴드를 사용한 스케일 특징과 고주파 에너지를 이용한 회전-불변 특징을 이용하여 복사-이동 조작된 영상의 특징 맵 두 개를 생성하였다. 각각의 가지들은 서로 다른 특징으로 학습되었기 때문에 두 맵을 결합함으로써 두 가지 특징의 장점을 가진 맵을 만들 수 있다.

그림 7은 이를 위한 모듈의 구성을 나타낸다. 그림 7에서 보는 바와 같이, 각 특징 맵의 화소 정보를 유지하고 채널의 수를 줄이기 위해 1×1 필터 한 개로 연산을 진행한다. 생성된 각 특징은 화소 단위의 softmax를 통해 각 가중치를 정한다. 디코딩 모듈에서 얻어낸 특징 맵에 가중치를 화소별로 곱해준 후 BN-Inception을 적용하여 최종적인 복사-이동 영역 이진 맵을 생성한다.

6. 손실 함수

본 논문에서는 학습을 위한 손실 함수로 이진 교차 엔트로피를 다음과 같이 사용한다.

$$L = \frac{1}{m} \sum_{k=1}^m [y_k^t \log(y_k) + (1 - y_k^t) \log(1 - y_k)]. \quad (5)$$

여기서 L 은 손실 값, m 은 학습 영상의 개수, y_k^t 는 k 번째 실제 영상 그리고 y_k 는 k 번째 결과 영상이다.

III. 실험 및 결과

1. 실험 데이터

본 실험에서는 기존의 기계학습 기반 복사-이동 조작 검출 방법 [21, 23, 26]에서 사용되었던 것과 마찬가지로 100,000장의 테스트 영상 (조작 및 원본 각 100,000장씩)을 사용하였다. 이 영상들은 8: 1: 1의 비율로 나누어져 각각 훈련, 검증, 및 테스트에 사용되었다.

2. 실험 환경 및 세부 사항

본 실험은 학습 및 훈련을 TensorFlow v2.4.0 환경에서 진행하였다. 최적화 기법으로는 0.001의 학습률 (learning rate)을 가진 Adam (adaptive moment estimation)을 사용하였다. 배치 크기는 16, 에포크는 100회로 설정하였다. 또한, 제안 방법의 정확한 성능 평가를 위하여 교차 검증을 10회 실시하였다. 학습 환경은 Intel i7-11700K @ 3.60 GHz CPU, NVIDIA GeForce RTX 3080 Ti 12GB GPU, 64GB RAM, Windows 10 OS 기기를 사용하였다.

3. 성능 평가

검출 성능 평가 지표로써 화소 단위에서 Precision, Recall 및 F1-score를 사용하였다. 평가 지표를 계산하기 위한 TP, FP 및 FN은 각각 실제 정답을 정답이라 예측한 횟수, 실제 오답을 정답이라 예측한 횟수 그리고 실제 정답을 오답이라 예측한 횟수를 말한다. Precision은 예측 결과 정답 중 실제 정답의 비율, Recall은 실제 정답 중 예측 결과가 정답인 비율이며, F1-Score는 Precision과 Recall의 조화 평균이다. 위의 세 가지 성능 평가 지표는 다음과 같다.

$$\text{Precision} = \frac{T_p}{T_p + F_p}, \quad (6)$$

$$\text{Recall} = \frac{T_p}{T_p + F_N}, \quad (7)$$

$$\text{F1} = 2 \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (8)$$

4. 실험 결과

제안된 복사-이동 검출 네트워크의 성능을 평가하기 위하여 회전 변환이 없는 테스트 영상 집합, 회전 변환이 있는 영상 집합을 구분하여 결과를 도출하였다. 성능 비교를 위해 세 가지의 SIFT 기반 방법들 즉, SIFT+FM (SIFT and feature matching) [13], SIFT+Seg (SIFT and adaptive segmentation) [16] 및 SIFT+RLBP (SIFT and reduced local binary pattern) [17]을 비교하였다. 또한, BusterNet

표 1. 회전 변환이 없는 테스트 영상 집합에 대한 복사-이동 조작 검출 성능 비교

Table 1. Performance comparison of copy-move forgery detection for test images without rotation attack

Method	Precision	Recall	F1-Score
SIFT+FM [13]	0.105	0.040	0.054
SIFT+Seg [16]	0.249	0.028	0.045
SIFT+RLBP [17]	0.426	0.358	0.370
BusterNet [21]	0.608	0.508	0.511
SeNet [23]	0.725	0.655	0.657
RIWCNN [26]	0.494	0.609	0.507
Proposed	0.534	0.576	0.531

표 2. 회전 변환이 있는 테스트 영상 집합에 대한 복사-이동 조작 검출 성능 비교

Table 2. Performance comparison of copy-move forgery detection for test images with rotation attack

Method	Precision	Recall	F1-Score
SIFT+FM [13]	0.119	0.049	0.067
SIFT+Seg [16]	0.346	0.073	0.108
SIFT+RLBP [17]	0.359	0.321	0.326
BusterNet [21]	0.467	0.481	0.420
SeNet [23]	0.495	0.421	0.412
RIWCNN [26]	0.791	0.770	0.765
Proposed	0.817	0.802	0.801

표 3. 모든 테스트 영상 집합에 대한 복사-이동 조작 검출 성능 비교

Table 3. Performance comparison of copy-move forgery detection for all test images

Method	Precision	Recall	F1-Score
SIFT+FM [13]	0.112	0.045	0.061
SIFT+Seg [16]	0.302	0.053	0.079
SIFT+RLBP [17]	0.389	0.338	0.346
BusterNet [21]	0.531	0.493	0.461
SeNet [23]	0.599	0.527	0.522
RIWCNN [26]	0.657	0.697	0.648
Proposed	0.689	0.700	0.679

[21], SeNet (serialized network) [23]과 RIWCNN (rotation invariant wavelet CNN) [26] 등의 세 가지 기계학습 기반 방법의 결과를 비교하였다.

표 1은 회전 변환이 없는 조작 영상에 대한 성능 평가 지표를 나타낸다. 표 1에서 볼 수 있는 바와 같이, SIFT 기반 방법들은 F1-Score가 모두 0.4 이하로 대체로 낮은 성능을 보인다. SeNet이 가장 높은 F1-Score 0.657을 도출하고 있고, 제안 방법은 두 번째로 높은 F1-Score 0.531을 가진다. 결론적으로 SeNet은 회전 변환이 없는 조작 영상의 검출에 대하여 제안 방법보다 우수한 결과를 보임을 알 수 있다. 그림 8은 회전 변환이 없는 조작 영상의 검출 결과의 예를 나타낸다.

표 2는 회전과 스케일 변환이 동시에 있는 조작 영상에 대한 검출 결과를 나타낸다. 표 2에서 보는 바와 같이, 회전



그림 8. 회전 변환이 없는 테스트 영상에 대한 복사-이동 조작 영역 검출 결과
Fig. 8. Copy-move forgery localization results for test images without rotation attack

변환이 없는 경우 0.657의 F1-Score를 가졌던 SeNet은 회전 변환이 있는 경우 F1-Score가 0.412로 매우 감소하였다. 반면에, 회전 불변성을 고려한 RIWCNN 및 제안 방법은 F1-Score가 각각 0.507 및 0.531에서 0.765 및 0.801로 상당히 증가하였다. SIFT 기반의 세 가지 검출 방법은 회전 변환이 있는 경우 모두 F1-Score가 감소하였음을 알 수 있다. 따라서 복사-이동 조작 검출에 대하여 회전 변환에 강인한 특징을 사용한 방법들은 회전 변환에 상당히 강인한 결과를 보인다는 것을 알 수 있다. 회전 불변에 대한 고려의 반대급부로 스케일 변환만 있을 때, 성능이 떨어지는 단점이 있지만, 일반적으로 복사-이동 조작을 할 경우 회전 변환을 많이 사용하기 때문에, 회전 불변성을 고려하는 것이 전체적인 검출 성능을 높이는데 효과적이라 할 수 있다. 그림 9는 회전 변환과 스케일 변환 조작이 동시에 가해진 경우의 조작 영상에 대한 검출 결과의 예를 나타낸다.

표 3은 전체 테스트 영상에 대한 검출 성능을 종합한 것이다. 표 3에서 볼 수 있는 바와 같이, 제안 방법의 F1-Score가 0.679로 가장 높으며, RIWCNN과 SeNet이 각

각 0.648 및 0.522로 두 번째 및 세 번째로 높은 F1-Score를 가진다. 또한, 제안 방법의 성능은 예측 결과 정답 중 실제 정답의 비율을 표현하는 Precision이 0.689이고, 실제 정답 중 예측 결과가 정답인 비율을 나타내는 Recall 값이 0.700이기 때문에, 다른 모든 방법들에 비하여 좋은 Precision 및 Recall의 균형을 이루고 있다.

IV. 결론

본 논문에서는 이중 브랜치를 가지는 CNN을 이용한 복사-이동 조작 검출 방법을 제안하였다. 복사-이동 조작을 할 때 회전 또는 스케일 변환이 일반적으로 동시에 이루어지기 때문에, 기존의 CNN은 조작을 검출하기에 효과적이지 않다. 따라서 제안 방법은 웨이블릿 계수를 회전에 불변한 부밴드와 스케일 변환에 불변한 특징으로 나누었다. 각각의 특징들을 동일한 구조를 갖는 서로 다른 브랜치에 입력하여, 결합 모듈에서 융합하였다. 각 브랜치는 특징 추출, 상관분석, 디코딩 모듈로 구성되며, 각 브랜치의 최종 디코딩 모듈

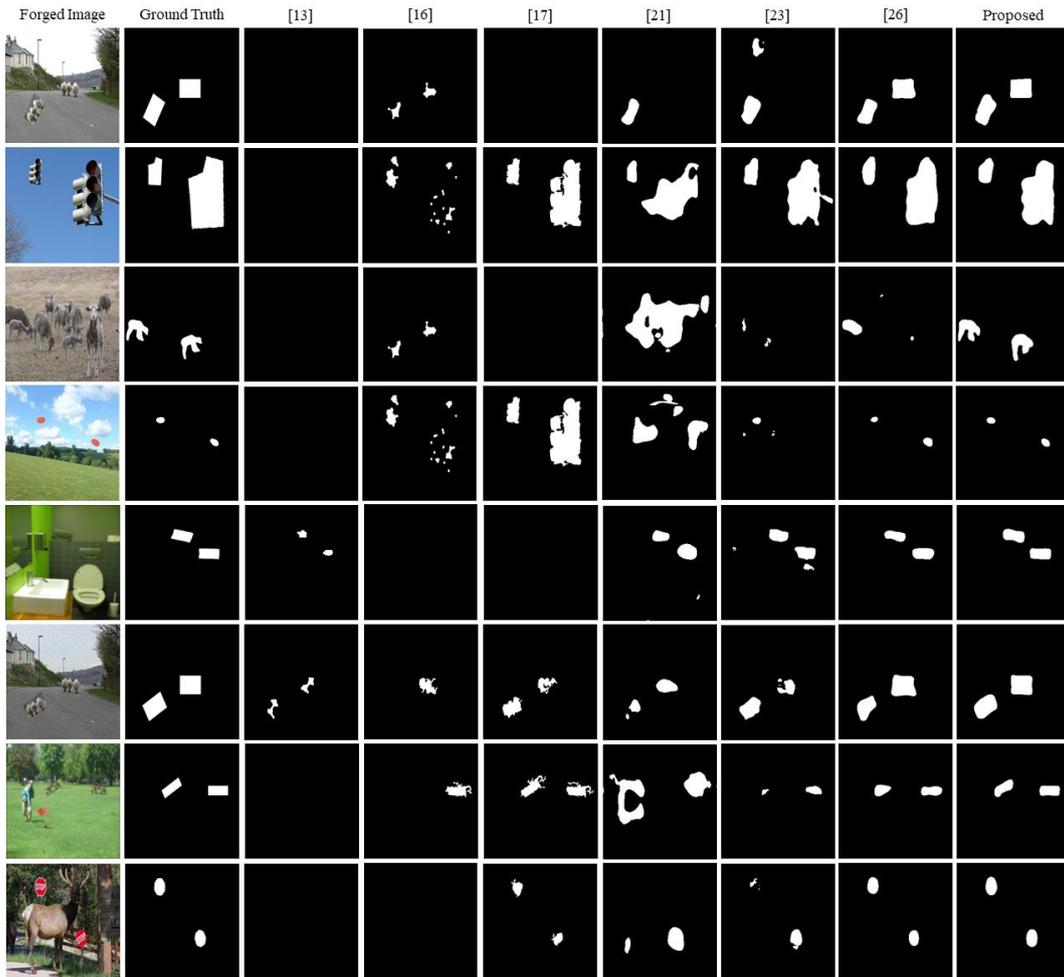


그림 9. 회전 변환이 있는 테스트 영상에 대한 복사-이동 조작 영역 검출 결과
 Fig. 9. Copy-move forgery localization results for test images with rotation attack

의 결과는 결합 모듈에서 융합된다. 융합된 결과로부터 복사-이동 조작된 영역을 추출하였다. 제안 방법의 성능을 평가하기 위하여 기존의 복사-이동 검출 방법들과 비교를 수행하였다. 실험 결과 제안된 네트워크는 회전 변환이 없는 조작 영상에 대하여 두 번째로 높은 F1-Score를 얻었고, 회전 변환이 가해진 조작 영상에 대해서는 가장 높은 성능을 보였다. 종합적으로 전체 테스트 영상에 대하여 가장 높은 Precision, Recall 및 F1-Score 값을 얻었다.

References

[1] S. Teerakanok, T. Uehara, "Copy-move Forgery Detection: A State-of-the-art Technical Review and Analysis," *IEEE Access.*, Vol. 7, pp. 40500-40568, 2019.
 [2] R. Thakur, R. Rohilla, "Recent Advances in Digital Image Manipulation Detection Techniques: A Brief Review," *Forensic Sci. Int.*, Vol. 312, 110311, 2020.
 [3] G. Lynch, F. Y. Shih, H. Y. Liao, "An Efficient Expanding Block Algorithm for Image Copy-move

Forgery Detection," *Inf. Sci.*, Vol. 239, pp. 2253-265, 2013.
 [4] Y. Sun, R. Ni, Y. Zhao, "Nonoverlapping Blocks Based Copy-move Forgery Detection," *Secur. Commun. Netw.*, Vol. 2018, Article ID 1301289, 2018.
 [5] Y. Gao, T. Gao, L. Fan, Q. Yang, "A Robust Detection Algorithm for Copy-move Forgery in Digital Images," *Forensic Sci. Int.*, Vol. 214, No. 1, pp. 33-43, 2011.
 [6] Y. Li, "Image Copy-move Forgery Detection Based on Polar Cosine Transform and Approximate Nearest Neighbor Searching," *Forensic Sci. Int.*, Vol. 224, No. 1-3, pp. 59-67, 2013.
 [7] R. Dixit, R. Nakar, "Copy -move Forgery Detection Utilizing Fourier - Mellin Transform Log-polar Features," *J. Electron. Imaging.*, Vol. 27, No. 2, 023007, 2018.
 [8] K. M. Hosny, H. M. Hamza, N. A. Lashin, "Copy-move Forgery Detection of Duplicated Objects Using Accurate PCET Moments and Morphological Operators," *Imag. Sci. J.*, Vol. 66, No. 6, pp. 330-345, 2018.
 [9] M. H. Alkawas, G. Sulong, T. Sabas, A. Rehman, "Detection of Copy-move Image Forgery Based on

- Discrete Cosine Transform,” *Neural Comput. Appl.*, Vol. 30, pp. 183–192, 2016.
- [10] T. Mahmood, A. Irtaza, Z. Mehmood, M. T. Mahmood, “Copy - move Forgery Detection Through Stationary Wavelets and Local Binary Pattern Variance for Forensic Analysis in Digital Images,” *Forensic Sci. Int.*, Vol. 279, pp. 8–21, 2017.
- [11] M. Zandi, A. Mahmoudi-Aznavah, A. Talebpour, “Iterative Copy-move Forgery Detection Based on a New Interest Point Detector,” *IEEE Trans. Inf. Forensics Secur.*, Vol. 11, No. 11, pp. 2499–2512, 2016.
- [12] C. M. Pun, J. L. Chung, “A Two-stage Localization for Copy-move Forgery Detection,” *Inf. Sci.*, Vol. 463–464, pp. 33–55, 2018.
- [13] X. Pan and S. Lyu, “Region Duplication Detection Using Image Feature Matching,” *IEEE Trans. Inf. Forensics Secur.*, Vol. 5, No. 4, pp. 857–867, Dec. 2010.
- [14] I. Amerini, L. Ballan, R. Caldelli, A. Del Bimbo, G. Serra, “A SIFT-based Forensic Method for Copy - move Attack Detection and Transformation Recovery,” *IEEE Trans. Inf. Forensics Secur.*, Vol. 6, No. 3, pp. 1099–1110, 2011.
- [15] V. Christlein, C. Riess, J. Jordan, C. Riess, E. Angelopoulou, “An Evaluation of Popular Copy-move Forgery Detection Approaches,” *IEEE Trans. Inf. Forensics Secur.*, Vol. 7, No. 6, pp. 1841–1854, 2012.
- [16] C. M. Pun, X. C. Yuan, X. L. Bi, “Image Forgery Detection Using Adaptive Oversegmentation and Feature Point Matching,” *IEEE Trans. Inf. Foren. Secur.*, Vol. 10, No. 8, pp. 1705–1716, 2015.
- [17] J. Y. Park, T. A. Kang, I. K. Eom, “Copy-move Forgery Detection Using Scale Invariant Feature and Reduced Local Binary Pattern Histogram,” *Symmetry*, Vol. 12, No. 4, 492, 2020.
- [18] G. Jin, X. Wan, “An Improved Method for SIFT-based Copy-move Forgery Detection Using Non-maximum Value Suppression and Optimized J-Linkage,” *Signal Process. Image Commun.*, Vol. 57, pp. 113–125, 2017.
- [19] D. M. Uliyan, H. A. Jalab, A. W. A. Wahab, S. Sadeghi, “Image Region Duplication Forgery Detection based on Angular Radial Partitioning and Harris Key-points,” *Symmetry*, Vol. 8, No. 7, pp. 62, 2016.
- [20] N. B. A. Warif, A. W. A. Wahab, M. Y. I. Idris, R. Salleh, F. Othman, “SIFT-symmetry: A Robust Detection Method for Copy-move Forgery with Refection Attack,” *J. Vis. Commun. Image Represent.*, Vol. 46, pp. 219–232, 2017.
- [21] Y. Wu, W. Abd-Almageed, P. Natarajan, “BusterNet: Detecting Copy-move Image Forgery with Source/target Localization,” *Proceedings of the European Conference on Computer Vision, Munich, Germany*, pp. 168 - 184, 2018.
- [22] J. L. Zhong, C. M. Pun, “An End-to-end Dense-InceptionNet for Image Copy-move Forgery Detection,” *IEEE Trans. Inf. Forensics Secur.*, Vol. 15, pp. 2134–2146, 2020.
- [23] B. Chen, W. Tan, G. Coatrieux, Y. Zheng, Y. Q. Shi, “A Serial Image Copy-move Forgery Localization Scheme with Source/target Distinguishment,” *IEEE Trans. Multimed.*, Vol. 23, pp. 3506–3517, 2021.
- [24] K. Simonyan, A. Zisserman, “Very Deep Convolutional Networks for Large-scale Image Recognition,” *Proceedings of 3rd International Conference on Learning Representations*, pp. 1 - 14, 2015.
- [25] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A. L. Yuille, “DeepLab: Semantic Image SSegmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs,” *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 40, No. 4, pp. 834 - 848, 2018.
- [26] S. I. Lee, J. Y. Park, I. K. Eom, “CNN-based Copy-move Forgery Detection Using Rotation-invariant Wavelet Feature,” *IEEE Access*, Vol. 10, pp. 106217–106229, 2022.
- [27] G. Cheng, J. Han, P. Zhou, D. Xu, “Learning Rotation-invariant and Fisher Discriminative Convolutional Neural Networks for Object Detection,” *IEEE Trans. Image Process.*, Vol. 28, No. 1, pp. 265–278, 2019.
- [28] S. Woo, J. Park, J.-Y. Lee, I. S. Kweon, “CBAM: Convolutional Block Attention Module,” *Proceedings of European Conference on Computer Vision*, pp. 3–19, 2018.

Jun Young Park (박 중 영)



2017 Department of Electronics Engineering from Pusan National University (BS.)

2017~Department of Electronics Engineering from Pusan National University (Ph.D.)

Field of Interests: Image enhancement, Computer Vision, Copy-Move Forgery Localization & Machine Learning
Email: jyp9917140@pusan.ac.kr

Sang In Lee (이 상 인)



2021 Department of Electronics Engineering from Pusan National University (BS.)

2021~Department of Electronics Engineering from Pusan National University (MS.)

Field of Interests: Image Processing, Copy-Move Forgery Localization & Machine Learning
Email: silee96@pusan.ac.kr

Il Kyu Eom (엄 일 규)



1990 Department of Electronics Engineering from
Pusan National University (BS.)

1992 Department of Electronics Engineering from
Pusan National University (MS.)

1998 Department of Electronics Engineering from
Pusan National University (Ph.D.)

2006~Department of Electronics Engineering from Pusan National
University (Prof.)

Field of Interests: Image Processing, Computer Vision, Digital
Image Forensic & Machine Learning

Email: ikeom@pusan.ac.kr

논문 2022-17-38

객체 탐지 과업에서의 트랜스포머 기반 모델의 특징점 분석 연구 (A Survey on Vision Transformers for Object Detection Task)

하 정 민, 이 현 종, 엄 정 민, 이 재 구*
(Jungmin Ha, Hyunjong Lee, Jungmin Eom, Jaekoo Lee)

Abstract : Transformers are the most famous deep learning models that has achieved great success in natural language processing and also showed good performance on computer vision. In this survey, we categorized transformer-based models for computer vision, particularly object detection tasks and perform comprehensive comparative experiments to understand the characteristics of each model. Next, we evaluated the models subdivided into standard transformer, with key point attention, and adding attention with coordinates by performance comparison in terms of object detection accuracy and real-time performance. For performance comparison, we used two metrics: frame per second (FPS) and mean average precision (mAP). Finally, we confirmed the trends and relationships related to the detection and real-time performance of objects in several transformer models using various experiments.

Keywords : Object Detection, Transformer, Inductive Bias, Computer Vision, Deep Learning

1. 서론

객체 탐지 (Object Detection) 혹은 검출은 그림 1과 같이 복수 객체의 위치를 찾고 이를 분류 (Classification) 하는 과업이다. 과거의 객체 탐지 기법은 대부분 합성곱 신경망을 사용하여 격자형 데이터인 이미지의 특성을 반영하였다. 따라서 귀납적 편향 (Inductive Bias)이 강하다.

대표적인 합성곱 신경망 기반 객체 탐지 방식은 2-Stage 와 1-Stage 방식이 있다. Faster R-CNN [1]과 같은 2-Stage 방식은 영역 제안 (Region Proposal)과 객체 분류를 따로 진행한다. 특히, 객체 탐지 과업은 실시간성이 필요한 자율 주행, 지능형 공장 등에 사용되기 때문에 실시간성 또한 중요한 요소이다. 따라서 영역 제안과 객체 분류를 결합하여 한 번에 진행하는 YOLO [2]와 같은 1-Stage 방식이 제안 되었다. 1-Stage 방식은 속도가 빠르지만 2-Stage 방식보다 상대적으로 정확성이 낮은 단점이 있다.

하지만, 합성곱 신경망을 사용한 방식들은 구조적으로 지역성 (Locality)에 대한 귀납적 편향이 높은 문제가 존재하였다. 이미지의 지역성처럼 귀납적 편향이란 모델을 학습시키기 위해 가정하는 데이터 특성을 의미한다. 귀납적 편향을 활용하여 이미지 분야에서 좋은 성능을 보여준 합성곱

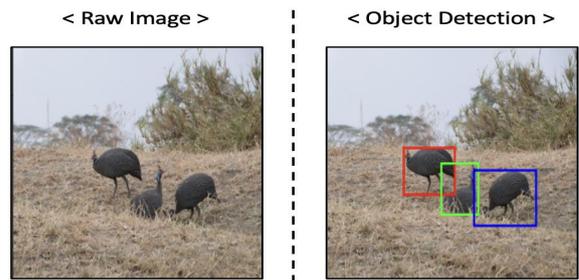


그림 1. 원본 이미지와 객체 탐지 결과
Fig. 1. Original Image and the Result of Object Detection

신경망은 구조적으로 지역성에 대한 유도 편향 (Deductive Bias)이 존재하기 때문에 강한 폐색 (Occlusion)이나 극심한 조명 조건에서 성능이 감소하는 문제가 존재한다. 이를 극복하기 위해 최근에는 귀납적 편향을 최소화하여 성능의 한계를 높인 트랜스포머 기반 모델이 대두되고 있다 [3-5].

본 논문은 객체 탐지 분야에서 트랜스포머 기반 모델의 구조와 특징점을 파악하고, 인식과 실시간 성능을 포괄적으로 비교 분석하였다.

기존 트랜스포머 모델 분류 [3]는 과업에 따른 분류 방법과 다르게 본 논문에서는 그림 2와 같이 객체 탐지 특화된 트랜스포머 모델의 구조적 특징 분류를 제시하였으며, 기본 트랜스포머 (Standard Transformer), with Key Points Attention, Adding Attention with Coordinates와 같이 구조적인 특징으로 분류하고, 각 분류별 특징점들을 분석하였다.

정량적 지표로는 객체 탐지 분야에서 객체 인식 정확도를 측정하기 위해 널리 사용되는 mAP (Mean Average Precision)를 사용하였다. 또한 각 모델의 FPS (Frame Per Sec

*Corresponding Author (jaekoo@kookmin.co.kr)
Received: Oct. 12, 2022, Revised: Nov. 16, 2022, Accepted: Nov. 21, 2022.
J. M. Ha: Kookmin University (M.S. Student)
H. J. Lee: Kookmin University (M.S. Student)
J. M. Eom: Kookmin University (M.S. Student)
J. K. Lee: Kookmin University (Assoc. Prof.)
* 이 논문은 2022년도 정부 (과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No.2021-0-00994, 지속가능하고 견고한 자율주행 인공지능 교육/개발 통합 플랫폼과 No.RS-2022-00167194,미션 크리티컬 시스템을 위한 신뢰 가능한 인공지능).



그림 2. 트랜스포머 기반 객체 탐지 모델 분류
Fig. 2. Taxonomy of Transformer based Object detection models

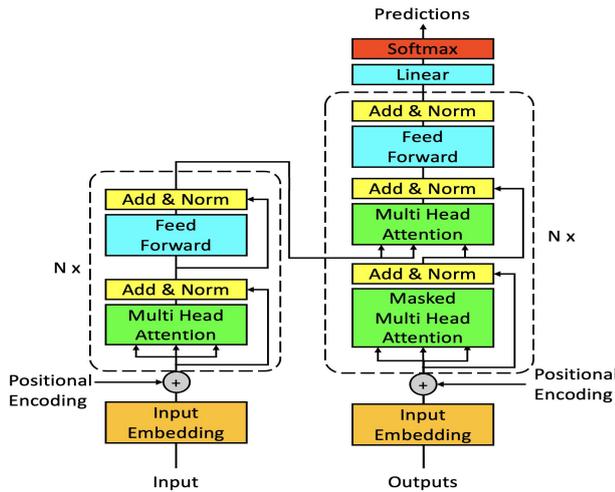


그림 3. 기본 트랜스포머의 구조
Fig. 3. Model Architecture of Basic Transformer

ond)를 통해 실시간 객체 탐지 사용 가능성을 확인했다. 본 논문은 인식 성능을 비교하기 위해 대표적인 객체 인식 공개 데이터인 MS-COCO (MicroSoft-Common Object in Context) [10]와 Pascal VOC [11]를 사용하였다. 또한 같은 실험 환경 조건을 활용하여 정량적 성능을 측정하였다. 그 결과 대부분 모델이 탐지 정확도 측면에서 높은 성능을 보였지만, 실시간성을 보장하기 어려운 경향을 확인할 수 있었다.

II. 모델 설명

본 논문에 사용된 모델들은 트랜스포머 기반 객체 탐지 모델이다. 일반적인 트랜스포머는 어텐션 (Attention) [12] 기반으로 모델이 확장되는 구조이다. 어텐션은 문장 전체에서 핵심 부분을 추상화하며 특징을 학습하기 위해 개발되었다. 트랜스포머는 어텐션을 사용해 전체 문장의 문맥을 잘 파악하여 자연어 처리 분야에서 좋은 성능을 내었다. 그래서 NLP (Natural Language Processing) 분야에서 최근 개발된 모델들은 대부분 트랜스포머 구조로 되어 있다 [13, 14]. NLP 분야 이외에 최근 트랜스포머 구조가 이미지 분야에서 처음으로 사용된 모델이 ViT (Vision Transformer) [4]이다. ViT는 그림 3의 트랜스포머 구조를 이미지 분류

과업에 적용해 좋은 성능을 보여준 모델이다. ViT에서는 이미지를 작은 패치들로 나누어 서열 형태로 처리하는 방법을 제시하였다. ViT는 컴퓨터 비전 분야에서 트랜스포머 구조가 사용될 수 있다는 가능성을 제시했으며 이후 개발된 비전 트랜스포머 모델들에게 큰 영향을 주었다 [3, 5, 6]. 본 논문에서 사용한 모델들은 기본 트랜스포머를 사용하는 Standard Transformers, 여러 해상도의 특징맵 (Feature Map)에서 특징 핵심점들 (Key Points)만 어텐션을 적용한 Transformers with Key Points Attention, 객체의 위치를 임베딩하여 학습에 사용하는 Transformer Adding Attention with Coordinates로 분류하였다.

1. 기본 트랜스포머 (Standard Transformers)

기본 트랜스포머는 인코더만 사용하는 모델과 인코더와 디코더 모두를 사용하는 모델이 있다. 두 방법 모두 ViT의 기본 인코더에 기반하고 있어 ViT처럼 이미지 전체에 대한 문맥을 잘 파악할 수 있다는 장점이 있다.

1.1. DETR (DEtection TRansformer)¹⁾ [5]

DETR은 자연어 처리 분야의 트랜스포머 구조를 객체 탐지 과업에 특화하여 기존의 합성곱 모델들보다 좋은 성능을 보여준 대표적인 모델이다. 또한, 객체 탐지 과업에서 합성곱 신경망이 주를 이룬 패러다임 (Paradigm)을 바꾼 초기 모델로 이후 DETR로부터 파생된 Deformable DETR [7], Sparse DETR [8], Conditional DETR [9]과 같은 객체 탐지 모델들이 연구 개발되었다.

DETR의 큰 특징은 그림 4에서 보이듯 객체 쿼리들 (Object Queries)로 객체를 탐지하는 것이다. 하나의 객체 쿼리에 하나의 객체가 이분 매칭 (Bipartite Matching)되어 학습하게 된다. 이미지 추론 시에는 하나의 객체 쿼리가 하나의 객체를 탐지하게 된다. 이분 매칭 기법을 통해 기존에 합성곱 신경망 모델에서 사용되었던 Anchor Box와 NMS (Non Maximum Suppression)를 대체하여 연산량을 줄였다.

그리고 별도의 ‘객체 없음’ 클래스를 생성하여 객체 쿼리의 개수를 충분히 크게 설정한다. 하지만 이런 방식은 객체 쿼리가 ‘객체 없음’ 클래스에 연결되는 비율을 증가시켜 클래스 불균형을 발생시킨다. 따라서 객체 쿼리가 ‘객체 없음’에 연결되면 역전파되는 값에 0.1을 곱하는 방법으로 클래스 불균형을 해소한다.

1) <https://github.com/facebookresearch/detr>

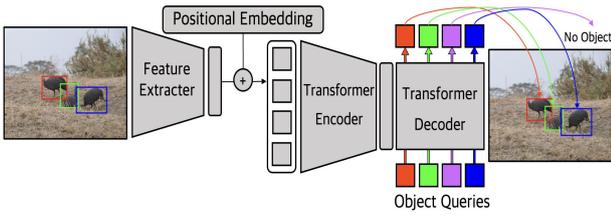


그림 4. DETR의 구조
Fig. 4. Overview of DETR

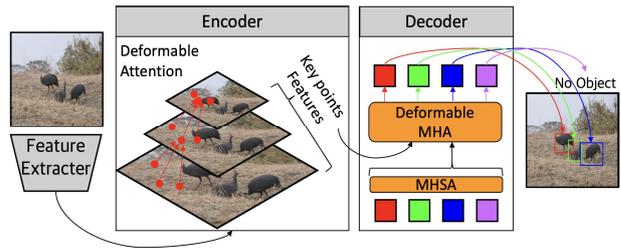


그림 6. Deformable DETR의 구조
Fig. 6. Overview of Deformable DETR

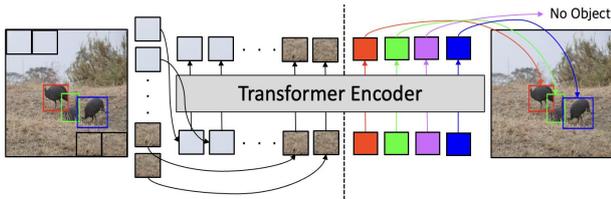


그림 5. YOLOS의 구조
Fig. 5. Overview of YOLOS

1.2. YOLOS (You Only Look One Sequence)²⁾ [6]

트랜스포머는 자연어 처리 분야에서 지식 전달을 목적으로 개발되었다. 자연어 처리 분야에서는 방대한 말뭉치에서 사전학습 (Pretraining)한 트랜스포머 모델을 사용하고자 하는 과업에 맞춰 미세조정 (fine-tuning)을 수행한다. 그리고 이미지 분류 분야에서도 방대한 이미지 데이터를 통해 학습한 ViT를 과업 전이하여 사용한다.

그러나 객체 탐지 과업은 이미지 분류보다 어려운 과업이며, 트랜스포머 기반의 객체 탐지 모델인 DETR은 ViT와 다른 구조를 갖는다. 따라서 ViT의 구조를 그대로 가져오며 ViT가 학습한 지식을 이용해서 객체 탐지를 할 수 있는 가능성을 확인하기 위해 YOLOS가 제안되었다.

YOLOS는 ViT 구조를 객체 탐지 과업으로 확장한 모델이다. 따라서 그림 5에서 보이는 것과 같이 ViT에서 사용한 트랜스포머 인코더를 사용했다. ViT에서 분류 예측을 위해 사용한 [CLS] 토큰을 없애고 DETR의 객체 쿼리의 역할을 하는 [DET] 토큰을 생성하여 트랜스포머 인코더의 입력으로 넣어준다. 트랜스포머 인코더를 통과한 [DET] 토큰들을 각자 하나의 객체에 연결해주어 학습을 진행한다. 그 결과 ViT의 지식을 활용하여 성공적으로 객체 탐지를 할 수 있다는 것을 확인하였다.

2. Transformers with Key Points Attention

Key Points Attention은 어텐션을 계산할 때 여러 해상도의 특징맵에서 몇 개의 핵심 특징점으로 어텐션 연산을 하는 것이다. 이런 방법은 초기 어텐션 가중치가 고르게 분포하는 것을 방지하여 학습 수렴 속도가 향상하는 장점이 있다.

2.1. Deformable DETR³⁾

DETR의 단점은 처음에 어텐션 가중치들 (Attention Weights)

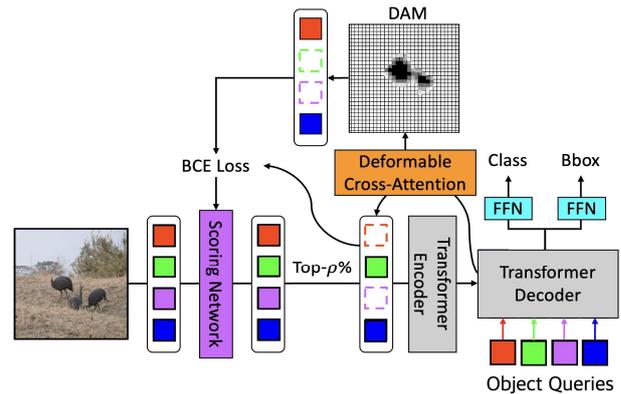


그림 7. Sparse DETR의 구조
Fig. 7. Overview of Sparse DETR

ghts)이 모든 픽셀의 평균값을 갖고 진행되기 때문에 학습의 수렴 속도가 느리다는 것이다. 또한 단일 해상도 특징맵을 사용하기 때문에 작은 물체에 대한 탐지 성능이 좋지 않다. Deformable DETR은 이런 문제를 해결하기 위해 Multi Scale Deformable Attention을 제안했다.

Multi Scale Deformable Attention은 백본망을 통해 여러 해상도의 특징맵을 추출한다. 추출된 특징맵들에서 몇 개의 핵심점들을 선택한다. 그 후 그림 6과 같이 선택된 핵심점들을 사용해서 어텐션 연산을 수행한다.

여러 해상도에서 어텐션을 연산하기 때문에 연산량이 증가하는 단점이 있다. 그러나 학습 초기에 어텐션 가중치 값이 고르게 분포하는 것을 방지해 학습 수렴 속도를 높였다. 또한 여러 해상도의 특징맵에서 객체에 대한 정보를 활용해 작은 물체에 대한 탐지 능력이 향상되는 경향을 보였다.

2.2. Sparse DETR⁴⁾

기존의 DETR 계열의 객체 탐지 모델들은 객체 쿼리를 사용하여 객체를 탐지했다. 그러나 이 과정에서 각 객체 쿼리들이 객체 탐지를 위해 더 비중 있게 참조하는 패치 토큰이 존재함을 발견했다. 또한 객체 쿼리가 참조하지 않는 패치 토큰을 제외하고 학습시켰을 때 전체 패치 토큰을 사용하여 학습했을 때와 성능 차이가 나지 않음을 확인하였다.

Sparse DETR에서는 인코더의 패치 토큰 중 어느 토큰

2) <https://github.com/hustvl/YOLOS>

3) <https://github.com/fundamentalvision/Deformable-DETR>

4) <https://github.com/kakaobrain/sparse-detr>

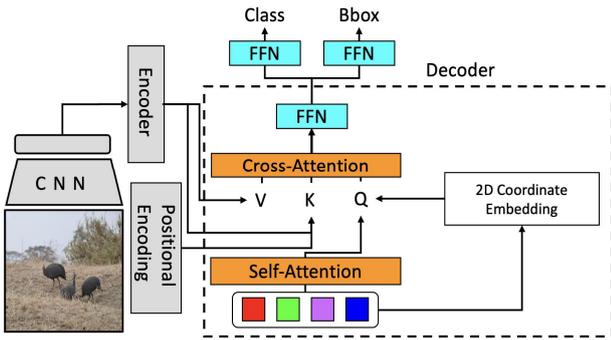


그림 8. Conditional DETR의 구조
Fig. 8. Overview of Conditional DETR

을 디코더에서 참조하여 학습에 사용할지 판단하는 Scoring Network를 사용하였다. 패치 토큰에 대한 참조 점수를 구하고 전체 패치 중 비율을 통해 참조 점수가 높은 상위 토큰들을 결정한 후 선택된 토큰만 사용하여 탐지하는 방법을 제안했다. 또한 학습 시 그림 7과 같이 만들어지는 어텐션 맵들을 더해서 학습에 영향을 주는 패치들을 나타내는 DAM (Decoder cross-Attention Map)을 생성한다. 이후 Scoring Network에서 생성한 참조 점수들과 BCE (Binary Cross Entropy) 손실함수를 사용해 Scoring Network를 학습한다.

그 결과 일부 패치만을 사용하기 때문에 기본 DETR보다 10배 빠른 학습 수렴 속도와 성능 향상을 보여주었다.

3. Transformers Adding Attention with Coordinates

Add Coordinates는 객체의 좌표를 임베딩하여 어텐션의 입력에 더해주는 것이다. 좌표 정보를 더해주면 정확한 바운딩 박스 영역을 예측하여 객체를 탐지하며 학습 수렴 속도가 향상하는 장점이 있다.

3.1. Conditional DETR⁵⁾

DETR은 학습이 느리게 수렴하는 문제점이 있다. 이는 처음 어텐션 맵의 값이 모두 비슷하게 계산되어 특정 부분에 집중하도록 학습되기까지 오래 걸리기 때문이다. 따라서 Deformable DETR에서는 어텐션 맵을 계산할 때 특정 핵심 점들을 이용했고 Sparse DETR은 참조하는 패치를 선택적으로 결정했다. 그러나 Conditional DETR은 객체가 존재할 수 있는 Anchor Points를 이용하는 방법을 사용했다.

객체 좌표를 전달하기 위해 그림 8과 같은 구조를 사용했다. 먼저, 객체 쿼리를 통해 학습 가능한 Anchor Points를 생성한다. 또한 이전 디코더를 통해 생성된 출력값을 완전

연결층을 이용하여 임베딩한 후 Anchor Points와 곱해준다. 이렇게 생성된 좌표 임베딩 값은 트랜스포머에서 Cross Attention을 계산할 때 입력에 더해준다. 객체의 좌푯값을 전달하는 방법을 통해 기존의 DETR보다 학습 수렴 속도를 약 6.7배 빠르게 향상했다. 또한 객체의 위치에 대한 정보를 제공해 탐지 성능이 향상했다.

III. 데이터 집합

1. MS-COCO

MS-COCO는 객체 탐지 과업에서 널리 사용되는 벤치마크 (Benchmark) 데이터 집합이다. 총 80개의 분류가 있으며, 사람, 버스, 자전거와 같은 일상에서 흔히 볼 수 있는 객체들을 포함하고 있다.

표 1과 같이 MS-COCO의 학습 이미지는 118,287개이며 검증 이미지와 테스트 이미지는 각각 5,000개, 40,670개이다.

2. Pascal VOC

Pascal VOC는 객체 탐지 과업의 벤치마크 데이터 집합이며 20개의 분류가 있다. 데이터는 2007년 제공된 2501개의 학습 이미지, 2510개의 검증 이미지, 4952개의 테스트 이미지와 2012년에 제공된 5717개의 학습 이미지, 5823개의 검증 이미지, 10,991개의 테스트 이미지가 있다.

본 논문에서는 2007년과 2012년의 학습 이미지를 합쳐 학습을 진행했다. 또한 테스트 이미지에 대한 주석이 없기 때문에 2007년과 2012년 검증 이미지를 통해 검증을 진행했다.

IV. 평가 지표

1. mAP (mean Average Precision)

본 논문에서는 모델의 평가 지표로 객체 탐지 과업에서 널리 사용되는 mAP를 사용하였다. mAP는 각 클래스의 AP 값들의 평균을 낸 값을 의미한다. 모델의 성능을 평가할 때 정밀도 (Precision)와 재현율 (Recall)을 이용한다. 한 클래스의 정밀도는 $\frac{TP}{TP+FP}$, 재현율은 $\frac{TP}{TP+FN}$ 를 통해 구한다.

객체를 탐지할 때 검출이 되었으면 Positive, 검출되지 않았으면 Negative로 구분한다. 또한 정답값과 검출 결과와

표 1. 실험에 사용된 데이터 집합의 상세 정보
Table 1. Details of used Datasets in Experiments

Dataset	Classes	Train Data		Validation Data		Test Data
		Images	Instances	Images	Instances	Images
MS-COCO	80	118,287	860,001	5,000	36,781	40,670
Pascal VOC	20	8,218	23,618	8,333	23,605	15,943

5) <https://github.com/Atten4Vis/ConditionalDETR>

표 2. COCO 검증 데이터집합에서 실험 결과
Table 2. Experimental Results on COCO Dataset

Model	Metric	mAP	mAP_50	mAP_75	mAP_s	mAP_m	mAP_l	FPS
DETR		0.421	0.623	0.442	0.214	0.460	0.611	29.1
YOLOS-Tiny		0.289	0.474	0.293	0.100	0.296	0.460	124.8
YOLOS-Small		0.361	0.565	0.371	0.154	0.386	0.562	11.9
YOLOS-Small (dwr)		0.375	0.575	0.391	0.158	0.401	0.573	13.5
YOLOS-Base		0.420	0.622	0.445	0.195	0.453	0.622	4.96
Deformable DETR (Single Scale)		0.394	0.597	0.422	0.207	0.429	0.558	27.0
Deformable DETR (Single Scale dc5)		0.414	0.618	0.449	0.237	0.453	0.560	23.1
Deformable DETR (base)		0.445	0.635	0.487	0.268	0.477	0.595	18.5
Deformable DETR (Refine)		0.463	0.650	0.501	0.285	0.492	0.615	18.0
Deformable DETR (Refine Two-Stage)		0.469	0.657	0.511	0.296	0.503	0.616	17.2
Sparse DETR (Swin Transformer $\rho=10\%$)		0.483	0.693	0.525	0.300	0.514	0.642	18.0
Sparse DETR (Swin Transformer $\rho=20\%$)		0.488	0.693	0.530	0.308	0.519	0.646	17.6
Sparse DETR (Swin Transformer $\rho=30\%$)		0.492	0.695	0.535	0.313	0.526	0.651	17.0
Sparse DETR (RetinaNet $\rho=10\%$)		0.455	0.659	0.464	0.286	0.482	0.601	18.2
Sparse DETR (RetinaNet $\rho=20\%$)		0.457	0.658	0.496	0.287	0.486	0.604	17.7
Sparse DETR (RetinaNet $\rho=30\%$)		0.461	0.661	0.498	0.291	0.492	0.608	17.3
Conditional DETR		0.409	0.619	0.434	0.207	0.442	0.595	26.0
Conditional DETR (dc5)		0.437	0.644	0.467	0.238	0.476	0.602	12.9
Average		0.430	0.630	0.460	0.241	0.460	0.600	24.0

같은 True, 다르다면 False로 구분한다. 모델이 객체를 제대로 검출하면 TP (True Positive), 모델이 객체를 오검출했을 때 FP (False Positive), 모델이 검출했어야 하는 객체를 검출하지 못했을 때 FN (False Negative)이라고 한다.

재현율과 정밀도는 일반적으로 반비례 관계를 갖는다. 그래서 모델의 전체적인 성능 분석을 위해 x축을 재현율, y축을 정밀도로 설정한 PR (Precision-Recall) 곡선을 사용한다.

2. FPS (Frame Per Second)

FPS는 초당 프레임 (Frame)을 얼마나 처리할 수 있는지 나타내는 지표로 모델의 추론 시간을 정량적으로 나타낼 때 사용하는 평가 지표이다. 객체 탐지는 실시간에서 사용되는 경우가 많으므로 모델의 정확성과 더불어 중요한 지표이다.

V. 실험 결과

실험에서 사용한 데이터 집합은 COCO 데이터 집합과 Pascal VOC 데이터 집합이며 GPU는 NVIDIA RTX A6000 1개를 사용하여 성능을 측정하였다. 표 2의 결과를 보면 모델의 용량이 커짐에 따라 객체의 특징을 더 잘 학습하기 때문에 mAP가 증가했다. 그러나 모델의 용량이 증가하면 연산량 또한 증가하기 때문에 FPS는 낮아짐을 확인하였다.

각 모델은 성능이 수렴할 때까지 학습을 진행하였기 때문에 모델마다 학습한 에폭 (epoch)이 다르다. 또한 백본망을 사용하는 모델들은 공통적으로 ResNet50 [15]을 사용했다.

이를 통해 mAP와 FPS가 상충관계에 있다는 것을 알 수 있다. 모델의 용량이 커지면 모델의 일반화 성능이 좋아지지만 연산량이 증가하여 FPS는 감소하기 때문이다. 이를 통해 트랜스포머 구조를 사용한 객체 탐지 모델들의 mAP와 FPS 성능 사이에 상충 관계가 있음을 알 수 있다. 따라서 효율적인 성능을 찾기 위해 최적의 mAP와 FPS를 갖는 모델의 용량을 찾는 것이 중요하다.

1. COCO 데이터 집합

DETR은 DETR 계열 모델의 기본이 되는 모델이기 때문에 성능은 표 2의 평균에 비해 높지 않은 것을 볼 수 있다. 그러나 성능을 높이기 위해 추가적인 모듈 (Module)을 더한 다른 모델들과 비교했을 때 FPS가 높게 측정된 것을 확인할 수 있었다.

표 2의 YOLOS 모델들 중 Tiny 모델의 FPS가 124.8로 가장 높게 측정되었다. 이는 YOLOS-Tiny가 ViT Tiny 모델에 [DET]토큰만 추가한 모델이기 때문에 연산량이 크지 않아 나타난 결과이다. 그러나 Small, Small (dwr), Base 모델은 연산량이 많이 늘어나기 때문에 FPS가 급격히 감소하는 것을 확인할 수 있다.

또한 Small 모델에는 Depth, Width, Resolution을 조절하는 최적화 기법을 적용했다 [16]. 그 결과 Small (dwr) 모델의 성능이 Small 모델보다 mAP가 0.014 향상되었다.

Deformable DETR에서는 Deformable Attention을 사용해 기본 DETR보다 성능을 향상했다. 이를 통해 다양한 해상도를 사용하는 것이 성능 향상에 중요한 요소라는 것을 확인할

표 3. Pascal VOC 검증 데이터집합에서 실험 결과
Table 3. Experimental Results on Pascal VOC Dataset

Model	Metric	mAP	mAP_50	mAP_75	mAP_s	mAP_m	mAP_l	FPS
	DETR		0.470	0.725	0.496	0.058	0.315	0.621
YOLOS-Tiny		0.260	0.460	0.259	0.007	0.075	0.387	141.2
YOLOS-Small		0.383	0.608	0.396	0.027	0.165	0.531	12.0
YOLOS-Small (dwr)		0.385	0.608	0.398	0.033	0.179	0.532	18.9
YOLOS-Base		0.424	0.662	0.445	0.049	0.220	0.585	5.7
Deformable DETR (Single Scale)		0.558	0.802	0.614	0.192	0.438	0.668	29.0
Deformable DETR (Single Scale dc5)		0.569	0.810	0.629	0.225	0.452	0.671	14.8
Deformable DETR (base)		0.586	0.814	0.653	0.228	0.469	0.693	18.7
Deformable DETR (Refine)		0.603	0.824	0.665	0.240	0.474	0.707	24.4
Deformable DETR (Refine Two-Stage)		0.605	0.830	0.669	0.252	0.482	0.713	22.8
Sparse DETR (Swin Transformer $\rho=10\%$)		0.520	0.788	0.562	0.154	0.377	0.629	20.5
Sparse DETR (Swin Transformer $\rho=20\%$)		0.529	0.794	0.570	0.159	0.377	0.637	20.5
Sparse DETR (Swin Transformer $\rho=30\%$)		0.533	0.796	0.578	0.173	0.384	0.638	19.7
Sparse DETR (RetinaNet $\rho=10\%$)		0.587	0.825	0.647	0.256	0.476	0.691	23.4
Sparse DETR (RetinaNet $\rho=20\%$)		0.589	0.828	0.655	0.250	0.474	0.693	23.1
Sparse DETR (RetinaNet $\rho=30\%$)		0.590	0.831	0.648	0.257	0.481	0.697	22.8
Conditional DETR		0.562	0.795	0.617	0.154	0.409	0.682	28.0
Conditional DETR (dc5)		0.579	0.802	0.634	0.186	0.426	0.692	14.9
Average		0.540	0.756	0.563	0.161	0.370	0.637	27.3

수 있다. 그러나 여러 해상도에서 어텐션을 진행하기 때문에 기본 어텐션보다 연산량이 증가하여 FPS가 낮게 측정된다.

Deformable DETR은 이전 디코더 층에서 예측한 바운딩 박스를 앵커박스 (Anchor Box)로 사용하는 Iterative Refinement 방법을 사용하여 성능을 향상했다. 또한 인코더에서 영역 제안을 사용하는 Two-Stage Deformable DETR을 통해 성능을 향상했다.

Sparse DETR은 임베딩된 패치 중 객체 쿼리에서 주로 참조하는 이미지 패치만을 사용해 Deformable DETR에서 성능을 향상했다. 그 중 Swin Transformer [17]를 백본망으로 사용하고 상위 30%의 패치만 참조하는 모델이 mAP 0.492로 전체 모델 중 가장 좋은 성능을 보였다. 또한 백본망으로 Swin Transformer를 사용하는 모델들이 백본망으로 RetinaNet을 사용하는 모델들보다 좋은 성능을 보였다. 이는 Swin Transformer가 RetinaNet [18]보다 이미지의 특징을 추출하는 백본망의 역할을 잘 수행하는 것을 의미한다. 또한 이미지의 패치 중 일부 패치만을 참조하여 Deformable DETR에 비해 연산량을 감소시켰다.

표 2에서 전체적으로 참조하는 패치의 개수가 많아질수록 FPS가 감소하는 경향을 보였다. 이는 사용하는 패치의 개수가 많아지는 만큼 연산량이 증가하기 때문에 나타나는 결과이다.

Conditional DETR은 객체의 좌표에 대한 정보를 제공하여 기본 DETR보다 성능이 향상되었다. 그러나 좌표에 대한 정보만 추가로 제공해주기 때문에 FPS는 기본 DETR과 비슷

하게 측정되었다. 또한, IoU (Intersection over Union)가 0.5 이상, 0.75 이상인 객체에 대한 성능을 측정하는 mAP_50과 mAP_75의 성능도 차이가 없었다.

2. Pascal VOC 데이터 집합

표 3의 결과를 보면 대체로 성능이 높게 나온 것을 볼 수 있으며 FPS 또한 높게 측정된 것을 확인할 수 있다.

Pascal VOC는 한 이미지 안에 많은 객체가 존재하지 않기 때문에 COCO 데이터 집합과 비교했을 때 평균적으로 더 높은 성능을 보여준다. 또한 이미지의 크기가 비교적 작기 때문에 이미지 크기에 영향을 받는 연산량이 감소하여 FPS가 높게 측정되었다.

또한 표 3의 작은 객체에 대한 성능을 측정하는 mAP_s를 보면 단일 해상도의 특징 맵만을 사용하는 모델이 대체로 성능이 낮은 것을 볼 수 있다.

그리고 특징으로 볼 수 있는 것은 트랜스포머의 인코더만 사용하는 YOLOS는 COCO 데이터 집합에서의 성능과 비슷하거나 더 낮은 성능을 보였다. 이는 Pascal VOC 이미지의 크기가 작아서 패치를 나눌 때 동일한 개수로 나누면 하나의 패치가 이미지에서 차지하는 영역이 넓어지게 된다. 따라서 이미지에 대한 세밀한 예측 (Dense Prediction)이 어려워진다.

또한 YOLOS는 [DET]토콘과 함께 Self Attention 연산이 이뤄지기 때문에 이미지 내부에서 객체를 분리해지지 못하게 된다. 그러나 인코더-디코더의 형태로 사용되는 모델은 인코더에서 추출한 이미지 전체 정보와 객체 쿼리를 통해 각 객

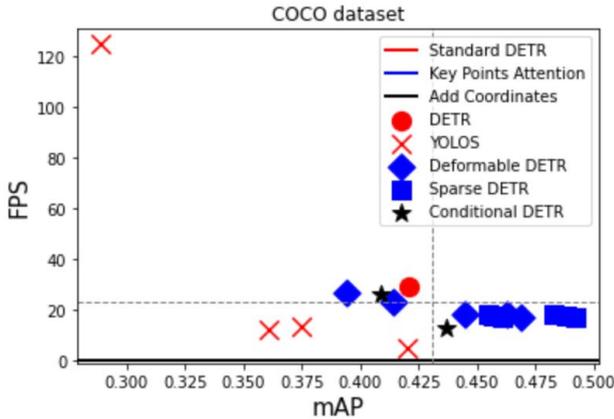


그림 9. 모델별 mAP, FPS의 관계 산점도

Fig. 9. mAP and FPS of various Vision Transformer Models for Object Detection

체 사이를 분리시켜주는 역할을 한다. 따라서 인코더만 사용하는 모델보다 세밀한 예측에서 좋은 성능을 보여준다.

표 3의 결과에서 Deformable DETR이 대체로 가장 높은 성능을 보여주었다. 이는 이미지의 크기가 작기 때문에 여러 해상도의 특징 맵을 활용하며 이미지 패치들을 모두 사용하기 때문이다. 반면 SparseDETR은 이미지의 패치 중 중요한 Top- p %를 사용한다. 그러나 이미지 크기가 작은 Pascal VOC 데이터 집합에서는 역설적으로 정보의 손실이 발생한다. 따라서 Deformable DETR이 가장 높은 성능을 보여준다.

VI. 논의 및 결론

본 논문에서는 트랜스포머 기반의 객체 탐지 모델들을 특징에 따라 분류하고 같은 데이터를 사용하여 mAP와 FPS를 측정하였다. 이를 통해 모델의 구조와 실시간 탐지 가능성의 관계에 대해 분석했다.

그림 9의 분포는 mAP와 FPS를 축으로 하는 모델들의 산점도이다. 그림 9의 점선은 각 축에서 실험에 사용된 전

체 모델의 평균을 의미한다. 그 결과 mAP와 FPS가 모두 평균 이상인 모델은 존재하지 않았다. 또한 DETR은 mAP, FPS 모두 평균에 가깝게 측정되었다.

기본 DETR에서 큰 변형을 주지 않고 객체의 좌표에 대한 정보만을 더해준 Conditional DETR도 평균에 가깝게 측정되었다. 또한 대부분 모델이 그래프의 오른쪽 아래에 있는 것을 볼 수 있다. 이는 기존의 DETR의 성능 향상을 위해 연산량을 고려하지 않았기 때문에 발생한 결과이다.

최근에는 모델의 성능 향상을 위해 여러 해상도를 사용하거나 핵심점들로 어텐션을 계산하는 방법을 사용한다. 또한 YOLOS의 dwr 최적화 기법, Deformable DETR의 Iterative Refinement와 Two-Stage 기법을 통해 성능 향상을 이루었다. 그 중 핵심점들로 어텐션을 사용한 모델들이 동일한 FPS에서 높은 mAP를 달성하는 경향을 확인했다.

그림 10은 GPU의 성능을 측정하기 위해 사용되는 GFLOPS (Giga Floating point Operations Per Second)를 통해 각 모델들의 연산량을 측정한 결과이다. 표 2와 표 3에서 가장 낮은 FPS를 갖는 YOLOS-base가 가장 높은 연산량을 갖고 FPS가 가장 낮은 YOLOS-tiny가 가장 낮은 연산량을

갖는 것을 볼 수 있다. 그러나 반대로 DETR과 Deformable DETR Single Scale을 비교했을 때는 연산량이 높은 모델이 FPS가 더 높게 나온 것을 확인할 수 있다. Deformable DETR Single Scale은 DETR에 Deformable Attention Module만 더해준 것이다. 따라서 연산량 자체는 낮게 나오지만 FPS가 느린 원인이 Deformable Attention Module의 최적화가 잘 이루어져있지 않아 발생하는 문제로 볼 수 있다. 위와 같은 결과를 통해 GFLOPS와 FPS 사이에 대략적인 상충관계를 갖지만 추가적인 모듈로 인한 FPS 저하가 있을 수 있다는 것을 알 수 있다.

그림 9를 통해 대체로 트랜스포머 구조의 모델들이 mAP와 FPS 사이에 상충관계가 있는 것을 확인했다. 그러나 정량적인 지표를 통해 어떤 상관관계가 있는지 확인하기 위해 FPS와 mAP사이의 피어슨 (Pearson) 상관계수를 계산하였다. 또한 연산량과 모델의 FPS와의 연관성을 확인하기 위해 m

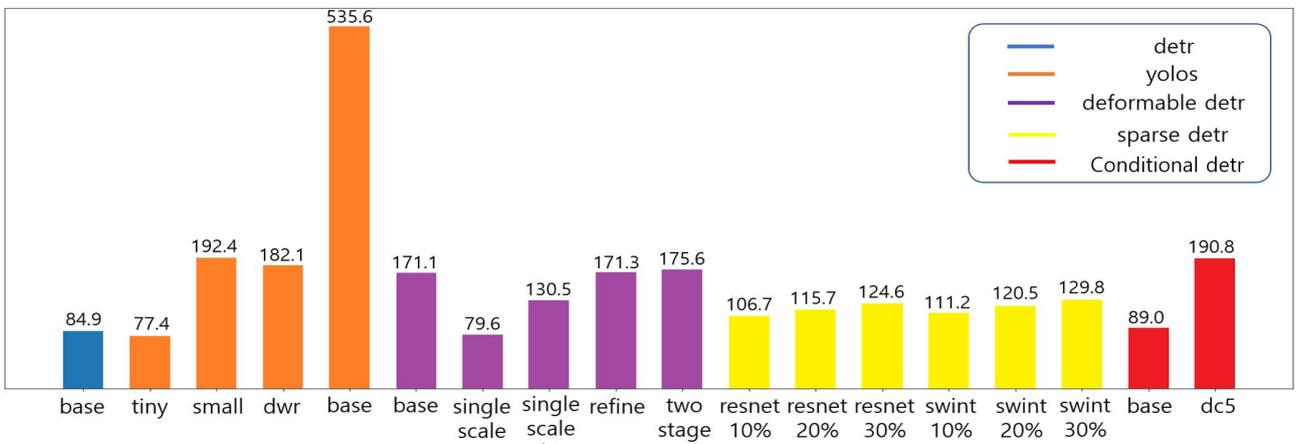


그림 10. 모델 별 GFLOPS

Fig. 10. GFLOPS of various Vision Transformer Models

표 4. FPS, mAP, GFLOPS 사이의 피어슨 상관계수
Table 4. Pearson Correlation Coefficient among FPS, mAP and GFLOPS

	FPS	mAP	GFLOPS
FPS	1.000	-0.667	-0.340
mAP	-0.667	1.000	-0.029
GFLOPS	-0.340	-0.029	1.000

AP, FPS, GFLOPS를 변수로 설정하여 서로와의 상관관계를 계산하였다. 그 결과 표 4와 같이 mAP와 FPS사이의 상관관계수가 -0.667로 강한 상충관계에 있음을 확인할 수 있었으며, GFLOPS와 다른 두 변수인 FPS의 상관관계수는 각각 -0.340으로 약한 상충관계에 있음을 알 수 있었다. 따라서 높은 성능을 달성한 모델이 효율적으로 성능을 유지하며 연산량을 감소시키는 방법에 관한 연구가 필요하다.

현재 개발되는 인공지능에 사용되는 임베디드 보드는 T FLOPS (Tera Floating point Operations Per Second)단위의 연산이 가능하다. 따라서 트랜스포머 기반의 모델들 또한 임베디드 보드에 적용하여 실생활에 사용 가능하다. 그러나 합성곱 신경망 기반의 모델들에 비해 아직까지 효율적이지 못하다.

본 논문에서는 두 개의 다른 데이터 집합에 다양한 트랜스포머 기반의 모델의 성능을 분석하였다. 이 결과를 기반으로 추후 실시간성을 보장하는 경량화된 트랜스포머 기반의 객체 탐지 모델 연구를 기대하는 바이다.

References

- [1] S. Ren, K. He, R. Girshick, J. Sun, "Faster r-cnn: Towards Real-time Object Detection with Region Proposal Networks," *Proceedings of Advances in Neural Information Processing Systems*, 2015.
- [2] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, "You Only Look Once: Unified, Real-time Object Detection," *Proceedings of Computer Vision and Pattern Recognition*, pp. 779-788, 2016.
- [3] K. Han, Y. Wang, H. Chen, X. Chen, J. Guo, Z. Liu, Y. Tang, A. Xiao, C. Xu, Y. Xu, Z. Yang, Y. Zhang, D. Tao, "A survey on Vision Transformer," *Journals of IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.45, No. 1, pp. 73-86, 2023.
- [4] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," *Proceedings of International Conference on Learning Representations*, 2021.
- [5] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, S. Zagoruyko, "End-to-end Object Detection with Transformers," *Proceedings of European Conference on Computer Vision*, pp. 213-229, 2020.
- [6] Y. Fang, B. Liao, X. Wang, J. Fang, J. Qi, "You Only Look at one Sequence: Rethinking Transformer in Vision Through Object Detection," *Proceedings of Advances in Neural Information Processing Systems*, Vol. 34, pp. 26183-26197, 2021.
- [7] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, J. Dai "Deformable Detr: Deformable Transformers for End-to-end Object Detection," *Proceedings of International Conference on Learning Representations*, 2021.
- [8] B. Roh, J. W. Shin, W. Shin, S. Kim, "Sparse DETR: Efficient End-to-End Object Detection with Learnable Sparsity," *Proceedings of International Conference on Learning Representations*, 2022.
- [9] D. Meng, X. Chen, Z. Fan, G. Zeng, H. Li, Y. Yuan, L. Sun, J. Wang "Conditional DETR for Fast Training Convergence," *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3651-3660 2021.
- [10] T. Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, "Microsoft COCO: Common Objects in Context," *Proceedings of European Conference on Computer Vision*, pp. 740-750, 2014.
- [11] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," *International Journal of Computer Vision*, Vol. 88, No. 2, pp. 303-338, 2010.
- [12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, "Attention is all you Need," *Advances in Neural Information Processing Systems*, pp. 6000-6010, 2017.
- [13] J. Devlin, M. W. Chang, K. Lee, K. Toutanova, "Bert: Pre-training of Deep Bidirectional Transformers for Language Understanding," *Proceedings of Conference of the North American Chapter of the Association for Computational Linguistics*, Vol. 1, pp. 4171-4186, 2019.
- [14] D. W. Otter, J. R. Medina, J. K. Kalita, "A Survey of the Usages of Deep Learning for Natural Language Processing," *Journal of IEEE transactions on neural networks and learning systems*, Vol. 32, No. 2, pp. 604-624, 2020.
- [15] K. He, X. Zhang, S. Ren, J. Sun, "Deep Residual Learning for Image Recognition," *Proceeding of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770-778, 2016.
- [16] P. Dollár, M. Singh, R. Girshick "Fast and Accurate Model Scaling," *Proceedings of Computer Vision and Pattern Recognition*, pp. 924-932, 2021.
- [17] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, "Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows," *Proceedings of International Conference on Computer Vision*, pp. 10012-10022, 2021.
- [18] T. Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollar, "Focal Loss for Dense Object Detection," *Proceedings of IEEE International Conference on Computer Vision*, pp. 2980-2988, 2017.

Jungmin Ha (하 정 민)



2022 Automobile and IT Convergence from Kookmin University (B.S.)
2022~Computer Science from Kookmin University (M.S.)

Field of Interests: Object Detection, Computer Vision, Transformer
Email: 20173430@kookmin.ac.kr

Hyunjong Lee (이 현 종)



2022 Automobile and IT Convergence from Kookmin University (B.S.)
2022~Computer Science from Kookmin University (M.S.)

Field of Interests: Autonomous Driving, Vision Segmentation
Email: dlrhkswhd10@kookmin.ac.kr

Jungmin Eom (엄 정 민)



2021 Automotive Engineering from Kookmin University (B.S.)
2021~Computer Science from Kookmin University (M.S.)

Field of Interests: Computer Vision, Continual Learning
Email: sozzso@kookmin.ac.kr

Jaekoo Lee (이 재 구)



2018 Electrical and Computer Engineering from Seoul National University (Ph.D)
2018~College of Computer Science at Kookmin University (Assistant Professor)

Career:
2011~2013 Assistance Research Engineer, LG Electronics R&D Campus
2018 Data Scientist, SK Telecom Research Center
Field of Interests: Artificial Intelligence(AI), Machine Learning, Deep Learning and Data Science, Autonomous Driving, Object Detection
Email: jaekoo@kookmin.ac.kr

논문 2022-17-39

잡음 제거 기술 기반의 비디오 인페인팅 성능 연구

(A Study on the Video Inpainting Performance using Denoising Technique)

서 정 윤, 백 한 결, 박 상 효*

(Jeong-yun Seo, Han-gyul Baek, Sang-hyo Park)

Abstract : In this paper, we study the effect of noise on video inpainting, a technique that fills missing regions of video. Since the video may contain noise, the quality of the video may be affected when applying the video inpainting technique. Therefore, in this paper, we compare the inpainting performance in video with and without denoising technique DAVIS dataset. For that, we conducted two experiments: 1) applying inpainting technique after denoising the noisy video and 2) applying the inpainting technique to the video and denoising the video. Through the experiment, we observe the effect of denoising technique on the quality of video inpainting and conclude that video inpainting after denoising would improve the quality of the video subjectively and objectively.

Keywords : Deep learning, Noise reduction, Video denoising, Video inpainting

I. 서 론

현재 컴퓨터 비전 분야는 빠른 속도로 성장하고 있다. 컴퓨터 비전의 분야 중 인페인팅 (inpainting) 기술은 이미지의 누락되거나 손상된 영역을 주변 배경과 유사하게 채우는 기술로 이미지 또는 영상 편집과 객체 제거, 손상 복원, 워터마크 제거 등의 용도로 쓰일 수 있다. 인페인팅 기술은 크게 이미지 인페인팅 (image inpainting)과 비디오 인페인팅 (video inpainting) 두 개의 분야로 나누어진다. 지금까지 인페인팅에 관한 많은 연구가 이루어졌지만, 완벽한 인페인팅 알고리즘을 구현하는 데는 어려움을 겪고 있다. 이미지 인페인팅 기법은 고전적인 비학습 기반 (non-learning based) 인페인팅 기술 [1-4]과 학습 기반 (learning based) 기법으로 나눌 수 있다 [5-8]. 학습 기반의 이미지 인페인팅을 위해서는 생성적 적대 신경망 (Generative Adversarial Network, GAN) [9]이 주로 사용된다. 비디오 인페인팅은 영상의 누락된 영역을 채우거나 특정 물체를 제거하고 손상된 영역을 복원하는 데 사용된다는 점이 이미지 인페인팅과 유사하다. 하지만 인페인팅 기술을 영상의 영역으로 확장하기 때문에 영상의 연속된 프레임 간 시공간적인 일관성을 보존하는 기술이 요구된다. 비디오 인페인팅 기술을 위해 프레임 간 시공간적 일관성을 보존하려는 시도는 많았지만, 현실 세계의 노이즈가 포함된 영상에 인페인팅을 적용할 때

나타나는 문제점은 거의 다루어지지 않았다. 그러나 어두운 장소에서 촬영된 영상과 같이 현실 세계에서 촬영된 영상에는 불가피하게 노이즈가 포함되어 있고, 이것은 영상에 인페인팅 기술을 적용할 때 영상 품질에 영향을 줄 수 있다. 따라서 본 논문에서는 인페인팅 기술을 사용할 때 영상의 잡음 제거 여부가 인페인팅 결과에 미치는 영향에 대해 연구하여 비디오의 품질을 개선할 방법을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서 인페인팅과 관련된 기존 연구와 한계를 소개한다. 3장에서는 제안하는 인페인팅 기법과 실험 방법을 설명하고, 4장에서 그 결과를 분석했다. 마지막으로 5장에서는 결론으로 마무리한다.

II. 관련 연구

1. 이미지 인페인팅

1.1 비학습 기반 이미지 인페인팅

고전적인 비학습 기반의 인페인팅 기술은 크게 패치 기반 방법과 확산 기반 방법으로 나눌 수 있다. 패치 기반의 인페인팅 모델 [1, 2]은 알려진 영역 (known region)의 픽셀 정보를 통해 이미지의 훼손된 영역을 채운다. 확산 기반의 인페인팅 모델 [3, 4]은 손상된 영역 주변의 정보를 손상된 영역 (missing region)으로 전파하여 이미지를 완성한다. 그러나 이러한 비학습기반의 인페인팅 기법들은 계산 복잡도가 높고, 손상된 영역이 클 때는 의미론적인 정보를 파악할 수 없다는 한계가 있다.

1.2 학습 기반 이미지 인페인팅

딥러닝 기술이 비약적으로 발전함에 따라 컴퓨터 비전 분야에도 딥러닝을 도입하게 되었다. 최근 인페인팅 기술에도 딥러닝을 적용한 연구가 활발하게 이루어지고 있다. [5]는

*Corresponding Author (s.park@knu.ac.kr)

Received: Oct. 19, 2022, Revised: Oct. 25, 2022, Accepted: Nov. 26, 2022.

J. Y. Seo: Kyungpook National University (B.S. Student)

H. K. Baek: Kyungpook National University (M.S. Student)

S. H. Park: Kyungpook National University (Asst. Prof.)

※ 이 논문은 2020년도 정부 (교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업인 (No. 2020R111A3072227).

※ 이 논문의 연구 결과 중 일부는 한국통신학회 “제3회 한국 인공지능 학술대회”에서 발표한 바 있음.

기존 이미지 인페인팅 기술을 개선하기 위해 부분 컨볼루션 층을 도입했다. 이전의 딥러닝 기반 인페인팅 기술은 이미지에 컨볼루션 필터를 사용해 이미지의 빈 영역을 고정 값으로 채웠다. 하지만 초기의 이미지 구멍 값에 의존하기 때문에 구멍의 질감이 부족하고, 복원 가능한 손상 영역이 직사각형 모양에 한정되며 색상의 대비와 부자연스러운 가장 자리를 가진다는 문제가 있었다. 따라서 부분 컨볼루션을 통해 손상된 부분에 대한 마스크 업데이트와 재정규화를 진행했다. 그 결과 커널을 통과하는 픽셀이 손상된 부분이면 계산을 수행하지 않고 통과하여 구멍이 아닌 영역에만 컨볼루션 연산을 수행하여 이미지를 완성하도록 했다.

먼저, [6]에서는 부분 컨볼루션 연산은 각 층을 지날 때마다 유효하지 않은 픽셀이 사라지면서 마스크의 모든 값이 1로 변하기 때문에 인페인팅을 위한 정보를 제공할 수 없으며 모든 채널이 같은 마스크를 공유하기 때문에 유연성이 제한된다는 문제를 제기했다. 따라서 게이트된 합성곱(gated convolution) 연산을 통해 최적의 마스크를 자동으로 학습하는 방법을 제안했다. 이미지의 공간 위치와 이미지 특성(feature)에 가중치를 곱한 값을 연산하여 마스크를 학습하고, Spectral-Normalized Markovian Discriminator (SN-PatchGAN)를 도입해 이진 마스크와 함께 사용자 주도의 인페인팅(user-guided inpainting)을 가능하게 했다. 또한, [7]의 인페인팅 모델은 크게 완성 네트워크(completion network), 전역 판별 네트워크(global discriminator network), 지역 판별 네트워크(local discriminator network) 세 개의 네트워크로 이루어진다. 먼저 이미지와 마스크를 완성 네트워크에 입력해 손상 이미지를 복구하고 두 개의 판별 네트워크를 통해 복구된 이미지의 진위 여부를 판별하여 인페인팅 결과가 손상된 영역 주변뿐만 아니라 이미지 전체적으로 일관성을 가지도록 한다. 완성 네트워크는 일반적인 컨볼루션 연산이 아닌 확장된 컨볼루션 연산을 통해 더 넓은 부분을 파악하여 연산하고 계산량을 줄이도록 개선했다. 생성적 적대 모델을 사용하는 또 다른 기법인 [8]의 이미지 인페인팅 모델은 거친 네트워크(coarse network)와 개선 네트워크(refinement network)로 이루어져 있어 손상된 이미지를 coarse-to-fine 기법으로 인페인팅 했다. 기존의 합성곱 신경망을 사용하는 기법들은 빈 영역과 떨어져 있는 픽셀 정보를 참고할 수 없었기 때문에 문맥 주의(contextual attention)를 사용해 알려진 영역의 유의미한 특성을 손상된 영역에 반영하여 생성 이미지의 품질을 개선했다.

2. 비디오 인페인팅

비디오 인페인팅 기술은 이미지 인페인팅 기술과 달리 프레임 간의 시공간적인 일관성을 유지해야 한다는 문제가 있다. Deep Flow-Guided Video Inpainting [10] 모델은 영상 흐름 기반의 픽셀 전파를 통해 영상의 시공간적인 일관성을 보존하고자 했다. Deep Flow Completion Network (DFC-Net)을 통해 영상의 흐름을 얻어 이 완성된 흐름을 기반으로 픽셀을 전파하여 손상된 영역을 채웠다.

Deep Video Inpainting [11]은 일시적인 특징 집합(temporal feature aggregation)과 시간적 일관성 보존(temporal consistency preserving)을 핵심 기능으로 가지는 ViNet이라는 모델을 제안했다. 이미지 기반의 인코더-디코더 모델을 기반으로 여러 개의 이웃 프레임들로부터 특징을 수집하여 이를 바탕으로 하나의 프레임의 누락된 영역을 채웠다. 또한 시간적인 일관성을 보존하기 위해 반복적인 피드백(recurrent feedback)을 통해 이전 단계의 출력값을 다음 단계의 입력값으로 주었고, 메모리 층을 추가하여 결과 프레임의 단기 및 장기 일관성을 보장했다.

이처럼 시각적으로 자연스러운 인페인팅 결과를 위한 계산법과 시공간적인 일관성을 보존하기 위한 모델에 관한 연구는 활발히 진행되고 있지만 인페인팅 데이터인 영상의 품질과 노이즈 제거 여부가 인페인팅 성능에 미치는 영향에 대해서는 거의 연구가 미미할 실정이다.

III. 제안 기법

1. 제안 기법

본 논문에서 진행한 두 가지 실험은 다음과 같다.

- 실험 1. 노이즈가 포함된 프레임의 노이즈를 디노이징 기법으로 먼저 제거하고 인페인팅 한다.
- 실험 2. 노이즈가 포함된 프레임을 먼저 인페인팅 한 후 디노이징 기법으로 노이즈를 제거한다.

위 두 실험의 결과와 원본 프레임에 대한 인페인팅 결과를 비교하여 노이즈가 인페인팅 영상의 품질에 영향을 줄 수 있음을 확인한다. 또한 두 실험의 결과를 비교해 노이즈 제거 여부에 따른 인페인팅 성능을 비교하여 노이즈가 포함된 영상일 경우 노이즈 먼저 제거하는 것이 인페인팅 성능 향상에 도움이 될 수 있음을 제안한다.

2. 배경 기술

2.1 Deep Flow-Guided Inpainting

본 실험에서는 비디오 인페인팅에 Deep Flow-Guided Video Inpainting 모델 [10]을 사용했다. 그림 1은 인페인팅 모델의 구조를 나타낸다. 전체 네트워크는 3개의 Deep Flow Completion Subnetwork (DFC-S)가 DFC-Net를 구성하고 있다. 각 DFC-S에서 추출한 영상 프레임 간 흐름(flow)을 기반으로 전체 영상의 흐름을 완성하고, 이 흐름(flow-file)을 양방향으로 전파해 프레임의 누락된 영역을 채운다. 원

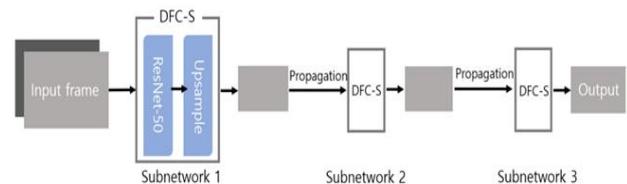


그림 1. Deep Flow-Guided Video Inpainting 모델 구조
Fig. 1. An Architecture of Video Inpainting Method

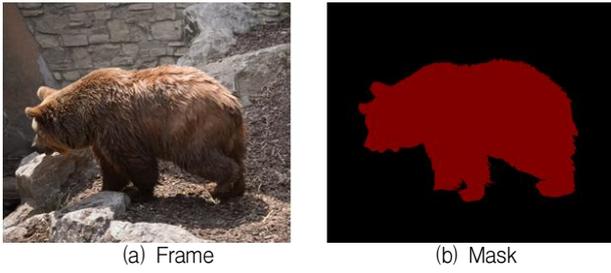


그림 2. Deep Flow-Guided Video Inpainting 입력 데이터
Fig. 2. Input Images of Deep Flow-Guided Video Inpainting

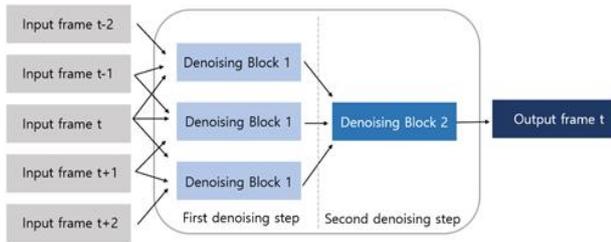


그림 3. FastDVDnet 구조
Fig. 3. An Architecture of FastDVDnet

본 모델의 흐름 추출에는 FlowNet2.0 [12] 모델이 사용되었지만, 본 실험에서는 전체 모델의 경량화를 위해 LiteFlowNet2.0 [13] 모델을 사용했다.

그림 2는 영상의 객체를 제거하기 위해 인페인팅 모델에 주어지는 입력의 예시다. 물체를 포함한 프레임과 물체의 위치를 나타내는 마스크가 입력으로 주어지면 DFC-Net은 프레임과 마스크가 겹치는 부분을 누락된 영역으로 인식하여 픽셀의 전파를 통해 마스크 영역을 배경과 유사하게 채운다.

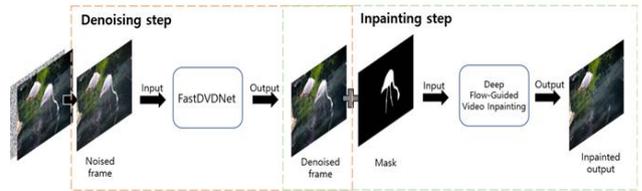
2.2 FastDVDnet

프레임의 잡음 제거에는 Fast Deep Video Denoising Network (FastDVDnet) 모델 [14]을 사용했다. 그림 3은 FastDVDnet의 구조를 나타낸다. 이 네트워크는 영상의 흐름을 추출할 필요 없이 두 번의 노이즈 제거 단계를 거쳐 이미지의 노이즈를 제거한다. 전체 네트워크는 5장의 연속적인 입력 프레임과 표준편차가 σ 인 노이즈맵을 입력값으로 가지며 첫 번째 디노이징 블록에는 3장의 입력 프레임과 노이즈맵이 입력으로 주어진다. 첫 번째 디노이징 단계를 구성하는 세 개의 디노이징 블록에서 각 1장의 노이즈 제거 프레임이 출력된다. 두 번째 디노이징 단계는 첫 번째 디노이징 단계에서 출력된 3장의 프레임을 입력값으로 가져 가운데 프레임의 노이즈를 제거하고 출력한다.

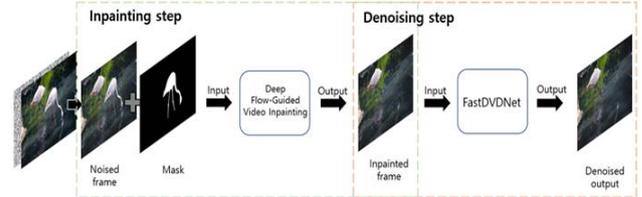
본 실험에서는 노이즈맵의 표준편차와 이미지에 추가한 노이즈의 표준편차를 5로 동일하게 설정하였으며 실험 방법에 따라 임의로 가우시안 노이즈를 추가한 원본 프레임 또는 노이즈를 추가한 후 인페인팅을 적용한 프레임을 입력값으로 주어 영상의 노이즈를 제거했다.

3. 데이터 세트

DAVIS 데이터 세트 (Densely Annotated Video Segmen-



Experiment 1. Inpaint frames after denoising



Experiment 2. Denoise frames after inpainting

그림 4. 제안 모델의 성능 평가를 위한 실험 방법
Fig. 4. Experimental Methods for Evaluating Our Proposed Method

tation Dataset) [15]는 영상의 객체 분할 (Object Segmentation)을 위해 제안된 데이터 세트로 총 150개의 영상 데이터를 제공한다. 각 영상은 객체를 포함한 프레임과 그에 대한 주석 (Annotation)으로 구성되어있다. 이 중 breakdance-floor, rollerblade, bmx-trees 데이터와 Deep Flow-Guided Video Inpainting 모델의 데모 버전에서 사용된 flamingo 영상에 대해 실험을 진행했다. 각 데이터 세트의 프레임에 노이즈를 넣거나 제거한 결과를 제안 기법의 입력 프레임으로 사용했으며 데이터세트의 객체 위치 정보를 가진 Annotation 데이터를 각 프레임의 이진 마스크 (Binary mask)로 입력했다.

4. 실험 방법

모든 실험 과정은 Ubuntu 환경에서 Python 3.7 버전, PyTorch 1.4.0 버전과 CUDA 11.3 버전으로 동일한 환경에서 진행되었다. 두 실험 모두 원본 비디오에 동일한 강도의 노이즈를 임의로 추가하여 실험을 진행했다. 원본 프레임에 추가한 노이즈는 가우시안 노이즈 (Gaussian noise)로 정규 분포를 따르며 평균이 0, 표준편차가 5인 노이즈를 생성하여 프레임에 추가했다. 그림 4는 두 가지 실험의 과정을 나타낸다. 첫 번째 실험은 노이즈를 추가한 각 프레임에 대해 먼저 FastDVDnet 모델을 사용해 노이즈를 제거하고 결과 프레임의 객체를 인페인팅했다. 두 번째 실험은 먼저 노이즈를 가진 프레임의 객체를 인페인팅 모델로 제거한 후 FastDVDnet 모델을 사용해 프레임의 노이즈를 제거했다.

IV. 실험 결과

1. 정성 평가

그림 5~그림 8은 각기 다른 환경에서의 인페인팅 결과이다. 인페인팅을 적용하는 과정에서 단계별 프레임 출력값과 원본 프레임 대비 출력값의 Peak Signal-to-Noise Ratio 값



그림 5. 원본 프레임과 마스크에 대한 인페인팅 결과
 Fig. 5. Inpainting Results of Original Frames and Masks

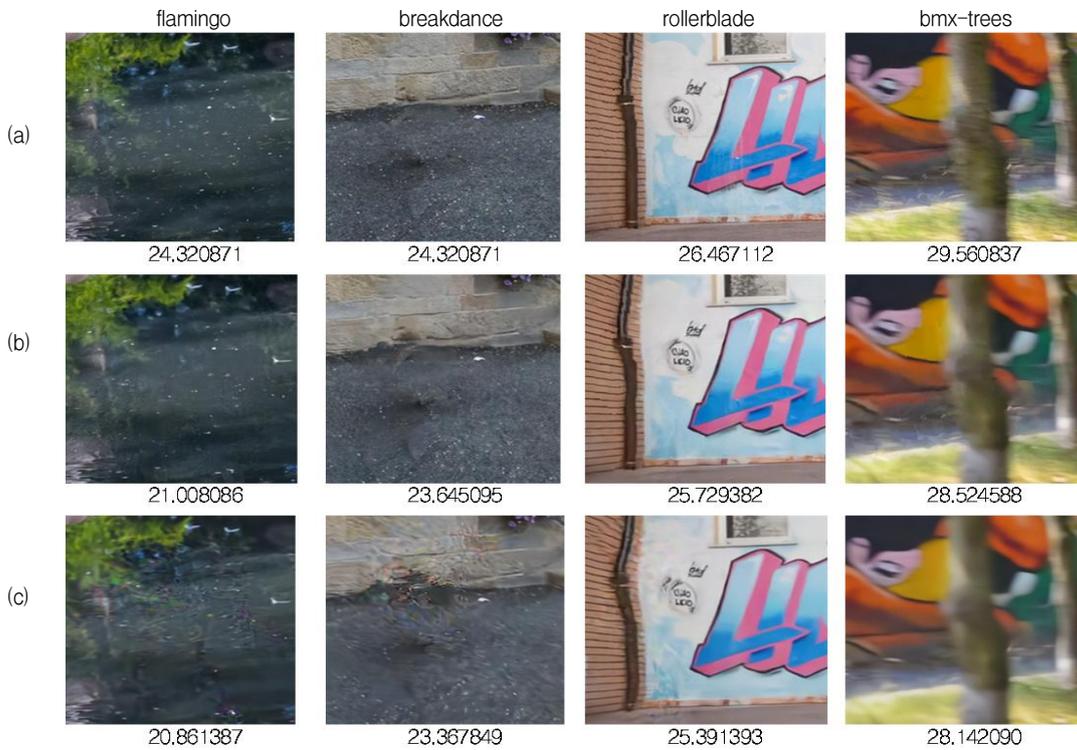


그림 6. 실험 1의 인페인팅 결과
 Fig. 6. Inpainting Results of Experiment 1



(a) Noised frames (b) Denoised frames (c) Inpainting results

그림 7. 실험 2의 인페인팅 결과
Fig. 7. Inpainting Results of Experiment 2



(a) Inpainting results of original frames (b) Inpainting results of experiment 1
(c) Inpainting results of experiment 2

그림 8. 실험 1과 실험 2의 인페인팅 결과 비교
Fig. 8. Comparison of Inpainting Results in Experiment 1 and Experiment 2

표 1. 두 실험에 대한 정량평가 및 비교 결과
Table 1. Performance Comparison of Inpainting Results in Experiment 1 and Experiments 2

Dataset	Method	Evaluation metric	
		PSNR ↑	SSIM ↑
break dance	(a)	23.645095	0.804672
	(b)	23.367849	0.755008
flamingo	(a)	21.008086	0.850027
	(b)	20.861387	0.806348
rollerblade	(a)	25.729382	0.877829
	(b)	25.391393	0.862348
bmx-trees	(a)	28.524588	0.899922
	(b)	28.142090	0.888364
(a) Experiment 1 : Denoising → Inpainting			
(b) Experiment 2 : Inpainting → Denoising			

을 담고 있다. 그림 5는 노이즈를 추가하지 않은 원본 프레임에 대한 인페인팅 결과다. 각 데이터 별로 마스크 영역의 객체가 제거된 결과를 확인할 수 있다. 그림 6은 실험 1의 결과다. 단계별로 노이즈를 넣은 프레임, 노이즈를 제거한 프레임, 인페인팅 결과가 순서대로 담겨있다. 그림 7은 실험 2의 결과로 노이즈를 넣은 프레임, 인페인팅 결과, 노이즈를 제거한 결과를 순서대로 나타냈다.

그림 8은 위에서부터 차례로 (a) 원본 프레임의 인페인팅 결과, (b) 실험 1의 결과, (c) 실험 2의 결과를 객체가 있던 부분 중심으로 확대한 사진이다. (a)의 결과에 비해 (b), (c)의 결과는 노이즈 제거 작업으로 인해 사진의 질감 (texture)이 부족함을 확인할 수 있다. (b)와 (c)를 비교해보면 대체적으로 노이즈 제거 작업을 먼저 수행한 (b)에 비해 인페인팅 후 노이즈를 제거한 (c)의 노이즈가 제대로 제거되지 않아 영상이 부자연스러운 것처럼 보인다.

2. 정량 평가

표 1은 앞서 수행한 실험 1, 2의 결과를 각각 DAVIS dataset의 원본 프레임과 비교한 정량적 평가값을 담고 있다. 평가 결과에 따르면 인페인팅 후 노이즈를 제거했던 실험 (b)보다 노이즈를 제거하고 인페인팅을 적용했던 실험 (a)가 Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM) 값에 대해 더 높은 성능을 보였다.

V. 결론

본 논문에서는 잡음이 포함된 영상의 잡음 제거 여부가 인페인팅 기술에 미치는 영향에 대해 실험을 통해 비교해 보았다. 앞서 진행한 두 실험을 비교해 봤을 때, 노이즈가 포함된 영상을 인페인팅한 후 노이즈를 제거한 영상은 노이즈가 완벽하게 제거되지 않았다. 반면에 노이즈를 제거한 후 인페인팅을 적용한 영상은 노이즈가 비교적 깨끗하게 제거되

어 앞의 실험 결과보다 자연스러운 것을 확인했다. 이는 노이즈가 포함된 프레임을 인페인팅할 경우 노이즈가 포함된 픽셀 정보를 통해 빈 영역을 채웠기 때문에 노이즈가 완벽히 제거되지 않은 것으로 추측된다. 이 실험 결과를 통해 잡음 제거 여부가 인페인팅 결과에 영향을 미칠 수 있으며, 영상에 노이즈가 존재하는 경우 먼저 노이즈를 제거한 후 인페인팅을 진행하는 것이 비디오의 품질을 개선할 가능성이 있음을 알 수 있다. 본 제안 기법을 실제 야간에 촬영된 영상이나 의도치 않게 노이즈가 발생한 영상에 적용한다면 실제 인페인팅 기술의 품질을 높이는 데 도움이 될 것으로 기대된다. 향후 본 실험을 확장시켜 다른 데이터셋에 적용해 보고 다양한 인페인팅 모델과 노이즈 제거 모델을 사용하여 추가 연구를 진행할 예정이다.

References

- [1] M. Bertalmio, L. Vese, G. Sapiro, S. Osher, "Simultaneous Structure and Texture Image Inpainting," *IEEE Transactions on Image Processing*, Vol. 12, No. 8, pp. 882–889, 2003.
- [2] S. Darabi, E. Shechtman, C. Barnes, D. Goldman, P. Sen, "Image Modeling: Combining Inconsistent Images Using Patch-based Synthesis," *ACM Transactions on Graphics*, Vol. 31, No. 4, Art. 82, pp. 1–10, 2012.
- [3] M. Bertalmio, G. Sapiro, V. Caselles, C. Ballester, "Image Inpainting," *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 417–424, 2000.
- [4] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, J. Verder, "Filling-in by Joint Interpolation of Vector Fields and Gray Levels," *IEEE Transactions on Image Processing*, Vol. 10, No. 8, pp. 1200–1211, 2001.
- [5] G. Liu, F. A. Reda, K. J. Shih, T. C. Wang, A. Tao, B. Catanzaro, "Image Inpainting for Irregular Holes Using Partial Convolutions," *Proceedings of the European Conference on Computer Vision*, pp. 85–100, 2018.
- [6] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, T. Huang, "Free-form Image Inpainting with Gated Convolution," *Proceedings of IEEE/CVF International Conference on Computer Vision*, pp. 4470–4479, 2019.
- [7] S. Iizuka, E. Simo-Serra, H. Ishikawa, "Globally and Locally Consistent Image Completion," *ACM Transactions on Graphics*, Vol. 36, No. 4, pp. 1–14, 2017.
- [8] J. Yu, J. Yang, X. Shen, X. Lu, T. S. Huang, "Generative Image Inpainting with Contextual Attention," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5505–5514, 2018.
- [9] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, "Generative Adversarial Nets," *Advances in Neural Information Processing Systems*, pp. 2672–2680, 2014.
- [10] R. Xu, X. Li, B. Zhou, C.C. Loy, "Deep Flow-guided Video Inpainting," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*,

pp. 3718-3727, 2017.

[11] D. Kim, S. Woo, J. Y. Lee, I. S. Kweon, "Deep Video Inpainting," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5792-5801, 2019.

[12] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, T. Brox, "FlowNet 2.0: Evolution of Optical Flow Estimation with Deep Networks." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2462-2470, 2017.

[13] T. W. Hui, X. Tang, C. C. Loy, "LiteFlowNet: A Lightweight Convolutional Neural Network for Optical Flow Estimation," Proceedings of the IEEE Conference

on Computer Vision and Pattern Recognition, pp. 8981-8989, 2018.

[14] M. Tassano, J. Delon, T. Veit, "FastDVDnet: Towards Real-time Deep Video Denoising Without Flow Estimation," Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1351-1360, 2020.

[15] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. V. Gool, M. Gross, A. Sorkine-Hornung, "A Benchmark Dataset and Evaluation Methodology for Video Object Segmentation," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 724-732, 2016.

Jeong-yun Seo (서 정 윤)



2019~Computer Science and Engineering from Kyungpook National University (B.S.)

Field of Interests: Deep Learning, Computer Vision, Inpainting
Email: yooniverse@knu.ac.kr

Han-gyul Baek (백 한 결)



2021 Electrical and Computer Engineering from Daejeon University (B.S.)
2021~Computer Science and Engineering from Kyungpook National University (M.S.)

Field of Interests: Deep Learning, Computer Vision, Inpainting
Email: qorgksruffl23@knu.ac.kr

Sang-hyo Park (박 상 효)



2011 Computer Engineering from Hanyang University (B.S.)
2017 Computer Science from Hanyang University (Ph.D.)

Career:

2017~2018 Postdoctoral position, Korea Electronics Technology Institute (KETI)

2018 Research Fellow, Yonsei University

2019~2020 Postdoctoral position, Ewha Womans University

2020~Assistant Professor with the School of Computer Science and Engineering, Kyungpook National University

Field of Interests: VVC, encoding complexity, immersive video, model optimization

Email: s.park@knu.ac.kr

논문 2022-17-40

마스크 생산 라인에서 다중 영상 기반 마스크 이어링 검사 방법 (Multi-Vision-based Inspection of Mask Ear Loops Attachment in Mask Production Lines)

우 지 명, 이 상 현, 이 현 철*
(JiMyeong Woo, SangHyeon Lee, Heoncheol Lee)

Abstract : This paper addresses the problem of vision-based ear loops and attachment inspection in mask production lines. This paper focuses on connections with ear loops and mask filter by an efficient combined approach. The proposed method used a template matching, shape detection and summation of histogram with preprocessing. We had a parameter for detecting defects heuristically. If the shape vertices are lower than the parameters our proposed method will find defective mask automatically. After finding normal masks in mask ear loops attachment status inspection algorithm our proposed method conducts attachment amount inspection. Our experimental results showed that the precision is 1 and the recall is 0.99 in the mask attachment status inspection and attachment amount inspection.

Keywords : Multi-vision-based inspection, Image processing, Manufacturing automation

1. 서 론

마스크는 대량 생산을 목적으로 자동화된 마스크 생산 시설에서 생산된다. 마스크 생산 라인에는 마스크 필터를 만들기 위해 마스크 원단을 공급하는 원단 공급 장치와, 마스크 원단을 성형해서 필터를 만드는 마스크 성형 장치, 그리고 마스크 필터에 이어링을 부착하는 이어링 부착 장치가 포함된다. 마스크를 생산할 때, 마스크 필터에 이어링을 부착하여 마스크를 제조한 뒤, 마스크를 최종적으로 포장하기 전에 마스크가 양품인지 불량인지를 검사하는 검사 과정이 필요하다. 이어링 부착 여부 검사에서는 이어링이 마스크 필터에 부착되어 있는지를 확인하고, 그 후 이어링 부착량에 관한 검사를 진행하여 마스크 필터에 이어링이 얼마나 견고하게 부착되어 있는지를 확인하게 된다. 이러한 과정들을 통해 양품과 불량품이 선별되고, 불량품은 마스크 흡착 장치에 의해서 폐기 처리가 된다. 마스크의 양불 판정을 사람의 육안에 의해 검수하게 되면, 시간과 피로의 문제로 인해 효율성이 떨어지게 된다. 그에 따라 본 논문에서는 이와 같은 문제점들을 해결하기 위해 영상 기반 마스크 이어링 및 부착량 검사기법을 제시하고, 그림 1과 같은 마스크 생산 시스템에서 멀티 비전을 사용함으로써 인해 마스크 검사의 정확도 및 검사 효율을 높인다.

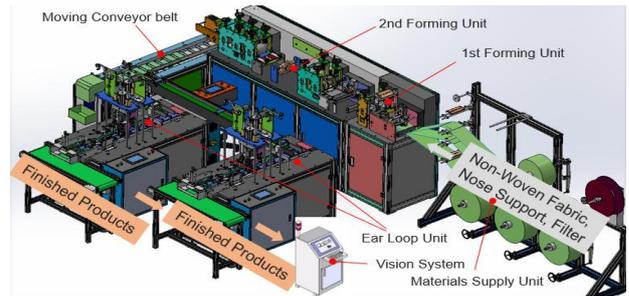


그림 1. 제안된 기법은 멀티 비전 시스템을 이용하여 마스크 생산 공장의 마지막 단계에서 사용된다.
Fig. 1. Our proposed method is used in final process of mask manufacturing factory with Vision System

이전 연구에서 계층적 피어슨 상관관계 계수를 기반으로 한 마스크 필터의 이물을 탐지하는 연구를 수행하였다 [1]. 이 방법은 실제 마스크 생산 시스템을 고려하여 다양한 이미지 처리 기술을 결합하여 수행된다. 마스크의 방향을 일정하게 x축에 맞추기 위한 이미지 회전, 마스크 필터 부분을 제외한 귀걸이 부분과 배경을 잘라내어 정확도를 높이기 위한 이미지 크롭을 한 후, 블러링을 하여 노이즈를 제거한다. 마지막으로, 계층적 피어슨 상관관계 계수를 이용하여 정상 마스크와 불량 마스크간의 상관관계를 계산하여 양불 판정을 하였다.

본 논문에서는 회전되어 있는 마스크 이미지에서도 알고리즘 처리를 하였으며 여러 가지 전통 컴퓨터 비전 기반 기법들을 활용하여 다양한 종류의 마스크 이어링 부착상태를 검사한다. 또한, 여러 이미지 전처리 과정과 히스토그램, 마스크의 특성을 기반으로 한 방법을 활용하여 마스크 이어링

*Corresponding Author (hcleee@kumoh.ac.kr)
Received: Oct. 14, 2022, Revised: Nov. 11, 2022, Accepted: Nov. 18, 2022.
J. M. Woo: Department of IT Convergence Engineering, Kumoh National Institute of Technology (M.S. Candidate)
S. H. Lee: Research Institute of Manufacturing Technology, Kumoh National Institute of Technology (Researcher)
H. C. Lee: Department of IT Convergence Engineering, Kumoh National Institute of Technology (Asst. Prof.)
※ 이 연구는 2020년 국립대학 육성사업비로 지원되었음.

표 1. 이미지 프로세싱 관련 연구

Table 1. Related works to image processing methods

Related Works	Description	Method	Real Data	Mask Application
[2-3]	Image detection and classification by CNN	DL	No	No
[6-7]	Image recognition by ANN	DL	No	No
[4-5]	Object detection and anomaly detection by CNN	DL	Yes	No
[8-9]	LBP for getting image features	NDL	No	No
[10-11]	Image detection by SIFT	NDL	Yes	No
[12-13]	Image feature detection by SURF	NDL	Yes	No
[1]	Mask anomaly detection for the status of mask filter	NDL	Yes	No
Proposed	Mask inspection for the status of mask ear loops and attachment	NDL	Yes	Yes

부착량 검사를 수행하며 정규화를 통해 다양한 마스크 이미지에도 강인한 모습을 보여준다.

II. 관련 기술 및 문제점

1. 관련 연구

표 1과 같이 Deep Learning (DL) 혹은 Non-Deep Learning (NDL)을 사용해서 이미지 탐지를 하는 많은 방법들이 있다. DL을 사용하는 방법에는 Convolutional Neural Network (CNN) [2-5], Artificial Neural Network (ANN) [6-7]등의 방법들이 있다. CNN은 인공 신경망의 한 종류이며 이미지 처리의 분야에 주로 이용된다. CNN은 Convolution layer, Rectified Linear Unit (ReLU), pooling, Fully connected layer 등으로 이루어진다. 이미지는 이런 식으로 구성된 모델을 통과하여 학습을 진행하고 결과를 나타낸다. ANN은 사람의 뇌의 정보처리 방식을 모사한 알고리즘이다. 예측, 분류 등의 의사결정 문제를 해결하는 기법이며 input layer, hidden layer, output layer로 구성되어 있다. 이러한 DL을 사용한 기법들은 이미지를 잘 탐지 해내지만 많은 양의 학습 데이터를 요구하기 때문에 시간이 많이 들고 높은 성능의 Graphic Processing Unit (GPU)을 요구하기 때문에 비용이 비싸다는 단점이 있다. NDL을 사용한 방법에는 Local Binary Patterns (LBP) [8-9], Scale Invariant Feature Transform (SIFT) [10-11], Speeded Up Robust Features (SURF) [12-13]등과 같은 방법들이 있다. LBP는 이미지의 질감을 표현하는데 사용되며 매우 간단하

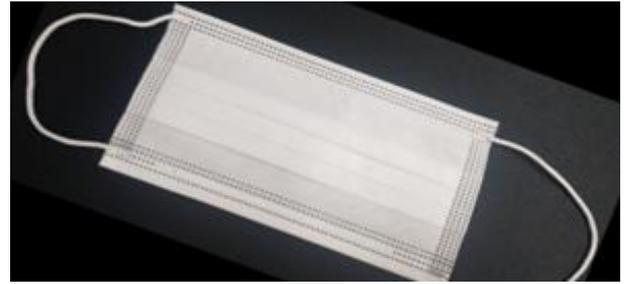


그림 2. 회전된 마스크는 마스크 이미지 처리를 할 때 문제를 일으킨다.
Fig. 2. Rotated Mask occurs problem when the mask image is processed

고 효과적이다. 하지만 마스크 필터 부분이 서로 다른 질감을 가지고 있기 때문에 LBP는 마스크 불량 탐지에 적합하지 않다. SIFT는 이미지의 변하지 않는 크기와 회전에서 특징을 뽑아내는 알고리즘이다. 두 개의 다른 이미지에서 SIFT특징을 추출한 다음 가장 유사한 특징과 일치시키면 해당 두 이미지의 일부를 찾을 수 있다. 하지만 이미지 피라미드를 사용하여 크기 변화에 따른 특징을 감지하기 때문에 처리 속도가 느리다는 단점이 있다. 그러므로 SIFT는 실시간으로 불량 마스크를 검출해야 하는 알고리즘에 사용하기에 적합하지 않다. SURF는 SIFT의 단점을 보완하기 위해 만들어진 알고리즘이다. SURF는 BOX 필터로 LOG를 근사하는 방법을 사용하며 key point 및 descriptor를 계산하는 각 단계에서 많은 특성 요소를 추가하여 처리 속도가 빠르다. 그러나 시점이나 조명이 바뀌면 이미지의 특징을 제대로 감지할 수 없기 때문에 마스크 생산 시스템에서 사용하기에 적합하지 않다. 표 1과 같이 실제로 생산된 데이터를 사용하여 NDL방식으로 마스크 불량을 탐지하는 연구가 이루어지지 않았기 때문에 본 논문에서는 마스크 이어링 검사 방법에 대하여 제시한다.

2. 비전 기반 마스크 양불 판정의 문제점

마스크 이어링 불량 검출에서 정상 마스크와 불량 마스크를 비교하기 위해서는 이미지의 크기, 면적이 동일한 조건이어야 한다. 또한 마스크 이미지의 원활한 불량 탐지를 위해서는 마스크 가로 길이의 방향이 x축과 평행이 되도록 각도를 균등하게 조절하는 것이 중요하다. 하지만 실제 마스크 생산 시스템에서는 그림 2와 같이 라인에 배치된 마스크가 라인 방향과 일치하지 않는 경우가 발생한다. 이러한 점은 마스크 이미지를 처리함에 있어서 문제를 야기한다. 따라서 마스크가 생산 라인에 놓일 때의 각도 차이를 없애는 기법이 필요하다.

III. 제안된 기법

본 논문에서는 그림 3과 같이 마스크 이어링 양불 판정을 할 때 마스크 이어링 부착 상태 여부를 판단한 다음에 마스크

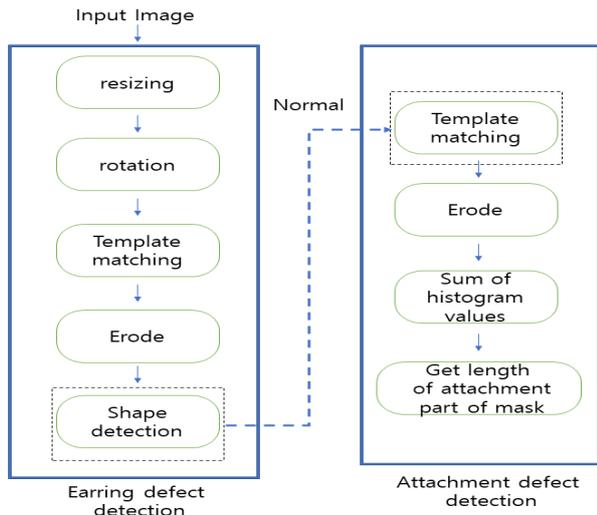


그림 3. 마스크 이어링 검사 알고리즘 순서도
Fig. 3. Mask ear loops attachment inspection algorithm flow chart

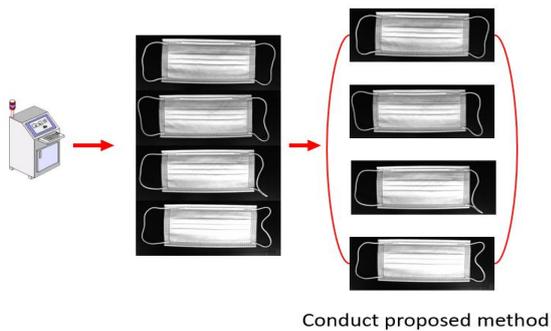


그림 4. 다중 영상 시스템에서의 이미지 처리
Fig. 4. Image processing in multi-vision system

크 이어링 부착량 검사를 통해 얼마나 견고하게 부착되어 있는가를 판단하고 이 두 가지 기준을 다 통과하면 양품으로 판단한다. 또한, 다중 영상 기반 시스템에서 촬영한 이미지들은 4개의 카메라로 4개의 마스크를 한 번에 촬영한 이미지이며 알고리즘을 수행 할 때는 그림4와 같이 각각의 마스크들로 분리해서 처리한다.

1. 마스크 이어링 부착 상태 검사

본 논문에서는 마스크 이어링 부착 상태 불량률 그림 5 (a)와 같이 완전히 이어링이 떨어진 것과 그림 5 (b)와 같이 한쪽만 이어링이 붙어 있는 것, 그리고 그림 5 (c), 5 (d)와 같이 이어링이 한쪽이 완전히 붙어 있지 않은 것으로 정의하였다. 이어링 부착 상태 불량 검사는 전처리 과정을 거친 후 template matching을 통해 마스크 필터만의 이미지를 추출해낸다. 그 후, 마스크 이미지 가장자리의 4개의 모서리를 일정 범위만큼 크롭을 진행한 뒤 shape detection을 통한 마스크 가장자리 모서리의 수를 일정 threshold로 비교를 해서 검사를 하게 된다.

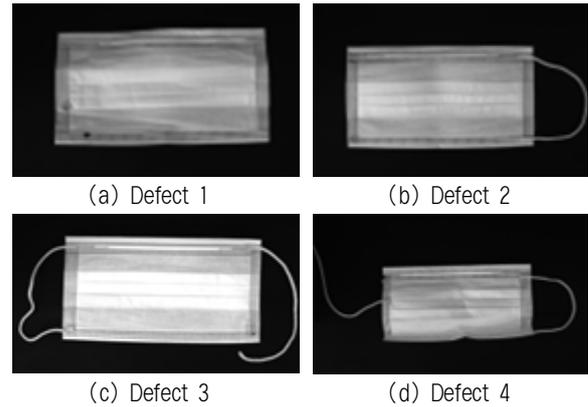


그림 5 마스크 이어링 부착 상태 불량 이미지
Fig. 5. Mask ear loops attachment defect images

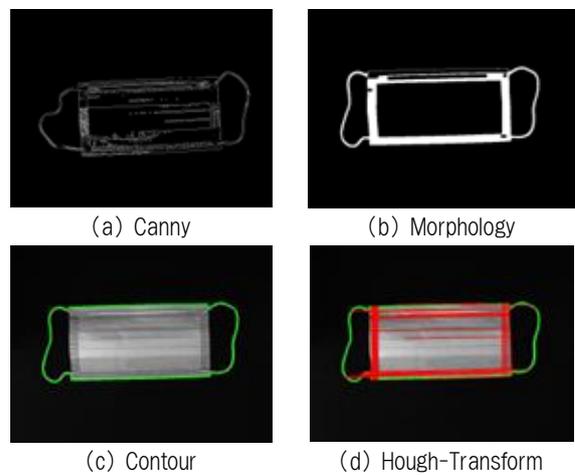


그림 6. 마스크 이어링 부착상태 검사를 위한 마스크 이미지 전처리 (초록색 선: 마스크 이미지의 바깥 테두리, 빨간색 선: 마스크 이미지 가장자리의 직선 성분)

Fig. 6. Image preprocessing of mask image for mask ear loops attachment inspection (Green line: Represents the outline of the image, Red line: Represents the linear component)

1.1 마스크 이미지 전처리

고해상도 이미지에 마스크 이미지 회전을 적용하면 마스크 바깥쪽 라인에 빛의 반사로 인해 다수의 outlier가 생기며 이로 인해 외곽 검출과 선 검출에 문제가 발생한다. 그리고, 고해상도 이미지는 상대적으로 연산속도가 느리기 때문에 원본 이미지를 리사이징을 통해 저해상도로 크기를 바꾸고 난 후 진행한다. 리사이징을 마친 이미지는 마스크 가장자리 부분을 탐지하기 위해 그레이스케일 이미지로 바뀌게 된다. 마스크의 가장자리를 찾기 위해 그림 6 (a)와 같이 Canny [14-15]함수를 사용한다. 그 후 그림 6 (b)와 같이 Morphology의 close연산을 사용하여 가장자리 부분을 확장시키고 그림 6 (c)와 같이 Contour [16-17]함수를 사용하여 마스크의 가장자리 부분을 리사이징을 한 원본 이미지에 나타낸다. 다음으로, 그림 6 (d)와 같이 OpenCV의 HoughlinesP [18-19] 함수를 사용하여 마스크 이미지 가장자리 직선의 시작점과 끝부분의 각도를

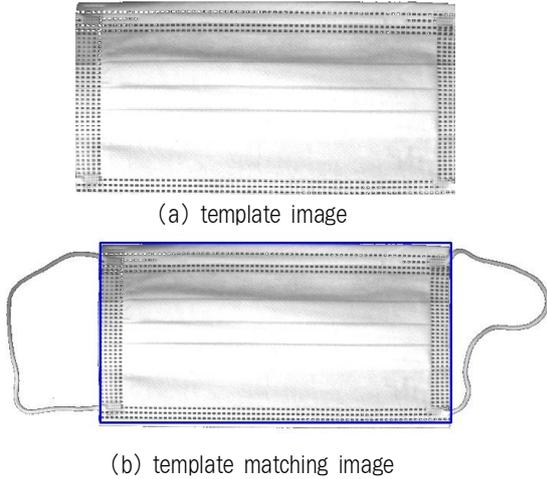


그림 7. Template image를 사용해서 original image에서 template image와 유사한 부분을 찾는다.
Fig. 7. Template matching image is derived from template image

이용해 rotation matrix로 틀어진 각도만큼 회전을 시킨다. 파라미터를 조정하면서 마스크의 직선 성분이 가로가 아닌 세로가 탐지 될 경우도 있다. 그러한 경우에는 각도에서 90도 만큼 뺀 각도를 회전한다.

1.2 Template Matching

Template matching [20-21]은 template image를 사용해서 그림 7 (a), 7 (b)와 같이 original image에서 template image와 가장 비슷한 부분을 찾게 만드는 OpenCV의 함수이다. template matching의 연산 방법으로는 The Minimum Square Difference (TM_SQDIFF)를 사용하였다.

TM_SQDIFF는 template image와 original image 사이의 최소차이를 위해 사용된다. template image가 original image에 적용되면 전체 영역에서 탐지를 시작하며 match가 되고 난 후에는 OpenCV에 있는 minMaxLoc 함수를 이용하여 마스크 필터 이미지의 4 꼭짓점의 좌표를 알아내 그림 8 처럼 크롭한다.

1.3 침식 연산

본 논문에서 침식 연산은 Template matching 후 얻은 4개의 마스크 가장자리 이미지의 노이즈를 제거 하기 위해 사용하였다. 침식 연산은 morphology 변환 방법 중 하나이다. 침식 연산은 커널 영역 안에 존재하는 모든 픽셀의 값은 커널 내부의 극솟값으로 대체하며 식 (1)과 같이 표현될 수 있다.

$$erode(x,y) = \min_{(i,j) \in ker \neq i} src(x+i, y+j). \tag{1}$$

즉, 구조 요소를 활용해 이웃한 픽셀을 최소 픽셀값으로 대체한다. 침식 연산을 적용하면 밝은 영역이 줄어들고 어두운 영역이 늘어난다. 침식 연산을 적용하지 않으면 크롭된 마스크 이미지에 울퉁불퉁한 부분으로 인하여 너무 많은 꼭짓점들이 발생하게 된다.

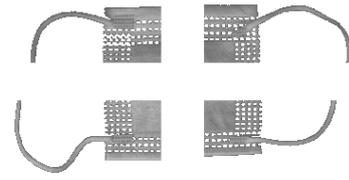


그림 8. 마스크 필터의 4가지 가장자리를 크롭한 이미지
Fig. 8. Cropped image of the four deges of the mask filter

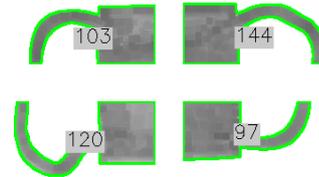


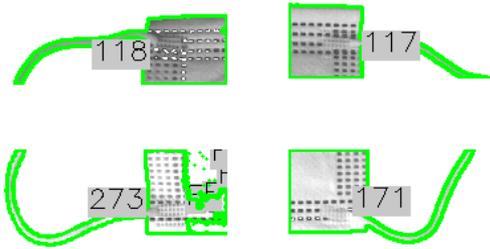
그림 9. Shape detection 적용 후 나타난 크롭된 이미지에서 꼭짓점의 개수
Fig. 9. Number of vertices in the cropped image that appears after applying shape detection

1.4 Shape Detection

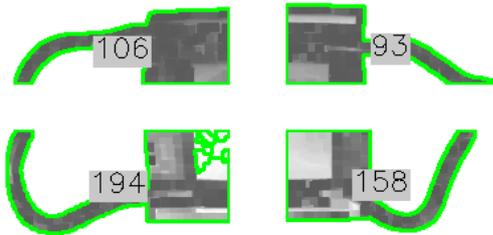
본 논문에서는 shape detection을 크롭된 마스크 4가지 이미지에서의 꼭짓점의 개수를 탐지하기 위하여 사용하였다. contour 함수를 통해 얻은 마스크 가장자리 부분으로 approxPolyDP 함수를 사용해 그림 9와 같이 크롭된 마스크 이미지에서의 꼭짓점의 개수를 찾아낸다. approxPolyDp 함수는 외곽선을 근사화 하기 위하여 사용된다. contour 근사는 곡선이 연속된 짧은 선분으로 근사화 될 수 있다는 가정에 근거한다. 이는 원래 곡선으로 정의된 점들의 집합으로 구성된 근사 곡선으로 이어진다. 마스크 이미지의 contour는 꼭짓점들의 집합으로 이루어져 있으므로 이미지의 꼭짓점의 개수를 shape detection을 통해 알아 낼 수 있다.

1.5 Defect Detection

Shape detection을 이용하여 마스크 이어링 부착상태의 불량률 결정 할 때, 꼭짓점의 수가 일정 threshold 미만일 때 불량 마스크를 결정하기 위해 epsilon 파라미터와 threshold 값을 결정해야 한다. 또한 침식 연산에서 커널 사이즈를 설정하는 것도 중요하다. 본 논문에서는 커널 사이즈를 7x7로 설정하였다. 만약 커널 사이즈가 이 값보다 낮으면 shape detection을 하는 동안 불필요한 마스크 가장자리들이 생겨서 그림 10 (a)처럼 안 좋은 성능을 가지게 된다. 마스크 이미지 가장자리의 길이는 미리 정해 둔 파라미터인 epsilon 값에 의해 결정된다. 그림 11 (a), 11 (b)와 같이 epsilon 값이 작을수록 contour 선이 original image와 가깝게 된다. 따라서 본 논문에서는 파라미터를 0.0005로 설정하였다. 그림 12과 같이 마스크에 이어링이 붙어 있다면 붙어 있지 않은 모서리 보다 꼭짓점의 개수가 많게 된다. 이로 인해, 일정 임계값 아래의 꼭짓점의 개수가 되면 불량으로 처리 하였다. Shape detection은 이미지의 도형마다 동작한다. 이 특징은 그림 13와 같이 마스크 이어링이 살짝만 뜯어져있는 경우에 사용

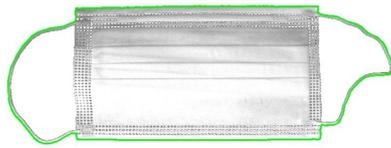


(a) without erosion calculation

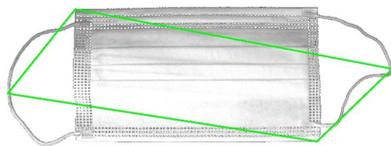


(b) using erosion calculation

그림 10. Shape detection에 침식연산을 적용
Fig. 10. Apply erosion calculation in shape detection



(a) epsilon parameter = 0.0005



(b) epsilon parameter = 0.05

그림 11. 마스크 이미지에 epsilon parameter 적용. 값이 작을수록 original image에 가까워진다.

Fig. 11. Epsilon parameter with mask images. The smaller the parameter, the closer the original image

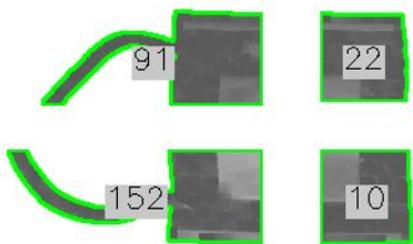


그림 12. Shape detection을 사용하여 마스크 이어링 부착 상태 불량 탐지. 초록색 선: contour선 근사, 숫자: 꼭짓점의 개수

Fig. 11. Detecting defective mask ear loops with shape detection. Green line: approximate contour line, Number; Number of vertices

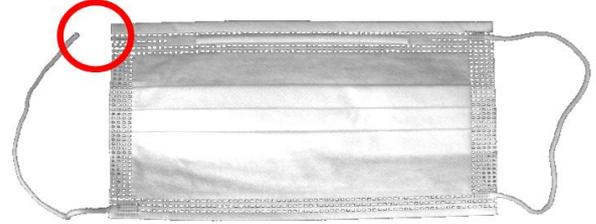
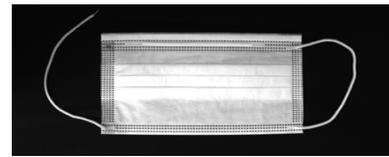
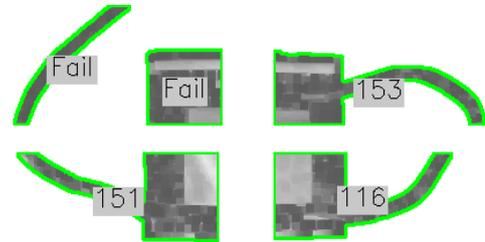


그림 13. 크롭된 이미지에서 2가지 도형이 탐지 되는 문제
Fig. 13. Shape detection problem when two shape is detected in one cropped image



(a) Slightly disconnected mask image



(b) Shape detection

그림 14. 마스크 이어링이 살짝 떨어져 있는 경우의 shape detection의 특징을 이용해 하나의 크롭된 이미지에 두가지 물체가 있을 때 문제 해결

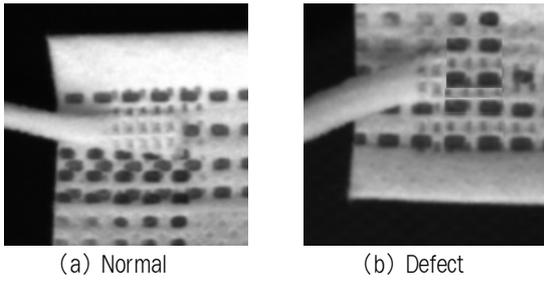
Fig. 14. Shape detection when the mask ear loops is slightly disconnected. Using a characteristic of shape detection solves the problem when two shape is detected in one cropped image

된다. 따라서 본 논문에서는 그림 14 (a), 14 (b)와 같이 하나의 이미지에 두 개의 도형이 탐지 되면 불량으로 처리하였다.

2. 마스크 이어링 부착량 검사

본 논문에서는 마스크 이어링 부착량 양불 기준을 부착부 길이가 5mm 이하인 경우 불량, 그 외에는 정상으로 정의하였고, 총 4개의 부착부가 전부 정상일 경우 해당 마스크를 정상으로 정의한다. 또한 이어링은 수평으로 부착된다고 가정하였다. 그림 15 (a), 15 (b)는 각각 마스크 부착부 정상과 불량량의 예시 이미지이다.

부착부 길이를 계산하기 위해 부착부 우측 지점과 좌측 지점의 x축 좌표를 추출한 후 그 차를 사용한다. 따라서 두 지점의 추출을 위한 이미지 전처리 과정이 개별적으로 진행된다. 이미지 전처리 과정 이후 template matching을 통해 마스크 부착부 4개의 범위를 추출하고 해당 범위를 크롭하게 된다. 이후 크롭된 이미지의 정규화 히스토그램을 계산하고 이를 통해 부착부의 우측, 좌측 지점을 판별하며 부착부 길이를 계산한다.



(a) Normal (b) Defect

그림 15. 마스크 이어링 부착부
Fig. 15. Mask ear loops attachment

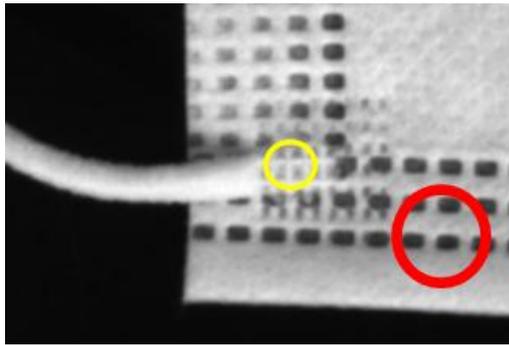


그림 16. 부착부 추출을 위해 사용되는 마스크의 특성
노란색: 패턴 A, 빨간색: 패턴 B

Fig. 16. Characteristics of the mask used for attachment amount extraction
yellow: pattern A, red: pattern B

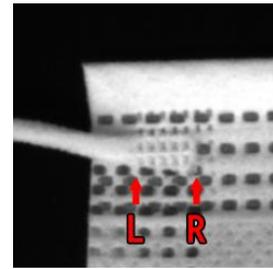
2.1 마스크 이미지 전처리

부착부 우측 지점과 좌측 지점 추출을 위해 사용되는 마스크 패턴의 특성이 서로 다르기 때문에 두 지점 추출을 위한 이미지 전처리 과정이 개별적으로 수행된다. 마스크 패턴은 크게 2가지 패턴이 있으며, 그림 16과 같이 주로 이어링 부착부에 생성되는 작은 사각형 패턴 (패턴 A), 그리고 마스크 본체에 생성되는 큰 사각형 패턴 (패턴 B)으로 이루어진다.

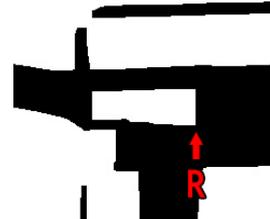
부착량 탐지를 위한 이미지 전처리에는 opening, closing 연산을 사용한다. 이는 erode와 dilate 연산으로 이루어지며 opening 연산의 경우 erode 이후 dilate, closing 연산의 경우 반대로 dilate 이후 erode를 수행하는 연산이다. erode와 dilate는 각각 식 (1), (2)로 표현되며 특정 패턴을 제거하거나 확장하는데 사용된다. x, y는 이미지의 좌표, i, j는 커널의 좌표를 나타내며 이미지에서 커널을 한 픽셀씩 이동시켜가며 연산이 수행된다. 본 논문에서는 3×3 크기의 커널을 사용하였다.

$$Dilate = \max_{(i,j) \in k} \{src(x+i, y+j)\}. \tag{2}$$

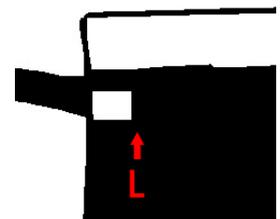
부착부 우측 지점 추출을 위한 이미지 전처리 과정은 다음과 같다. 먼저 입력 마스크 이미지를 바이너리 이미지로 변환시킨 후 closing 연산을 통해 패턴 A를 흰색으로 채운다. 이후 opening 연산을 통해 패턴 B를 확장시켜 이어링을



(a) Source



(b) Process 1



(c) Process 2

그림 17. 마스크 부착부 이미지 전처리
Fig. 17. Image preprocessing of mask attachment point



그림 18. 마스크 부착부의 크롭 범위
Fig. 18. Crop range of mask attachment

제외한 배경을 검정색으로 제거한다. 이러한 연산을 통해 부착부를 포함한 이어링이 흰색으로 강조되며, 그 결과는 그림 17 (b)와 같다.

부착부 좌측 지점 추출을 위한 이미지 전처리 과정은 우측 지점과는 다르게 closing 연산을 수행하지 않는다. 따라서 패턴 A, B 전부가 검은색으로 제거되어 부착부를 제외한 이어링 부분만이 강조되며, 그 결과는 그림 17 (c)와 같다.

2.2 부착부 크롭

전처리된 이미지를 이용하여 히스토그램 계산 시, 마스크 부착부 이외의 마스크 패턴이 히스토그램에 노이즈로 반영되어 정확도가 저하되는 문제를 방지하기 위해 부착부 크롭 과정을 수행한다.

부착부 크롭 범위 계산을 위해 먼저 template matching을 사용하여 마스크 본체의 꼭짓점 좌표 4개를 추출한다. 이후 해당 꼭짓점으로부터 특정 범위를 크롭하는데, 이 때 마스크 상단 부착부 2개와 하단 부착부 2개의 특성이 다르기 때문에 서로 다른 크기로 크롭을 수행한다. 상단 부착부 2개의 경우 패턴B가 4행으로 생성되는 반면, 하단 부착부 2개의 경우 패턴B가 3행으로 생성된다. 앞선 이미지 전처리

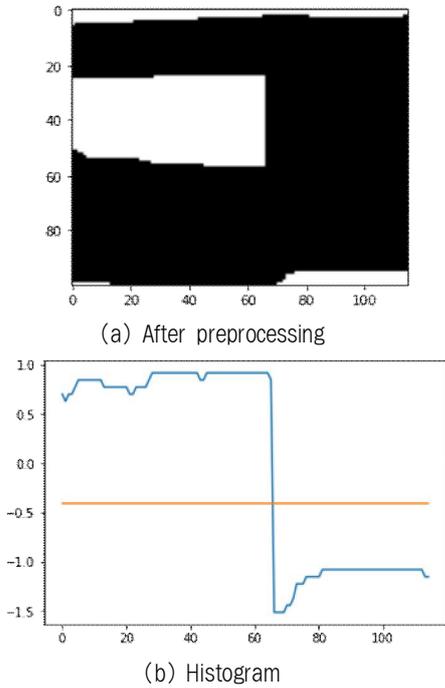


그림 19. 마스크 부착부 우측 지점 검출을 위한 히스토그램 생성
 파란색: 히스토그램, 주황색: 임계값
 Fig. 19. Histogram for detecting the right side of the mask attachment
 blue: histogram, orange: threshold

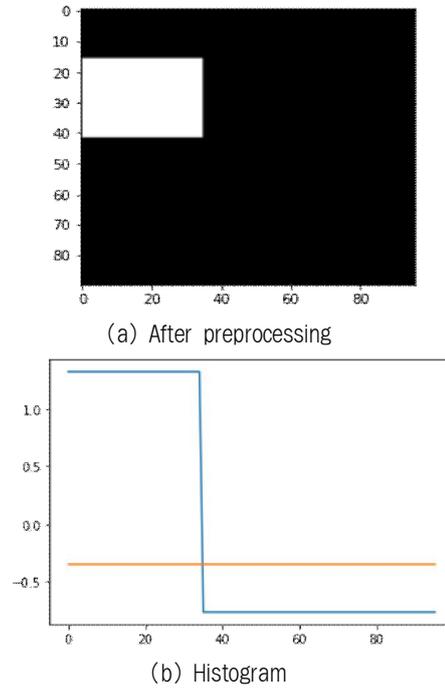


그림 20. 마스크 부착부 좌측 지점 검출을 위한 히스토그램 생성
 파란색: 히스토그램, 주황색: 임계값
 Fig. 20. Histogram for detecting the left side of the mask attachment
 blue: histogram, orange: threshold

과정에서 알 수 있듯이 해당 알고리즘은 패턴 B에 의존하여 수행된다. 따라서 하단 부착부를 크롭할 경우 상단과 달리 더 작은 높이로 크롭을 진행한다. 그림 18은 마스크에서 각각의 부착부 크롭 범위가 표시된 모습을 보여준다. 상단 2개의 부착부 범위는 가로 150px, 세로 100px 크기로 크롭되며, 하단 2개의 부착부 범위는 가로 150px, 세로 70px 크기로 크롭된다. 이후 하나의 알고리즘을 4개의 크롭 이미지에 일괄적으로 적용시키기 위해 대칭 연산을 수행한다. 따라서 상단 우측, 하단 좌측, 하단 우측 크롭 이미지는 상단 좌측 부착부와 동일한 형상을 다르게 변환된다.

2.3 부착량 검출

전처리 후 크롭된 이미지에서 부착부 우측, 좌측 지점을 추출하기 위해 히스토그램을 사용한다. 이 때 히스토그램은 정규화를 시켜, 고정된 임계값으로 부착부의 우측, 좌측 지점을 판별하게 된다.

부착부 우측 지점의 검출 과정은 다음과 같다. 먼저 앞서 전처리 후 크롭된 가로 150px의 이미지를 다시 한번 크롭하는 과정을 거친다. 좌측 35px를 잘라내어 가로 115px의 이미지로 변환시키며, 그 결과는 그림 19 (a)와 같다. 이 크롭 과정은 부착부 좌측 지점의 일부를 제거하여 이후 히스토그램을 이용한 우측 지점 판별의 정확도를 높이는 역할을 한다. 다음으로 히스토그램을 계산하기 위해 크롭된 이미지를 y축 방향으로 sum 연산을 수행한다. 마지막으로 해당 값에 평균을 뺀 후 표준편차로 나누어 정규화를 수행한다. 정규

화된 히스토그램은 그림 19 (b)의 파란색 그래프와 같으며, 가로축은 이미지의 x축 좌표, 세로축은 정규화된 히스토그램 값을 의미한다. 그림 19 (b)의 주황색 선은 임계값이며 -0.4를 적용했고, 임계값을 지나는 히스토그램의 가로축 값이 해당 부착부의 우측 지점 x축 좌표를 의미하게 된다.

부착부 좌측 지점의 검출 과정은 부착부 우측 지점 검출과 유사하다. 먼저 크롭 과정을 한번 더 거치게 되는데, 가로 150px의 이미지를 좌측 10px 지점부터 부착부 우측 지점까지 크롭을 하게 되며, 그 결과는 그림 20 (a)와 같다. 이후 정규화된 히스토그램은 그림 20 (b)의 파란색 그래프와 같으며, 주황색 선은 임계값으로 -0.35를 적용했다. 임계값을 지나는 히스토그램의 가로축 값이 해당 부착부의 좌측 지점 x축 좌표를 의미하게 된다.

부착부 우측 지점 x축 좌표와 부착부 좌측 지점 x축 좌표의 차가 픽셀 단위의 부착 길이가 되며, 픽셀 당 길이를 곱하여 최종적으로 부착 길이를 계산한다. 픽셀당 길이는 0.13mm로 정의한다.

IV. 실험 결과

본 논문에서는 python 3.7.9, opencv 4.5.3.56, 11th Gen Intel (R) Core (TM) i9-11900F의 환경에서 마스크 이어링 부착 상태 검사와 부착량 검사를 진행하였다.

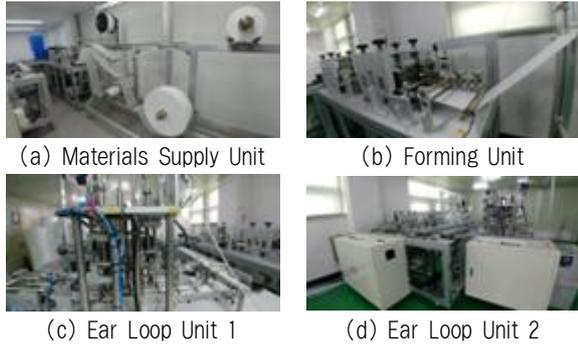


그림 21. 마스크 생산 공장. 부직포, 필터는 (a)에서 만들어진다. 마스크 성형은 (b)에서 이루어진다. 이어링은 (c)와 (d)에서 만들어진다.
 Fig. 21. Mask production factory. Non-Woven Fabric, Filter is made in (a). Mask Forming is conducted in (b). Ear loops are attached in (c), (d).

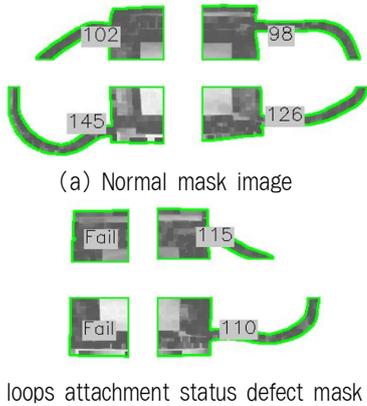


그림 22. 마스크 이어링 부착 상태 정상과 불량 이미지에 대한 이어링 부착 상태 검사 알고리즘 수행
 Fig. 22. Perform mask ear loops attachment status inspection algorithm for mask ear loops normal and defect image

1. 데이터 획득

공개된 마스크 데이터 셋이 없기 때문에 본 논문에서는 그림 21 (a), 21 (b), 21 (c), 21 (d)의 마스크 생산 공장에서 비전 시스템으로 촬영한 마스크 이미지 데이터 셋을 사용하였다. 실험에 사용된 정상 마스크 이미지는 800장이고 이어링 부착 상태 불량 마스크 이미지 100장, 부착량 불량 마스크 이미지 100장이다.

2. 마스크 이어링 부착 상태 검사 결과

실험은 앞서 그림 3과 같은 순서대로 shape detection을 통해 모서리의 개수를 얻고, 그림 22 (a), 22 (b)와 같이 임계값을 기반으로 정상과 불량을 판단하여 이어링 부착 상태 검사를 수행한 다음, 정상으로 판단된 이미지들을 다시 이어링 부착량 탐지를 수행하여 0.5cm이하의 길이를 불량으로 처리하는 방식으로 수행한다. 이어링 부착 상태 검사를 수행할 때, shape detection을 수행하기 위해 template matching을

표 2. 혼동 행렬을 통한 마스크 이어링 부착 상태 탐지 기법 결과 비교

Table 2. Comparison of results of mask ear loops attachment inspection technique through confusion matrix

	TP	FN	FP	TN	Precision	Recall
Proposed method	899	1	0	100	1	0.99
H-T based inspection	687	213	35	65	0.95	0.75
M-T based inspection	892	8	29	71	0.96	0.99

통해 정확한 마스크 4 모서리의 좌표를 얻어내는 과정이 중요하다. 이어링 부착 상태 검사 기법의 비교를 위해 hough transform (H-T)을 사용해 마스크의 직선성분의 좌표를 얻어내 실제 마스크의 가로, 세로 길이를 더해 4 모서리의 좌표를 얻어내는 방법과 moment function (M-F)을 통해 이미지의 무게 중심 값을 구하여 마스크의 4 모서리의 좌표를 가로 세로 길이 만큼 더하여 얻어내는 방법을 수행하였다. 정상 800장, 마스크 이어링 부착 상태 불량 100장, 이어링 부착량 불량 100장에 대하여 마스크 이어링 부착 상태 검사를 진행하였을 때의 결과는 표 2와 같다. H-T를 기반으로 한 방법에서 Hough transform을 사용할 때는 직선 성분을 검출하기 위한 임계값을 고정해야하는데 이미지마다 동일한 임계값을 사용하면 직선 성분을 제대로 검출 하지 못하는 경우가 생긴다. 이러한 경우에는 직선성분의 최솟값 최댓값이 잘못 측정되기 때문에 잘못 크롭된 이미지가 생성되어 상대적으로 정상과 불량을 잘 구분해내지 못하는 모습을 보여준다. M-T를 기반으로 한 방법에서 moment function은 contour함수로 구한 마스크 이미지 외곽성분의 무게중심을 구해야 하는데 한쪽 이어링이 없는 이미지에 대해서는 무게중심이 치우쳐져서 잘못된 크롭 이미지를 생성하게 된다. 이로 인해, 정상일 때는 판단을 잘하지만 불량일 때는 상대적으로 정확도가 낮은 모습을 보인다. 본 논문에서 제안한 기법으로 이어링 부착 상태 검사를 수행할 경우 참 값을 올바르게 분류하는지에 대한 지표인 정밀도 (Precision)는 1이 나오며 실제 참 값인 것들 중에 참값으로 예측이 잘된 비율인 재현율 (Recall)은 0.99가 나온다. 수행시간은 단일 이미지를 기준으로 하였을 때 0.12초가 소요된다. 다중영상 기반으로 이미지를 처리하지 않았을 때는 4개의 이미지에 알고리즘을 수행하는데 1.03초가 소요되지만 다중영상 기반으로 알고리즘을 수행할 경우 0.48초가 소요되어 수행시간을 훨씬 단축 시킬 수 있었다.

3. 마스크 이어링 부착량 검사 결과

그림 23 (a), 그림 23 (b)는 각각 정상 마스크와 불량 마스크의 이어링 부착량 검사 알고리즘 수행 예시를 보여준다. 빨간선은 부착부의 우측 지점을 의미하고, 파란선은 부착부의 좌측 지점을 의미하며, 해당 부착부가 정상일 경우 녹색 텍스트, 불량일 경우 적색 텍스트로 부착 길이가 표시



(a) Normal mask



(b) Defective mask

그림 23. 정상 마스크와 불량 마스크의 부착량 검사 결과
Fig. 23. The attachment amount inspection result of normal mask and defective mask

표 3. 혼동 행렬을 통한 마스크 이어링 부착량 탐지 결과
Table 3. Results of mask ear loops attachment amount inspection through confusion matrix

	TP	FN	FP	TN	Precision	Recall
Proposed method	798	1	0	100	1	0.99

되었다. 정상 마스크인 그림 23 (a)에 대해서는 4개의 부착부 부착 길이가 모두 5mm를 초과하여 정상으로 분류 되었다. 반면 불량 마스크인 그림 23 (b)에 대해서는 하단 좌측 부착부가 다른 부착부에 비해 부착 길이가 짧은 것을 볼 수 있으며, 본 논문이 제안하는 마스크 부착량 검사 알고리즘이 해당 부착부를 불량으로 옳게 식별한 것을 볼 수 있다.

마스크 이어링 부착량 검사는 마스크 이어링 부착 상태 검사가 선행된 후 수행되는 알고리즘이다. 따라서 마스크 이어링 부착량 검사 알고리즘의 성능을 분석하기 위하여, 마스크 이어링 부착 상태 검사를 선행해 정상으로 판별된 마스크에 대하여 마스크 이어링 부착량 검사 알고리즘을 수행하고 그 결과를 분석한다.

본 논문에서는 총 1000장의 마스크 중 앞서 마스크 이어링 부착 상태 검사에서 정상으로 판별된 899장의 마스크를 대상으로 알고리즘 결과를 분석하며, 전체 899장의 마스크는 정상 마스크 799장과 마스크 이어링 부착량 불량인 마스크 100장으로 구성되어 있다. 마스크 이어링 부착량 검사 알고리즘 수행 결과는 표 3과 같다. Precision은 1이 나오며 Recall은 0.99가 나온다. 수행시간은 단일 이미지를 기준으로 하였을 때 0.07초가 소요된다. 다중영상 기반으로 이미지를 처리하지 않았을 때는 4개의 이미지에 알고리즘을 수행하는데 0.42초가 소요되지만 다중영상 기반으로 알고리즘을 수행할 경우

0.28초가 소요되어 수행시간을 훨씬 단축 시킬 수 있었다.

V. 결론

본 논문에서는 여러 가지 전처리 과정과 shape detection을 활용해 다양한 종류의 마스크 이어링 검사를 수행하였다. 다른 기법들과 비교하였을 때 본 논문에서 제시한 방법을 사용한 방법을 수행할 경우 가장 높은 정확도를 보여줄 수 있었다. 마스크 이어링 부착 상태 검사에서 정상으로 분류된 이미지를 히스토그램과 이미지 전처리를 활용한 방법으로 0.5cm이하의 마스크 이어링 부착량 검사도 수행하였다. 마스크 이어링 부착상태 검사와 마스크 이어링 부착량 검사에서 precision의 값은 1, recall의 값은 0.99가 나온다. 수행시간의 경우 다중영상 기반으로 알고리즘을 수행할 경우 단일영상 기반의 알고리즘 수행에 비하여 속도가 개선된 것을 확인할 수 있었다. 실험은 실제 마스크 생산 공장에서 생산된 마스크 이미지 데이터를 기반으로 수행하였으며 수행 결과 높은 정확도를 보여주고 있다.

References

- [1] H. Lee, H. Lee, "Average Blurring-based Anomaly Detection for Vision-based Mask Inspection Systems," 2021 21st International Conference on Control, Automation and Systems (ICCAS), 2021, pp. 2144-2146, doi: 10.23919/ICCAS52745.2021.9649945.
- [2] S. Li, W. Song, L. Fang, Y. Chen, P. Ghamisi, J. A. Benediktsson, "Deep Learning for Hyperspectral Image Classification: An Overview," in IEEE Transactions on Geoscience and Remote Sensing, Vol. 57, No. 9, pp. 6690-6709, Sept. 2019, doi: 10.1109/TGRS.2019.2907932.
- [3] R. Chauhan, K. K. Ghanshala, R. C. Joshi, "Convolutional Neural Network (CNN) for Image Detection and Recognition," 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC), 2018, pp. 278-282, doi: 10.1109/ICSCCC.2018.8703316.
- [4] S. B. Yang, S. J. Lee, "Improved CNN Algorithm for Object Detection in Large Images," Journal of the Korea Society of Computer and Information, Vol. 25, No. 1, pp. 45 - 53, 2020.
- [5] N. Nasaruddin, K. Muchtar, A. Afdhal, A. P. J. Dwiyanoro, "Deep Anomaly Detection Through Visual Attention in Surveillance Videos," Journal of Big Data Vol. 7, No. 1, pp. 1-17, 2020.
- [6] S. Mayannavar, U. Wali, V. M. Aparanji, "A Novel ANN Structure for Image Recognition", arXiv:2010.04586, 2020.
- [7] K. Madani, "Artificial Neural Networks Based Image Processing & Pattern Recognition: From Concepts to Real-World Applications," 2008 First Workshops on Image Processing Theory, Tools and Applications, pp. 1-9, doi: 10.1109/IPTA.2008.4743797, 2008.

- [8] C. Song, F. Yang, P. Li, "Rotation Invariant Texture Measured by Local Binary Pattern for Remote Sensing Image Classification," 2010 Second International Workshop on Education Technology and Computer Science, pp. 3-6, doi: 10.1109/ETCS.2010.37, 2010.
- [9] D. Huang, C. Shan, M. Ardabilian, Y. Wang, L. Chen, "Local Binary Patterns and Its Application to Facial Image Analysis: A Survey," in IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), Vol. 41, No. 6, pp. 765-781, doi: 10.1109/TSMCC.2011.2118750, 2011.
- [10] F. Guo, J. Yang, Y. Chen, B. Yao, "Research on Image Detection and Matching Based on SIFT Features," 2018 3rd International Conference on Control and Robotics Engineering (ICCRE), pp. 130-134, doi: 10.1109/ICCRE.2018.8376448, 2018.
- [11] R. N. Satare, S. R. Khot, "Image Matching with SIFT Feature," 2018 2nd International Conference on Inventive Systems and Control (ICISC), pp. 384-387, doi: 10.1109/ICISC.2018.8399100, 2018.
- [12] M. Muthugnanambika, S. Padmavathi, "Feature Detection for Color Images Using SURF," 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS), pp. 1-4, doi: 10.1109/ICACCS.2017.8014572, 2017.
- [13] Z. Zhu, G. Zhang, H. Li, "SURF Feature Extraction Algorithm Based on Visual Saliency Improvement." International Journal of Engineering and Applied Sciences, Vol. 5, No. 3, 257267, Mar. 2018.
- [14] J. Canny, "A Computational Approach to Edge Detection," in IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-8, No. 6, pp. 679-698, 1986.
- [15] R. J. Danos, A. R. Frey, Y. Wang, "Canny Algorithm: A New Estimator for Primordial Non-Gaussianities", arXiv:1108.2265, 2011.
- [16] J. Malik, S. Belongie, T. Leung, J. Shi, "Contour and Texture Analysis for Image Segmentation", International Journal of Computer Vol. 43, No. 1, pp. 7-27, 2001.
- [17] S. Liu, X. Peng, Z. Liu, "Image Quality Assessment through Contour Detection," 2019 IEEE 28th International Symposium on Industrial Electronics (ISIE), pp. 1413-1417, doi: 10.1109/ISIE.2019.8781416, 2019.
- [18] P. Mukhopadhyay, B. B. Chaudhuri, "A survey of Hough Transform", Pattern Recognition, Vol. 48, No. 3, pp. 993-1010, 2015.
- [19] D. Duan, M. Xie, Q. Mo, Z. Han, Y. Wan, "An Improved Hough Transform for line Detection," 2010 International Conference on Computer Application and System Modeling (ICCSM 2010). Vol. 2. IEEE, pp. V2-354, 2010.
- [20] N. S. Hashemi, R. B. Aghdam, A. S. B. Ghiasi, P. Fatemi, "Template Matching Advances and

Applications in Image Analysis", arXiv:1610.07231, 2016.

- [21] Y. Yu, Z. Tu, L. Lu, X. Chen, H. Zhan, Z. Sun, "An Improved Template Matching Method for Object Detection", Proceedings of the 29th ACM International Conference on Multimedia, pp. 2753, 2021.

JiMyeong Woo (우지명)



2021 Electronic Engineering from Kumoh National Institute of Technology (B.S.)

2021~Department of IT Convergence Engineering, Kumoh National Institute of Technology (M.S. Candidate)

Field of Interest: AI, Image Processing

Email: k2361505@naver.com

SangHyeon Lee (이상현)



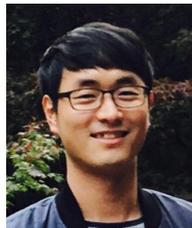
2022 Electronic Engineering from Kumoh National Institute of Technology (B.S.)

2022~Research Institute of Manufacturing and Productivity, Kumoh National Institute of Technology (Researcher)

Field of Interest: AI

Email: freeg159@kumoh.ac.kr

Heoncheol Lee (이현철)



2006 Electronic-Electrical Engineering and Computer Sciences from Kyungpook National University (B.S.)

2008 Electrical Engineering and Computer Sciences from Seoul National University (M.S.)

2013 Electrical Engineering and Computer Sciences from Seoul National University (Ph.D.)

2019~Department of IT Convergence Engineering, School of Electronic Engineering, Kumoh National Institute of Technology (Assist. Prof.)

Career:

2011 Researcher, ASRI, Seoul National University

2013 Senior Researcher, Agency for Defense Development

2019 Technical Adviser, LG Electronics

Field of Interests: Vision-based Anomaly Detection, SLAM, Path Planning, Algorithm Acceleration, Deep Learning

Email: hclee@kumoh.ac.kr

논문 2022-17-41

텍스트-비디오 검색 모델에서의 캡션을 활용한 비디오 특성 대체 방안 연구

(A Study on the Alternative Method of Video Characteristics Using Captioning in Text-Video Retrieval Model)

이 동 훈*, 허 찬*, 박 혜 영*, 박 상 호*

(Dong-hun Lee, Chan Hur, Hyeyoung Park, Sang-hyo Park)

Abstract : In this paper, we propose a method that performs a text-video retrieval model by replacing video properties using captions. In general, the existing embedding-based models consist of both joint embedding space construction and the CNN-based video encoding process, which requires a lot of computation in the training as well as the inference process. To overcome this problem, we introduce a video-captioning module to replace the visual property of video with captions generated by the video-captioning module. To be specific, we adopt the caption generator that converts candidate videos into captions in the inference process, thereby enabling direct comparison between the text given as a query and candidate videos without joint embedding space. Through the experiment, the proposed model successfully reduces the amount of computation and inference time by skipping the visual processing process and joint embedding space construction on two benchmark dataset, MSR-VTT and VATEX.

Keywords : Multimodal Deep Learning, Video-Captioning, Text-Video Retrieval

1. 서론

텍스트-비디오 검색은 텍스트가 입력으로 주어질 때 데이터베이스안의 후보 비디오들에서 의미적으로 일치하는 비디오를 찾아내는 과제이다. 그림 1은 텍스트-비디오 검색에 대한 예시로 왼쪽의 텍스트 쿼리를 입력으로 받았을 때 그에 맞는 비디오를 찾는 텍스트-비디오 검색 결과를 시각적으로 설명하고 있다. 최근 급격히 성장한 Netflix, YouTube 등의 비디오 스트리밍 시장과 더불어 많은 주목을 받고 다양한 연구가 이루어지고 있는 분야이다. 이러한 과제를 해결하기 위해 많은 연구들이 제안되고 있는데 가장 널리 사용되는 방법은 임베딩 기반 접근방법이다. 임베딩 기반 접근 방법은 비디오와 텍스트의 특징을 추출하여 같은 공간상에서 매핑하는 방법으로 비디오와 텍스트라는 종류가 다른 두 데이터를 특징 추출 모듈을 통해 특징을 추출한 후 공통되는 공간상에 임베딩시킨다.

기존의 공통 임베딩 기반 연구에서는 임베딩 하고자 하는 비디오와 텍스트를 각각 별도의 특징 추출 모듈을 이용하여 특징을 추출한다. 예를 들어, 비디오의 경우 CNN 계열 모델

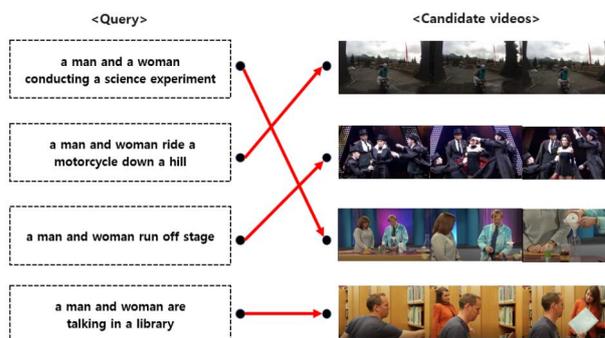


그림 1. 텍스트-비디오 검색에 대한 예시
Fig. 1. Examples of text-video retrieval

을 인코더로 사용하여 프레임 단위로 이미지의 시각적인 특징벡터를 추출하고, 텍스트의 경우에는 순차정보를 얻을 수 있는 RNN 계열 인코더를 기반으로 하여 문장의 특징벡터를 추출한다. 이러한 특징벡터들을 임베딩 공간상에서 맵핑하는 경우 각각 텍스트와 이미지라는 다른 데이터의 형태를 공통 공간에 임베딩하기 때문에 두 데이터의 분포 차이에서 오는 격차문제 [1]가 발생하게 된다. 또한 이와 같은 구조를 이용하는 선행 연구들 [2-4]에서는 텍스트 인코더 대비 비디오의 시각정보 처리 과정에서의 연산량이 급격히 증가하게 된다. 비디오의 경우 프레임 단위별로 이미지를 기반으로 한 시각적인 정보를 담고 있을 뿐만 아니라 프레임의 연속적인 내용을 담고 있는 시간적인 정보도 담고 있어 인코딩시 많은 연산량을 요구한다. 그로 인해 비디오 시각 정보

* These authors contributed equally to this work.

*Corresponding Authors (hypark@knu.ac.kr; s.park@knu.ac.kr)
Received: Oct. 20, 2022, Revised: Nov. 11, 2022, Accepted: Nov. 26, 2022.
D. Lee: Kyungpook National University (Undergraduate Student)
C. Hur: Kyungpook National University (PhD Student)
H. Park: Kyungpook National University (Prof.)
S. Park: Kyungpook National University (Asst. Prof.)
* 본 논문은 2020년도 정부 (교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임 (No. 2020R111A3072227).
* 이 논문의 연구 결과 중 일부는 한국통신학회 "제3회 한국 인공지능 학술회"에서 발표한 바 있음.

를 처리하기 위해 여러 복잡한 과정이 더해지게 되어 전체 모델의 크기가 커지게 된다. 우리는 이러한 비디오 시각처리 부분에서 생기게 되는 많은 연산 과정을 줄이고자 비디오의 특성을 생성모듈을 통한 캡션으로 대체하여 텍스트-비디오 검색을 진행하는 모델을 제안한다.

제안 모델은 비디오의 특성을 비디오의 정보를 표현하는 캡션(텍스트)으로 변환한 뒤 이 캡션과 입력으로 받는 쿼리 텍스트를 비교하여 검색을 하게 된다. 쿼리와 동일한 유형의 텍스트 데이터를 처리할 수 있게 되므로, 복잡한 비디오 시각처리 과정을 요구하지 않기 때문에 기존에 비디오와 텍스트를 임베딩할 때 각각의 특성의 차이에서 생기는 격차 문제를 줄일 수 있게 된다. 이를 통해 모델을 경량화시킬 수 있으며 그로 인해 추론 단계에서 사용하는 연산량 및 연산시간도 줄어들게 된다. 제안 방법의 성능을 검증하기 위해서 MSR-VTT, VATEX 2개의 벤치마크 데이터셋에서 실험 및 분석을 진행하였다.

II. 관련 연구

1. 텍스트-비디오 검색

텍스트-비디오 검색은 크게 개념기반, 공통 임베딩 공간 기반 방법으로 나뉘게 된다. 개념 기반 방법의 대표적인 예시는 Ad-hoc Video Search (AVS) [5]가 있다. AVS는 시각적 개념 분류기와 언어 규칙에 의존하며, 텍스트 쿼리와 특정 비디오 사이의 유사성을 개념 일치률 통해 계산을 한다. AVS의 경우 비디오와 텍스트 쿼리를 개념으로 표현해 주기 때문에 어느 정도 해석이 가능하다는 장점이 있지만, 비디오와 쿼리 둘 다를 통해 전체적인 관계가 매핑된 정보를 설명하는 것이 매우 어렵다는 단점도 있다.

이러한 문제를 해결하기 위해 가장 많이 사용되는 방법은 공통 임베딩 기반 방법 [2-4, 6]이다. 공통 임베딩 기반 방법은 비디오 및 텍스트 쿼리를 인코딩 한 후 공통의 잠재된 공간상에 위치시켜 학습 및 추론에 사용하는 방법이다. 많은 연구들이 이 과정에서 사전 훈련된 CNN 기반 모델을 이용한 정교한 비디오 인코딩 과정을 통해 비디오를 특정 프레임 단위로 나눠 이미지 레벨에 있는 시각적 특징들을 추출해 비디오 피처로 정의하는 인코딩된 결과물로 표현하였다. 초기 연구에는 비디오 프레임간의 시간 순서와 가중치를 고려하지 않고 평균 풀링 [2]을 통해 집계하는 방법을 사용하였으나 점차 비디오 프레임의 시간 정보를 명시적으로 모델링하기 위해 LSTM [4], GRU [5], 최댓값 풀링 [7, 8] 또는 멀티헤드-셀프 어텐션 [9] 등을 사용하여 시간 순서와 프레임간의 중요도를 모델링하였다.

상술한 많은 방법들은 비디오의 시각적인 정보와 시간적인 정보를 담기 위해 인코딩 과정에서 여러 모듈을 사용하게 된다. 이런 과정은 비디오의 길이에 따라 유동적으로 바뀔 뿐 아니라 많은 연산량을 가지게 되어 모델이 복잡해져 추론 과정에서조차 시간과 자원을 많이 소모하는 단점을 가진다. 우리가 제안하는 모델은 비디오의 특성을 캡션으로

대체하는 방법을 사용하기 때문에 입력으로 받는 쿼리 텍스트와 후보 비디오의 표현 형태가 동일한 형태를 가지게 된다. 이러한 같은 형태의 데이터의 이용은 비디오 시각 인코더를 생략하기 때문에 비디오 입력에 대한 연산량을 상당히 감소시켰다.

2. 비디오-캡셔닝

비디오-캡셔닝이란 비디오가 주어질 때 이를 언어적으로 설명하는 캡션 문장을 생성하는 과제이다. 최근 딥러닝을 활용한 자연어 처리와 컴퓨터 비전의 발전을 통해 많은 비디오 캡셔닝 모델들이 제안되고 있으며 많은 연구 [10-12]가 인코더-디코더 기반의 구조를 사용하고 있다. 인코더-디코더 기반의 방법은 입력으로 받은 텍스트와 비교 대상이 되는 비디오를 각각 별도의 인코더에 넣어서 인코딩 과정을 거친 뒤 비교를 하여 나온 결과물을 디코더에 넣어서 보여 주는 방법이다.

인코더에서 비디오의 시각적인 정보를 인코딩하기 위해 주로 CNN 계열 모델들을 인코딩 모듈로 활용한다. CNN을 활용한 방법 [13]은 비디오를 프레임 단위로 나누어 특성을 추출한다. 이때 일부 모델 [14, 15]은 어텐션 메커니즘을 사용하여 각 프레임간의 연관성을 찾기도 하며, 오디오의 정보도 활용하는 방식도 있다. 또한 모든 프레임이 아닌 특정 프레임만을 채택해서 캡션을 만드는 방법도 있다. 디코더에서 텍스트의 특성을 추출하는 과정에는 주로 RNN [13] 계열 방법들을 사용한다. 문장에서의 단어의 순서를 결정하는데 있어 연속적인 데이터의 특징을 잘 추출하는 RNN 기반의 방법을 주로 사용하며, 최근에는 어텐션 기법을 사용하거나 Transformer 모델 [7]을 사용하여 비디오 캡션을 생성하는 방법들 [10, 16, 17]도 존재한다.

본 논문에서는 Transformer를 사용한 인코더-디코더 구조의 비디오 캡션 모델 [10]을 이용하여 비디오의 특성으로 활용하였다. 우리는 이러한 비디오-캡셔닝의 결과로 생성된 캡션이 비디오의 내용과 의미적으로 같지만 텍스트로 표현한다는 점에 주목하였다. 다른 관점에서 비디오를 표현하는 생성된 캡션 정보를 이용하여 텍스트-비디오 검색 문제에서 비디오의 특징을 효율적으로 표현하는 방법을 제안하였다.

III. 제안 방법

3.1. 특징 추출 과정

본 논문에서는 텍스트-비디오 검색에서 인코딩 과정을 위한 기본 모델로 듀얼 인코더 모델 [18]의 구조를 사용하였다. 그림 2 하단부에서 볼 수 있듯이, 텍스트가 입력으로 주어질 때, 특징 추출 모듈 $\phi(t)$ 는 3단계의 특징추출과정을 거치며 다음과 같이 표현된다.

$$\phi(t) = [L_1(t), L_2(L_1(t)), L_3(L_2(L_1(t)))]. \quad (1)$$

단, 여기서 t 는 입력으로 받는 텍스트 쿼리, L_1, L_2, L_3 는

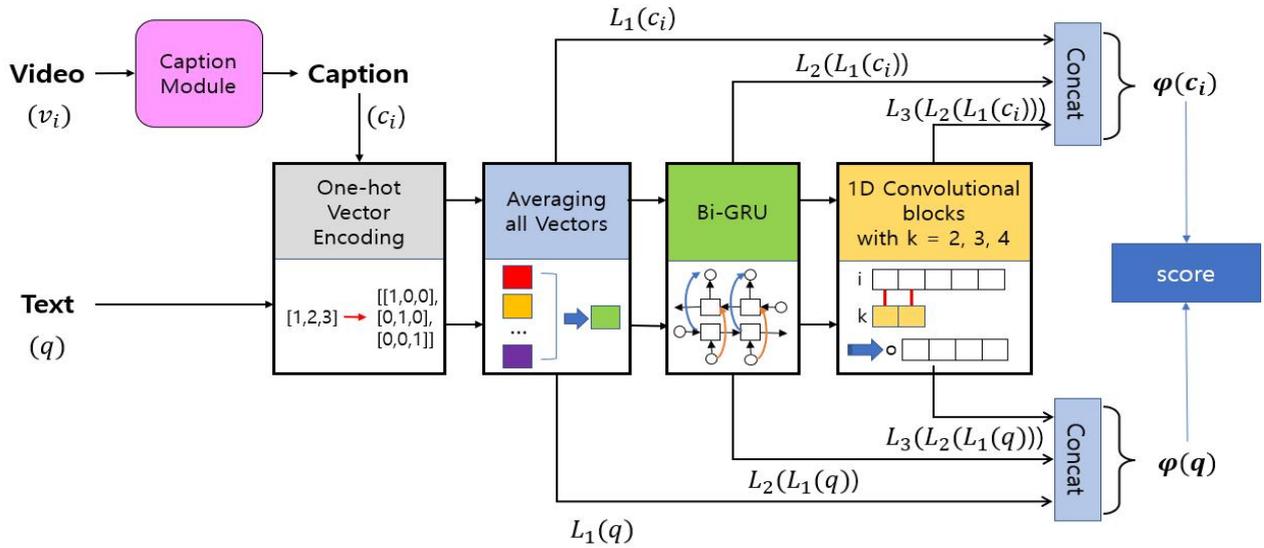


그림 2. 제안하는 텍스트-비디오 검색 모델
Fig. 2. Proposed text-video retrieval model

텍스트의 특징 추출과정 3단계를 말하며, $[\cdot, \cdot]$ 는 L_1, L_2, L_3 의 결합한다는 것을 의미한다. 우선 입력으로 텍스트(t)가 주어지면 이를 원-핫 벡터로 표현한다. 그리고 첫 번째 단계 (L_1)에서 이 벡터의 평균을 구해 텍스트를 전체적으로 표현하는 글로벌 인코딩 벡터 값을 얻는다. 다음, 두 번째 단계 (L_2)에서는 L_1 을 통과한 글로벌 벡터를 bi-GRU를 통과시켜 텍스트의 순차적인 정보를 표현하는 벡터를 얻는다. 세 번째 단계 (L_3)에서는 L_2 를 통과한 벡터를 1-D CNN을 통과시켜 텍스트의 지역정보를 반영하는 벡터를 얻는다. 최종적인 인코딩 결과로는, 이 세 단계에서 나온 벡터들을 결합하여 특징 벡터로 정의한다. 이러한 정교한 특징 추출 과정은 3.2장에서 도입한 비디오 특징을 캡션으로 대체한 생성 캡션들에 대해서도 같은 방법으로 진행되어진다.

3.2. 캡셔닝 모듈 선택

비디오 정보를 시각적으로 표현하는 방법 대신, 우리는 비디오의 정보를 생성 모델을 통해 얻어진 언어적 정보를 이용하여 표현하였다. 이러한 표현 방법은 입력 비디오가 쿼리와 같은 특성 (텍스트)으로 맞춰진다는 점에서 착안되었고 이를 표현 방법을 통해 얻은 장점을 3.3장에서 설명하도록 한다.

캡셔닝 생성 모델로는 UniVL [10]을 사용하였다. UniVL은 비디오 캡셔닝에 사용하는 인코더-디코더 기반의 사전학습 모델로 비디오와 텍스트를 이해하고 캡션을 생성하기 위해 4개의 모듈로 설계되었다. 4개의 모듈은 3개의 인코더와 1개의 디코더로 구성되어 있으며, 비디오와 텍스트를 각각 트랜스포머 인코더로 처리한 다음, 전처리한 비디오와 텍스트를 결합해 또 다른 트랜스포머를 기반해 만든 교차 인코더에 넣는다. 마지막으로 인코딩한 값을 디코더에 넣어 최종적인 캡션을 생성하게 된다. 실험에 사용하는 학습데이터셋으로 UniVL에 파인-튜닝해 얻어진 캡션 결과물

을 듀얼 인코딩 모델의 비디오 특성으로 넣어 사용하였다.

3.3. 캡션정보를 이용한 추론 과정

3.2절에서 선택된 캡션 모듈을 통해 추론 과정에서 후보 비디오들은 생성한 캡션들로 대체된다. 이는 그림 2에서 비디오가 Caption Module로 캡션을 생성하는 부분에 해당한다. i 번째 후보 비디오 데이터 캡션을 생성하는 식은 다음과 같다.

$$c_i = Gen(v_i). \tag{2}$$

여기서 Gen 는 캡션 생성 모듈, v_i 는 입력 비디오 데이터를 말하며, 결과적으로 c_i 는 i 번째 입력 비디오를 텍스트 정보로 표현하는 캡션이다. 이를 통해 생성한 캡션은 입력으로 받는 텍스트와 동일한 형태를 지닌 데이터로 표현되기 때문에 별도의 임베딩 공간 없이도 유사도 계산 커널을 통해 유사성을 계산할 수 있게 된다. 따라서 쿼리로 주어진 캡션과 후보 데이터셋의 각 데이터와의 유사도를 비교하는 과정은 다음과 같다.

$$score_i = \psi(\phi(q), \phi(c_i)). \tag{3}$$

여기서 q 는 질의로 주어진 텍스트 쿼리, ϕ 는 텍스트 인코더를 의미하고 $\psi(\cdot, \cdot)$ 는 두 텍스트의 유사도를 비교하기 위한 코사인 유사도 커널을 의미한다. 주어진 쿼리에 대해 이러한 과정을 후보 데이터셋의 모든 후보 비디오에 대해 실행하여 얻은 결과 중 가장 높은 스코어 값을 가지는 후보 비디오를 쿼리에 대한 검색 결과로 판단한다.

기존의 복잡한 시각적 인코딩 모델을 이용하여 임베딩 공간을 학습시켜 추론에 이용한 것과 다르게, 제안 모델은 유사도를 계산할 때 텍스트들을 입력으로 받기 때문에 별도 임베딩 공간에 학습할 필요 없이 추출된 특징을 바로 매칭에 사용할 수 있다. 그러므로 임베딩 공간을 이용하는 모델

에서 지적되어 왔던 이중 간의 데이터 (비디오, 텍스트)가 임베딩 공간에서 표현될 때 분포 차이가 나는 문제 [19]를 완화할 수 있다는 장점을 가진다. 또한 비디오 표현을 위해 시각 인코더 등 많은 모듈을 사용하는 모델 [2, 3] 에 비해 경량화된 모델이기 때문에 연산속도의 향상을 볼 수 있다.

IV. 실험 결과

4.1 실험 환경

실험을 위해 텍스트-비디오 모델에서 사용하는 대표적인 데이터셋인 MSR-VTT와 VATEX 데이터셋을 사용하였다. MSR-VTT 데이터 세트는 10,000개의 비디오 클립과 이 클립을 설명하는 200,000개의 문장으로 구성되며, 한 클립당 20개의 문장으로 구성되어 있다. VATEX에는 유튜브에서 수집된 10초정도 길이를 가진 34,991개의 동영상 데이터 세트이다. 비디오 한 개당 10개의 영문과 10개의 중국어 문장이 할당되는데 이 실험에서는 영어 문장만 사용한다.

우리는 각각의 학습 데이터 셋을 UniVL을 통해 학습시켰다. 이후 추론 과정에서 테스트 데이터 셋에 대한 캡션들을 만든 뒤 이 캡션들을 비디오의 특성 부분 대신 사용하였다. 실험은 ubuntu 18.04 환경에서 NVIDIA RTX3060 1개를 GPU로 두고 진행하였다. 성능평가의 방법으로 R@K (K = 1, 5, 10), MedR, mAP 까지 총 5개의 순위 기반 측정 지표를 사용했다. R@K는 검색된 상위 K개의 결과 중 하나 이상의 관련 항목이 발견된 테스트 쿼리의 백분율이며, Medr은 검색 결과에서 첫 번째 관련 항목의 중간값이다.

4.2 벤치마크 데이터셋 실험 및 정량적 평가

표 1에서는 MSR-VTT 데이터셋에 대한 실험결과를 보여주고 있다. MSR-VTT를 데이터셋으로 하여 제안모델과 W2VV [20], MEE [1]를 비교 모델로 두어 R@1, R@5, R@10, MedR, mAP를 평가지표로 선정해 실험을 진행하였으며, W2VV와 MEE의 실험결과는 Dual Encoding for Video Retrieval by Text [18]의 실험을 참고하였다.

실험결과를 보면 R@1과 MedR 지표에 대해서는 W2VV, MEE 모델과 다르게 제안 모델이 가장 좋은 성능을 보여주는 것을 확인할 수 있다. 또한 R@5, R@10, mAP지표도 가장 높은 성능을 보이는 MEE모델에 약간 낮은 성능을 보이고 있지만, MEE 모델은 비디오의 특성을 다루기 위해 복잡한 시각 인코딩 과정을 포함하고 있기 때문에 캡션정보로 대체하여 비디오와 텍스트를 한 종류의 데이터로 연산하는 제안 모델이 연산량과 연산 시간에서 강점을 보이며, 다른 모델과 유사하거나 높은 성능을 보인다고 해석할 수 있다.

표 2에서는 VATEX 데이터셋에 대한 실험결과를 보여주고 있다. VATEX를 데이터셋으로 하여 이전과 동일하게 제안모델과 W2VV [20], VSE++ [21]를 비교 모델로 두어 R@1, R@5, R@10을 평가지표로 선정해 실험을 진행하였으며, W2VV와 VSE++의 실험결과는 Dual Encoding for Video Retrieval by Text [18]의 실험을 참고하였다.

실험결과를 보면 가장 높은 성능을 보이는 VSE++ 모델

표 1. MSR-VTT 데이터셋에 대한 텍스트-비디오 검색 결과
Table 1. Text-video retrieval results on MSR-VTT dataset

Model	R@1	R@5	R@10	MedR	mAP
W2VV [19]	1.1	4.7	8.1	23.6	0.037
MEE [20]	6.8	20.7	31.1	28.0	0.1470
Proposed method	6.9	20.1	29.3	36.0	0.1426

표 2. VATEX 데이터셋에 대한 텍스트-비디오 검색 결과
Table 2. Text-video retrieval results on VATEX dataset

Model	R@1	R@5	R@10
W2VV [19]	14.6	36.3	46.1
VSE++ [21]	31.3	65.8	76.4
Proposed method	31.0	64.0	75.2

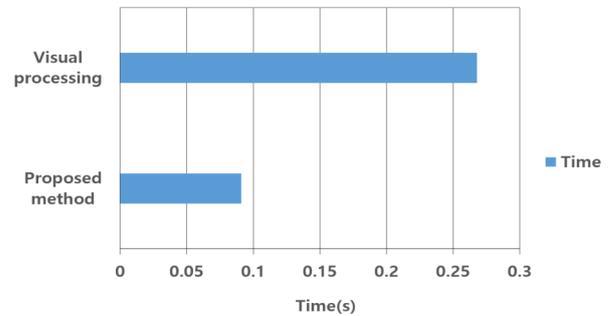


그림 3. 시각정보 처리모델과 제안모델의 텍스트-비디오 검색 시간 비교 결과

Fig. 3. Comparison result of text-video retrieval time between a visual processing model and the proposed model

에 비해 큰 차이를 보이지 않지만, VSE++는 추가적인 샘플링을 통한 학습과정과 시각 인코딩 과정을 거치며 많은 연산이 필요하다. 따라서 캡션 정보를 이용하여 어느 정도 성능을 보장하며 연산량을 줄인 제안 모델이 마찬가지로 큰 강점을 보이는 것을 확인할 수 있다.

그림 3은 MSR-VTT 데이터셋에 대한 시각 정보 처리 모델과 제안 모델의 텍스트-비디오 검색 시간 즉, 모델의 추론시간 비교 그래프를 보여주고 있다. 시각정보 처리 모델은 제안 모델과 동일한 추출단계를 이용해 비디오에서 시각적인 특징을 추출해 비디오의 특성으로 이용하는 모델이다. 1개의 쿼리 텍스트를 입력해 텍스트의 내용을 가장 잘 담는 비디오를 검색하는데 걸리는 시간까지를 측정하여 비교한 결과를 보여주고 있는데, 제안모델이 0.09초 듀얼 인코딩 모델이 0.26초가 측정되어 제안모델이 시각정보를 처리하는 모델에 비해 약 3분의 1 적은 검색 시간 값을 가진다. 이러한 결과는 임베딩 환경 등의 연산 속도나 연산량이 중요한 분야에서 비디오 검색 문제를 다룰 때 복잡한 비디오 시각처리 인코더 대신 캡션 정보를 사용할 수 있다는 가능성을 시사한다.

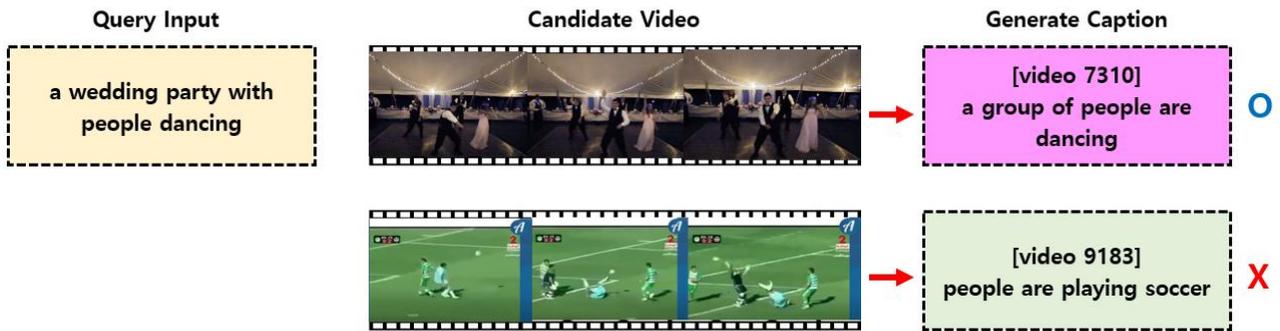


그림 4. MSR-VTT 데이터셋의 추론 결과 예시
 Fig. 4. Example of inference result on MSR-VTT dataset

4.3 정성적 평가

그림 4에서는 MSR-VTT 테스트 데이터셋에 넣어서 나온 모델의 추론결과 예시를 보여주고 있다. 텍스트 쿼리와 후보 비디오 셋 (2990개)를 입력으로 넣었을 때 캡션을 이용한 유사도를 산출한 영상의 결과를 보여주고 있으며, 쿼리로 'a wedding party with people dancing'를 넣은 결과 video7310을 통해 만든 캡션인 'a group of people are dancing'이 가장 높은 score를 얻어 다른 비디오 특징보다 검색 결과에 가깝다고 예측하였고 정답 비디오 클립과 일치하였다. 이러한 결과를 통해 video의 시각적 인코더를 사용하지 않더라도, 캡션 생성 모델을 통하여 비디오를 잘 표현하는 텍스트를 얻을 수만 있다면 텍스트-비디오 검색에서 좋은 특징으로 사용가능함을 볼 수 있다.

V. 결론

본 논문에서는 비디오를 이용해 만든 캡션 정보를 비디오의 특성으로 활용해 텍스트-비디오 검색 문제를 해결하는 모델을 제안하였다. 기존모델의 경우 비디오의 특성을 처리하기 위해 다양한 처리 방법을 사용한 탓에 두 데이터간 차이에서 오는 격차 문제나 연산량이 복잡하다는 단점을 가지고 있었다면, 제안 모델에서는 비디오의 정보를 담아낼 수 있는 캡션을 생성하고, 비디오의 특성을 캡션으로 정의함으로써 텍스트의 특성과 직접적인 비교를 할 수 있어 공통 임베딩 공간을 구축할 필요가 없다는 장점을 가진다. 또한 많은 양의 연산을 요구하는 비디오의 시각 인코딩 과정을 생략함으로써 기존의 모델 대비 경량화된 장점을 보여주었다.

제안 모델을 더 개선하기 위해, 텍스트 특징 추출을 큰 코퍼스를 이용해 학습된 BERT [22] 등의 사전 학습한 모델을 활용한다면 더욱 좋은 성능을 낼 수 있을 것으로 기대된다. 또한 생성 모델을 이용한 캡션이 의미 있는 정보로 활용할 수 있다는 것을 실험을 통해 확인하게 되었고, 이를 비디오의 시각적 정보와 결합하여 더 발전된 검색 모델을 만들어 낼 수 있는 가능성을 보여준다.

References

- [1] A. Miech, I. Laptev, J. Sivic, "Learning a Text-video Embedding from Incomplete and Heterogeneous Data," arXiv preprint arXiv:1804.02516, 2018.
- [2] N. C. Mithun, J. Li, F. Metze, A. K. Roy-Chowdhury, "Learning Joint Embedding with Multimodal Cues for Cross-modal Video-text Retrieval," in ICMR, pp. 19-27, 2018.
- [3] X. Li, C. Xu, G. Yang, Z. Chen, J. Dong, "W2VV++: Fully Deep Learning for Ad-hoc Video Search," in ACM Multimedia, pp. 1786-1794, 2019.
- [4] A. Torabi, N. Tandon, L. Sigal, "Learning Language-visual Embedding for Movie Understanding with Natural-language," arXiv preprint arXiv:1609.08124, 2016.
- [5] G. Awad, J. Fiscus, D. Joy, M. Michel, A. Smeaton, W. Kraaij, G. Quenot, M. Eskevich, R. Aly, R. Ordelman, G. Jones, B. Huet, M. Larson, "TRECVID 2016: Evaluating Video Search, Video Event Detection, Localization, and Hyperlinking," in TRECVID Workshop, 2016.
- [6] X. Yang, J. Dong, Y. Cao, X. Wang, M. Wang, T. S. Chua, "Tree-augmented Cross-modal Encoding for Complex-query Video Retrieval," in SIGIR, pp. 1339-1348, 2020.
- [7] A. Miech, D. Zhukov, J. B. Alayrac, M. Tapaswi, I. Laptev, J. Sivic, "Howto100m: Learning a Text-video Embedding by Watching Hundred Million Narrated Video Clips," in ICCV, pp. 2630-2640, 2019.
- [8] M. Wray, D. Larlus, G. Csurka, D. Damen, "Fine-grained Action Retrieval Through Multiple Parts-of-speech Embeddings," in ICCV, pp. 450-459, 2019.
- [9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, "Attention is all you need," in NIPS, pp. 5998-6008, 2017.
- [10] H. Luo, L. Ji, B. Shi, H. Huang, N. Duan, T. Li, J. Li, T. Bharti, M. Zhou, "Univ: A Unified Video and Language Pre-training Model for Multimodal Understanding and Generation," arXiv preprint arXiv:2002.06353, 2020.

- [11] B. Pan, H. Cai, D. A. Huang, K. H. Lee, A. Gaidon, E. Adeli, J. C. Niebles, "Spatio-temporal Graph for Video Captioning with Knowledge Distillation," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10870-10879, 2020.
- [12] W. Pei, J. Zhang, X. Wang, L. Ke, X. Shen, Y. W. Tai, "Memory-attended Recurrent Network for Video Captioning," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8347-8356, 2019.
- [13] S. Venugopalan, M. Rohrbach, J. Donahue, R. J. Mooney, T. Darrell, K. Saenko, "Sequence to Sequence - Video to Text," in Proc. IEEE Int. Conf. Comput. Vis., pp. 4534-4542, 2015.
- [14] L. Gao, Z. Guo, H. Zhang, X. Xu, H. T. Shen, "Video Captioning with Attention-based LSTM and Semantic Consistency," IEEE Trans. Multimedia, Vol. 19, No. 9, pp. 2045-2055, 2017.
- [15] J. Song, Z. Guo, L. Gao, W. Liu, D. Zhang, H. T. Shen, "Hierarchical Lstm with Adjusted Temporal Attention for Video Captioning," arXiv preprint arXiv:1706.01231, 2017.
- [16] L. Zhou, Y. Zhou, J. J. Corso, R. Socher, C. Xiong, "End-to-end Dense Video Captioning with Masked Transformer," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., pp. 8739-8748, 2018.
- [17] L. Huang, W. Wang, J. Chen, X. Wei, "Attention on Attention for Image Captioning," in Proc. IEEE Int. Conf. Comput. Vis., pp. 4634-4643, 2019.
- [18] J. Dong, X. Li, C. Xu, X. Yang, G. Yang, X. Wang, M. Wang, "Dual Encoding for Video Retrieval by Text," IEEE Transactions on Pattern Analysis and Machine Intelligence . Vol. 44, No. 8, pp. 4065-4080, 2021.
- [19] X. Wang, L. Zhu, Y. Yang, "T2vIad: Global-local Sequence Alignment for Text-video Retrieval," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5079-5088, 2021.
- [20] J. Dong, X. Li, C. G. Snoek, "Predicting Visual Features from Text for Image and Video Caption Retrieval," IEEE Transactions on Multimedia, Vol. 20, No. 12, pp. 3377-3388, 2018.
- [21] F. Faghri, D. J. Fleet, J. R. Kiros, S. Fidler, "VSE++: Improved Visual-semantic Embeddings," in BMVC, 2018, pp. 1-13.
- [22] J. Devlin, M. W. Chang, K. Lee, K. Toutanova, "Bert: Pre-training of Deep Bidirectional Transformers for Language Understanding," arXiv preprint arXiv:1810.04805 (2018).

Dong-hun Lee (이 동 훈)



2017~Computer Science and Engineering from
Kyungpook National University (B.S.)

Field of Interests: Video-Text Retrieval, Video Captioning
Email: hy05205@naver.com

Chan Hur (허 찬)



2019~2021 Computer Science and Engineering
from Kyungpook National University
(M.S.)

2021~Computer Science and Engineering from
Kyungpook National University (Ph.D.)

Field of Interests: Video-Text Retrieval, Multimodal Learning
Email: chanhur94@gmail.com

Hyeyoung Park (박혜영)



2004~Computer Science and Engineering from
Kyungpook National University (Prof.)

Career:

2000~2004 Researchers at Brain Science Institute, RIKEN, Japan

2009~2010 Visiting Professor, Texas A&M University

Field of Interests: Neural Networks, Learning Theory

Email: hypark@knu.ac.kr

Sang-hyo Park (박상효)



2011 Computer Engineering from Hanyang University (B.S.)

2017 Computer Science from Hanyang University (Ph.D)

Career:

2017~2018 Postdoctoral position, Korea Electronics Technology Institute (KETI)

2018 Research Fellow, Yonsei University

2019~2020 Postdoctoral position, Ewha Womans University

2020~ Assistant Professor with the School of Computer Science and Engineering, Kyungpook National University

Field of Interest: VVC, encoding complexity, immersive video, model optimization

Email: s.park@knu.ac.kr

논문 2022-17-42

실시간 탄도 궤적 목표물 추적을 위한 GPU 기반 병렬적 입자군집최적화 기법

(Parallelized Particle Swarm Optimization with GPU for Real-Time Ballistic Target Tracking)

한 윤 호, 이 현 철*, 권 혁 훈, 최 원 석, 정 보 라

(Yunho Han, Heoncheol Lee, Hyeokhoon Gwon, Wonseok Choi, and Bora Jeong)

Abstract : This paper addresses the problem of real-time tracking a high-speed ballistic target. Particle filters can be considered to overcome the nonlinearity in motion and measurement models in the ballistic target. However, it is difficult to apply particle filters to real-time systems because particle filters generally require much computation time. This paper proposes an accelerated particle filter using graphics processing unit (GPU) for real-time ballistic target tracking. The real-time performance of the proposed method was tested and analyzed on a widely-used embedded system. The comparison results with the conventional particle filter on CPU (central processing unit) showed that the proposed method improved the real-time performance by reducing computation time significantly.

Keywords : Ballistic target tracking, Graphics processing unit, Particle swarm optimization, Real-time systems

1. 서 론

탄도 궤적을 가진 목표물 추적 및 요격의 성능은 얼마나 정확하게 목표물을 추적할 수 있는지에 따라 판단할 수 있다. 목표물의 위치, 각도 등의 상태를 정밀하게 추정할 수 있는 알고리즘을 채택하여 표적을 추적하여야 한다. 일반적으로 목표물의 상태 추정 시 발생하는 모델 노이즈는 수학적 단순성을 고려할 때 가우스 분포를 가진다고 가정된다. 그러나 레이돔 탐색기 (Seeker radome)와 섬광 (Scintillation) 등에 의하여 발생하는 측정 모델 노이즈는 비선형 및 비가우스 특성을 가지기 때문에 노이즈의 가우스 분포에 대한 가정은 적용되기 힘들다 [1, 2]. 비선형 및 비가우스 특성의 불확실성으로 인하여 기존의 몇몇 필터링 알고리즘들은 만족스럽지 못한 성능이 나타날 수 있다는 것으로 알려져 있다. 선형 칼만 필터 기반의 표적 추적 알고리즘들은 표적의 상태를 추정하는 도중 적절한 값에 수렴하지 않거나 심지어 값이 분기될 수 있다.

노이즈의 비선형 및 비가우스 특성으로 인하여 발생하는 문제들을 해결하기 위하여 Extended Kalman filter (EKF), Particle filter (PF), unscented Kalman filter (UKF) 등의 다양한 비선형 필터들이 목표 상태 추정을 위하여 적용되었

다. 또한 최적화 기법도 비선형 및 비가우스 노이즈가 있는 환경에서 목표의 상태 추정을 위하여 적용될 수 있다. 그중 입자 군집 최적화 기법은 다양한 유형의 오류 분포를 처리할 수 있는 능력을 가지고 있어 목표물의 상태 추정에 적용하여 연구되고 있다. 또한 입자 군집 최적화 기법은 사용되는 입자들이 서로 정보를 교환하면서 최적점을 찾아 나아가기 때문에 몇 개의 입자들이 지역적 최소점 (local minimum)에 빠지더라도 전체적으로는 전역 최적점 (global optimum)을 수렴할 수 있다는 장점이 있다. 그러나 입자 군집 최적화 기법 (Particle Swarm Optimization)을 사용할 때 가장 큰 제한점은 최적의 값을 찾기 위해서는 입자의 수와 알고리즘의 계산 반복 횟수에 비례하여 성능이 나타난다는 점이다. 따라서 실시간 시스템에 입자 군집 최적화 기법을 적용하기 위해서는 계산 문제에 대한 해결책이 마련되어야 한다는 제약점이 있다.

본 논문에서는 실시간 탄도 목표물의 정밀한 추적 및 요격을 위하여 비선형 및 비가우스 노이즈 환경에서 상태 추정이 가능하도록 입자 군집 최적화를 사용한다. 그러나 입자 군집 최적화 기법을 사용하여 목표물의 상태를 정밀하게 추정하기 위해서는 많은 수의 입자들과 알고리즘의 반복 횟수가 필요하다. 입자 군집 최적화 기법은 샘플링 기반의 알고리즘으로써, 사용되는 입자 수와 반복 횟수가 커질수록 알고리즘 실행 시 소요되는 시간도 늘어난다. 따라서 실시간성을 갖도록 하기 위해서는 입자 군집 최적화 기법의 가속화가 필수적이다. 본 논문에서 입자 군집 최적화 중 계산 시간이 많이 소요되는 부분을 식별한 후, 그 부분에 대하여 GPU와 CUDA를 사용한 병렬화를 적용하였다. 또한 탄도 목표물 요격체에서 입자 군집 최적화 기법을 사용하여 상태

*Corresponding Author (hcleee@kumoh.ac.kr)

Received: Oct. 14, 2022, Revised: Nov. 15, 2022, Accepted: Nov. 26, 2022.

Y. Han: Department of IT Convergence Engineering, Kumoh National Institute of Technology (M.S. Candidate)

H. Lee: Department of IT Convergence Engineering, Kumoh National Institute of Technology (Assist. Prof.)

H. Gwon: PGM R&D Lab, LIGNEX1 (Chief Research Engineer)

W. Choi: PGM R&D Lab, LIGNEX1 (Chief Research Engineer)

B. Jeong: PGM R&D Lab, LIGNEX1 (Research Engineer)

* 이 연구는 광역방어 특화연구센터 프로그램(UD200043CD)의 일환으로 국방과학연구소와 방위사업청의 지원으로 수행되었음.

추적을 한다는 가정하에 PC환경이 아닌 온보드 환경에 적합하도록 NVIDIA Jetson Xavier를 사용하여 가속화가 진행되었다. 따라서 임베디드 환경에서 입자 군집 최적화를 CPU와 GPU의 상호설계하여 실시간성을 향상시키며 탄도 목표물 상태 추정에 성공하였다.

표 1은 그중 대상 추적, 객체 추적, 모션 추적 등의 다양한 목표 상태 추적 연구에 입자 군집 최적화 기법이 사용된다는 것을 보여준다. 미사일 응용에 사용되는 관련 연구도 발견된다. GPU를 이용한 가속화 방법은 여러 분야에서 적용할수 있다 [3]. 입자 군집 최적화 기법의 가속화는 연구들은 실시간 임무 완수를 위하여 GPU를 CUDA와 함께 사용하여 수행되며, 입자 군집 최적화 기법의 가속화 연구는 대부분 모션 추적 연구에서 진행되었다. 입자 군집 최적화 기법은 목표물의 상태 추적이 아닌 검출 등 다른 방면에서도 가속화 연구가 진행되고 있다 [4, 5]. GPU를 사용하여 입자 군집 최적화를 가속화한 연구들은 대부분 사용된 입자들이 가지고 있는 정보의 에러를 계산하는 부분이나 비용함수를 계산하는 부분을 가속화 하였다. 본 논문에서의 입자 군집 최적화 기법은 탄도 목표물을 추적하는 목적에 적용되어 다른 연구들과 어플리케이션이 차이가 있다. 때문에 다른 연구들과 달리 알고리즘 중 연산시간이 가장 많이 소요된 입자들이 어느 지점으로 움직여야 하는지 판단하는 부분과 우도함수 계산부분을 가속화 하였다.

본 논문의 나머지 부분은 다음과 같이 구성되어 있다. 우선 입자 군집 최적화 기법과 최적화 기법을 사용하여 탄도 궤적 목표물 추적과 입자 군집 최적화 기법이 실시간성을 가지기에 제한점에 대하여 설명한다. 다음으로 입자 군집 최적화 알고리즘의 블록 단위로 프로파일링하여 각 부분마다 소요되는 연산시간을 구한 후 각 입자가 가지는 정보를 업데이트하는 부분과 입자들이 가지는 정보를 종합하여 최적의 값을 찾는 부분에 대한 병렬화 방법이 제안된다. 마지막으로 임베디드 시스템에서 제안된 방법의 결과를 병렬화하지 않은 알고리즘과 제안된 방법을 사용한 알고리즘의 결과를 비교한 후 결과들에 대한 결론을 제시한다.

표 1. 목표 추적에 적용된 입자 군집 최적화 관련 연구들
Table 1. Related works to PSO for target tracking

Related Works	GPU-based parallelization	Parallelization part	Missile Application
[6], [7], [8], [9]	X	-	X
[10], [11]	X	-	O
[12]	O	calculation of the error	X
[13]	O	calculation of the cost func	X
Ours	O	predicted measurement / associated likelihood func	O

II. 탄도 궤적 목표물 추적

1. 표본 추출 기반 탄도 궤적 목표물 추적

특정 대상을 추적하는 알고리즘의 주 목표는 실제 추적하는 대상의 상태를 실시간으로 추정하는 것이다. 본 연구의 대상인 탄도 미사일의 추적 알고리즘의 성능을 평가하기 위해서는 먼저 탄도 미사일 궤적의 모사가 요구된다. 대기권 밖에서의 비행과 달리 대기권으로의 재돌입 한 후에는 중력과 더불어 항력 등의 공기역학적 힘이 탄도 미사일의 경로를 결정하는 데 중요한 영향력을 미친다. 본 연구에서는 탄도 미사일을 점 질량으로 가정하였으며, 3차원 데카르트 좌표계에서 기술하였다. 다음은 중력 및 공기역학적 힘을 포함한 3차원 비선형 운동 방정식을 나타낸다.

(1), (2), (3)의 함수에서 x, y, z 는 위치를 나타내며 V 는 속도 나타내고, γ 와 ψ 는 각각 비행경로각과 헤딩각을 의미한다. 또한, m 은 질량, g 는 중력 상수를 나타내며, T, D, L 은 각각 추력, 항력, 양력을 나타낸다. 공기역학적 힘은 공기 밀도 ρ , 항력 계수 C_D , 양력 계수 C_L 그리고 기준 넓이 S 로 이루어져 있다. 그리고, ϕ 는 양력을 발생 방향을 나타낸다. 본 연구에서 고려하는 재진입 단계에서는 일반적으로 추진기관의 연소가 끝난 상태이기 때문에 추력은 0으로 설정하였으며, 질량도 일정하다고 가정하였다. 또한, 대기권 내에서 일반적으로 탄도 미사일의 기동이 미미하고 항력에 비하여 상대적으로 영향이 작기 때문에 양력을 0으로 가정하였다.

$$\begin{aligned} \dot{x} &= V \cos \gamma \cos \psi \\ \dot{y} &= V \cos \gamma \sin \psi \\ \dot{z} &= -V \sin \gamma \end{aligned} \tag{1}$$

$$\begin{aligned} \dot{V} &= \frac{(T - D - mg \sin \gamma)}{m} \\ \dot{\gamma} &= \frac{(L \cos \phi - mg \cos \gamma)}{mV} \end{aligned} \tag{2}$$

$$\begin{aligned} \dot{\psi} &= \frac{L \sin \phi}{mV \cos \gamma} \\ D &= \frac{1}{2} \rho V^2 \cdot C_D \cdot S \\ L &= \frac{1}{2} \rho V^2 \cdot C_L \cdot S \end{aligned} \tag{3}$$

2. 목표물의 움직임 및 측정 모델

목표물 추적은 기준 운동 모델을 기반으로 추정 알고리즘을 설계하므로 몇 가지 모델을 고려할 수 있다. 본 논문에서는 잘 알려져 있는 Singer 모델을 기반으로 연구되었다 [14, 15]. Singer 모델은 목표물의 가속도가 평균이 0인, 1차 정적 마르코프 프로세스라고 가정한다. 연속적인 시간에서 Singer 모델의 상태 공간 표현은 (4)와 (5)의 식으로 나타낼 수 있다. (4)의 식에서 구할 수 있는 x 는 추적 목표물의 상태를 의미한다. (4)와 (5)의 식에서 w 는 평균이 0이며, 시정수 τ 로 얻어지는 백색 가우스 노이즈이다. 그리고 F 와 G 행렬의 I_3 와 τ 는 3차 항등 행렬과 기동 상수이다. w 의 이산 시간 방정식은 (6)과 (7)의 식으로 나타낸다. 여기서 Φ_k 와 Δt 는 상태 전이 행렬과 샘플링 시간 간격을 의미한다.

$$\dot{x} = Fx + Gw \quad (4)$$

$$F = \begin{bmatrix} 0_3 & I_3 & 0_3 \\ 0_3 & 0_3 & I_3 \\ 0_3 & 0_3 & -\frac{I_3}{\tau} \end{bmatrix}, \quad G = \begin{bmatrix} 0_3 \\ 0_3 \\ I_3 \end{bmatrix} \quad (5)$$

$$x_k = \Phi_{k-1} x_{k-1} + w_{k-1}, \quad w_{k-1} \sim \mathcal{N}(0, Q_k) \quad (6)$$

$$\Phi_k \approx I + F\Delta t \quad (7)$$

$$Q_k \approx S_w Q_0 = S_w \begin{bmatrix} \frac{\Delta t^5}{20} I_3 & \frac{\Delta t^4}{8} I_3 & \frac{\Delta t^3}{6} I_3 \\ \frac{\Delta t^4}{8} I_3 & \frac{\Delta t^3}{3} I_3 & \frac{\Delta t^2}{2} I_3 \\ \frac{\Delta t^3}{6} I_3 & \frac{\Delta t^2}{2} I_3 & \Delta t I_3 \end{bmatrix} \quad (8)$$

(6)의 공분산 Q_k 은 전력 스펙트럼 밀도 (Power Spectral Density)인 S_w 와 백색 노이즈 jerk 모델인 Q_0 로 구성된다. 일정 시간 동안 가속이 증가하는 크기는 해당 시간 동안의 jerk의 적분을 나타낸다.

(4)의 식에서 상태 변수 x 는 다음 (9)의 식처럼 데카르트 좌표계에서 위치, 속도 및 가속도를 나타내는 벡터인 P , V 그리고 A 를 사용하여 정의한다. (10)의 식에서 $[x, y, z]$ 은 데카르트 좌표계에서 목표물의 위치를 의미한다.

$$x = [P^T \ V^T \ A^T]^T \quad (9)$$

$$P = [x \ y \ z]^T \quad (10)$$

목표물의 측정값인 고도, 방위각, 그리고 거리는 탄도탄

탐지 레이더에 의하여 측정된다고 가정한다. 이 측정값들은 표적과 레이더의 상대 위치에 따라서 얻을 수 있다. (11)의 식에서의 아래 첨자 r 는 목표 대상과 레이더 사이의 상대적 위치를 나타내고, m 은 레이더 위치를 의미한다. 따라서 두 개의 베어링 각도 z_θ 와 z_ψ , 상대 거리 z_R 는 레이더 측정 오차 등을 포함하여 다음 식 (12)와 같이 나타낼 수 있다. 레이더 측정 오차는 가우시안 잡음인 레이더 수신기 잡음 n_θ , n_ψ 그리고 n_R 와 비가우시안 잡음인 글린트 (Glint) 잡음 $n_{G,\theta}$ 와 $n_{G,\psi}$ 로 구성된다 [16].

$$[x_r \ y_r \ z_r]^T = [x \ y \ z]^T - [x_m \ y_m \ z_m]^T \quad (11)$$

$$\begin{bmatrix} z_\theta \\ z_\psi \\ z_R \end{bmatrix} = \begin{bmatrix} \tan^{-1} \left(\frac{z_r}{\sqrt{x_r^2 + y_r^2}} \right) + n_{G,\theta} + n_\theta \\ \tan^{-1} \left(\frac{y_r}{x_r} \right) + n_{G,\psi} + n_\psi \\ \sqrt{x_r^2 + y_r^2 + z_r^2} + n_R \end{bmatrix} \quad (12)$$

III. 문제점 기술

1. 목표물 추적 문제

탄도 미사일과 같은 고속의 목표물 경우 추적 알고리즘의 업데이트율과 추정 정확도가 매우 중요한 부분이다. 성공적인 요격을 위한 정밀 유도 및 제어는 얼마나 정확히 표적을 추적할 수 있는지에 크게 영향을 받는다. 본 연구에서는 입자 군집 최적화 알고리즘이 목표 추적의 높은 추정 정확도

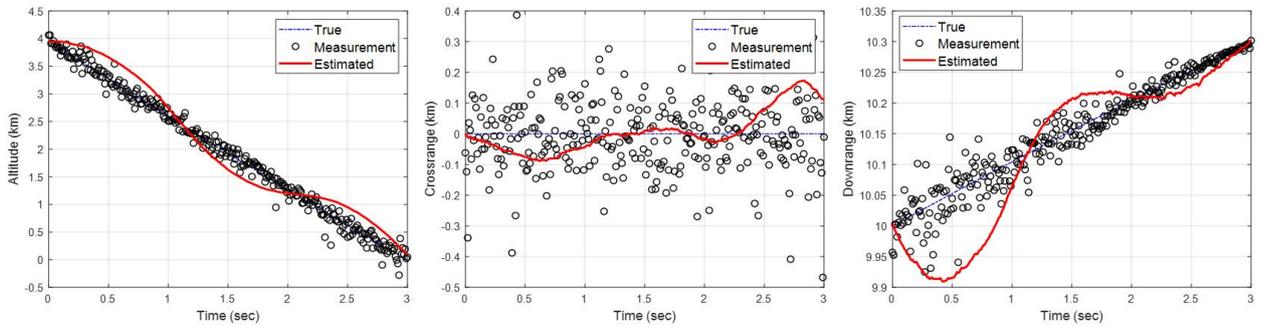


그림 1. 100개의 입자를 사용한 입자 군집 최적화 탄도 목표물 추적 결과
Fig. 1. Result of the ballistic target tracking with PSO using 100 particles

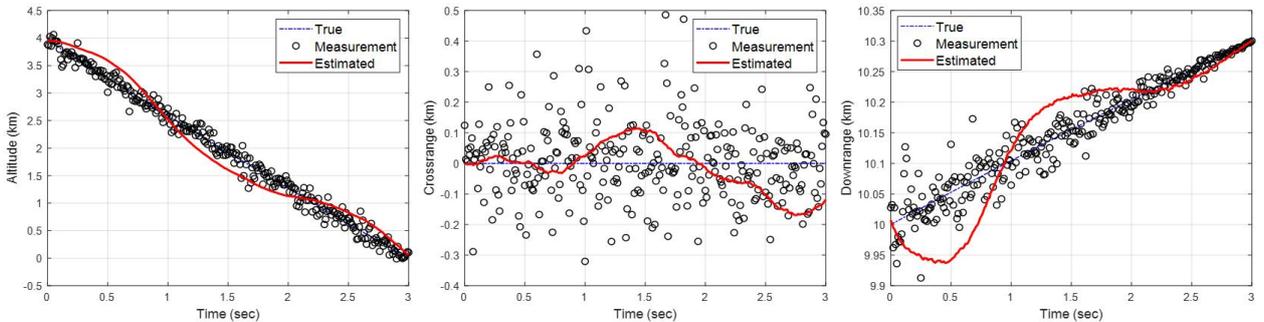


그림 2. 200개의 입자를 사용한 입자 군집 최적화 탄도 목표물 추적 결과
Fig. 2. Result of the ballistic target tracking with PSO using 300 particles

를 위하여 사용되었다. 탄도 궤적을 가지는 목표물을 추적하기 위하여 입자 군집 최적화를 적용하였을 때, 충분한 수의 입자를 사용하여야 높은 정확도를 가지는 추정을 할 수 있다. 입자 군집 최적화에서 충분하지 않은 수의 입자들을 사용하였을 때의 목표물 추적 결과는 그림 1과 그림 2에 나타난다. 그림 1은 탄도 목표물 추적에 100개의 입자를 사용한 입자 군집 최적화 결과이며 그림 2는 200개의 입자를 사용하였을 때이다. 두 그림 모두 파란 선으로 나타난 목표물의 실제 상태를 정확하게 추정하는데 실패하였다. 그림 1보다 그림 2에서 목표물의 실제 값과 추정 값 간 차이가 좀더 작았다. 따라서 목표물의 정확한 상태 추정을 위하여 보다 많은 수의 입자가 필요하다는 것을 알 수 있다.

2. 실시간성 문제

탄도 궤적 목표물을 요격하기 위해서는 대상 목표물을 요격할 물체가 실시간으로 상태를 추정할 수 있어야 한다. 그러나 입자 군집 최적화 알고리즘 특성상 정확한 추정을 위하여 계산 시에 시간이 많이 소요된다는 커다란 제한점이 있다. 입자 군집 최적화 알고리즘은 입자들이 가지는 속도, 가속도 그리고 위치 정보를 계산한 후 그 중의 최적값을 찾아 움직이며 최적점을 찾아내는 알고리즘이다. 입자들이 가지는 정보를 계산하는 부분이 한 사이클마다 반복해서 진행되며, 각각의 입자마다 가지는 정보를 계산하기 때문에 알고리즘 수행 시 많은 계산시간이 소요된다. CPU를 이용하여 입자 군집 최적화 알고리즘을 진행하게 되면 반복 횟수만큼 순차적으로 계산이 수행된다. 정확한 목표 상태 추정을 위하여 입자 수가 증가할수록 동일한 계산을 더 많이 반복하여야 하므로 알고리즘 수행 중 계산시간 또한 증가한다. 목표물의 상태 추정을 위하여 입자 군집 최적화 기법을 적용하여 고속으로 움직이는 탄도 목표물을 추적하여 요격할 때, 이러한 제한점으로 인하여 실시간으로 변경되는 목표물의 상태를 제대로 추적할 수 없으므로 요격에 실패할 수 있다. 따라서 고속으로 움직이는 탄도 목표물에 대하여, 추적 시 실시간으로 목표물의 상태 추정할 수 있어야 하므로 이 문제를 해결하기 위하여 입자 군집 최적화 기법에 대한 GPU 기반 가속화 방법을 제안한다.

탄도 목표물을 추적하여 요격하기 위한 물체는 목표물 상태 추정을 위한 장치가 탑재되어야 한다. 해당 물체에 PC 환경을 탑재하기는 전력, 발열 등의 문제로 인하여 제한이 된다. 때문에, 주로 목표물 상태 추정의 계산을 진행할 수 있는 보드 등을 사용하게 된다. 이를 고려하여 본 논문에서의 실험은 임베디드 시스템에서 CPU와 GPU를 상호 설계하여 병렬화 및 가속화를 진행하였다. 탄도 목표물의 실시간 추적 및 요격을 위하여 입자 군집 최적화 기법을 온보드 환경에서 가속화하여 실시간성을 향상하였다.

IV. 제안하는 기법

1. 데이터 획득

우선 CPU에서의 전체 알고리즘의 흐름도는 그림 3에 나타난다. 먼저 앞서 설명한 식들을 근거로 하여 추적하기 위한 탄도 목표물의 모델을 만들고 추적 시 발생하는 노이즈를 생성한다. 그 후 생성된 탄도 목표물에 대한 상태를 추정하기 위하여 입자 군집 최적화 기법을 적용하여 목표를 추적할 수 있도록 한다. 입자 군집 최적화 기법의 알고리즘은 먼저 입자들을 생성하고 값을 초기화한다. 생성된 입자들에 최적값을 찾고자 하는 공간에 랜덤한 값들을 부여한다. 입자들이 가지는 값들은 그 지점에서의 위치, 속도 그리고 가속도의 정보를 의미한다. 이 정보를 사용하여 목표물에 대한 상태를 추정한다. 다음 추정된 결과값을 우도 함수에 적용하여 가장 가능성 있는 추정치를 구한다. 각각의 입자가 가지고 있는 정보 중 가장 최적점으로 판단되는 값을 국소 최적점 (Local optimal point)이라 하고, 모든 입자가 가지고 있는 정보 중에서 가장 최적점으로 판단되는 값을 전역 최적점 (Global optimal point)이라 한다. 우도함수로 구한 추정치와 전역 최적점, 국소 최적점 그리고 입자의 정보를 사용하여 입자 정보에 대하여 업데이트를 진행한다.

이 과정이 각각의 입자마다 적용되며 임의의 반복 횟수만큼 반복하며 최적점을 찾아간다. 입자 군집 최적화 알고리즘이 완료되면 탄도 목표물의 실제값과 비교하여 얼마나 정밀하게 목표물의 상태를 추정하였는지 확인한다.

2. 계산 시간 프로파일링

입자 군집 최적화 알고리즘을 병렬화하기 전, 알고리즘 중 어느 부분에서 계산 시간이 많이 소요되는지 식별하는 작업이 병렬화를 하기 전 선행되어야 한다. 입자 군집 최적화를 사용하여 탄도 목표물을 추적하는 알고리즘을 보여주는 그림 3의 흐름도에서 각 부분마다 소요되는 계산 시간을 그림 4에서 보여주고 있다. 알고리즘 각각의 부분마다 소요되는 시간을 식별하기 위하여 입자 군집 최적화 알고리즘의 입자는 1000개로 정의하였고 한 입자가 최적점을 향하여 이

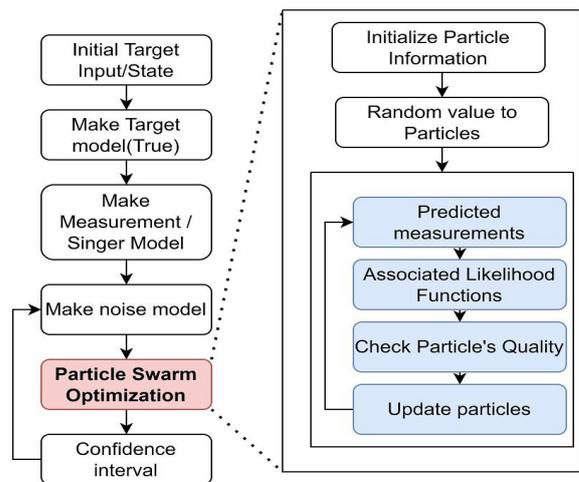


그림 3. 목표물 추적 알고리즘의 흐름도
Fig. 3. Flowchart of target tracking algorithm

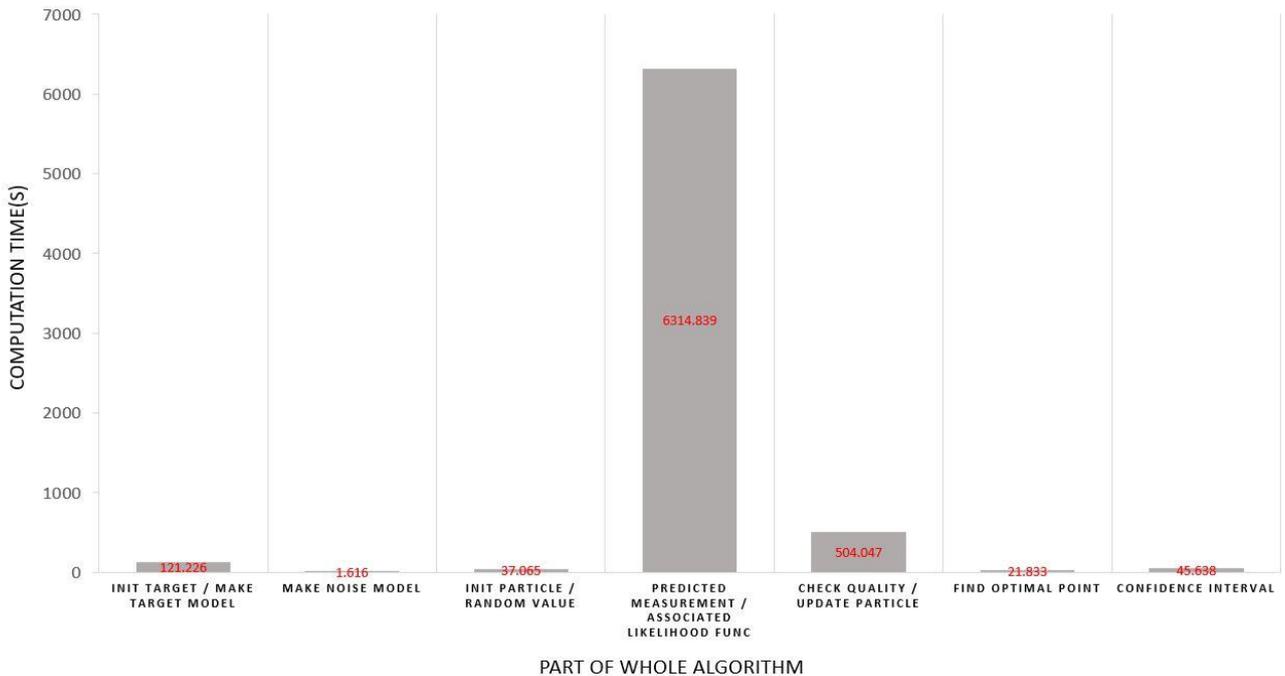


그림 4. 입자 군집 최적화 기법을 적용한 목표물 추적 알고리즘의 각 파트별 수행시간
 Fig. 4. Computation time of the Target tracking algorithm using Particle Swarm Optimization

동하는 횟수는 10번으로 정의하였다. 그림 4에서 추적 대상이 되는 탄도 목표 모델 생성 부분과 노이즈 모델을 생성하는 부분들은 목표물을 추적하는 알고리즘에서 소요되는 시간에 비하여 매우 적은 시간을 필요로 한다. 목표물의 상태를 추정하는 입자 군집 최적화 부분에서 시간이 많이 소요되는 것을 알 수 있다. 특히 시간이 가장 많이 소요되는 부분은 각 입자들이 가지는 정보를 사용하여 입자들이 어느 지점으로 움직여야 하는지 판단하는 부분과 우도함수를 계산하는 부분으로 식별되었다. 이 부분의 계산 시간은 약 6314.839초 소요되었으며 이 부분에 대하여 GPU를 사용한 병렬화를 진행하고 결과적으로 CPU에 비하여 실시간성을 가질 수 있도록 가속화 하였다.

3. 병렬적 입자 군집 최적화 기법

3.1 목표의 대한 상태 추정 부분 병렬화

그림 4에서 제일 시간이 오래 걸리는 부분이라고 식별된 부분 중 먼저 입자들이 가지고 있는 정보들을 사용하여 목표의 상태라고 예측되는 측정값을 구하는 부분을 병렬화한다. 데카르트 좌표계 안에서 목표의 위치와 상태를 추정하기 위하여 목표의 거리와 각도와 목표의 회전 각도를 다음의 식을 사용하여 구한다.

$$Range[n] = \sqrt{x_{p1}^2[n] + x_{p2}^2[n] + x_{p3}^2[n]} \quad (13)$$

$$\theta[n] = -atan(x_{p3}[n] / \sqrt{x_{p1}^2[n] + x_{p2}^2[n]}) * \frac{180}{\pi} \quad (14)$$

$$\psi[n] = atan(x_{p2}[n] / x_{p1}[n]) * \frac{180}{\pi} \quad (15)$$

(13), (14) 그리고 (15)의 식에서 거리, 각도, 회전 각도를 추정하기 위하여 사용되는 x_{p1} , x_{p2} , x_{p3} 은 후에 서술할 입자 군집 최적화 알고리즘에서 최적점을 찾아가기 위하여 알고리즘 안에서 구해지는 입자의 위치 정보이다. 목표 상태의 추정을 위하여 알고리즘상의 입자 위치 정보는 행이 3개이고 열은 입자 군집 최적화에 사용되는 입자의 개수 (n)만큼의 크기를 가지는 행렬로 얻어진다. (13)~(15)의 식은 입자의 개수인 n 번 만큼 반복되며 각각 열이 n 개인 행렬로 나타난다. 입자의 개수만큼 위의 식들이 반복되면서 목표 상태를 추정하기 때문에 입자가 많을수록 정확도가 높은 상태를 추정할 수 있지만, 계산 또한 많이 반복되기 때문에 병렬화를 진행하여 가속화 하였다.

CUDA 커널들을 정의하기 전, 커널들에 사용될 스레드와 블록들에 대하여 정의하여야 한다 [17]. 스레드의 크기는 1024로 정의하였으며, 스레드들을 포함하는 블록은 ($N \times$ 입자의 수)로 정의된다. 커널에 들어갈 데이터 행렬들의 행의 크기에 따라 N 을 1, 3, 9로 3개를 정의하였다. 커널에서 입력 받은 데이터들의 주소를 찾기 위한 id 는 아래 (16)의 식을 사용하여 정의된다. (16)의 식에서 $blockIdx$ 는 GPU 메모리 블록의 x 축 차원의 개수이며, $blockDim$ 는 블록의 행의 개수, 그리고 $threadIdx$ 는 스레드의 수를 의미한다. 본 연구에서 한 블록이 가질 수 있는 스레드의 수는 1024개로 정의하였다.

$$id = blockIdx * blockDim * threadIdx \quad (16)$$

CUDA를 사용하여 GPU 상에서 병렬 계산하기 위하여 (13)~(15)의 식에 사용되는 함수들을 커널로 정의하여야 한다.

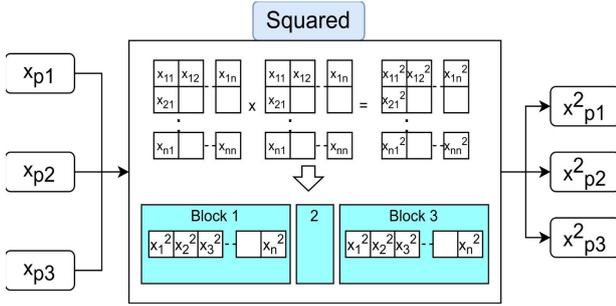


그림 5. 거리, 각도, 회전 각도를 구하기 위한 입자들의 위치정보 제공 커널

Fig. 5. Square kernel of positional information of the particles to obtain Range, Theta, and Psi

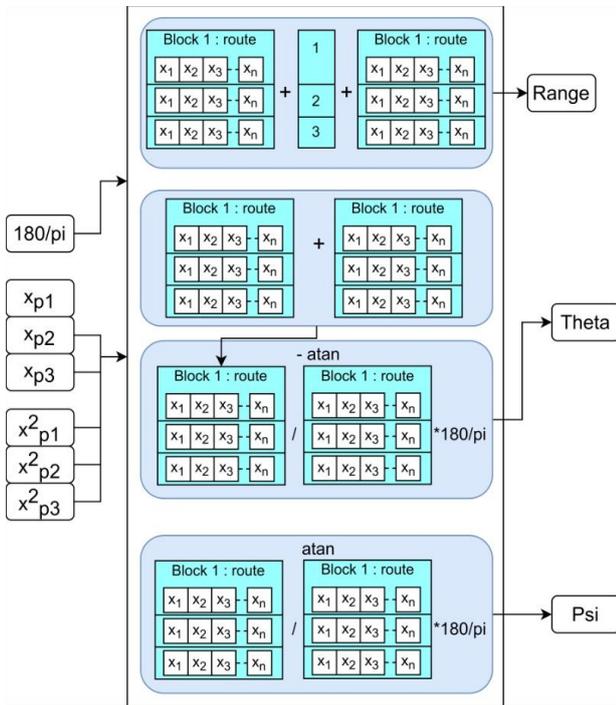


그림 6. 거리, 각도, 회전 각도를 계산하는 커널

Fig. 6. Kernels for obtaining Range, Theta, and Psi

알고리즘 상의 입자의 위치 정보 값이 본래의 값과 제공 값이 사용되기 때문에 위치 정보 값을 제공해주는 커널을 정의한다. (13), (14), (15)의 수식들을 각각 커널로 정의하여 CUDA를 사용하여 병렬 계산할 수 있게 하였다. 먼저 그림 5와 같이 x_p 행렬들의 제공을 구하는 커널을 정의하여 입자 정보의 제공을 병렬 계산하여 구한다.

그 후 거리, 각도, 회전 각도를 구하는 (13)~(15)의 식들은 서로의 결과에 영향을 끼치지 않아 독립적으로 연산이 가능하다. 그림 6과 같이 거리, 각도, 회전 각도를 구하는 커널들을 동시에 병렬 수행하여 계산시간을 단축한다.

3.2 입자 정보 업데이트 부분 병렬화

두 번째 부분은 입자가 가지고 있는 정보를 업데이트하는 부분에서 계산시간이 많이 소요된다. 이 부분을 크게 두 부

분으로 나누면 입자가 가지고 있는 정보를 업데이트하는 부분과 업데이트하기 전 업데이트에 사용되는 값을 구하는 부분으로 구분한다. 먼저 입자 군집 최적화 기법에서 입자가 가지고 있는 정보를 업데이트하는 부분은 아래의 식으로 구할 수 있다.

$$P_{a+1} = kai * ((c * eps * (OP_g - P_x)) + (c * eps * (OP_l - P_x))) - (1 - kai) * P_v \quad (17)$$

$$P_{v+1} = P_v + P_a \quad (18)$$

$$P_{x+1} = P_x + P_v \quad (19)$$

(17)의 식에서 입자 군집 최적화 알고리즘 내의 하나의 입자가 가지고 있는 위치 정보 P_x , 속도 정보 P_v , 국소 최적점 OP_l , 모든 입자들이 가지는 정보 중 전역 최적점 OP_g 를 사용하여, 다음 사이클에 하나의 입자가 가질 가속도 P_{a+1} 정보를 구할 수 있다. (18)과 (19)의 식 또한 하나의 입자가 가지는 속도, 가속도, 위치 정보를 사용하여 다음 사이클에 하나의 입자가 가질 속도 P_{v+1} 와 위치 P_{x+1} 정보를 구한다. 식 (13)~(15)에서 사용되는 입자의 위치 정보는 식 (17)~(19)을 사용하여 업데이트 한다.

입자가 가지고 있는 정보들을 위의 식들을 적용하여 업데이트하기 전, 한 사이클에서 (13)~(15)의 식을 사용하여 목표의 상태를 추정한 후 추정된 결과값을 우도 함수에 적용하여 가장 가능성 있는 추정치를 구하여 다음 사이클의 입자 정보 업데이트에 사용한다.

$$Range_A = \frac{1}{\sqrt{2\pi} * sig_R} * \exp\left(\frac{-(mea_R - Range)^2}{(2 * sig_R)^2}\right) \quad (20)$$

거리의 대한 추정치 $Range_A$ 는 (20)의 식을 사용하여 구한다. sig_R 는 추적할 목표물에 대하여 추정된 거리 데이터에서 발생할 수 있는 노이즈이며, mea_R 은 목표물에 대하여 추정된 거리 측정값이다. 본 논문에서는 sig_R 를 1로 설정하였다. 또한 (20)과 (21)~(24) 식에서 사용되는 $Range$ 는 (13)의 식을 사용하여 얻은 목표의 거리 추정치이다.

$$temp_1 = (1 - ep) * (1 / \sqrt{2\pi * (sig_{t,p}^2 + (sig_{g1} / Range^2))}) \quad (21)$$

$$temp_2 = \exp\left(\frac{-(mea_{t,p} - \theta)^2}{2 * ((sig_{t,p} + sig_{g1})^2 / Range^2)}\right) \quad (22)$$

$$temp_3 = (ep) * (1 / \sqrt{2\pi * (sig_{t,p}^2 + (sig_{g2} / Range^2))}) \quad (23)$$

$$temp_4 = \exp\left(\frac{-(mea_{t,p} - \theta)^2}{2 * ((sig_{t,p} + sig_{g2})^2 / Range^2)}\right) \quad (24)$$

각도와 회전 각도의 추정치는 (21)~(24)의 식들을 사용하여 구할 수 있다. 각도의 측정 노이즈 sig_t 와 회전 각도의 측정 노이즈 sig_p 는 0.1로 설정하였고 추정 시 생길 수 있는 노이즈 sig_{g1} 과 sig_{g2} 는 각각 0.5와 1로 설정하였다. 노이즈 값들과 각각의 측정값들을 사용하여 각도와 회전 각도에 대한 추정치는 위의 식에서 구할 수 있는 $temp$ 들의 합으로 나

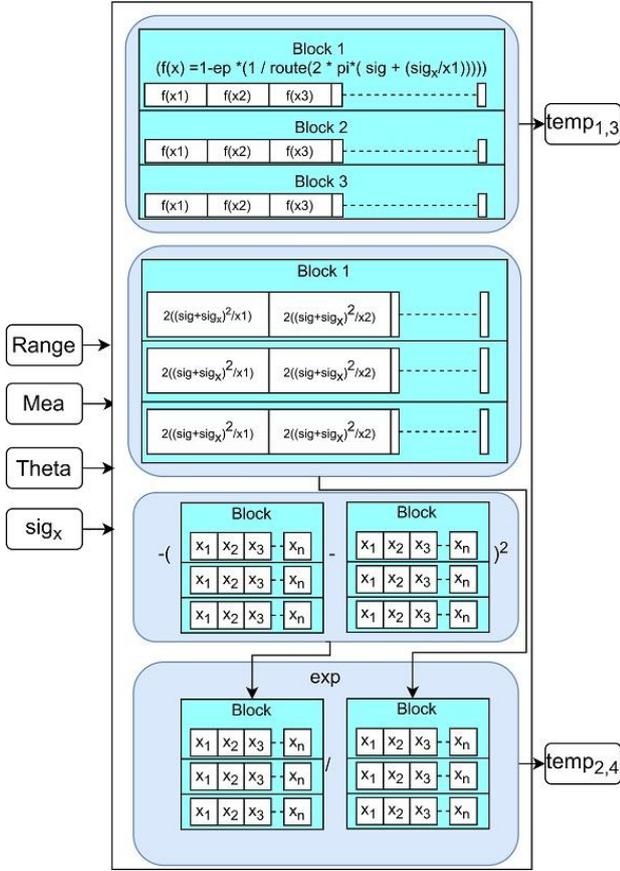


그림 7. 우도함수를 계산하기 위한 커널들
Fig. 7. Kernels for calculating the likelihood functions

타낼 수 있다. 이러한 과정을 각각의 입자마다 반복하여 입자가 가지는 추정치를 구하여야 하므로 입자의 정보를 업데이트하는 과정보다 이 과정에서 많은 계산 시간이 소요된다.

(20)~(24)의 연산을 병렬 계산을 위하여 CUDA 커널로 정의할 때, 공통적인 연산들은 같은 커널을 사용할 수 있다. 그림 7에서 나타나듯이, (22)와 (24)의 식에서 $temp_2$ 과 $temp_4$ 를 구하는 연산은 사용되는 노이즈의 값만 다르기 때문에, 노이즈 값만 다르게 하여 병렬 계산한다. 마찬가지로 (21)과 (23)의 식 또한 같은 형태를 띠고 있어 커널에 들어가는 값만 차이를 주면 되어 동일한 커널을 사용할 수 있다.

그림 8에서 CUDA를 사용하여 병렬 계산을 진행할 때, 사용되는 커널들의 전체적인 흐름도를 보여준다. 먼저 목표에 대한 상태 추정 부분을 병렬 계산한다. 여기서 얻어지는 거리 정보에 대한 추정을 사용하여, 입자들이 가진 정보를 업데이트 하기 전 단계인 가장 가능성 있는 추정치를 구하는 우도 함수 부분을 병렬 계산하여 가속화한다.

V. 실험 결과

1. 탄도 궤적 목표물 추적 결과

논문에서 입자 군집 최적화 알고리즘 중 목표의 상태 추

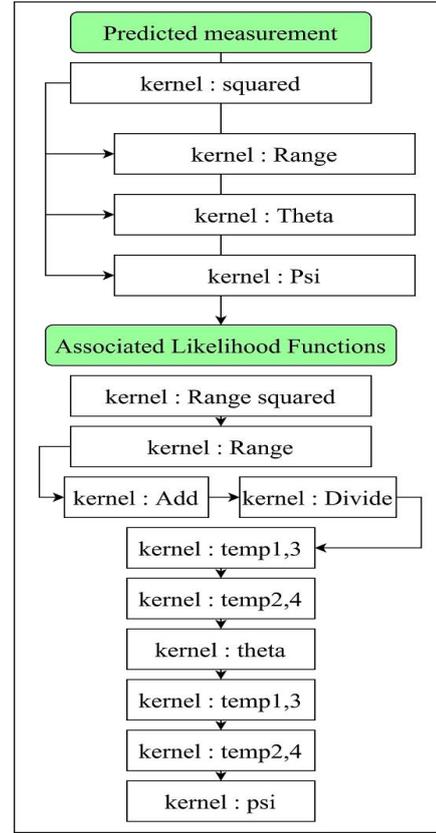


그림 8. 입자 군집 최적화에 사용되는 CUDA 커널 흐름도
Fig. 8. Flowchart of CUDA kernels in PSO

정 부분과 추정치를 구하는 부분을 병렬화하여 고속 표적 추적을 위한 GPU로 입자 군집 최적화 기법을 가속화 한다. 제한된 가속화 된 알고리즘의 추적 결과는 탄도 표적 추적 시나리오에서 평가한다. 수치적 시뮬레이션을 위하여 식 (1)과 (2)의 동적 모델을 사용하여 실제 기준 궤적을 생성한다. 비산물의 공기역학적 항력과 중량은 [18]을 참고하여 설정하였다. 샘플링 간격은 $\Delta t = 0.01$ 초로 설정하였으며 총 시뮬레이션 시간은 3초가 소요되었다. 레이더 수신기 소음 모델의 표준 편차 n_R, n_θ, n_ψ 는 각각 1m, 0.1deg, 0.1deg이며 글린트 노이즈 $n_{G,\theta}, n_{G,\psi}$ 는 (25)의 식과 같은 가우시안 분포를 따른다.

$$p = (1 - \epsilon)p_{G_1} + \epsilon p_{G_2} \quad (25)$$

위의 식에서 ϵ 는 글린트 확률을 의미하며, P_{G_1} 은 $p_{G_1} \sim N(0, 0.1^2)$, P_{G_2} 는 $p_{G_2} \sim N(0, 1^2)$ 를 따르는 가우스 모델이다. 추적 모션 모델은 식 (5)를 따르는 Singer 모델이며 측정 모델은 식 (12)를 사용하여 얻는다. 레이더의 위치는 지상에 고정되어 있는 것으로 가정하고, 탄도 표적은 중력 및 공기역학적 항력을 고려하여 고속으로 움직인다고 가정 한다. 따라서 탄도 표적의 속도는 시뮬레이션 시간에 따라서 달라진다.

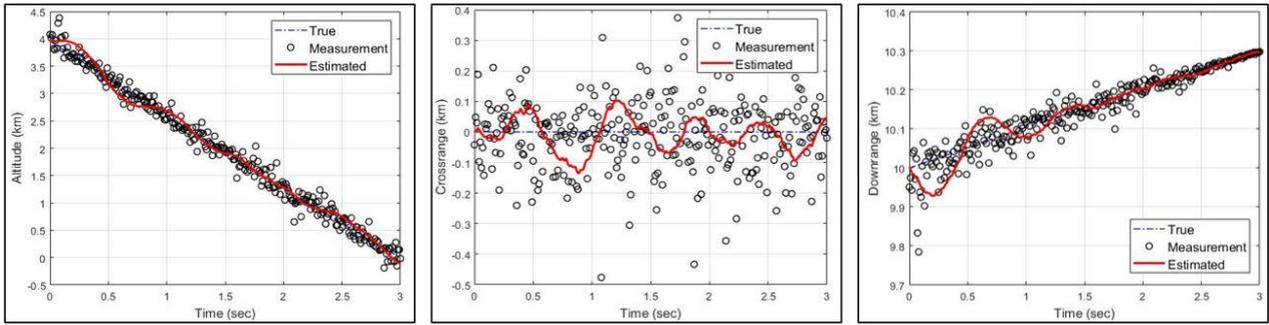


그림 9. 한 입자의 이동 횟수가 5일 때의 레이더 측정 및 트루 모델과 추정된 목표 결과 범위 비교
 Fig. 9. When the number of epoch used in PSO was 5, Estimated target model compared to calculated by radar measurements and true position.

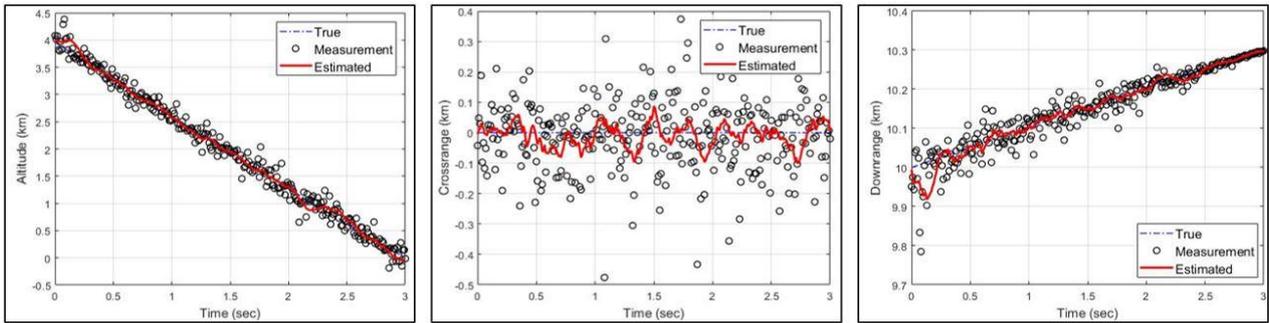


그림 10. 한 입자의 이동 횟수가 10일 때의 레이더 측정 및 트루 모델과 추정된 목표 결과 범위 비교
 Fig. 10. When the number of epoch used in PSO was 10, Estimated target model compared to calculated by radar measurements and true position.

궤적 및 상태 추정의 결과는 그림 10에 나타난다. 입자 군집 최적화 알고리즘에 사용되는 입자 수와 입자의 이동횟수가 너무 적으면 수렴이 잘되지 않거나 에러 바운드가 크게 튀는 현상이 발생한다. 목표물 추적 알고리즘이 잘 수렴하도록 입자 군집 최적화 알고리즘의 입자는 1000개로 정의하였고 한 입자가 최적점을 향하여 이동하는 횟수는 10번으로 정의하였다. 초반 부분의 에러 바운드가 튀는 현상이 있지만 이것은 입자 군집 최적화 기법의 특성상 처음에는 랜덤한 값이 들어가기 때문이고 반복이 진행될수록 최적의 값을 찾아가기 때문에 후에는 목표 상태 추정이 제대로 동작하는 것을 알 수 있다. 비교를 위하여 한 입자의 이동횟수를 5번으로 조정하였을 때의 하향 범위 비교와 고도 범위 비교 결과를 그림 9에 나타내었다. 그림 9, 10의 왼쪽 그림은 트루 모델과 추정된 고도 범위를 (Altitude) 비교한 그림이며 중간 그림은 교차 범위 (Cross-range), 오른쪽 그림은 목표 하향 범위 (Down-range)의 결과 비교를 보여준다.

2. 병렬화 및 가속화 결과

본 논문에서의 병렬화 및 가속화 실험은 NVIDIA Jetson Xavier 환경에서 진행되었으며, CPU는 6-core NVIDIA Carmel ARM v8.2 64-bit GPU는 NVIDIA Volta architecture with 384 NVIDIA CUDA cores and 48 Tensor cores 사양을 가진다. 실험에서는 CPU만을 사용한 병렬화 하지 않은 입자 군집 최적화 기법과 CPU와 GPU를 함께 사용하여 병렬화

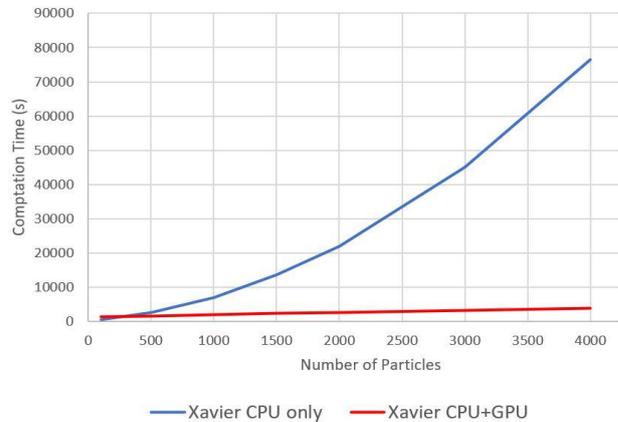


그림 11. 전체 목표 추적 알고리즘의 계산 시간 비교
 Fig. 11. Computation time of Whole target tracking algorithm

된 입자 군집 최적화 알고리즘을 비교한다.

그림 11에서는 목표 추적 알고리즘 전체 수행시간 비교가 나타나 있으며 그림 12과 표 2에서는 그 중 입자 군집 최적화 알고리즘만의 CPU와 GPU 사용 간의 수행시간 비교를 나타낸다. 목표 추적을 위한 알고리즘은 300번의 반복을 진행하여 수행되었으므로, 표 2의 계산시간은 그중 1번의 반복에서 입자 군집 최적화 기법이 수행된 시간을 의미한다. 그림 11, 12과 표 2에서 입자 군집 최적화 기법에 사용된 입자 수가 100, 500개 즉, 입자 수가 적을 때는 CPU만을

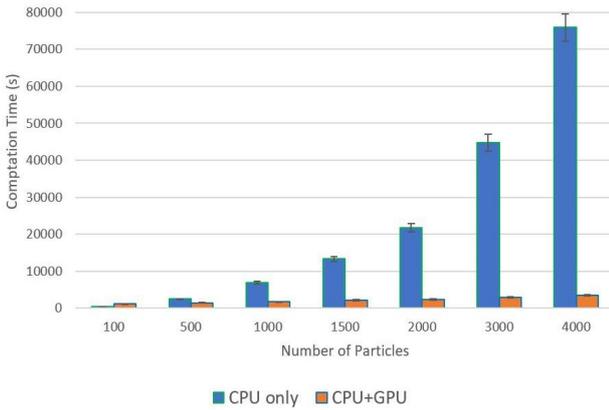


그림 12. 목표 추적 알고리즘 중 입자 군집 최적화 기법의 계산 시간 비교
Fig. 12. Computation time of Particle Swarm Optimization

표 2. 한 step 당 입자 군집 최적화 기법의 계산 시간 비교
Table 2. Computation time comparison of Particle Swarm Optimization per one step

Number of Particles	only CPU	CPU+GPU
100	1.2031	3.8219
500	8.1129	4.7291
1000	22.8975	5.4873
1500	44.4302	7.0893
2000	72.3309	7.7308
3000	149.1618	9.5788
4000	252.9826	11.4243

사용했을 때 계산시간이 더 적게 소요되거나 비슷하게 소요된다. 하지만 입자 수가 1000개 이상으로 사용되었을 때에는 CPU만을 사용하여 알고리즘을 진행하였을 때보다 GPU를 사용하여 앞서 설명한 연산시간이 많이 걸리는 부분을 병렬화하여 진행한 부분이 더 시간이 적게 걸리는 것을 알 수 있다. 병렬 계산을 위하여 CUDA를 사용하면 필연적으로 오버헤드가 발생한다. 오버헤드 시간은 CUDA 사용을 위한 CUDA 초기화, 선언된 커널들을 불러오는 시간 등을 포함하게 된다. 그림 11, 12 그리고 표 2에서 GPU를 사용하여 계산 시간을 진행하였을 때 오버헤드 시간까지 포함한 결과이다. 오버헤드 시간은 입자 군집 최적화 기법의 입자 수와 관계없이 평균적으로 97초 정도 소요되었다.

표 3과 그림 13에서는 입자 군집 최적화 기법을 입자 수를 1000개로 설정하여 진행된 탄도 목표 추적 알고리즘의 전체 계산 시간을 파트별로 보여준다. 앞서 설명한 계산 시간이 많이 소요된다고 식별된 목표에 대한 상태 추정 부분과 입자 정보 업데이트 부분을 GPU를 사용하여 병렬화하였을 때, 해당 파트들이 CPU만을 사용하였을 때보다 계산 시간이 현저히 줄어 가속화가 성공하였다는 것을 알 수 있다. CUDA 사용을 위한 오버헤드를 포함하더라도 전체 계산 시간은 GPU를 사용하여 병렬화한 방법이 계산 시간이 적게 소요되었다.

표 3. 목표 추적 알고리즘의 파트별 계산 소요 시간
Table 3. Computation time for each part of the target tracking algorithm.

Part	Computation time(s)	
	Only CPU	CPU + GPU
CUDA init delay	-	94.559
Others	746.68	819.849
predict measurement / associated likelihood func	6314.839	1011.639
Total	7061.519	1926.047
Without CUDA init delay	7061.519	1831.488

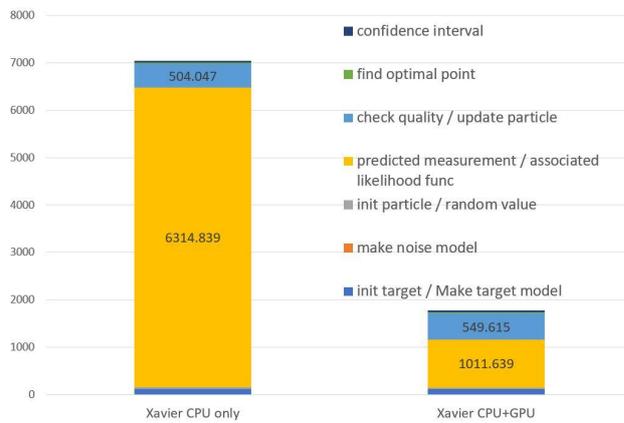


그림 13. 목표 추적 알고리즘의 파트별 계산 소요 시간
Fig. 13. Computation time for each part of the target tracking algorithm.

VI. 결론

본 논문에서는 고속으로 움직이는 탄도 목표물을 추적하는 문제에 대하여 입자 군집 최적화 기법을 사용하였으며 실시간성을 가지기 위하여 GPU를 사용한 가속화를 진행하였다. 입자 군집 최적화 기법을 사용하여 목표로 하는 탄도 목표물의 움직임과 각도 등의 상태 추적을 하였다. 입자 군집 최적화 기법의 특성상 알고리즘의 반복이 진행되지 않은 초반에는 입자들이 최적의 값의 정보를 가지고 있지 않기 때문에 에러 바운드가 튀는 부분이 있으나, 알고리즘이 진행되어 반복이 진행되면서 입자들이 가지고 있는 전역, 국소 최적점의 정보를 따라 최적점을 찾아 움직이면서 상태 추적이 정상적으로 진행되었다.

탄도 목표물의 추적 알고리즘에서 입자 군집 최적화 기법에서 대부분의 시간이 소요되었으며, 특히 그 중 입자가 가지고 있는 정보를 이용한 목표에 대한 상태 추정 부분과 입자 정보 업데이트 부분에서 계산 시간이 많이 소요되는 것을 식별하였다. 입자 정보 업데이트 부분에서도 입자 정보를 업데이트 하기 전 업데이트에 필요한 값을 구하는 부분에서 시간이 많이 소요되었다. 계산 시간이 많이 소요된다고 식별된 두 부분을 CUDA를 사용하여 GPU를 이용한 병

렬화하였으며 CUDA 사용을 위하여 필연적으로 발생하는 오버헤드 시간을 고려하더라도 CPU만을 사용하여 알고리즘을 진행하였을 때보다 계산 시간이 줄어들었다. 탄도 목표물 추적 시 CPU만을 사용하여 목표물의 상태 추정을 하는 것보다 GPU를 사용한 본 논문에서 제안된 병렬화 된 입자군집 최적화 기법을 적용한 알고리즘이 계산 시간을 더 적게 소요하기 때문에 CPU만을 사용하였을 때보다 실시간성을 더 가지게 된다.

References

- [1] G. M. Siouris, "Missile Guidance and Control Systems," Verlag, Berlin: Springer, pp. 113 - 119, 2004.
- [2] G. A. Hewer, R. D. Martin, J. Zeh, "Robust Preprocessing for Kalman Filtering of Glint Noise," IEEE Transactions on Aerospace and Electronic Systems, Vol. 23, No. 1, pp. 120 - 128, 1987.
- [3] G. Oh, H. Lee, H. Lee, "Hierarchical Correlation-based Anomaly Detection for Vision-based Mask Filter Inspection in Mask Production Lines," IEMEK J. Embed. Sys. Appl., Vol. 16, No. 6, pp. 277 - 283, 2021 (in Korean).
- [4] Y. Zhou, Y. Tan "GPU-based Parallel Particle Swarm Optimization", 2009 IEEE Congress on Evolutionary Computation, pp. 1493-1500, 2009.
- [5] L. Mussi, S. Cagnoni, F. Daolio, "GPU-Based Road Sign Detection Using Particle Swarm Optimization", 2009 Ninth International Conference on Intelligent Systems Design and Applications. IEEE, pp. 152-157, 2009.
- [6] A. Keshavarz-Mohammadiyan, H. Khaloozadeh, "PSO-PF Target Tracking in Range-based Wireless Sensor Networks with Distance-dependent Measurement Noise", 2015 23rd Iranian Conference on Electrical Engineering (ICEE), pp. 911-915, 2015.
- [7] W. Ding, W. Fang "Target Tracking by Sequential Random Draft Particle Swarm Optimization Algorithm", 2018 IEEE International Smart Cities Conference (ISC2), 2018.
- [8] S. S. Rayala, N. A. Kumar, "Particle Swarm Optimization for Robot Target Tracking Application", materialstoday PROCEEDINGS, Vol 33, No. 7, pp. 3600-3603, 2020.
- [9] S. Yang, Q. Ma, W. Huang, "Particle Swarm Optimized Unscented Particle Filter for Target Tracking", 2009 2nd International Congress on Image and Signal Processing, pp. 1-5, 2009.
- [10] Z. Cheng, L. Fan, Y. Zhang, "Multi-agent Decision Support System for Missile Defense Based on Improved PSO Algorithm" Journal of Systems Engineering and Electronics, Vol. 28, No.3, pp. 514-525, 2017.
- [11] X. Zheng, Y. Gao, W. Jing, Y. Wang, "Multidisciplinary Integrated Design of Long-range Ballistic Missile Using PSO Algorithm" Journal of Systems Engineering and Electronics, Vol. 31, No. 2, pp. 335-349, 2020.
- [12] J. G. dos Santos Júnior, J. P. S. do Monte Lima, "Particle Swarm Optimization for 3D Object Tracking in RGB-D Images", Computers & Graphics, Vol. 76, pp. 167-180, 2018.
- [13] B. Rymut, B. Kwolok, "Real-time Multiview Human Pose Tracking Using Graphics Processing Unit-accelerated Particle Swarm Optimization", Concurrency and Computation: Practice and Experience, Vol. 27, pp. 1551-1563, 2014.
- [14] R. A. Singer, "Estimating Optimal Tracking Filter Performance for Manned Maneuvering Targets," IEEE Transactions on Aerospace and Electronic Systems, Vol. 4, pp. 473-483, 1970.
- [15] X. R. Li, V. P. Jilkov, "Survey of Maneuvering Target Tracking. Part I. Dynamic Models," IEEE Transactions on Aerospace and Electronic Systems, Vol. 39, No. 4, pp. 1333-1364, 2003.
- [16] J. Kim, M. Tandale, P. K. Menon, "Particle Filter for Ballistic Target Tracking with Glint Noise," Journal of Guidance, Control, and Dynamics, Vol. 33, No. 6, pp. 1918 - 1921, 2010.
- [17] H. M. Park, J. S. Kwon, T. H. Hwang, D. S. Kim, "Implementation of Integrated CPU-GPU for Efficient Uniform Memory Access Method and Verification System," IEMEK J. Embed. Sys. Appl., Vol. 11, No. 2, pp. 57 - 65, 2016 (in Korean).
- [18] D. C. Wright, D. Kadyshchev, "An Analysis of the North Korean Nodong Missile," Science & Global Security, Vol. 4, No. 2, pp. 129-160, 1994.

Yunho Han (한 용 호)



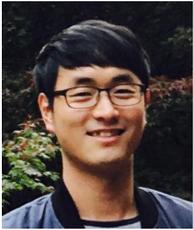
2021 Electronic Engineering from Kumoh National Institute of Technology (B.S.)

2021~Department of IT Convergence Engineering, Kumoh National Institute of Technology (M.S. Candidate)

Field of Interests: Algorithm acceleration with GPU and FPGA, embedded system, Path Planning

Email: gksfbsgh@kumoh.ac.kr

Heoncheol Lee (이 현철)



2006 Electronic-Electrical Engineering and Computer Sciences from Kyungpook National University (B.S.)

2008 Electrical Engineering and Computer Sciences from Seoul National University (M.S.)

2013 Electrical Engineering and Computer Sciences from Seoul National University (Ph.D.)

2019~Department of IT Convergence Engineering, School of Electronic Engineering, Kumoh National Institute of Technology (Assist. Prof.)

Career:

2011 Researcher, ASRI, Seoul National University

2013 Senior Researcher, Agency for Defense Development

2019 Technical Adviser, LG Electronics

Field of Interests: Vision-based Anomaly Detection, SLAM, Path Planning, Algorithm Acceleration, Deep Learning

Email: hclee@kumoh.ac.kr

Hyeokhoon Gwon (권혁훈)



2002 Aerospace Engineering from KAIST University (B.S.)

2005 Aerospace Engineering from KAIST University (M.S.)

2020 Aerospace Engineering from KAIST University (Ph.D.)

Career: 2009 Researcher, LIGNEX1

Field of Interests: Convex optimization, optimal control, Guidance and autopilot design and nonlinear control

Email: hhwon22@lignex1.com

Wonseok Choi (최원석)



2006 Electronic-Electrical Engineering from Hongik University (B.S.)

2017 Defense Engineering from Yonsei University (M.S.)

Career: 2007 Researcher, LIGNEX1

Field of Interests: FPGA, Filter, Sensor, Munition Design

Email: wonseok.choi@lignex1.com

Bora Jeong (정보라)



2020 Electronic-Electrical Engineering from Hanyang University (B.S.)

Career: 2020 Researcher, LIGNEX1

Field of Interests: FPGA, Filter, Sensor, Munition Design

Email: bora.jung@lignex1.com

논문 2022-17-43

FPGA기반 뉴럴네트워크 가속기에서 2차 타일링 기반 행렬 곱셈 최적화

(Optimizing 2-stage Tiling-based Matrix Multiplication in FPGA-based Neural Network Accelerator)

권진세, 이제민, 권용인, 박제만, 유미선, 김태호, 김형신*
 (Jinse Kwon, Jemin Lee, Yongin Kwon, Jeman Park, Misun Yu, Taeho Kim, Hyungshin Kim)

Abstract : The acceleration of neural networks has become an important topic in the field of computer vision. An accelerator is absolutely necessary for accelerating the lightweight model. Most accelerator-supported operators focused on direct convolution operations. If the accelerator does not provide GEMM operation, it is mostly replaced by CPU operation. In this paper, we proposed an optimization technique for 2-stage tiling-based GEMM routines on VTA. We improved performance of the matrix multiplication routine by maximizing the reusability of the input matrix and optimizing the operation pipelining. In addition, we applied the proposed technique to the DarkNet framework to check the performance improvement of the matrix multiplication routine. The proposed GEMM method showed a performance improvement of more than 2.4 times compared to the non-optimized GEMM method. The inference performance of our DarkNet framework has also improved by at least 2.3 times.

Keywords : Versatile Tensor Accelerator (VTA), Neural Processing Unit (NPU), General Matrix Multiplication, GEMM Optimization

1. 서론

뉴럴네트워크의 경량화 및 가속화가 컴퓨터 비전 분야의 중요한 주제로 다루어지고 있다 [1-2]. 뉴럴네트워크 경량화를 위해 네트워크 구조 탐색 (Network Architecture Search), 지식증류 (Knowledge Distillation), 가지치기 (Pruning), 재매개변수화 (Reparameterization), 양자화 (Quantization) 등이 연구되고 있다 [3-5]. 하지만 단순히 뉴럴네트워크 경량화만으로는 성능 향상을 기대할 수 없으며, 뉴럴네트워크를 효율적으로 실행할 수 있는 하드웨어가 수반되어야 한다 [6]. 뉴럴네트워크 가속이 가능한 상용 하드웨어는 구글의 TPU (Tensor Processing Unit) [7], 인텔의 NCS (Neural Compute Stick) [8] 등의 ASIC (Application Specific Integrated Circuit) 기반 가속기가 있다. FPGA (Field Programmable Gate Array) 기반의 연구용 가속 하드웨어로는 TVM 백엔드로 공개된 오픈소스 VTA (Versatile Tensor Accelerator) [9] 가속기가 대표적이다. CNN (Convolutional Neural Network:CNN)의 핵심 연산인 컨볼루션 연산은 Im2col (Im

age to Columnize)변환을 통해 행렬 곱셈으로 차원축소가 가능하다 [10]. 행렬 곱셈 연산으로 차원을 낮추게 되면 이미 잘 정의되어있는 BLAS (Basic Linear Algebra Subprogram) 라이브러리를 행렬 곱셈에 적용할 수 있다. 하지만, 기존 BLAS 라이브러리는 범용 CPU, GPU를 타겟으로만 최적화 되어, 새로운 하드웨어에 적용이 어렵다 [10-15]. 또한, 정확도 표현을 낮추어 데이터양을 줄이는 양자화 기법을 적용하고자 할 때 INT8 과 같은 저정확도 데이터 표현을 대부분의 BLAS에서 지원하지 않고 있다. 따라서, 기존의 BLAS를 새로운 가속 하드웨어 바로 적용하는 것은 현실적으로 불가능하다.

본 논문에서는 FPGA 기반 뉴럴네트워크 가속기 VTA와 같은 행렬 곱셈기반의 가속 회로에서 적용 가능한 2차 타일링 GEMM 기법을 제안하고자 한다. 본 논문의 기여를 정리하면 다음과 같다.

- 2차 타일링 기반 최적화 행렬 곱셈 : 행렬 곱셈에서 행렬들의 크기가 커질수록 메모리 접근에 비용이 커지게 된다. 이러한 메모리 접근의 비효율성을 개선하고자 타일링 기법을 사용하게 되는데, 본 논문에서는 하드웨어에 최적화된 타일 크기와 메모리에 최적화된 크기 2가지의 타일로 메모리를 구성함으로써 연산 성능을 극대화하였다. 제안하는 기법은 비정방행렬, 정방행렬 케이스에서 평균 2.4배 이상의 성능 개선을 보였다.

- DarkNet [16] 딥러닝 프레임워크 개선 : 제안하는 행렬 곱셈 가속 기법의 성능을 확인하기 위해서 오픈소스 딥

*Corresponding Author (hyungshin@cnu.ac.kr)
 Received: Oct. 21, 2022, Revised: Nov. 11, 2022, Accepted: Nov. 23, 2022.
 Jinse Kwon: Chungnam National University (Ph.D. Student)
 J. M. Lee, Y. I. Kwon, J. M. Park: ETRI (Senior Researcher)
 M. S. Yu: ETRI (Principal Researcher)
 T. H. Kim: ETRI (Assistant Vice President)
 H. S. Kim: Chungnam National University (Prof.)
 ※ 이 논문은 2022년도 정부 (과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No.2018-0-00769,인공지능 시스템을 위한 뉴로모픽 컴퓨팅 SW 플랫폼 기술 개발).

러닝 프레임워크인 DarkNet의 코드를 수정하여 최적화된 VTA 백엔드를 구현하였다. DarkNet 프레임워크는 Im2Col 기반의 행렬 곱셈은 지원하지만, 특정 가속 하드웨어를 지원하지 않으며 양자화 같은 가속 기능들이 적용되어있지 않다. 따라서 본 연구를 위해 DarkNet 프레임워크에 VTA 백엔드를 지원하도록 하였고, 양자화 일부 기능을 지원하도록 수정하였다. 제안하는 기법을 적용하였을 때 ResNet-18 모델 [17]의 추론 성능이 다크넷에서 기본 지원되는 CPU 대비 VTA 타겟에서 2.3배의 성능 향상을 달성하였다.

본 논문의 2장에서는 뉴럴네트워크 및 행렬곱셈과 관련된 연구들을 정리하여 연구의 목적을 설명하고자 한다. 3장에서는 제안하는 기법인 2차 타일링 행렬 곱셈 가속과 DarkNet 프레임워크 개선에 대해 자세히 소개한다. 4장에서는 제안하는 기법의 성능 개선을 실험 및 평가하여 연구의 성과를 제시하고, 5장에서 본 연구의 결론을 맺는다.

II. 관련연구

1. 컨볼루션 연산의 차원 축소 기법

CNN의 핵심 연산은 그림 1 위와 같이 컨볼루션 연산으로써 입력이미지 ($C \times H \times W$)와 가중치값 ($N \times C \times R \times S$)의 컨볼루션 곱셈으로 출력 이미지 ($N \times H \times W$)의 결과를 도출하게 된다. 직접 컨볼루션 (Direct Convolution) 연산의 경우 커널의 위치에 맞는 입력 이미지의 메모리를 접근하게 되는데, 이 경우 메모리 접근이 연속적이지 않으므로 캐시미스가 빈번하게 발생하므로 성능 저하가 발생한다. 이러한 문제를 해결하기 위해 그림 1 중간과 같이 입력 이미지에서 커널 레이아웃이 접근할 위치의 값을 미리 복사하여 새로운 메모리에 저장하므로 메모리 사용량은 증가하지만, 고차원 컨볼루션 연산이 저차원 행렬 곱셈으로 차원 축소가 가능해진다. 이러한 변환 기법을 Im2Col (Image to Column) 라고 부른다. 차원 축소 기법은 입력 이미지의 메모리 사용량이 커널 크기를 곱한 만큼 증가하지만 연속적인 메모리 접근이 가능하므로 성능이 향상되고, 행렬 곱셈 가속 코드를 포함하는 선형대수 연산 라이브러리인 BLAS를 사용할 수 있게 된다. Im2Col과 비슷한 방식으로는 모델 파라미터를 분리하여 계산한 후 최종 출력 이미지에서 쉬프트 누적을 통한 컨볼루션 연산을 수행하는 Kn2Row (Kernel to Row) 기법이 존재한다 [18]. 이 방식은 그림1의 아래와 같이 결과를 미리 알 수 없는 입력 이미지 대신 상수값을 갖는 가중치 (weight) 파라미터의 메모리 레이아웃을 재배치하여 연산한다. 하지만, 출력 이미지의 쉬프트 덧셈 방식의 성능 저하와 출력 이미지의 메모리 사용량 증가로 성능 개선이 Im2Col 대비 크지 않아 범용적으로 사용되지 않는다 [19].

기존의 Im2Col의 최적화 기법으로는 MEC (Memory Efficient Convolution) [20], Indirect Convolution [21] 등이 있다. MEC 연산은 Im2Col 변환을 기반으로 하여, 메모리 접근 방향을 열 방향에서 행 방향으로 데이터 레이아웃을 변경함으로써 커널의 재사용성을 높이므로 메모리 사용량을 줄

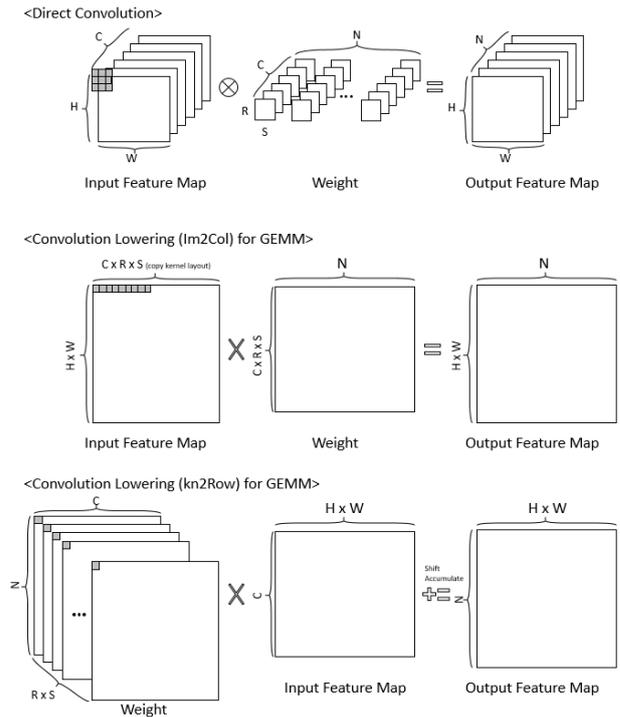


그림 1. (위) 직접 컨볼루션 연산 과정 도식, (중간) Im2Col 차원 축소를 통한 행렬 곱셈 변환 (아래) kn2Row 차원 축소를 통한 행렬 곱셈 (누적) 변환
Fig. 1. (Above) Direct Convolution Method (Middle) Im2Col Convolution Lowering Transformation (Bottom) Kn2Row Convolution Lowering Transformation

일 수 있는 기법이다. Indirect Convolution은 Im2Col과 방식은 동일하지만 메모리를 복사하지 않고, 포인터 버퍼를 통해 메모리를 접근하도록 하여 메모리 사용량을 줄이는 기법을 제안하였다. MEC의 경우 커널의 크기가 크고 stride가 1인 경우에는 성능 이득이 발생하지만, 커널의 크기가 작거나, stride가 2 이상인 경우에 오히려 성능이 저하되는 구조적 한계가 있다.

2. BLAS, BLIS 라이브러리

BLAS 라이브러리는 타겟 CPU, GPU에서 선형대수 연산 루틴들이 최적의 성능을 갖도록 연구 개발되고 있다. 범용 CPU에 대한 최적화 루틴을 갖는 BLAS 라이브러리는 intel-MKL (Math Kernel Library) [15], LAPACK [12], OpenBLAS [11] 등이 있고, ARM 계열의 CPU BLAS는 직접 컨볼루션 루틴을 포함하는 BLIS (BLAS-like Library Initiation Software) 형태로 발전하고 있다. 대표적인 예로 ARM-CCL (Compute Library) [13], NNPACK [14] 등이 있다. GPU BLAS는 NVIDIA 계열의 CuBLAS [22], 범용 GPU 프로그래밍을 위한 OpenCL를 이용한 CLBlast [23], cBLAS [24] 등이 있다. 오랜 기간 잘 정의된 BLAS, BLIS 라이브러라고 할지라도, 새로운 딥러닝 가속기에는 바로 적용하는 것은 불가능하며, 새로운 가속기를 위한 연산 라이브러리를 새롭게 개발해야 하는 어려움이 있다.

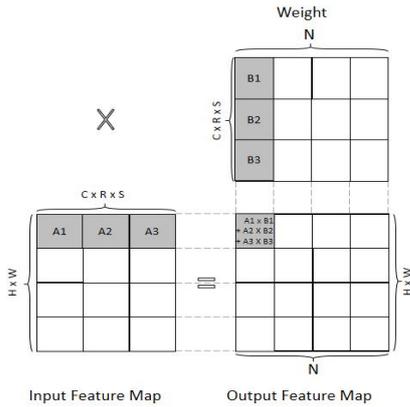


그림 2. 타일링 기반 행렬 곱셈
Fig. 2. Tiling-based matrix multiplication

3. 행렬 곱셈의 타일링 최적화

그림 2는 일반 행렬 곱셈에서 일반적으로 사용하는 타일링 기법을 도식하였다. 일반 행렬 곱셈은 좌측 행렬의 한 행과 우측 행렬의 한 열을 곱하여 출력 행렬의 하나의 값을 계산하게 되는데, 행렬의 크기가 커질수록 캐쉬라인을 넘어가는 메모리 접근이 빈번하게 발생하므로 성능 저하가 발생하게 된다. 이러한 캐쉬 미스를 방지할수 있는 방법으로 타일링 기법이 제안되어졌다 [25]. 캐쉬 미스를 최소화 하는 타일 크기의 서브 행렬로 나누고, 서브 행렬들의 곱셈을 수행한 후에 출력 서브 행렬에 누적하는 방식으로 연산하게 된다.

단순 타일링 기법이 모든 크기의 행렬에서 항상 성능 개선이 보이지는 않는다. A행렬 (MxK)과 B (KxN)행렬이 있다고 하자. 충분히 큰 정방형 행렬 (M=K=N)에서 타일 크기에 따른 성능 개선이 선형적으로 증가하지만, 일정 크기 이상으로 커지게 되면 오히려 성능이 저하된다. 비정방형 행렬에서 A행렬이 열 방향으로 길고, B 행렬이 행 방향으로 긴 경우 (M,N ≫ K) 오히려 타일의 빈번한 교체가 발생하므로 성능 저하가 발생할 수 있다.

4. VTA 하드웨어 구조

VTA 하드웨어는 오픈 소스로 공개되어 있어, 그림 3과 같이 VTA 하드웨어 여러 모듈과 세부요소들에 대해 상세히 살펴볼 수 있다 [9]. DRAM과 FPGA 메모리인 Buffer 사이의 데이터 흐름을 제어하기 위한 큐 (Queue)가 존재한다. LOAD 모듈은 DRAM으로부터 입력데이터와 가중치 데이터를 읽어오며, STORE 모듈은 결과 데이터를 DRAM으로 저장한다. 각 입력 버퍼와 가중치 버퍼는 16의 배수로 COMPUTE 모듈의 입력 텐서와 가중치 텐서에 로드된다. COMPUTE 모듈은 16x16 행렬과 16x1 벡터 연산을 한 번의 명령어로 수행할 수 있다. VTA 각 명령어는 Instruction Fetch 모듈을 통해 COMPUTE 모듈로 전송되며, 각 명령어는 OP-code와 Operand로 구성되어있다. Operand는 버퍼의 인덱스로 구성되는데 입력 버퍼는 16의 배수, 가중치 버퍼는 16x16을 한 개의 인덱스만으로 접근이 가능하다. 입력 버퍼, 가중치 버퍼, 출력 버퍼의 크기는 명령어의 Operand 길이에 의존하게

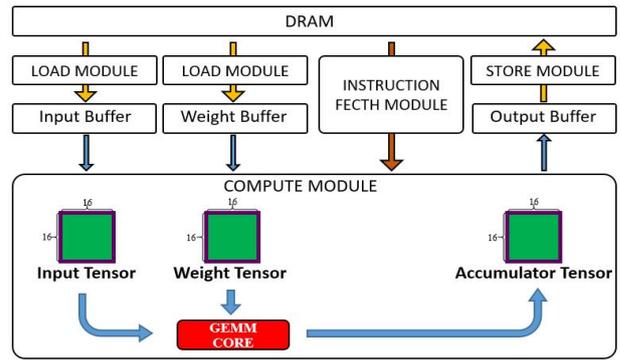


그림 3. VTA 하드웨어 구조 [9]
Fig. 3. VTA Hardware Architecture [9]

된다. 만약 인덱스를 초과 할 경우 기존의 버퍼를 DRAM으로 저장하고 비운 후에, 접근하고자 하는 위치의 데이터를 DRAM으로 새로 로드한 후에 사용해야 한다.

III. 제안하는 기법

1. 2차 타일링 기법

그림 4는 단순 타일링 기법에 추가로 서브 행렬을 타일링 하는 2차 타일링 방식을 도식하였다. 2차 타일링 기법은 1차 타일로 구획을 나누고, 2차 세부 타일로 실제 연산 부분을 구분하게 된다. 그림 5의 알고리즘과 같이 행렬 A,B를 block_size로 구획을 나누고 block_size에 대해 세부 타일 크기인 16x16의 크기로 잘게 쪼갤 수 있게 된다. 16x16 행렬의 곱셈에만 집중하므로 최적화 문제를 쉽게 만들 수 있다. 본 논문에서 제안하는 2차 타일링 기법은 VTA 하드웨어 뿐만 아니라 텐서 기반의 뉴럴네트워크 가속회로에도 바로 적용이 가능할 것으로 기대한다.

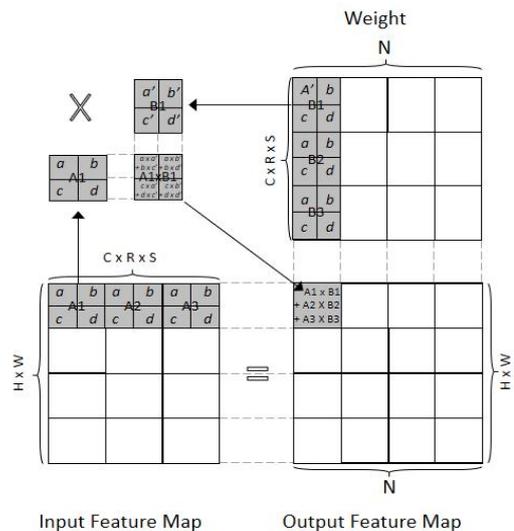


그림 4. 2차 타일링 기반 행렬 곱셈
Fig. 4. 2-stage tiling-based matrix multiplication

Algorithm 1 : 2-stage Tile-based baseline GEMM on CPU

```

1 for lm = 1:M/block_size do
2   for lk = 1:K/block_size do
3     for ln = 1:N/block_size do
4       for bm = 1:block_size/16 do
5         for bk = 1:block_size/16 do
6           for bn = 1:block_size/16 do
7             for h = 1:16 do
8               for w = 1:16 do
9                 for x = 1:16 do
10                  C[h,w](lm,ln,bn) += A[h,x](lm,lk,bm,bk) * B[w,x](lk,ln,bn,bk)
11                end
12              end
13            end
14          end
15        end
16      end
17    end
18  end
19 end
    
```

그림 5. 2차 타일링 기반 CPU 기본 행렬곱셈 알고리즘
Fig. 5. 2-stage tiling-based GEMM algorithm on CPU

Algorithm 2 : 2-Stage Tile-based GEMM on VTA

```

1 for lm = 1:M/block_size do
2   for lk = 1:K/block_size do
3     for ln = 1:N/block_size do
4       LOAD_BLOCK( input_buffer, block_size * block_size)
5       LOAD_BLOCK( weight_buffer, block_size * block_size)
6       for bm = 1:block_size/16 do
7         for bk = 1:block_size/16 do
8           for bn = 1:block_size/16 do
9             VTA_GEMM(C[16x16](lm,ln) += A[16x16](lm,lk) * B[16x16](ln,bk))
10            end
11          end
12        end
13      STORE_BLOCK( output_buffer, block_size * block_size)
14    end
15  end
16 end
    
```

그림 6. 2차 타일링 기반 VTA 행렬곱셈 알고리즘
Fig. 6. 2-stage tiling-based GEMM algorithm on VTA

그림 6의 알고리즘은 VTA 하드웨어에 적용하기 위해 그림 5의 알고리즘을 개선한 버전이다. 그림 5의 7~13번 라인의 16x16 행렬 곱셈 연산을 위한 반복문 3개가 그림 6의 9번 라인의 VTA_GEMM 명령어 한 줄로 대체된다. 버퍼 로드와 연산 과정의 파이프라인이 형성되지 않을 경우 메모리 병목현상이 발생하게 된다. 따라서 그림 6에서 4~5번 라인의 입력 이미지와 가중치 파라미터의 로드 및 연산, 13번 라인의 저장을 연속적으로 수행하도록 작성하였다.

2. VTA 하드웨어에서 2차 타일링 최적화

VTA 하드웨어는 16x16 행렬과 16x1 벡터를 한 번에 연산 할 수 있다. 그림 7 에서와 같이 행렬 곱셈을 위해서는 DRAM에서 FPGA의 온 칩 메모리 (Buffer)로 데이터가 1차적으로 로드된다. 그리고 GEMM 코어 모듈 내의 레지스터 파일로 존재하는 텐서에 2차적으로 데이터가 로드되게 된다. 이러한 VTA 하드웨어 특성을 최대한 이용하려면 입력, 가중치 버퍼에 충분히 많은 데이터를 가지고 와야 한다. 그

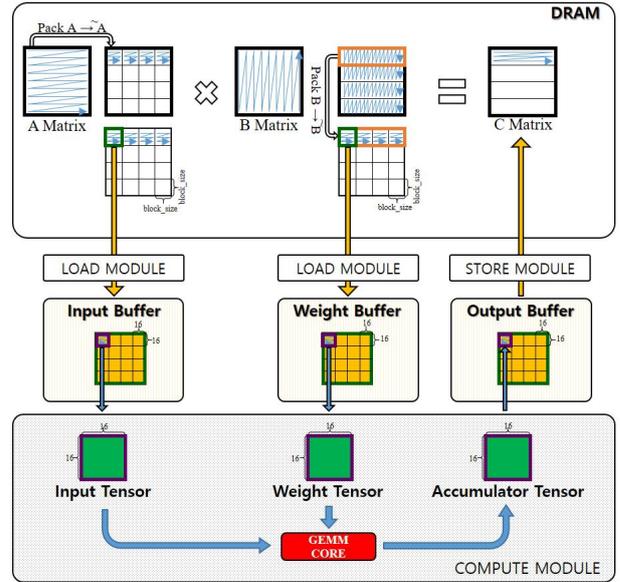


그림 7. VTA의 하드웨어 구조를 이용한 2차 타일링 기반 행렬 곱셈 최적화 기법
Fig. 7. 2-stage tiling-based matrix multiplication optimization technique on VTA

Algorithm 3 : Optimized 2-Stage Tile-based GEMM on VTA (Ours)

```

1 for lm = 1:M/ mblock_size do
2   for bm = 1: mblock_size/16 do
3     for lk = 1:K/ kblock_size do
4       LOAD_BLOCK( input_buffer, 16 * kBLOCK )
5       for n = 1: N/16 do
6         LOAD_BLOCK( weight_buffer, 16 * kBLOCK )
7         for bk = 1: kblock_size/16 do
8           VTA_GEMM(C[16x16](lm,ln) += A[16x16](lm,lk) * B[16x16](ln,bk))
9         end
10        end
11      end
12    end
13    // Flushing after filling the accumulator buffer
14    STORE_BLOCK(output_buffer, mBLOCK * n/16 )
15  end
    
```

그림 8. VTA에 최적화된 2차 타일링 기반 행렬곱셈 알고리즘
Fig. 8. Optimized 2-stage tiling-based GEMM algorithm on VTA

리고 입력,가중치 텐서에서는 재사용성을 높일 수 있는 방식으로 GEMM 코어의 가동률을 높여야 한다. 이 두 가지를 모두 반영하기 위한 기법을 그림 8의 알고리즘으로 작성하였다. 그림 6 알고리즘과 차이점은 1차 타일을 기존에는 동일한 타일 크기로 구획하였지만, 그림8의 최적 알고리즘에서는 M블록과 K블록 두가지 타일 형태로 구분하여 구획하였다. 두가지 크기로 1차 타일을 구분한 이유는 A행렬과 B행렬에서 K의 크기는 동일한 점을 이용하여 A행렬의 재사용성을 높이기 위한 전략이다. 그림 6 알고리즘의 6번 라인이 최적 알고리즘의 2번 라인으로 옮겨진 것 또한 2차 세부 타일 단위로 A행렬을 재사용하도록 의도한 것이다. 그림 8의 4번 라인은 A 행렬을 한번 로드 할 때 5, 6번 라인의 코

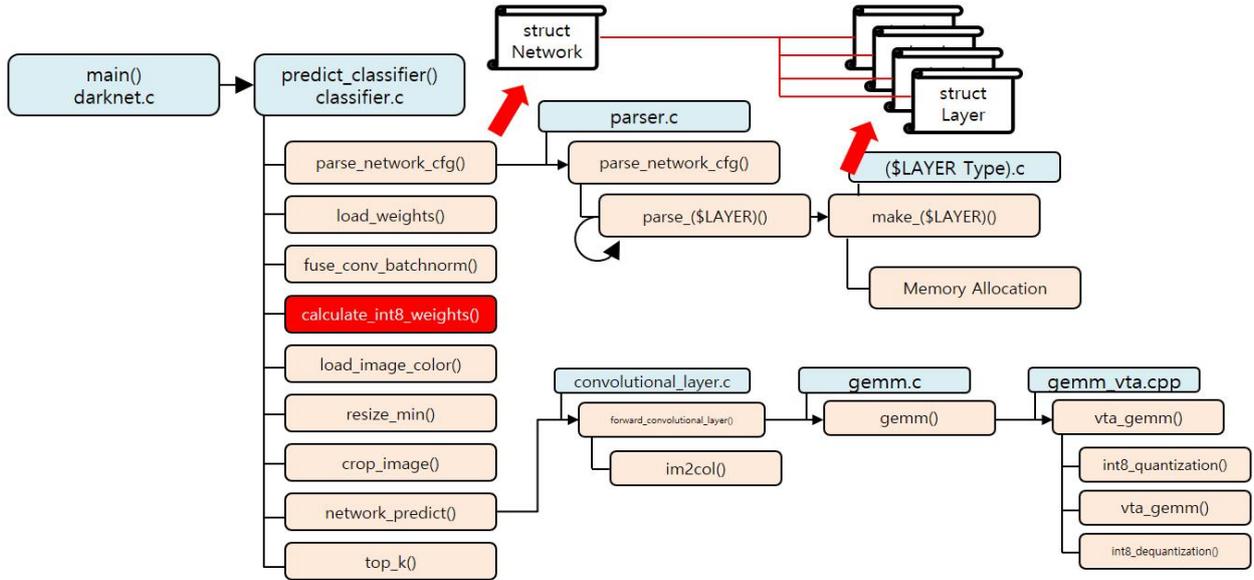


그림 9. DarkNet 프레임워크의 추론 코드 분석 및 VTA GEMM 루틴 추가
 Fig. 9. Analysis of Inference code and Adding VTA GEMM routine in Darknet

드처럼 가중치 버퍼 (16x16)를 N블록 만큼 로드하여 입력 버퍼를 재사용한다. 뿐만아니라 8번 라인의 VTA_GEMM 연산과 6번 라인의 가중치 버퍼의 파이프라인이 생성되므로 연산과 로드가 슬랙 없이 지속적으로 공급 연산 된다. 이러한 과정이 무한정 반복될 수는 없으며, 누적 버퍼의 인덱스가 한계값에 도달하게 되면 누적 버퍼를 출력 버퍼로 저장하고 누적 버퍼를 초기화하게 된다. 이러한 과정을 입력 버퍼를 모두 연산할 때 까지 반복하므로 VTA 가동율을 최대 로 올리게 되었다.

3. DarkNet 프레임워크 성능 최적화

DarkNet 프레임워크 [16]는 순수하게 C로만 작성된 오픈 소스 딥러닝 프레임워크이다. 지원가능한 타겟 하드웨어는 범용 CPU와 NVIDIA GPU로 다른 하드웨어에 대한 백엔드는 구현되어 있지 않다. 따라서 본 연구의 VTA 하드웨어에 최적화된 행렬 곱셈 연산을 DarkNet 프레임워크에 구현하였다. DarkNet 프레임워크는 이미지 분류와 물체 인식 두가지 타스크를 제공하고 있다. 하지만 이미지 분류 모델은 개발 지원이 중단되어 실험을 위해 ResNet-18 모델을 ONNX (Open Neural Network eXchange) 변환을 통해 Darknet으로 변환하였다.

DarkNet 이미지 분류를 위한 코드 흐름을 분석여 그림 9에 작성하였다. 먼저 darknet.c 파일에 main() 함수가 작성되어 있어, predict_classifier() 함수를 호출하여 이미지 분류를 수행하게 된다. classifier.c 파일에는 DarkNet 구성 메타 데이터를 담고 있는 cfg 파일을 파싱하는 parse_network_cfg()함수와 파라미터 파일을 불러오는 load_weights() 함수, 그리고 추론과정에서 BatchNorm 레이어를 생략하기 위한 fuse_conv_batchnorm() 함수등의 부수적인 함수를 통과하면 추론을 위한 준비과정이 완료된다.

입력 이미지는 load_image_color() 함수를 통해 OpenCV 또는 std_img 라이브러리를 통해 픽셀 단위의 조정을 통해 network_predict()함수의 입력으로 들어가게 된다. DarkNet은 Network, layer 두 가지 구조체를 기반으로 동작하게 되는데, Network 구조체에는 레이어 정보들을 배열형태로 가지고 있으며 뉴럴네트워크에 필요한 데이터들을 저장하게 된다. 그리고 Layer 구조체는 cfg 파일에 정의 되어진 레이어들을 실행하는데 필요한 정보들을 담고 있다.

VTA 하드웨어는 연산 가능한 데이터 형식이 INT8 이므로 제안하는 최적의 행렬 곱셈 알고리즘을 DarkNet에 붙이기 위해 calculate_int8_weight() 함수를 작성하여 가중치 파라미터를 미리 양자화 하는 루틴을 추가하여 런타임에 가중치는 양자화를 스킵하게 하였다. VTA 하드웨어에서 GEMM 연산이 수행되도록 하기 위해 convolutional_layer.c 파일의 forward_convolutional_layer() 함수를 포함한 하위 함수들을 수정하였다.

IV. 실험 및 평가

1. 실험 환경

- 타겟 보드 : Xilinx ZCU102 MPSoC
- Processing System (PS)
 - ◆ CPU : ARM Cortex-A53 quad-core
 - ◆ RAM : 4GB
- Programmable Logic (PL)
 - ◆ FPGA-기반 VTA 하드웨어
 - ◆ 클럭 주파수 : 333MHz
 - ◆ 입/출력데이터 형식 : 8비트 (양자화)

표 1. VTA 성능 검증을 위한 정방, 비정방형 행렬의 테스트 케이스

Table 1. Test Cases of Square and Non-square Matrix for VTA Performance Verification

		CNN Layers Dimension						Matrix Dimension		
								M	K	N
Square Matrix							128	128	128	
							256	256	256	
							512	512	512	
							1024	1024	1024	
		H	W	C	R	S	N	M	K	N
Non-square Matrix		64	64	64	3	3	64	4096	576	64
		32	32	128	3	3	128	1024	1152	128
		16	16	128	3	3	128	256	1152	128
		16	16	256	3	3	256	256	2304	256
		8	8	256	3	3	512	64	2304	512
	8	8	512	3	3	512	64	4608	512	

표 2. 정방형/비정방형 행렬 곱셈의 성능 비교 표
Table 2. Performance comparison for square and non-square matrix multiplication

(단위 : millisecond)

Method		Baseline on CPU	2-Stage Tiling on VTA	Optimized 2-stage Tiling on VTA
M, K, N				
Square Matrix	128, 128, 128	6.7	1.7	0.6
	256, 256, 256	51.3	4.9	2.6
	512, 512, 512	403.0	28.9	10.6
	1024, 1024, 1024	3208.9	184.8	58.4
Non-square Matrix	4096, 576, 64	905.0	59.9	19.1
	1024, 1152, 128	448.8	34.5	10.3
	256, 1152, 128	112.2	10.8	4.7
	256, 2304, 256	446.7	39.5	14.2
	64, 2304, 512	225.1	18.9	14.4
64, 4608, 512	452.3	42.5	28.1	

2. 실험 방법

VTA에 최적화된 2차 타일링 기반 행렬 곱셈의 성능을 검증하기 위한 방법으로 정방형 행렬 곱셈과 비정방형 행렬 곱셈을 실험하였다. 정방형 행렬은 행렬 곱셈을 위한 행렬 A(M x K), B(K x N)이 다음과 같을 때, M,K,N 크기가 모두 동일한 경우를 말한다. 비정방형 행렬의 경우 CNN 레이어의 차원축소를 통해 결정된 행렬의 크기로 실험하였다. 실험 데이터는 표 1과 같다. 정방형 행렬은 128 크기에서 2배씩 키워가며 행렬의 크기를 1024행렬 까지 실험하였다. 비정방형 행렬의 경우 ResNet-18 모델의 컨볼루션 레이어의 파라미터 및 입력 크기를 Im2Col 변환을 통해 행렬 차원을 구하였고, 해당 차원을 시험 데이터로 사용하였다.

DarkNet 프레임워크에서 제대로 동작하는지를 확인하기

```

Total BFLOPS 4.742
avg outputs = 14953
Loading weights from resnet18-v1-7.weights...
seen 64, trained: 0 K-images (0 Kilo-batches 64)
Done! Loaded 35 layers from weights-file

calculate_int8_weights Done!
classes = 1000, output in cfg = 1000
256 256
conv 0 : CPU_gemm
conv 1 : VTA_gemm - m k n : 4096 576 64 => 32.119 GOPS (time : 0.009394)
conv 2 : VTA_gemm - m k n : 4096 576 64 => 33.395 GOPS (time : 0.009035)
conv 3 : VTA_gemm - m k n : 4096 576 64 => 33.517 GOPS (time : 0.009002)
conv 4 : VTA_gemm - m k n : 4096 576 64 => 33.414 GOPS (time : 0.009030)
conv 5 : VTA_gemm - m k n : 1024 64 128 => 8.356 GOPS (time : 0.001922)
conv 6 : VTA_gemm - m k n : 1024 576 128 => 29.786 GOPS (time : 0.005065)
conv 7 : VTA_gemm - m k n : 1024 1152 128 => 35.368 GOPS (time : 0.008335)
conv 8 : VTA_gemm - m k n : 1024 1152 128 => 35.482 GOPS (time : 0.008507)
conv 9 : VTA_gemm - m k n : 1024 1152 128 => 35.440 GOPS (time : 0.008518)
conv 10 : VTA_gemm - m k n : 256 128 256 => 10.320 GOPS (time : 0.001619)
conv 11 : VTA_gemm - m k n : 256 1152 256 => 28.376 GOPS (time : 0.005319)
conv 12 : VTA_gemm - m k n : 256 2304 256 => 29.192 GOPS (time : 0.010343)
conv 13 : VTA_gemm - m k n : 256 2304 256 => 28.916 GOPS (time : 0.010441)
conv 14 : VTA_gemm - m k n : 256 2304 256 => 28.926 GOPS (time : 0.010438)
conv 15 : VTA_gemm - m k n : 64 256 512 => 10.863 GOPS (time : 0.001541)
conv 16 : VTA_gemm - m k n : 64 2304 512 => 20.895 GOPS (time : 0.007225)
conv 17 : VTA_gemm - m k n : 64 4608 512 => 22.165 GOPS (time : 0.013623)
conv 18 : VTA_gemm - m k n : 64 4608 512 => 22.235 GOPS (time : 0.013580)
conv 19 : VTA_gemm - m k n : 64 4608 512 => 22.224 GOPS (time : 0.013587)
conv 20 : CPU_gemm
data/egyptian-cat_285.JPEG: Predicted in 981.470000 milli-seconds.

top1. Egyptian cat : 0.501246 ( 285)
top2. Siamese cat : 0.083574 ( 284)
top3. coyote : 0.073890 ( 272)
top4. grey fox : 0.053618 ( 280)
top5. lynx : 0.051351 ( 287)

VTA memory free Done!
    
```

그림 10. Darknet 프레임워크에서 ResNet-18 모델의 수행 결과
Fig. 10. Results of ResNet-18 model in Darknet framework

위해서 MxNet에서 학습된 ResNet-18 모델을 ONNX 변환을 통해 DakNet cfg 파일과 weights 파일로 생성하였고 해당 모델을 양자화를 통해 Int8 모델로 변환하여 수행하도록 하였다. 정확도는 기존 FP32와 비교하여 검증하였다.

3. 실험 결과 및 분석

표 2는 Xilinx ZCU102 보드의 CPU, VTA에서 행렬 곱셈 성능을 측정한 결과이다. CPU에서는 직접 컨볼루션 연산을 수행한 시간을 밀리 초 단위로 기록하였다. VTA 하드웨어의 성능 비교를 위해 2차 타일링 행렬 곱셈과 최적화된 2차 타일 행렬 곱셈의 성능을 비교하였다.

정방형 행렬의 크기가 2배씩 증가 됨에 따라 CPU의 연산 시간은 2³ 만큼 증가하였지만, VTA 하드웨어를 이용할 경우 기본 루틴에서 2²³배씩 증가, 최적화된 루틴에서 2²¹배씩 증가하여 최적화된 루틴의 성능이 개선 되었음을 알 수 있다. 최적화 루틴은 비 최적화 루틴 대비 평균 2.6배 더 빨랐다.

비정방형 행렬은 CNN 레이어의 크기를 반영하므로 A, B 행렬의 크기가 일관되지 않으므로 행렬 곱셈 성능을 최대로 발휘되지 않는 경우들이 발생한다. 본 논문에서 제안한 기법은 비정방형 행렬의 경우에도 비최적화 루틴 대비 최적화 루틴의 성능이 평균 2.4배 더 개선되었음을 확인하였다.

그림 10은 DarkNet 프레임워크에서 ResNet-18 모델을 수행한 결과이다. 각 컨볼루션 레이어에서 평균 26 GOPS 이상의 성능을 보였다. 성능이 저하된 5, 10, 15번 레이어의 특징은 1x1 커널을 갖는 컨볼루션 레이어로써 연산량이 적은 부분으로써 기본적인 VTA 하드웨어를 구동하는 오버헤드가 오히려 크게 반영되어, 상대적으로 성능이 낮게 나오는 것 처럼 보인다. 하지만 실제 연산 수행시간은 CPU 대비 더 높은 성능을 보인다. DarkNet에서 제공하는 CPU 타

켓에서 OpenMP를 활성화 한 경우 ResNet-18은 평균 2.1초 이상의 추론 시간이 소요되었지만, VTA 타겟에서는 0.9초 미만의 추론시간을 보여 최소 2.3배 이상의 추론 성능 개선을 보였다.

V. 결론

본 논문에서는 FPGA 기반 뉴럴네트워크 가속 회로인 VTA 에서 최적의 2차 타일링 행렬 곱셈 루틴을 제안하였다. 비 최적화 루틴의 빈번한 버퍼 메모리 교체로 인한 병목현상 해결을 위해 ACCUMULATOR 버퍼를 가득 채울 수 있도록 타일 크기를 수정하였다. 명령어의 버퍼 인덱스가 허용하는 범위 내에서 입력 버퍼와 가중치 버퍼가 최대한 재사용 되도록 루틴을 최적화하였다. 비 최적화 루틴 대비 제안하는 기법은 정방향 행렬에서 평균 2.6배, 비 정방향 행렬에서 2.4배 개선되었다. 또한 DarkNet 프레임워크에 제안하는 VTA 최적 행렬 곱셈 루틴을 구현하여 ResNet-18 모델에서 기존 CPU 대비 성능이 개선됨을 실험을 통해 보였다.

References

[1] J. W. Bae, B. G. Han, "Implementation of Deep Learning-based Label Inspection System Applicable to Edge Computing Environments," IEMEK J. Embed. Sys. Appl, Vol. 17, No. 2, pp. 77-83, 2022 (in Korean).

[2] J. Y. Choi, H. J. Lee, C. W. Jeong, H. C. Jung, "Development of AI Service with Surgical Tools Segmentation and Action Recognition," IEMEK J. Embed. Sys. Appl, Vol. 16, No. 2, pp. 51-57, 2021 (in Korean).

[3] H. D. Kim, "Design of Speech Enhancement U-Net for Embedded Computing," IEMEK J. Embed. Sys. Appl, Vol. 15, No. 5, pp. 227-234, 2020 (in Korean).

[4] H. J. Kim, "Analysis of Reduced-Width Truncated Mitchell Multiplication for Inferences Using CNNs," IEMEK J. Embed. Sys. Appl, Vol. 15, No. 5, pp. 235-242, 2020 (in Korean).

[5] J. M. Lee, M. S. Yu, Y. I. Kwon, T. H. Kim, "Quantune: Post-training Quantization of Convolutional Neural Networks using Extreme Gradient Boosting for fast Deployment," Future Generation Computer Systems 132, pp. 124-135, 2022.

[6] G. Y. Kwon, S. W. Park, T. W. Suh, "Cycle-accurate NPU Simulator and Performance Evaluation According to Data Access Strategies," IEMEK J. Embed. Sys. Appl, Vol. 17, No. 4, pp. 217-228, 2022 (in Korean).

[7] <https://coral.ai/products/dev-board>

[8] <https://www.intel.com/content/www/us/en/developer/tools/neural-compute-stick/overview.html>

[9] T. Moreau, T. Chen, L. Vega, J. Roesch, E. Yan, L. Zheng, J. Fromm, Z. Jiang, L. Ceze, C. Guestrin, A. Krishnamurthy, "A Hardware - software Blueprint for Flexible Deep Learning Specialization," IEEE Micro, Vol.

39, No. 5, pp. 8-16, 2019.

[10] S. Chetlur, C. Woolley, P. Vanderersch, J. Cohen, J. Tran, B. Catanzaro, E. Shelhamer, "cudnn: Efficient Primitives for Deep Learning." arXiv preprint arXiv:1410.0759, 2014.

[11] <https://www.openblas.net/>

[12] <https://github.com/Reference-LAPACK/lapack>

[13] <https://www.arm.com/technologies/compute-library>

[14] <https://github.com/Maratyszczka/NNPACK>

[15] E. Wang, Q. Zhang, B. Shen, G. Zhang, X. Lu, Q. Wu, Y. Wang, "Intel Math Kernel Library," High-Performance Computing on the Intel® Xeon Phi TM, Springer, Cham, pp 167-188, 2014.

[16] <https://github.com/AlexeyAB/darknet>

[17] K. He, X. Zhang, S. Ren, J. Sun, "Deep Residual Learning for Image Recognition," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770-778, 2016.

[18] A. Anderson, A. Vasudevan, C. Keane, D. Gregg, "Low-memory Gemm-based Convolution Algorithms for Deep Neural Networks," arXiv preprint arXiv:1709.03395, 2017.

[19] J. S. Park, K. M. Bin, K. H. Lee, "mGEMM: Low-latency Convolution with Minimal Memory Overhead Optimized for Mobile Devices," Proceedings of the 20th Annual International Conference on Mobile Systems, Applications and Services, pp. 222-234, 2022.

[20] M .S. Cho, B. Daniel, "MEC: Memory-efficient Convolution for Deep Neural Network," International Conference on Machine Learning. PMLR, pp. 815-824, 2017.

[21] M. Dukhan, "The Indirect Convolution Algorithm." arXiv preprint arXiv:1907.02129, 2019.

[22] <https://docs.nvidia.com/cuda/cublas/index.html>

[23] C. Nugteren, "CLBlast: A Tuned OpenCL BLAS Library," Proceedings of the International Workshop on OpenCL, pp. 1-10, 2018.

[24] <https://github.com/clMathLibraries/clBLAS>

[25] K. Goto, V. D. G Robert, "High-performance Implementation of the Level-3 BLAS," ACM Transactions on Mathematical Software (TOMS) Vol. 35, No. 1, pp. 1-14, 2008.

Jinse Kwon (권진세)



2014 Mechanical Engineering from Chungnam National University (B.S.)
 2017 Computer Engineering from Chungnam National University (M.S.)
 2017~Computer Engineering from Chungnam National University (Ph.D. Student)

Field of Interests: Embedded AI Computing, Efficient AI, Embedded AI Acceleration
 Email: kwonse@cnu.ac.kr

Jemin Lee (이 제 민)

2011 Computer Engineering from Chungnam National University (B.S.)
 2017 Computer Engineering from Chungnam National University (Ph.D.)
 2017~2018 Knowledge Service Engineering from KAIST(Post-Doc)

Career:

2018~Senior Researcher, AI SoC Research Division from ETRI
 2019 Co-chair, ACM Mobisys
 2022~Committee Member, IeMeK

Field of Interests: Mobile Computing, Efficient AI, AI Compiler

Email: leejaymin@etri.re.kr

Yongin Kwon (권 용 인)

2008 Electrical and Electronic Engineering from KAIST (B.S.)
 2010 Electrical and Computer Engineering from Seoul National University (M.S.)
 2015 Electrical and Computer Engineering from Seoul National University (Ph.D.)

Career:

2015~2019 Principal Researcher, System LSI, Samsung Electronics

2019~Senior Researcher, AI SoC Research Division from ETRI
 Field of Interests: Mobile Cloud Computing, Compiler, Deep Learning, Embedded Systems

Email: yongin.kwon@etri.re.kr

Jeman Park (박 제 만)

2004 Electrical and Electronic Engineering from Hanyang University (B.S.)
 2006 Electronics and Computer Engineering from Hanyang University (M.S.)
 2014 Electronics and Computer Engineering from Hanyang University (Ph.D.)

Career:

2012~Senior Researcher, AI SoC Research Division from ETRI
 Field of Interests: Computer Network, Mobil Edge Computing (MEC), AI Compiler

Email: jeman@etri.re.kr

Misun Yu (유 미 선)

1999 Computer Engineering from Chungnam National University (B.S.)
 2002 Computer Engineering from Pohang University of Science and Technology (M.S.)

Career:

2002~Principal Researcher, AI SoC Research Division from ETRI
 2018~Director of Neuromorphic Computing Software Platform for AI System

Field of Interests: Deep Learning Compiler, Neural Network Partitioning, Parallel Processing

Email: msyu@etri.re.kr

Taeho Kim (김 태 호)

1995 Information Engineering from Sungkyunkwan University (B.S.)
 1997 Computer Science from KAIST (M.S.)
 2005 Computer Science from KAIST (Ph.D.)

Career:

2001~2002 International Fellow, USA SRI CSL

2004~2005 Principal Researcher, Digital Printing Division, Samsung Electronics

2005~Assistant Vice President, AI SoC Research Division, ETRI
 Field of Interests: AI SoC, Safety-critical, Intelligent CPS, Software Engineering, Artificial Intelligent, Code Generation

Email: taehokim@etri.re.kr

Hyungshin Kim (김 형 신)

1990 Computer Science from KAIST (B.S.)
 1991 Satellite Communications Engineering from Univ. of Surrey (M.S.)
 2003 Computer Science from KAIST (Ph.D.)
 2004~Dept. of Computer Science and Engineering, Chungnam National Univ. (Professor)

Career:

1992 Researcher, SaTReC, KAIST

2003 Post Doc Researcher, CMU

Field of Interests: Real-time Embedded Software, Embedded AI Computing

Email: hyungshin@cnu.ac.kr

논문 2022-17-44

차량용 임베디드 프로세서에서 저전력 반응적 제어를 위한 이기종 멀티코어 협력적 스트리밍 온-칩 소프트웨어 구조 (Collaborative Streamlined On-Chip Software Architecture on Heterogenous Multi-Cores for Low-Power Reactive Control in Automotive Embedded Processors)

권지수, 박대진*
(Jisu Kwon, Daejin Park)

Abstract : This paper proposes a multi-core cooperative computing structure considering the heterogeneous features of automotive embedded on-chip software. The automotive embedded software has the heterogeneous execution flow properties for various hardware drives. Software developed with a homogeneous execution flow without considering these properties will incur inefficient overhead due to core latency and load. The proposed method was evaluated on an target board on which a automotive MCU (micro-controller unit) with built-in multi-cores was mounted. We demonstrate an overhead reduction when software including common embedded system tasks, such as ADC sampling, DSP operations, and communication interfaces, are implemented in a heterogeneous execution flow. When we used the proposed method, embedded software was able to take advantage of idle states that occur between heterogeneous tasks to make efficient use of the resources on the board. As a result of the experiments, the power consumption of the board decreased by 42.11% compared to the baseline. Furthermore, the time required to process the same amount of sampling data was reduced by 27.09%. Experimental results validate the efficiency of the proposed multi-core cooperative heterogeneous embedded software execution technique.

Keywords : Automotive, Embedded system, Heterogeneous execution, Micro-controller unit, Multi-core

1. 서론 및 관련 연구

최근 자동차의 패러다임이 내연기관에서 모터, 배터리 제어를 비롯한 전자 제어 장치들로 전환되어감에 따라 차량 내부의 각종 센서 동작을 제어하는 임베디드 소프트웨어에 대한 관심이 높아지고 있다 [1-5]. 차량에서 사용되는 임베디드 소프트웨어는 내부의 여러 전자 제어 유닛 (ECU)에서 프로그래밍 대로 각각의 역할을 수행한다. 또한, 차량에서 ECU가 차지하는 비중도 점차 커지고 있다. 최근에는 차량의 원가에서 ECU 및 소프트웨어가 차지하는 비중이 50%에 육박하고 있다. 하지만, ECU를 구성하는 마이크로컨트롤러 유닛 (MCU)에 적체될 소프트웨어는 운영체제 위에서 수행되는 일반적인 프로그램이 아니기에 개발 단계에서 효율성 증대를 위해 차량용 어플리케이션에 대한 이해가 필요하다.

ECU의 차량용 소프트웨어는 코드 라인의 증가 및 제어

대상 센서 수의 증가에 따라 수행 복잡도 또한 증가하고 있다. 이러한 복잡성으로 인해 소프트웨어 실행과정에서 발생하는 지연 및 오류는 차량용 소프트웨어의 특성상 큰 피해로 이어질 수 있다. 따라서 차량용 ECU에 사용되는 MCU와 소프트웨어 특성을 바탕으로 효율적 실행이 가능케 하는 연구가 필요하다 [6-11].

본 논문에서는 소프트웨어의 실행 관점에서 동종 (homogeneous) 및 이기종 (heterogeneous) 특성을 정의한다. 소프트웨어 실행에서의 동종 특성은 동일한 제어 흐름에서 동일한 컴포넌트에 접근하고 데이터를 처리하는 것으로 정의한다. 반대로 이기종 특성은 실행 과정에서 데이터 처리 또는 연산을 위해 서로 독립적인 컴포넌트에 접근하는 제어 흐름으로 정의한다. 대부분의 차량용 MCU 임베디드 소프트웨어는 제어 대상을 센싱하고, 취득한 데이터를 처리하여 이를 전송하는 일련의 과정으로 구성된다. 하지만 이러한 동종 소프트웨어 실행 흐름으로 이루어진 상태에서는 다양한 하드웨어 제어 요소들로 인해 많은 오버헤드가 발생한다. 임베디드 소프트웨어에서는 analog-to-digital converter (ADC)를 사용한 샘플링이나, 인터럽트의 처리 등 소프트웨어 실행 흐름에 따라 개발자가 개입하여 최적화할 수 있는 여지가 매우 큰 이기종 특성을 보인다 [12-14].

본 논문은 범용 마이크로컨트롤러가 아닌 차량용 임베디

*Corresponding Author (boltanut@knu.ac.kr)

Received: Oct. 14, 2022, Revised: Oct. 25, 2022, Accepted: Nov. 15, 2022.

J. S. Kwon: Kyungpook National University (Ph.D. Student)

D. J. Park: Kyungpook National University (Assoc. Prof.)

※ 본 논문은 교육부의 재원으로 한국연구재단 (NRF-2018R1A6A1A03025109, NRF-2022R111A3069260)의 지원을 받아 수행된 연구임.

※ 본 논문은 과학기술정보통신부의 재원으로 정보통신기획평가원 (No. 2021-0-00944, No. 2022-0-00816, No. 2022-0-01170)의 지원을 받아 수행된 연구임.

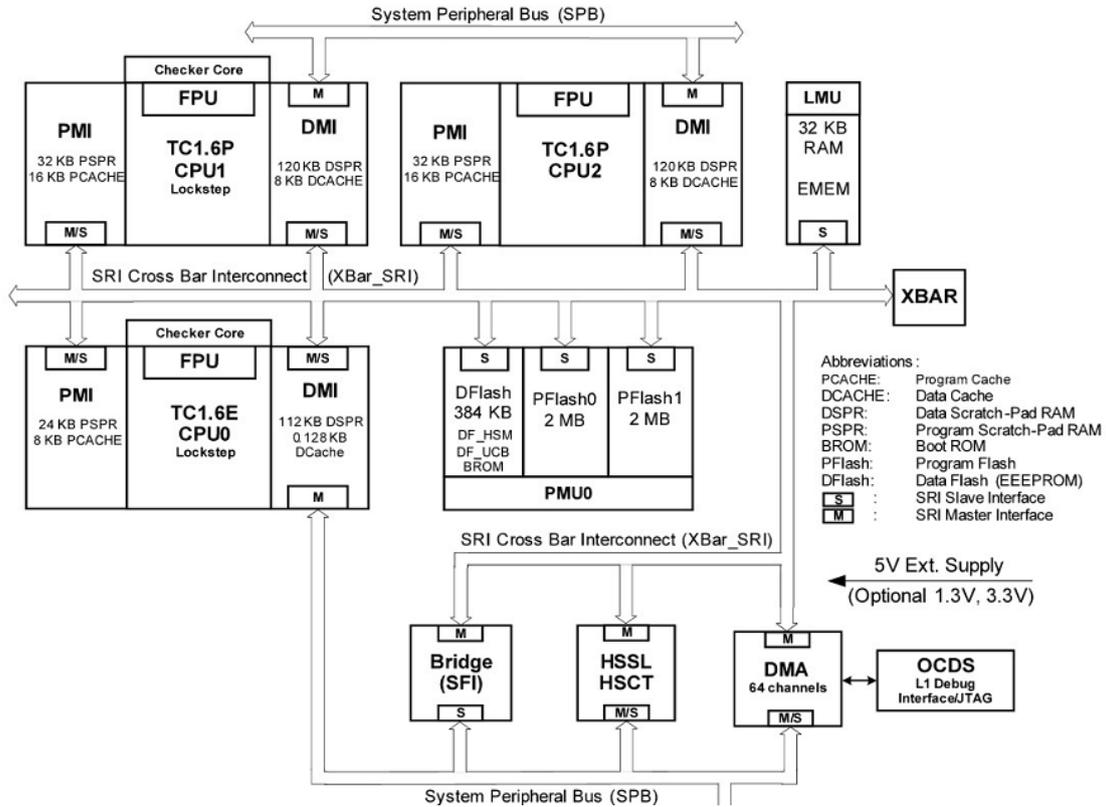


그림 1. TC275 MCU의 내부 구조 블록 다이어그램 [18]
Fig. 1. The internal architecture block diagram of the TC275 MCU

드 프로세서에 특화된 소프트웨어 구조를 제안하고자 한다. 차량용 임베디드 프로세서는 고온, 고압 등 열악한 환경에서 동작하기 위해 범용 MCU와 내부 구조, 개발 방법 등에서 차이가 존재한다. Infineon사의 AURIX™ TriCore™ 차량용 MCU는 독립적으로 소프트웨어 실행이 가능한 복수의 코어를 온-칩에 내장하고 있다. 차량용 MCU의 멀티-코어를 활용하여 한정된 자원을 효율적으로 사용하고자 하는 연구가 활발히 수행되고 있다 [15-17]. 본 논문에서는 멀티-코어 환경에서 동종 실행 흐름으로 구현된 소프트웨어의 태스크들을 차량용 소프트웨어의 이기종 실행 특성에 맞게 각각의 코어에 분담하는 기법을 제안한다. 임베디드 소프트웨어의 전체 실행과정은 크게 데이터 sensing, processing, collecting으로 나눌 수 있는데, 각각의 과정들을 태스크로 할당하고 이를 복수의 코어에서 연속적으로 실행할 수 있도록 코어 간의 코드 실행을 스케줄링한다. 결과적으로 태스크 간의 대기 시간으로부터 발생하는 오버헤드를 최소화함으로써 실행 시간 및 전력 소모를 개선할 수 있다.

본 논문은 다음과 같이 구성된다. 2장에서는 Infineon사의 AURIX™ TriCore™ 임베디드 소프트웨어 개발과정을 소개한다. 3장에서는 동종 실행 흐름으로 구현된 임베디드 소프트웨어에 제안하는 기법을 적용하는 과정을 설명한다. 4장에서는 타겟 보드에서 코어 간의 협력적 이기종 연산을 적용하여 실행 시간 및 전력 소모를 분석한 결과를 설명하고, 마지막 5장에서 결론을 맺도록 한다.

II. 차량용 임베디드 소프트웨어 개발

본 논문에서 사용하는 차량용 MCU는 Infineon사의 AURIX™ TriCore™ TC27x 멀티-코어 계열이다. TC27x 계열의 MCU 중, 본 논문에서는 TC275를 대상으로 설명 및 실험을 수행한다. 코어 3개로 이루어진 TC275의 전반적인 구조를 그림 1에서 나타내었다 [18].

하단 SPB (system peripheral bus)에 연결된 여러 가지 주변 장치 (peripheral)들은 그림에서 생략하였다. TC275에는 CPU0부터 CPU2까지 3개의 코어가 존재하는데, 각 코어는 독립적으로 수행할 코드를 저장하기 위한 플래시 메모리를 보유하고 있다. 모든 코어는 동일한 TriCore™ 1.6 아키텍처 기반으로 구현되었지만, 상대적으로 저전력에 초점을 맞춘 E-코어 (efficiency) 1개와 성능에 초점을 맞춘 P-코어 (performance) 2개로 구성된다. E-코어와 P-코어는 캐시 메모리의 크기나 파이프라인 구조에서 서로 다른 특징을 보인다.

멀티-코어 구조에서는 온-칩 메모리 요소에 대한 접근을 관리하는 것이 중요하다. TC275의 각 코어는 독립적으로 사용하는 scratch-pad (SPR), 캐시 메모리뿐만 아니라, cross-bar 버스 (SRI)를 통해서 공유 메모리 자원들에 접근할 수 있다. 코어가 사용할 수 있는 공유 메모리 자원들에는 PMU (Program Memory Unit)의 코드/데이터 플래시 메모리 (PFlash/DFlash), LMU (Local Memory Unit)의 SRAM이

표 1. Infineon AURIXTM TriCoreTM TC275 MCU 사양
Table 1. Specification of Infineon AURIXTM TriCoreTM TC275 MCU

	TC1.6E (Efficiency)	TC1.6P (Performance)
Operating Clock Frequency	up to 200 MHz	up to 200 MHz
Scratch-Pad RAM (SPR)	Data: up to 112 Kbyte (DSPR) Instruction: up to 24 Kbyte (PSPR)	Data: up to 120 Kbyte (DSPR) Instruction: up to 32 Kbyte (PSPR)
Cache	Instruction: up to 8 Kbyte (ICACHE)	Data: up to 8 Kbyte (DCACHE) Instruction: up to 16 Kbyte (ICACHE)
Flash Memory	Data: up to 384 Kbyte (DFLASH) Instruction: up to 4 Mbyte (PFLASH)	
SRAM	32 Kbyte	

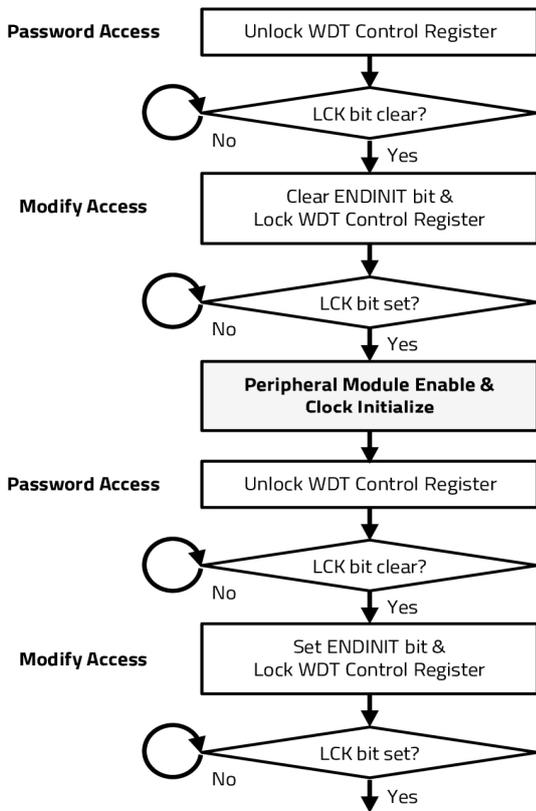


그림 2. 시스템 critical WDT 제어 레지스터 잠금 해제를 통한 주변 장치 모듈 초기화 순서도
Fig. 2. Peripheral module initializing flow chart by system critical WDT control register unlock

있다. TC275 MCU의 세부 사양은 표 1에 기재하였다. TC275의 주변 장치 제어는 SPB를 통한 레지스터 설정으로 이루어진다. 3개 코어 모두 SPB에 연결되어 동일한 조건에서 주변 장치들을 제어할 수 있다. 이때 후술할 초기화 과정에서 특정 주변 장치에 접근하는 코어에 따라 각 코어에 할당된 보호 레지스터 설정이 필요하다.

TC275는 차량용으로 설계된 MCU이기에 일반적인 범용 MCU와 다른 특징이 존재한다. 차량용뿐만 아니라, 범용 MCU에 적재되는 소프트웨어 설계는 대상 하드웨어에 대한

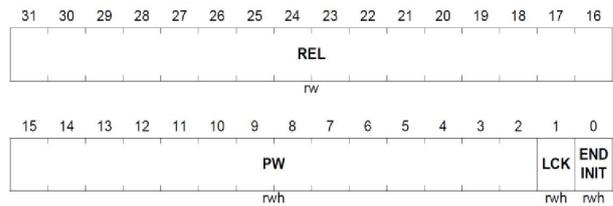


그림 3. 32비트 WDT 제어 레지스터 필드 구조
Fig. 3. 32-bit WDT control register field structure

이해를 바탕으로 개발되어야 한다. 본 논문에서 대상으로 삼고 있는 차량용 임베디드 프로세서의 멀티 코어 협력적 연산을 구현하기 위해서 MCU 하드웨어 제어에 대한 설명을 선행한다. 차량용 임베디드 프로세서의 다양한 특징 중, 주변 장치 제어를 하기 위한 프로그래밍 관점에 대해 설명하도록 한다. 주행 중인 차량의 내부는 높은 온도와 압력, 진동으로 인해 전자 제어 장치의 동작에 노이즈가 발생하기 쉬운 열악한 환경이다. 이러한 환경에서도 차량용 MCU는 외란의 영향을 최소화하며 강인하게 동작해야 차량의 피해가 발생하지 않는다. 따라서 TC275는 사용자의 의도와 달리 MCU의 모듈 컴포넌트가 비정상적으로 동작하는 상황을 방지하기 위해 복잡한 시퀀스 (sequence)를 통해 모듈을 초기화하도록 한다. TC275에서 하드웨어 모듈을 사용하기 위해 초기화하는 과정은 그림 2와 같다.

모듈을 구동하거나 클락 신호를 인가하는 설정에 해당하는 레지스터는 TC275에서 system critical 레지스터에 해당하기 때문에 이를 조작하려면 Password Access, Modify Access 두 단계의 시퀀스가 필요하다. TC275 멀티-코어 구조는 3개의 코어를 내장하는데, 각 코어에서 병렬적인 주변 장치 접근을 보장함과 동시에, 동일한 주변 장치 접근으로 인한 버스 충돌을 방지하기 위해서 코어마다 할당된 WDT (watchdog timer) 제어 레지스터가 존재한다. 그림 3은 첫 번째 코어 CPU0에 할당된 32비트 WDT 제어 레지스터이다.

임의의 주변 장치 모듈에 클락 신호를 공급하고 enable시키기 위해선 WDT 제어 레지스터의 ENDINIT 필드를 set 해야 하고, 이를 위해서는 Password Access를 통해 WDT 제어 레지스터의 잠금을 먼저 해제해야 한다. 이때 CPU0

표 2. 시스템 critical 레지스터 접근에 따른 LCK, ENDINIT 비트 조합
Table 2. LCK, ENDINIT bit combination according to system critical register access

	Password Access	Modify Access (Before Init.)	Modify Access (After Init.)
LCK	0	1	1
ENDINIT	1	0	1

에서 주변 장치를 enable하는 코드가 수행 중이라고 가정하면, 서로 다른 코어 사이의 충돌을 방지하기 위해 CPU0에 할당된 WDT 제어 레지스터 WDTCPU0CON0에서 잠금을 해제할 수 있다. Password Access를 위해서는 32비트 전체 WDT 제어 레지스터의 값을 읽되, PW 필드는 값을 반전하여 읽는다. 그리고 레지스터로부터 읽은 값을 그대로 다시 WDT 제어 레지스터에 write하는데 레지스터의 잠금을 풀고자 하는 것이 목적이므로 LCK 비트의 값을 0으로 clear, ENDINIT 비트의 값을 1로 set한다. 유의해야 할 점은 WDT 제어 레지스터에 32비트 전체를 한 번에 write해야 한다는 점이다. WDT 제어 레지스터의 LCK 비트가 0으로 읽히면 잠금이 해제되고 ENDINIT 비트를 수정할 수 있다.

WDT 제어 레지스터의 ENDINIT 비트가 0의 값이어야만 주변 장치 초기화 설정이 가능하므로, Modify Access를 통해 잠금이 해제된 레지스터의 ENDINIT 비트를 수정한다. 앞서 Password Access와 동일하게 WDT 제어 레지스터의 값을 읽고, 그대로 write하되, ENDINIT 비트의 값을 0으로 clear함과 동시에 LCK 비트를 잠그기 위해 1로 set한다. LCK 비트가 1로 읽히면 주변 장치를 제어하는 레지스터에서 해당 모듈을 enable하거나 클락을 인가하도록 설정할 수 있다. 주변 장치 초기화 설정을 완료한 후에는 이전과 동일하게 Password Access에서 WDT 제어 레지스터의 잠금을 해제하고, Modify Access에서 WDT 제어 레지스터의 ENDINIT 비트를 1로 set한다. WDT 제어 레지스터에 접근하는 목적에 따라 3가지의 조합이 존재한다. 각 조합에 따른 LCK, ENDINIT 비트의 값을 표 2에 나타내었다.

III. 멀티-코어 협력적 연산

일반적인 차량용 임베디드 소프트웨어의 역할은 센서가 데이터를 수집하고, 이를 연산하여 통신 인터페이스로 전송하는 3가지 단계로 구분할 수 있다. 그림 4는 일련의 과정이 동종 소프트웨어 실행 흐름으로 구현된 경우를 나타내었다. 하나의 소프트웨어는 서로 다른 역할을 하는 복수의 태스크들로 구성되어 간헐적으로 발생하는 이벤트에 대응하는 서비스 복잡도뿐만 아니라, 구동하는 하드웨어도 모두 다르다. 이러한 이기종 특성을 고려하지 않고, 단일 코어 기반 실행 흐름으로 프로그램을 개발하게 되면 각 태스크 사이의 idle, wait으로 인한 불필요한 오버헤드가 발생할 수 있다. 임베디드 소프트웨어 실행과정에서 데이터 센싱은 ADC의 샘플링으로 대응된다. ADC는 외부에서 입력되는 아날로그 신호를 샘플링하여 그 결과를 버퍼에 저장한다. 센싱 데이터에 digital signal processing (DSP) 등의 연산을 적용하는 경우는 많은 계산량으로 인해 오버헤드가 더 크게 작용할 수 있다. 본 논문에서 가정된 DSP 연산에는 이동 평균, FIR 필터 등 목표로 하는 도메인에 따라 다양한 알고리즘이 사용될 수 있다. 이러한 DSP 연산 처리 과정에서 ADC 샘플링 지연이나, DSP 연산 적용 범위의 데이터가 확보를 위한 대기로부터 오버헤드가 발생한다. 또한, DSP 처리가 완료된 데이터를 전송하는 통신 인터페이스로는 universal asynchronous receiver / transmitter (UART)가 사용되었다. DSP 연산이 완료되어 버퍼에 저장된 데이터들은 UART 통신으로 호스트에 전송된다. 전체 실행 흐름은 단일 코어에서 동종으로 구현되어 있지만, 세부적인 태스크들이 구동하는 하드웨어 요소, 코드 실행의 순서관계 등을 고려하면 이기종 실행 흐름 특성을 보인다. 이러한 특성을 고려하지 않은 상태로 차량용 임베디드 소프트웨어를 구현한다면 비효율적인 동작을 하게 된다.

본 논문에서 타겟으로 삼은 차량용 임베디드 프로세서에는 메모리 관리 유닛이 존재하지 않으므로 운영 체제 수준에서 제공하는 쓰레드, 멀티 프로세스 등, 기존 멀티 코어 활용을 위한 API를 사용하기 어렵다. 따라서 개발자의 타겟 차량용 임베디드 프로세서 하드웨어에 대한 이해를 바탕으

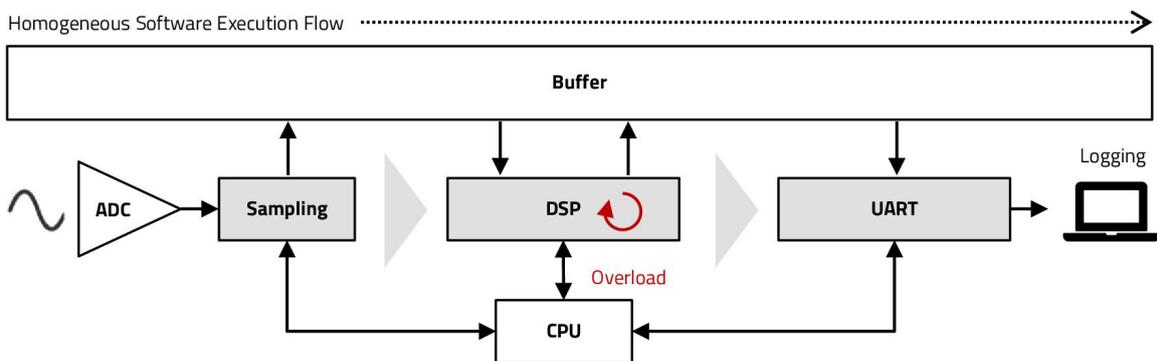


그림 4. 기존 동종 실행 흐름 임베디드 소프트웨어 구조
Fig. 4. Conventional homogeneous execution flow embedded software structure

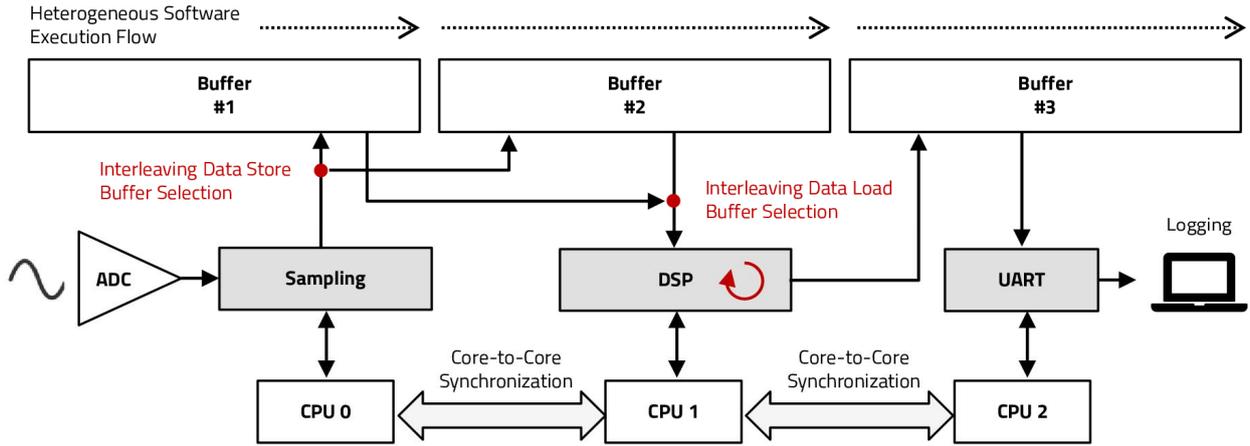


그림 5. 제안하는 멀티-코어의 협력적 연산 기반 이기종 임베디드 소프트웨어 실행 구조
 Fig. 5. Proposed multi-core cooperative operation-based heterogeneous embedded software execution structure

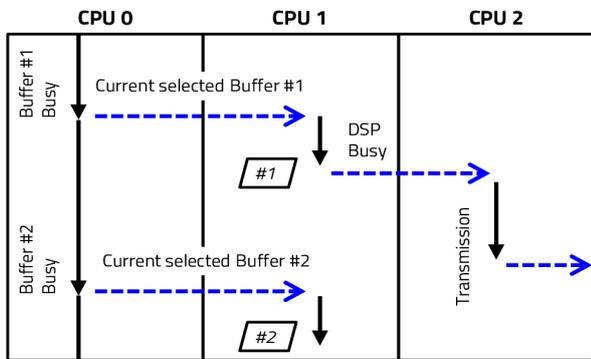


그림 6. 멀티-코어 동기화 개요
 Fig. 6. Multi-core synchronization overview

로 한 소프트웨어 설계가 필요하다. 따라서 그림 5와 같이 멀티-코어 구조에서 각 코어에 이기종 태스크들을 할당하여 오버헤드를 최소화하고 연속적인 프로그램 실행이 가능케 하였다. 이를 위해서 기존의 단일 코어가 모든 과정을 관리하는 구조에서 3개의 코어를 사용하고 각 코어가 샘플링, DSP, 전송 태스크들을 나누어 맡도록 하였다. 또한, 기존의 단일 버퍼 구조에서 3개의 버퍼를 사용하도록 변경하였다. 첫 번째 코어 CPU 0은 ADC 하드웨어를 구동하여 샘플링을 수행한다. 샘플링 수행 결과는 버퍼에 저장하되, 버퍼의 사용 여부를 확인하여 첫 번째 버퍼 #1과 두 번째 버퍼 #2를 번갈아가며 사용한다. 두 번째 코어 CPU 1에서는 앞서 CPU 0이 샘플링 데이터를 저장한 버퍼로부터 데이터를 읽어 DSP 연산을 수행한다. 이때 ADC 샘플링 데이터가 저장되는 버퍼가 2개이므로 CPU 0으로부터 idle 상태의 버퍼 정보를 확인하여 DSP를 수행한다. 예를 들어 현재 CPU 0이 첫 번째 버퍼에 샘플링 데이터를 저장 중이라면, CPU 1은 idle 상태인 두 번째 버퍼로부터 데이터를 읽어 DSP 연산 처리를 한다. CPU 1에서 DSP 연산이 완료된 데이터는 세 번째 버퍼 #3에 저장된다. 세 번째 코어 CPU 2는 버퍼 #3에 저장된 데이터를 UART 통신 인터페이스로 호스트 PC에 전송한다.

현재 멀티-코어를 사용하는 구조에서는 코어 간의 동기화 문제로 underrun, overrun이 발생할 수 있다. 이를 방지하기 위해 각 코어 간에 플래그를 사용하여 버퍼 및 DSP의 상태를 확인할 수 있도록 하였다. 각 코어 간의 동기화 관계를 그림 6에 나타내었다. CPU 1은 CPU 0으로부터 현재 ADC 샘플링 데이터가 저장되는 버퍼 상태를 확인하고, DSP 연산을 적용할 버퍼를 결정한다. 이때 더블 버퍼를 사용함으로써 버퍼 점유로 인해 발생할 수 있는 비효율적인 오버헤드를 제거하였다. 하나의 버퍼가 CPU 0에 의해 점유되는 동안, 다른 버퍼에 대해 CPU 1에서 접근하여 데이터 연산 처리를 위해 읽어감으로써 CPU 0의 연속적인 동작을 보장할 수 있다. CPU 2는 CPU 1로부터 DSP 연산 수행 상태를 확인하고 버퍼에 저장된 데이터 전송 시작 여부를 결정한다. 이러한 과정에서 CPU 1과 2는 CPU 0의 더블 버퍼 동작으로부터 발생하는 오버헤드 시간 동안에 idle 구간을 최소화할 수 있다.

IV. 실험

멀티-코어 차량용 MCU에서 이기종 연산의 효율적 실행을 검증하기 위해 사용한 보드는 그림 7과 같다. Infineon사의 TC275 MCU를 실장하고 있는 Hitex사의 ShieldBuddy TC275 보드를 사용하였다 [18]. 타겟 보드가 실장하고 있는 TC275 MCU는 TriCore™ v1.6.1 코어를 내장하고 있으며 최대 200 MHz 속도로 동작한다. 또한 온-칩에 포함된 3개의 코어는 각 코어의 명령어 실행을 위해 4Mbyte의 플래시 메모리를 갖추고 있다. RAM 용량은 472 Kbyte이다. 보드의 아날로그 핀으로부터 ADC 샘플링의 대상이 되는 신호를 입력받도록 하였다. 제안하는 기법을 검증하기 위한 버퍼의 크기는 100개로 설정하였고, 이는 각 코어에서 모두 동일하다. 실험에서 사용될 DSP 연산은 윈도우 크기가 10인 이동 평균 필터로 선택하였다. DSP 필터 연산을 거쳐 통신 인터페이스로 전송된 ADC 샘플링 데이

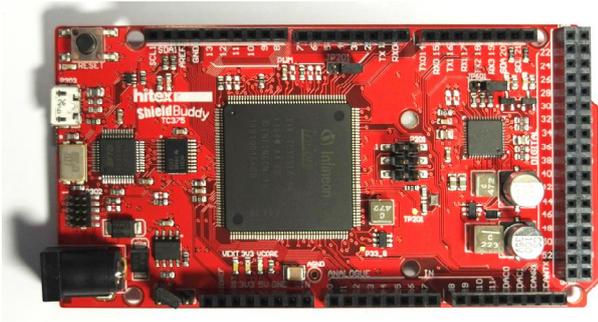


그림 7. 타겟 보드 (Hitex Shieldbuddy TC275)
Fig. 7. Target board (Hitex Shieldbuddy TC275)

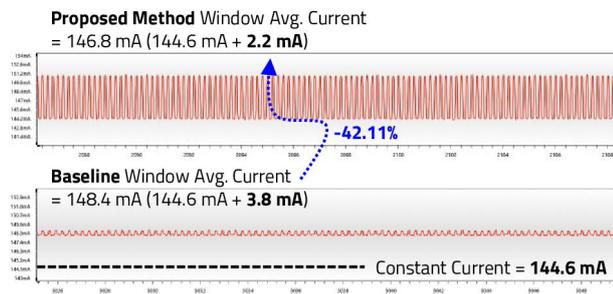


그림 8. 타겟 보드 전류 측정 시각화
Fig. 8. Target board current measurement visualization

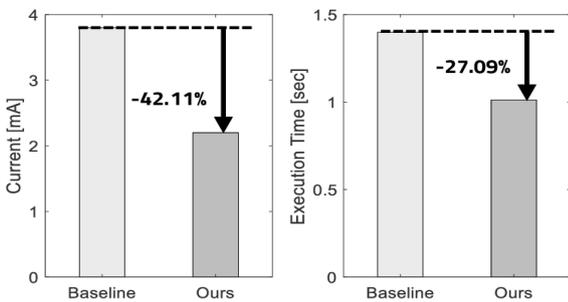


그림 9. 타겟 보드 전류 및 실행 시간 측정 결과
Fig. 9. Target board current and execution time measure result

터는 아날로그 신호에서 노이즈가 감소된 형태를 가진다. TC275 보드에서 호스트 PC로 데이터를 전송할 때는 UART 통신을 사용하도록 하였다.

멀티-코어 이기종 연산의 효율성을 검증하기 위해 비교될 baseline은 동종 실행 구조로 구현된 임베디드 소프트웨어이다. 비교 대상이 되는 baseline 소프트웨어는 ADC 샘플링 데이터가 가득 찬 버퍼에 대해 DSP 연산을 수행하고, DSP 연산이 완료된 버퍼에 대해 UART 인터페이스를 통한 전송을 수행하는 과정이 순차적으로 구현되어 있다. 제안하는 방법을 적용했을 때의 전력 소모량 변화를 비교하기 위해 TC275 보드의 전류를 측정하였다. 실험 과정에서의 전류 측정 구간이 연산 처리 중간에 위치할 경우, 전력 budget의 형태에 따라 잘못된 결과가 관측될 수 있다. 이를 방지하기

위해 전체 작업이 완료되어 충분한 반복이 이루어지기까지의 시간으로 설정하였다. 그림 8의 상단 그래프는 제안하는 기법을 적용한 경우, 하단 그래프는 baseline 경우의 보드에 흐르는 전류를 시각화한 결과이다. 10초 동안의 전류를 측정된 결과, 제안하는 기법을 적용한 경우는 146.8 mA, baseline은 148.4 mA의 평균 전류가 측정되었다. 이때 보드가 아무 동작도 하지 않는 경우 소모하는 상수 전류가 144.6 mA로 측정되어 이를 제외하면 평균 전류는 3.8 mA에서 2.2 mA로 42.11% 감소하였다. 또한, 실행 시간 개선을 검증하기 위해 두 가지 경우에서 500개 샘플을 출력하기까지의 소요 시간을 측정하였다. Baseline 경우는 1.399초가 소요됐지만, 제안하는 기법을 적용한 경우는 1.012초가 소요되어 27.09% 감소한 결과를 보였다. 이러한 결과는 그림 9에 나타내었다.

실험 결과로부터 멀티-코어 차량용 MCU에 임베디드 소프트웨어를 구현하는 경우, 이기종 실행 특성을 고려했을 때 더 효율적인 연산이 가능한 것을 확인할 수 있다. 성능 차이에 대한 원인을 분석하면 다음과 같다. 전력 그래프의 추이를 보면, baseline 경우의 전류는 TC275 보드가 아무 동작도 하지 않을 때의 상수 전류에 비해 큰 오프셋만큼 상승해있다. 반면, 제안하는 기법의 경우는 최대 전류는 baseline에 비해 다소 높지만, 태스크 사이에 idle한 구간에서는 보드의 상수 전류에 가까울 정도로 전류값이 낮아진다. 그 결과 멀티-코어 이기종 실행방식으로 임베디드 소프트웨어를 구현하면 평균적인 전류는 낮아질 수 있다.

V. 결론

본 논문에서는 차량용 MCU에 적재되는 임베디드 소프트웨어의 이기종 특징을 고려하여 멀티-코어의 협력적 연산 구조를 제안하였다. 또한, 멀티-코어가 내장된 차량용 MCU에서 사용되는 소프트웨어를 개발하는 일부 과정을 소개하였다. 차량용 소프트웨어는 다양한 하드웨어 구동을 위해 이기종 실행 흐름을 가지는 특성을 가진다. 이러한 특성에 대한 고려 없이 동종 실행 흐름으로 개발된 소프트웨어에서는 코어의 대기 시간이나 코어의 부하로 인해 비효율적인 오버헤드가 발생한다.

제안하는 방법은 멀티-코어를 내장한 차량용 MCU가 실장된 실제 보드에서 검증되었다. ADC 샘플링, DSP 연산, 통신 인터페이스 등 임베디드 시스템의 일반적인 태스크들이 포함된 소프트웨어가 동종 실행 흐름으로 구현된 경우, 제안하는 방법을 적용하여 멀티-코어 협력적 환경에 적합하게 이기종 실행 흐름으로 구현된 경우의 오버헤드 감소를 실증하였다. 제안하는 방법을 사용했을 때 이기종 태스크 사이에 발생하는 idle 상태를 활용하여 보드의 자원을 효율적으로 사용할 수 있었다. 실험 결과, baseline과 비교하여 보드의 소모 전력은 42.11% 감소하였다. 또한, 동일한 양의 샘플링 데이터를 처리하는데 소요된 시간은 27.09% 감소한 결과를 얻었다. 실험 결과로부터 제안된 멀티-코어 협력적

이기중 임베디드 소프트웨어 실행 방법의 효율성을 검증하였다. 향후 차량용 MCU에서 수행할 수 있는 다양한 어플리케이션을 대상으로 효율적인 연산을 가능케 하는 협력적 분산 처리 기법에 관해 연구를 진행할 예정이다.

References

- [1] S. Elashri, A. Azim, "An Energy-aware Optimization Model for Real-time Systems Analysis and Design: Work-in-progress," In Proceedings of the 2021 International Conference on Embedded Software (EMSOFT '21), pp. 45-46, 2021.
- [2] J. Chang, S. Oh, D. Park, "Accuracy-Area Efficient Online Fault Detection for Robust Neural Network Software-Embedded Microcontrollers," In Proceedings of the 2022 International Conference on Embedded Software (EMSOFT '22), 2022.
- [3] H. R. Faragardi, B. Lisper, K. Sandström, T. Nolte, "A Resource Efficient Framework to run Automotive Embedded Software on Multi-core ECUs," Journal of Systems and Software, Vol. 139, pp. 64-83, 2018.
- [4] M. Uelschen, M. Schaarschmidt, C. Fuhrmann, C. Westerkamp, "Work-in-Progress: PowerMonitor: Design Pattern for Modelling Energy-Aware Embedded Systems," 2019 International Conference on Embedded Software (EMSOFT), pp. 1-2, 2019.
- [5] S. H. Aldaajeh, S. Harous, S. Alrabaaee, "Fault-Detection Tactics for Optimized Embedded Systems Efficiency," IEEE Access, Vol. 9, pp. 91328-91340, 2021.
- [6] N. Navet, F. Simonot-Lion, Automotive Embedded Systems Handbook, CRC Press, 2017.
- [7] M. Ashjaei, L. L. Bello, M. Daneshtalab, G. Patti, S. Saponara, S. Mubeen, "Time-Sensitive Networking in Automotive Embedded Systems: State of the Art and Research Opportunities," Journal of Systems Architecture, Vol. 117, pp. 102137, 2021.
- [8] F. Salewski, S. Kowalewski, "Hardware/Software Design Considerations for Automotive Embedded Systems," in IEEE Transactions on Industrial Informatics, Vol. 4, No. 3, pp. 156-163, 2008.
- [9] G. L. Gopu, K. V. Kavitha, J. Joy, "Service Oriented Architecture based Connectivity of Automotive ECUs," 2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT), pp. 1-4, 2018.
- [10] C. Bradatsch, T. Ungerer, R. Zalman, A. Lajtkep, "Towards Runtime Testing in Automotive Embedded Systems," 2011 6th IEEE International Symposium on Industrial and Embedded Systems, pp. 55-58, 2011.
- [11] S. H. Lee, D. K. Lee, P. Choi, D. Park, "Efficient Power Reduction Technique of LiDAR Sensor for Controlling Detection Accuracy Based on Vehicle Speed," IEMEK J. Embed. Sys. Appl., Vol. 15, No. 5, pp. 215-225, 2020 (in Korean).
- [12] G. Xie, Y. Chen, Y. Liu, Y. Wei, R. Li, K. Li, "Resource Consumption Cost Minimization of Reliable Parallel Applications on Heterogeneous Embedded Systems," in IEEE Transactions on Industrial Informatics, Vol. 13, No. 4, pp. 1629-1640, 2017.
- [13] G. N. Khan, J. Levman, J. Alirezaie, "Hardware-Software Co-Synthesis of Heterogeneous Embedded Computer Systems," 2006 Canadian Conference on Electrical and Computer Engineering, pp. 1304-1307, 2006.
- [14] Y. H. Fan, J. O. Wu, S. F. Wang, "Software Synthesis of Middleware for Heterogeneous Embedded Systems," 2012 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet), pp. 2084-2087, 2012.
- [15] P. Gai, M. Violante, "Automotive Embedded Software Architecture in the Multi-core age," 2016 21th IEEE European Test Symposium (ETS), pp. 1-8, 2016.
- [16] E. Díaz, E. Mezzetti, L. Kosmidis, J. Abella, F. J. Cazorla, "Modelling Multicore Contention on the AURIX™ TC27x," 2018 55th ACM/ESDA/IEEE Design Automation Conference (DAC), pp. 1-6, 2018.
- [17] G. Xie, Y. Chen, Y. Liu, Y. Wei, R. Li, K. Li, "Resource Consumption Cost Minimization of Reliable Parallel Applications on Heterogeneous Embedded Systems," in IEEE Transactions on Industrial Informatics, Vol. 13, No. 4, pp. 1629-1640, 2017.
- [18] <https://www.infineon.com/cms/en/product/microcontroller/32-bit-tricore-microcontroller/32-bit-tricore-aurix-tc2xx/aurix-family-tc27xt/>
- [19] <https://www.hitex.com/microcontroller-support/aurix/shieldbuddy-tc275>

Jisu Kwon (권 지 수)

2019 Electronics Engineering from
Kyungpook National University
(B.S.)

2019~Electronic and Electrical Engineering
from Kyungpook National
University (Integrated Ph.D.
Student)

Field of Interests: machine learning, embedded system,
microcontroller

Email: kjisu96@knu.ac.kr

Daejin Park (박 대 진)

2001 School of Electronics Engineering
from Kyungpook National
University (B.S.)

2003 School of Electronics Engineering
from KAIST (M.S.)

2003~2014 Research Engineer, SK Hynix/
Samsung

2014 School of Electronics Engineering from KAIST (Ph.D.)

2016~School of Electronics Engineering from Kyungpook
National University (Associate Professor)

Field of Interests: Low-power SoC Design, Robust Embedded
Systems

Email: boltanut@knu.ac.kr

Guideline for IEMEK Manuscript

I. Guideline for Research Ethics

Research misconduct (hereinafter referred to as the “Misconduct”) should be never acceptable and refers to forgery, falsification, plagiarism and unfair acts of displaying theses and their authors executed in research proposals, conduct, results reporting and announcements as follows:

- ① Counterfeiting, which is the act of creating data or research results that do not exist
- ② Falsification, which is the act of distorting research information or results by artificially manipulating research materials, equipment, process, or by randomly transforming or deleting data
- ③ Plagiarism, which is the act of fraudulent use of other people’s ideas, research content and results without appropriate approval, citation or quotation.
- ④ Duplicate submission/publication, which is the academic act of showing the same or substantially similar academic work as the researcher’s own previous findings in another journal or literary work, without permission of the copyright holders or journal editors who first published it, or without displaying appropriate sources.
- ⑤ Display of unjustified paper authors, which is the act of granting paper authorship to persons without justifiable reason, who made no scientific and technical contribution in content or results of research, or granting paper authorship as a gesture of gratitude or courtesy to persons who did not make any scientific or technical contribution.
- ⑥ The act of willfully interfering with investigating allegations of one’s own or someone else’s Misconduct and harming the informant.
- ⑦ The act of seriously deviating from the range of science and technology that is commonly accepted
- ⑧ In addition, such Misconduct may also be included that is deemed necessary for its own investigation or prevention.

Upon receipt of research ethics issues in relation to the academic activities of the Society, the President of the Society shall inform the Research Ethics Committee of the matter for their investigation. The Committee shall investigate the Misconduct and submit the investigation report of research misconduct within three months in principle.

The Research Ethics Committee shall be responsible for verifying whether the Misconduct is true or not. If no research Misconduct is confirmed, the Society shall conduct appropriate follow-up action in order to restore the honor and reputation of related researchers. On the contrary, if the research Misconduct is confirmed, this result shall be published and each of the following actions shall be taken in accordance with the gravity of cases that shall be determined by a resolution of the Board of Directors:

- ① Demand for cancellation or modification of research results
- ② Cancellation of publication of research results
(Official written notice for cancellation shall be sent to relevant agencies.)
- ③ Disqualification of membership for a reasonable period
- ④ Removal of the name of relevant persons
- ⑤ Names of persons involved shall be provided to law enforcement authorities

The provisions not specified in this Regulation shall be in accordance with the resolution by the Board of Directors or the relevant laws and regulations. For all the other policies not stated in the IEMEK research ethics regulation, the International Standards for Editors and Authors ([http://publicationethics.org/international-standards -editors-and-authors](http://publicationethics.org/international-standards-editors-and-authors)) should be applied.

II. Aims and Scope

The aim of the IEMEK J. Embed. Sys. Appl. is to publish peer-reviewed research in the rapidly developing field of embedded technology and its industrial applications. The journal covers the original academic achievements related to embedded technology and the latest industrial outstanding developments in the following field but not limited to:

- Embedded Hardware Technologies: SoC, MCU, multi-core, low power system design, hardware architecture, cyber-physical systems.
- Embedded Software Technologies: Embedded OS, middleware, networking, development and verification tools, real-time computing, software architecture.
- Embedded Convergence Technologies: Intelligent vehicles, intelligent robots, healthcare and medical equipments, brain sciences.

III. Peer Review

Peer review is the critical assessment of manuscripts submitted to IEMEK journals by experts who are not part of the editorial staff. Every received manuscript is circulated to three peer reviewers. The author's name and affiliation should not be disclosed during review process to reviewers. The review process can be repeated till three times if the request of revision is suggested by reviewers. If two reviewers do not agree to accept the journal, it also may not be considered for publication. Peer review helps the editor's decision which manuscripts are suitable for their journals and helps authors and editors to improve the quality of publication. The final camera-ready manuscript will be reviewed by the manuscript editor for the consistency of the format and the completeness of references. The manuscript may be revised according to the opinion of the manuscript editor.

IV. Publication Type

Authors can submit manuscripts by different types of manuscript such as review paper, invited paper, regular paper, technical notes, and comment paper.

V. References

- Examples :
 - ① T.H. Yoon, Y.S. Song, "Multi-layer surveillance system based on wireless mesh networks," IEMEK J. Embed. Sys. Appl., Vol. 7, No. 5, pp.209-217, 2012(in Korean).
 - ② G.D. Hong, and J. Park, "Novel embedded OS for soccer robot system", Journal on Embedded Systems, Vol. 1, No. 1, pp.1-12, 2004.
 - ③ <http://kidbs.itfind.or.kr/WZIN/>

VI. Fee for Submission and Publication Charge

- Regular/Rapid manuscript is charged for the review processing fee of Korean Won Free/200,000.
- If the manuscript is accepted for publication, the publication processing fee of Korean Won 250,000 is charged, in addition to the extra page charge of Korean Won 30,000 per page in case of having more than 7 pages. Furthermore, if there is the acknowledgement of the financial support for research grants, the extra fee of Korean Won 100,000 is charged.

VII. Publication Interval

It is published 6 times in a year, The last day of say every even month, February, April, June, August, October, and December.

VIII. Copyright

The copyright and the transfer right of the online/offline published paper and journal is owned by the IEMEK.

IX. Contact Information on Inquiry

- Editorial Office of IEMEK, 12-6, Sampungan-gil, Gyeongsan-si, Gyeongsangbuk-do, 38544, KOREA
- Tel: +82-53-801-1475, Fax: +82-53-801-1476, Email: iemek@iemek.org

Checklist for Original Article

Title of the manuscript: _____

Please check below items as ✓ mark before submission of the manuscript.

-
1. Manuscript was written in good Korean or English.
-
2. Title, name of authors, affiliation, captions of figures and tables were described in Korean and in English.
-
3. Abstract and Keywords should be written in English.
-
4. The first letter of an English sentence was always capitalized. It is an exception in the case of the letters of proper nouns.
-
5. English name was written as the given name and the family name in order.
-
6. From cover page, title (English and Korean), name of authors (English and Korean), abstract (English: less than 300 letters), keywords (English: less than 5 words), References (English), main text, references, appendix were described in order.
-
7. The number of section and sub section was described as (I, II, III,...), (1, 2, 3, ...), respectively.
-
8. Abbreviation of the Figure and Table were described as Fig. 1. and Table 1. The title of a figure and a table was placed below the figure and above the table, respectively. Captions of figures and tables were described in Korean and in English.
-
9. Figure with sufficient resolution was placed in the main text so as to avoid interrupting the flow of text.
-
10. Every articles in References were cited in the main text and were numbered according to the following numeric order as [] type.
[1] G.D. Hong, and J. Park, "Novel embedded OS for soccer robot system", Journal on Embedded Systems, Vol. 1, No. 1, pp.1–12, 2004.
[2] T.H. Yoon, Y.S. Song, "Multi-layer surveillance system based on wireless mesh networks," IEMEK J. Embed. Sys. Appl., Vol. 7, No. 5, pp.209–217, 2012(in Korean).
[3] B. Boyd, L.E. Ghaoui, E. Feron, and V. Balakrishnan, Linear Matrix Inequalities in Systems and Control Theory, SIAM, Philadelphia, 1994.
-
11. According to above items 1 ~ 10, manuscript was prepared with proper font size for review. The document format for review process is accepted only in PDF. Acknowledgement is not appropriate in the manuscript for review. The accepted paper should be written by following the format of the final version (which is available in IEMEK homepage) in HWP files.
-

Copyright Transfer Agreement Form

Fill in the form and return to:

IEMEK Editorial Office
12-6, Sampungan-gil
Gyeongsan-si, Gyeongsangbuk-do, 38544, KOREA.
Phone: +82-53-801-1475
Email: iemek@iemek.org
[Http://iemek.org](http://iemek.org)

Corresponding author's name	
Article title	
Email address	
Affiliation	
Co-authors' names	

Copyright Transfer Statement

The copyright to this article is transferred to Institute of Embedded Engineering of KOREA (IEMEK) if and when the article is accepted for publication. The undersigned hereby transfers any and all rights in and to the article including without limitation all copyrights to IEMEK. The undersigned hereby represents and warrants that the paper is original and that he/she is the author of the paper, except for material that is clearly identified as to its original source, with permission notices from the copyright owners where required. The undersigned represents that he/she has the power and authority to make and execute this assignment.

I declare on behalf of myself and my co-authors (if any), that:

1. This article I have submitted to IEMEK J. Embed. Sys. Appl. is original and has not been published in the same form elsewhere.
2. It will not be submitted anywhere else for publication prior to acceptance/rejection by IEMEK
3. This paper does not contain any materials that violate any personal or proprietary rights.
4. A copyright permission is obtained for materials published elsewhere and which require this permission for reproduction.

Copyright to the above work (including without limitation, the right to publish the work in whole, or in part, in any and all forms) is hereby transferred to IEMEK, to ensure widest dissemination and protection against infringement. The copyright transfer covers the exclusive right to reproduce and distribute the article, including reprints, translations, photographic reproductions, microform, electronic form (mobile, offline, online) or any other reproductions of similar nature. No proprietary right other than copyright is proclaimed by IEMEK.

This agreement is to be signed by at least one of the authors who has obtained the assent of the co-author(s) where applicable. After submission of this agreement signed by the corresponding author, changes of authorship or in the order of the authors listed will not be accepted.

I hereby certify that I am authorized to sign this document either in my own right or as an agent of my employer, and have made no changes to the current valid document supplied by IEMEK.

Corresponding Author's Full Name and Signature

Name (print): _____

Date: _____

Original Signature (in ink): _____

(사)대한임베디드공학회 2023년 4월 특집호 논문 모집 안내

- 디지털 헬스케어 및 의료 인공지능 임베디드 소프트웨어 분야 -

(사)대한임베디드공학회에서는 최근 많은 관심을 끌고 있는 임베디드 시스템 연구의 일환으로 디지털 헬스케어 및 의료 인공지능 임베디드 소프트웨어 분야의 주요 기술 특집호(2023년 4월호)를 발간하고자 합니다. 관련 논문 모집 분야는 아래를 참조하시고, 아래의 분야가 아니더라도 임베디드 시스템과 연관되는 주요 기술 내용(하드웨어/소프트웨어, 자율주행, 인공지능, 영상처리 및 IoT 분야)이면 투고가 가능하오니 학회 회원 연구자분들의 많은 관심과 투고를 부탁드립니다.

※ 논문 모집 분야

Medical Information Systems/Medical Embedded Systems
Hospital of the Future Telemedicine
AI-based Health Monitoring System Architecture
Sensors and Robots in Next Generation Health Monitoring CPSs
Machine Learning and AI Algorithms in Health Monitoring
New Low-power Platforms and Sensors for Health Monitoring
Embedded Systems in Healthcare
Wired and Wireless Communication Technologies in AI-based Health Monitoring
Signal Processing and Fusion for Complex AI/ML-based Health Monitoring Environments
Security and Privacy for Next Generation AI/ML-based Health Monitoring

※ 주요 일정

- 2023년 1월 31일: 논문 투고 마감일
- 2023년 2월 28일: 논문 심사 결과 최종 통보
- 2023년 3월 10일: 논문 최종 본 제출 마감일

※ 담당 Guest Editor : 고영웅 교수 (한림대학교)

Co-guest editor: 장성봉 교수 (금오공과대학교)

※ 논문 투고 방법

- 학회 홈페이지: <http://www.iemek.org>
- 논문투고절차: 학회회원 가입 후, 학회 홈페이지 온라인 논문심사 메뉴를 통해 제출
- 주의사항
 - 투고용 논문 제출 시 원고형식 특집논문 **“2023년 4월 특집호” 체크**
 - 특집호 논문 편 수 및 편집 일정에 따라 다음 호에 분리되어 출판될 수 있음.
- 혜택
 - 투고료 : 면제
 - 게재료 : 게재료 총 금액에서 50% 감면
(7면 이하 25만원, 8면 이상 면당 3만원 추가, 사사문구 편당 10만원 추가별도)

※ 문의처 : 학회 사무국

- 주소: 경북 경산시 삼풍안길 12-6, 1층 대한임베디드공학회
- 전화: 053) 801-1475, 팩스: 053) 801-1476, 홈페이지: <http://www.iemek.org>

Date of the First Issue : 2006. 06. 30.

Printing Date 2022.12. 15.
Publication Date 2022. 12. 31.

Aims and Scope The aim of the IEMEK J. Embed. Sys. Appl. is to publish peer-reviewed research in the rapidly developing field of embedded technology and its industrial applications. The journal aims to cover the introduction of the academically related embedded original technology and the industrial latest outstanding developments in the field of embedded hardware, software and its application technologies but not limited to the related topics.

Publication Period It is published 6 times in a year.
(The last day of every even month, Feb, Apr, Jun, Aug, Oct, Dec.)

Publisher Yong Wan Park, Ph. D.
Institute of Embedded Engineering of Korea
12-6, Sampungan-gil, Gyeongsan-si
Gyeongsangbuk-do, 38544, KOREA.
Tel: +82-53-801-1475, Fax: +82-53-801-1476

Editor Ju Hyun Park, Ph. D.

ISSN 1975-5066

Printer Jeongmun Design [Tel:+82-53-817-1996]

Journal Website <http://journal.iemek.org/>

■ Contact Information on Inquiry :

Editorial Office of IEMEK, 12-6, Sampungan-gil, Gyeongsan-si, Gyeongsangbuk-do, 38544, KOREA
Tel: +82-53-801-1475, Fax: +82-53-801-1476, Email: iemek@iemek.org

■ Subscription : Subscription rates is 10,000 Korean Won.(Contact to IEMEK office for further information.)

■ Copyright © 2020 Institute of Embedded Engineering of Korea <http://www.iemek.org>

<p>This journal was supported by the Korean Federation of Science and Technology Societies Grant funded by the Korean Government (MEST)</p>

IEMEK Journal of Embedded Systems and Applications

Papers

Special Issue : System Optimizations and Applications of Artificial Intelligence–based Image and Signal Processing

- | | |
|------------|--|
| 309 | 37. Dual Branched Copy–Move Forgery Detection Network Using Rotation Invariant Energy in Wavelet Domain
Jun Young Park, Sang In Lee, Il Kyu Eom |
| 319 | 38. A Survey on Vision Transformers for Object Detection Task
Jungmin Ha, Hyunjong Lee, Jungmin Eom, Jaekoo Lee |
| 329 | 39. A Study on the Video Inpainting Performance using Denoising Technique
Jeong–yun Seo, Han–gyul Baek, Sang–hyo Park |
| 337 | 40. Multi–Vision–based Inspection of Mask Ear Loops Attachment in Mask Production Lines
JiMyeong Woo, SangHyeon Lee, Heoncheol Lee |
| 347 | 41. A Study on the Alternative Method of Video Characteristics Using Captioning in Text–Video Retrieval Model
Dong–hun Lee, Chan Hur, Hyeyoung Park, Sang–hyo Park |
| 355 | 42. Parallelized Particle Swarm Optimization with GPU for Real–Time Ballistic Target Tracking
Yunho Han, Heoncheol Lee, Hyeokhoon Gwon, Wonseok Choi, and Bora Jeong |
| 367 | 43. Optimizing 2–stage Tiling–based Matrix Multiplication in FPGA–based Neural Network Accelerator
Jinse Kwon, Jemin Lee, Yongin Kwon, Jeman Park, Misun Yu, Taeho Kim, Hyungshin Kim |
| 375 | 44. Collaborative Streamlined On–Chip Software Architecture on Heterogenous Multi–Cores for Low–Power Reactive Control in Automotive Embedded Processors
Jisu Kwon, Daejin Park |