

모바일을 위한 JavaCv를 이용한 Tensorflow모델 구동환경 개발

박진상^o, 오상권*, 이성진*

^o경상대학교 항공우주및소프트웨어공학전공,

*경상대학교 항공우주및소프트웨어공학전공

e-mail: {do_the_g, ooh7451541}@naver.com, insight@gnu.ac.kr

Tensorflow Model Environment with JavaCv for Mobile Devices

JinSang Park^o, SangGwon Oh*, SeongJin Lee*

^oDepartment of Aerospace and Software Engineering, Gyeongsang National University,

*Department of Aerospace and Software Engineering, Gyeongsang National University

● 요약 ●

현재 PC환경 뿐만 아니라 모바일 환경, 임베디드 환경에서 딥러닝 모델을 구동하기 위한 많은 연구들이 진행 중에 있다. 본 연구에서는 완성된 딥러닝 모델을 구동하는 환경을 Java로 구현하여 개발 접근성을 높이 고자 한다. 이미지, 영상처리를 위해 OpenCV를 사용시 C++ API문서는 보편화되어있는 반면에 JavaCv API 문서는 그렇지 못하다. 그러나 모바일 개발 환경 특성상 Java언어로 작업한 코드를 안드로이드 스튜디오 오에서 작업 시 그대로 가져올 수 있어 개발이 용이하다. 모델 구동을 위한 전반적인 이미지 처리 및 작업환경을 개발하였다.

키워드: JavaCv, 정규화(normalization), 복호화(decryption)

I. Introduction

최근 많은 연구에서 Tensorflow[1]와 Python을 사용하여 다양한 딥러닝 모델을 개발하고 있다. 개발된 모델들을 PC뿐만 아니라 모바일 환경에서 사용할 수 있도록 최적화하는 다양한 연구들이 진행 중에 있다. 딥러닝 모델을 모바일 환경에서 사용할 수 있도록 개발하는 방법에는 여러 가지 방법이 있는데, 그 중 모델의 크기를 최소화할 수 있도록 구조 및 변수들을 설계하는 방법과 개발된 모델을 Tensorflow Lite[2]나 양자화 기법을 활용하여 크기를 줄이는 방법이 대표적이다.

초기에 모델을 경량화 하도록 설계하는 방법은 모델개발에 있어 제한적이고 구현 난이도가 어려우며, Tensorflow Lite를 사용하여 양자화하는 기법의 경우 이미 생성된 모델의 구조에 따라 제공되는

양자화 기능이 호환되지 않아 경량화에 실패하는 경우가 있다. 그로 인해 본 연구에서는 이미 python언어로 생성된 모델을 모바일 환경에서 구동시키기 위해 Tensorflow Java API와 JavaCv API를 활용하여 이미지를 전 후처리하고, 모델을 불러와 동작시킬 수 있는 프로그램을 만들었다. 모델에 사용할 이미지 파일을 내부적으로 전처리하여 RGB값으로 이루어진 pixel 배열을 -1~1사이 값으로 조정된 float형식의 배열로 만들어 사용한다. 이렇게 전처리된 데이터를 모델에서 사용하고, 처리된 결과를 다시 복호화하여 RGB값을 가진 이미지 파일로 저장한다.

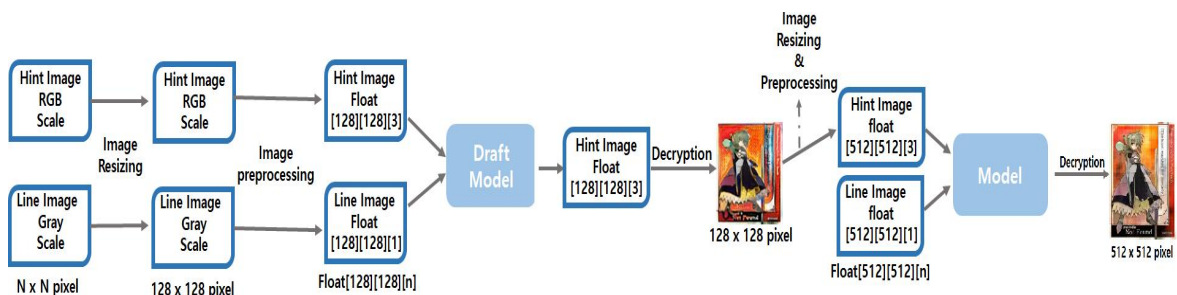


Fig. 1. 입력 이미지의 변환 과정

II. Preliminaries

1. Related works

1.1 Tensorflow Lite

Tensorflow Lite는 Tensorflow의 모바일과 임베디드 디바이스용 경량급 솔루션으로, 낮은 지연시간과 작은 바이너리 크기로 임베디드 디바이스에서 머신러닝 추론이 가능하다.

1.2 API

OpenCv는 방대한 문서, 여러 예제들이 존재한다는 장점이 있고, JavaCv는 사용률이 낮고 문서화가 부족하여 사용이 어렵지만 모바일 환경 작업에 호환성이 높다는 장점이 있다.

III. The Proposed Scheme

본 논문에서는 로드된 이미지가 Tensorflow모델에서 동작 하도록 크기수정과 정규화를 하며 float형으로 변환한다. 이후 출력을 위해 float형 데이터를 복호화하고 이미지로 변환한다.

1. Load

특정 파일에 저장되어있는 라인 이미지와 힌트 이미지를 JavaCv의 cvLoadImage()함수를 통해 로드한다. 이때 어떠한 크기의 이미지가 로드 되더라도 이미지 크기가 Tensorflow 모델의 입력으로 알맞도록, 이미지의 크기를 변환하는 resizedImg() 함수를 사용하여 조정한다. 이때 로드된 이미지들의 Type은 Iplimage형인데, 이를 Tensorflow 모델에 입력하기 위해서 float형 배열로의 변환과 정규화를 한다.

2. Conversion & Normalization

Iplimage는 3차원의 형태로 x축과 y축은 이미지의 좌우, 상하를 나타내며 z축은 각 픽셀의 RGB값을 의미한다. RGB값은 각각 0~255로 이뤄져 있으며, 변환 시 각 픽셀의 RGB값을 -1~1사이로 정규화(이미지 전처리) 한다. 이때 cvScalar 구조체의 val(0)부터 val(2)는 B,G,R을 의미 한다는 것에 주의해야 한다. 또한 Draft모델은 다른 모델과는 힌트이미지의 입력구조가 다르기 때문에, 힌트 이미지의 흰색부분을 2로 초기화 하는 과정이 필요하다.

3. Conversion & Decryption

이후 정규화한 float형 배열 이미지를 Tensorflow 모델에 입력하고 그 결과 값을 다시 resizedImg()함수에서 사용하기 위해 Iplimage형으로 변환(복호화)시킨다. 이때 -1~1 사이의 값을 가지고 있는 float형 배열의 각 값을 0~255사이의 값을 갖도록 복호화한 뒤, cvSet2D() 함수를 이용해 Iplimage형으로 변환시킨다.

IV. Conclusions

모바일 환경에서 이미지생성 모델을 구동하기 위해 JavaCv와 TensorFlow JAVA API를 사용해 모바일 이식작업을 진행하였다. 개발 결과 양자화나 TF-lite같은 경량화 기법과 비교해 실행속도가 느리고 용량이 크다는 단점이 있지만, 모바일 환경에 호환성이 높으며 기존 구현된 모델을 손쉽게 모바일에서 사용할 수 있는 장점이 있다.

Table. 1. 경량화 기법간 비교

기법	실행속도	모델용량	사용성	호환성
TensorFlow-lite	빠름	중간	낮음	낮음
양자화	빠름	낮음	높음	낮음
TensorFlow-CPP	중간	큼	낮음	높음
TensorFlow-JAVA	느림	큼	높음	높음

ACKNOWLEDGEMENT

이 논문은 2018년도 정부(교육부)의 재원으로 한국과학창의재단(2019년도 학부생 연구프로그램)의 지원을 받아 수행된 연구이며 또한, 이 성과는 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. 2019R1G1A1100455).

REFERENCES

- [1] Yeojin Chung, SungMahn Ahn, Jiheon Yang and Jaejoon Lee "Comparison of Deep Learning Frameworks: About Theano, Tensorflow, and Cognitive Toolkit" In Korea Intelligent Information System Society, pp. 6-7. 2017.
- [2] "TensorFlow Lite" TensorFlow, accessed DEC 27, 2019, <https://www.tensorflow.org/lite/>