KIPS Trans. Softw. and Data Eng.
Vol.11, No.1 pp.29~34
ISSN: 2287-5905 (Print), ISSN: 2734-0503 (Online)

https://doi.org/10.3745/KTSDE.2022.11.1.29

Single Shot Detector for Detecting Clickable Object in Mobile Device Screen

Min-Seok Jo† · Hye-won Chun† · Seong-Soo Han†† · Chang-Sung Jeong†††

ABSTRACT

We propose a novel network architecture and build dataset for recognizing clickable objects on mobile device screens. The data was collected based on clickable objects on the mobile device screen that have numerous resolution, and a total of 24,937 annotation data were subdivided into seven categories: text, edit text, image, button, region, status bar, and navigation bar. We use the Deconvolution Single Shot Detector as a baseline, the backbone network with Squeeze-and-Excitation blocks, the Single Shot Detector layer structure to derive inference results and the Feature pyramid networks structure. Also we efficiently extract features by changing the input resolution of the existing 1:1 ratio of the network to a 1:2 ratio similar to the mobile device screen. As a result of experimenting with the dataset we have built, the mean average precision was improved by up to 101% compared to baseline.

Keywords : Test Automation, Android Object Detection, Mobile Screen Detection, Computer Vision, Deep Learning

모바일 디바이스 화면의 클릭 가능한 객체 탐지를 위한 싱글 샷 디텍터

조 민 석† · 전 혜 원† · 한 성 수†† · 정 창 성†††

요 약

모바일 디바이스 화면상의 클릭 가능한 객체를 인지하기 위한 데이터셋을 구축하고 새로운 네트워크 구조를 제안한다. 모바일 디바이스 화면에서 클릭 가능한 객체를 기준으로 다양한 해상도를 가진 디바이스에서 여러 애플리케이션을 대상으로 데이터를 수집하였다. 총 24,937개의 annotation data를 text, edit text, image, button, region, status bar, navigation bar의 7개 카테고리로 세분화하였다. 해당 데이터셋을 학습하기 위한 모델 구조는 Deconvolution Single Shot Detector를 베이스라인으로, backbone network는 기존 ResNet에 Squeeze-and-Excitation block을 추가한 Squeeze-and-Excitation networks를 사용하고, Single shot detector layers와 Deconvolution module을 Feature pyramid networks 형태로 쌓아 올려 header와 연결한다. 또한, 기존 input resolution의 1:1 비율에서 오는 특징의 손실을 최소화하기 위해 모바일 디바이스 화면과 유사한 1:2 비율로 변경하였다. 해당 모델을 구축한 데이터셋에 대하여 실험한 결과 베이스라인에 대비하여 mean average precision이 최대 101% 개선되었다.

키워드 : 테스트 자동화, 안드로이드 객체 탐지, 모바일 화면 인지, 컴퓨터 비전, 딥러닝

1. 서 론

기술 발전 속도가 증가함에 따라 새로운 제품의 개발 및 출시 기간 또한 짧아졌다. 이에 따라 제품을 빠르지만 정확하게 테스트하는 기술이 시장에서의 경쟁력을 나타내고 있다. 더불어 제품의 종류와 수가 증가하여 여러 종류의 제품을 추가 구현 없이 테스트할 필요로 이어진다. 일반적으로 모바일 기반의 제품을 테스트할 때는 에이전트를 각 디바이스에 설치하여 테스트를 진행한다[1-3]. 따라서, 같은 애플리케이션일지라도 디바이스 마다 다른 test-script를 작성하여 개별 테스트를 진행해야 한다. 이는 디바이스 의존도가 높다. 해당 문제는 디바이스에 에이전트 설치 없이 device-independent한 테스트를 진행하는 필요로 이어진다. 해당 문제를 해결하기 위해서는 영상을 사용하여 테스트를 진행하는 인공지능 기반의 test-case를 찾고 필요한 정보들을 분류하는 특정 분야에 관한 기술이 필요하다. 본 논문에서는 모바일 애플리케이션을 각각 다른 해상도의 디바이스에서 device-independent한 테스트를 진행하기 위한 원천 기술을 제안한다. 모바일 디바이스 화면 상에서 클릭 가능한 객체에 대한 데이터셋을 구축하고 해당 객체를 탐지하기 위한 객체 탐지 모델을 설계한다. 해당 모델은 모바일 화면 이미지를 대상으로 설계되었다. 기준 모델의 input resolution 비율인 1:1을 모바일 화면 이미지

※ 이 논문은 2021년도 4단계 BK21 사업에 의하여 지원되었음.
※ 본 논문은 과학기술정보통신부 및 정보통신산업진흥원의 '고성능 컴퓨팅 지원' 사업으로부터 지원받아 수행하였음.
※ 이 논문은 2021년 한국정보처리학회 춘계학술발표대회의 우수논문으로 "CMDNet: 클릭 가능한 모바일 화면 객체 탐지를 위한 싱글 샷 아키텍처"의 제목으로 발표된 논문을 확장한 것임.
† 준 회 원 : 고려대학교 전기전자공학과 석사과정
†† 종신회원 : 강원대학교 자유전공학부 조교수
††† 정 회 원 : 고려대학교 전자공학과 교수
Manuscript Received : June 28, 2021
First Revision : August 11, 2021
Accepted : August 11, 2021
* Corresponding Author : Chang-Sung Jeong(csjeong@korea.ac.kr)

비율에 맞추어 가로, 세로 1:2 비율로 변경하여 이미지 전처리 과정에서 발생하는 특징의 손실과 변이를 최소화하였다. 모바일 화면 이미지의 경우 Video Graphics Array(VGA)와 같은 저해상도 이미지부터 Wide Quad High Definition (WQHD)과 Ultra High Definition(UHD)같은 고해상도 이미지까지 모두 존재하기 때문에, 넓은 범위의 해상도에서도 객체 탐지가 가능하도록 설계하였다. 넓은 범위의 해상도에서 객체 탐지가 가능한 모델을 만들기 위해 해상도, 시간, 정확도 면에서 trade-off가 필요하며 효율적인 trade-off 관계를 도출하였다. 본 논문에서 구축한 모바일 화면 이미지 데이터셋에 대해 설계한 모델로 실험한 결과 86.5 Mean Average Precision(mAP) 및 10.2 Frame Per Second(FPS)를 달성하였다.

2. 연구 배경

2.1 2-Stage Model

객체 탐지 모델은 주어진 이미지에 대하여 localization과 classification을 수행하는 딥러닝 모델이다. 객체 탐지 모델은 크게 2-stage model과 1-stage model로 나눌 수 있다. 2-stage model은 localization과 classification을 따로 차례로 수행하는 반면, 1-stage model은 localization과 classification을 같이 수행하여 결과를 도출해낸다. 2-stage model은 대표적으로 R-CNN[4]이 있다. R-CNN은 입력 이미지로부터 Selective search[5]를 통해 물체가 있을 법한 2000개의 후보 영역을 추출한다. 추출한 각 후보 영역들을 이미지로부터 잘라내고(crop) 같은 크기로 변경(warp)한다. 같은 크기로 변경한 모든 후보 영역을 classification을 수행하는 Convolutional Neural Network(CNN)에 차례로 통과시켜 각 후보 영역에 관한 특징을 추출한다. 추출한 특징을 Support Vector Machine(SVM)을 통해 classification을 진행하여 해당 영역이 어떤 물체인지 찾는다. 또한, 추출한 특징은 bounding box regression을 통해 물체의 정확한 위치를 찾는다. 하지만 R-CNN의 경우 Selective search를 통해 얻은 2000개의 후보 영역을 각각 CNN에 통과시켜야 하므로 수행 시간이 오래 걸리고 localization과 classification 과정 사이에 특징을 공유하지 못하여 End-to-End 학습이 불가능하다는 문제가 있다. Fast R-CNN[6]에서는 Region of Interest Pooling(RoI pooling)을 통해 해당 문제를 해결하였다. RoI pooling을 통해 모든 후보 영역에 대해 CNN을 통과하는 문제를 해결하였으며 후보 영역의 특징을 투영 시킴으로써 모든 과정을 한 번에 학습할 수 있게 되었다. Faster R-CNN[7]에서는 Fast R-CNN에서 가장 느린 부분인 후보 영역을 찾는 Selective search를 딥러닝 모델인 Region Proposal Network(RPN)으로 대체하여 더욱 빠른 모델을 구성하였다.

2.2 1-Stage Model

1-stage model은 localization과 classification을 한 번에 수행한다. 이미지 입력부터 결과까지 하나의 모델을 통해 한 번에 도출하는 특성상 일반적으로 2-stage model보다 연산 속도가 빠른 경향이 있다. 대표적으로 Single Shot Detector (SSD)[8]가 있다. SSD는 300 x 300의 input resolution을 가지며 Visual Geometry Group(VGG)[9]을 backbone network로 사용한다. 입력된 이미지를 VGG-16을 통해 특징을 추출한다. 추출한 특징을 여러 번의 Convolution layer를 통과하며 6개의 다중 특징 맵을 생성한다. 6개의 특징 맵은 Convolution layer를 통과하며 그 크기가 작아지고 모든 특징마다 정해진 서로 다른 크기와 비율의 Bounding box를 대입하여 최종 결과를 도출해낸다. SSD는 Yolo[10]와 비교하여 추론 시간은 빨라지고 mAP는 향상시켰다. 하지만 목표를 달성하는 과정에서 작은 물체에 대한 성능은 다소 뒤처진 경향을 보였다. 해당 문제를 해결하기 위해 설계된 모델이 Deconvolution Single Shot Detector(DSSD)[11]이다. 기존 SSD의 특징 추출에 사용되었던 backbone network인 VGG를 ResNet[12]기반의 Residual-101로 변경하여 모델의 추론 시간이 빨라졌다. 또한, 작은 객체들에 대한 객체 탐지 정확도를 높이면서 추론 시간이 많이 증가하지 않는 Deconvolution module을 설계 및 추가하였다.

3. 모델 구조

3.1 Backbone Network

본 논문에서 제안하는 모델은 DSSD를 베이스라인으로 한다. DSSD의 경우 backbone network로 ResNet을 사용한다. 모델의 정확도를 높이기 위해서 본 논문에서 제안하는 모델은 backbone network를 기존 ResNet 구조에서 Squeeze-and-Excitation block을 추가한 Squeeze-and-Excitation networks(SENet)[13]으로 변경하였다. SENet은 DSSD에서 backbone network로 사용한 ResNet에 비하여 특징 추출 과정에서 추가 연산량이 거의 없이 양질의 특징을 추출하였다. ResNet의 module을 개선한 SE-ResNet module은 Fig. 1과 같다. Residual block 통과 전·후의 특징을 더하는 ResNet module과 달리 SE-ResNet module은 Residual block, Global pooling, Fully Connected layer(FC), Rectified Linear Unit(ReLU), FC, Sigmoid를 차례로 거치고 Residual block 통과 후의 특징 크기와 같도록 scale 조정 후, SE-ResNet module 입력 전 특징과 다시 합치는 과정을 거친다.

3.2 Deconvolution Module

Deconvolution module에서는 비대칭의 Hourglass[14] 구조를 사용하여, 이미지 및 특징의 크기에 상관없는 high-context 정보를 효율적으로 추출 및 활용하였다. 비대칭의 구조를 선택함으로써, 대칭 구조를 가졌을 때의 단점인 추론 시간이 두 배로 늘어나는 것을 방지하였다. 또한, Deconvolution module에서 두 개의 인풋을 합치는 연산을 Bilinear pooling method[15]가 아닌 Element-wise product 연산을 사용하여 추론 시간을 줄였다.

3.3 Prediction Module

Prediction module은 기존 Residual block에서 한 번의 Convolution layer를 추가하였다. 또한, Skip connection을 통

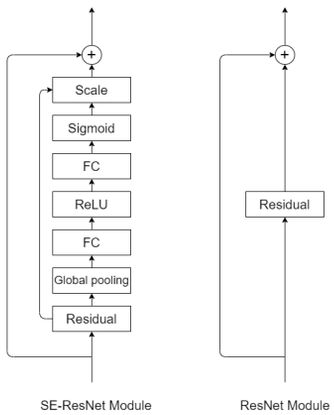

Fig. 1. SE-ResNet Module

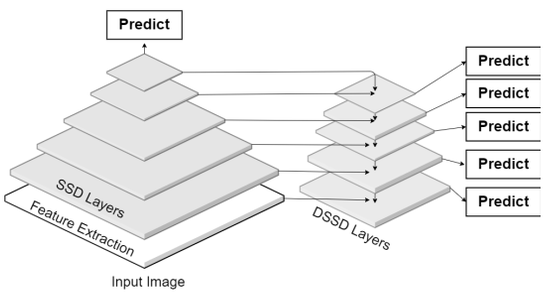

Fig. 2. Overall Architecture

해 바로 더하지 않고, module 입력 전 특징에 Convolution layer를 한 번 수행한 후 더함으로써, 기존 Residual block 보다 향상된 성능의 module을 설계하였다. 추론 중에는 배치 정규화 과정을 빼고 진행하여 기존 SSD와 비교하여 추론 시간을 1.2에서 1.5배 감소시켰다.

3.4 구현 세부사항

본 논문에서 제안하는 모델의 전체적 구조는 Fig. 2와 같다. SSD의 1-Stage 구조와 Feature Pyramid Networks (FPN)[16]의 특징을 층층이 쌓아 올린 구조를 섞은 형태를 사용한다. 모델의 구조는 크게 특징 추출을 수행하는 backbone network, 직접적인 결과를 도출하는 header, backbone과 header를 잇는 neck으로 나누어 구성되어 있다. backbone network를 통해 추출한 특징을 SSD layer를 통해 header로 연결한다. 해당 과정에서 FPN 구조를 통해 스테이지별로 추출한 특징을 Deconvolution module에 통과시킨다. 스테이지별로 추출된 특징들은 Deconvolution module을 통과함으로써 서로 다른 스테이지에서 온 특징들의 크기를 2*height x 2*width x 521로 같게 통일시킨다. header는 Prediction module을 사용하여 각 객체의 위치와 정확도를 직접 도출해낸다.

기존 모델의 input resolution은 가로·세로 비율이 1:1을 기준으로 크게 바뀌지 않는다. 하지만 본 논문에서 탐지하고자 하는 대상인 모바일 화면의 경우 대다수의 이미지가 1:2의 비율로 이루어져 있다. 비율의 격차에서 오는 문제를 해결하기 위해 모델의 input resolution의 비율을 1:2로 변경하였다.

Ground Truth(GT) box와 Intersection Over Union(IoU) 값이 임계 값(e.g. 0.5) 이상인 Predicted anchor box만 학습에 사용하였다. Loss 함수는 두 개의 loss 함수의 평균으로 사용하였다. 객체의 위치를 학습하기 위한 regression loss와 각 객체의 카테고리를 학습하기 위한 classification loss를 사용하였다. Smooth L1을 regression loss로 선택하였고, classification loss는 Cross entropy를 사용하여 학습을 진행했다.

구축한 데이터셋이 캡처된 모바일 화면이기 때문에 학습을 진행할 때 데이터셋을 증폭함에 있어서 random expansion augmentation trick은 사용하지 않았다. 카메라로 찍은 이미지 데이터와는 다르게 캡처된 모바일 화면 이미지는 주변 환경의 영향이 없으므로 이미지가 흔들리거나 휘는 등의 변이가 없다. 주변 환경의 영향이 없는 데이터셋으로 인해 학습을 진행하면서 모델이 데이터셋에 과적합되는 것을 방지하기 위해 색상, 채도, 명도를 랜덤하게 변경하고 vertical · horizontal flip을 사용하였다.

4. 데이터셋

모바일 디바이스 화면의 클릭 가능한 객체를 대상으로 데이터셋을 구축하였다. 다양한 해상도를 가진 디바이스들로부터 데이터를 수집하였으며, 다수의 애플리케이션 화면을 대상으로 수집하였다. 총 1,261장의 모바일 화면 이미지에 대하여 24,937개의 annotation data를 수집하였다. 해당 데이터셋을 통해 딥러닝 모델을 학습하고 유효성을 검증하기 위해 데이터셋을 train, validation, test set으로 나누었다. 전체 데이터셋 중에서 약 80%인 1,020장의 이미지는 train set으로 나누었다. 약 10%인 114장의 이미지는 유효성 검증을 위한 validation set으로 나누고, 나머지 약 10%는 test set으로 나누었다. 전체 annotation data 역시 각각 같은 비율로 train, validation, test set으로 나누었다. annotation data의 카테고리는 클릭 가능한 객체를 기준으로 선정하였다. text, edit text, image, button, region, status bar, navigation bar 의 7개 카테고리를 선정하였다. 데이터셋 규격은 Visual Object Classes(VOC)[17]규격을 준수하였다. 각 set 별 이미지와 annotation data는 Table 1과 같다.

카테고리별 annotation data의 수는 Fig. 3과 같다. text가 8,462개로 annotation data 중에서 가장 많다. image는 7,705개로 두 번째로 많았으며 region, button, status bar, edit text가 각각 3,563, 2,165, 1,228, 1,093개로 그 다음으로 많았다. navigation bar는 721개로 가장 적었다.

text와 image category의 경우 일반적으로 각 이미지에 최소 수 개의 annotation data가 있어 데이터 수집이 용이하였다. region category의 경우 여러 annotation data가 한 장의

Table 1. Annotation Data

	Images	Annotations
Train	1020	20268
Validation	114	2213
Test	127	2456
All	1261	24937

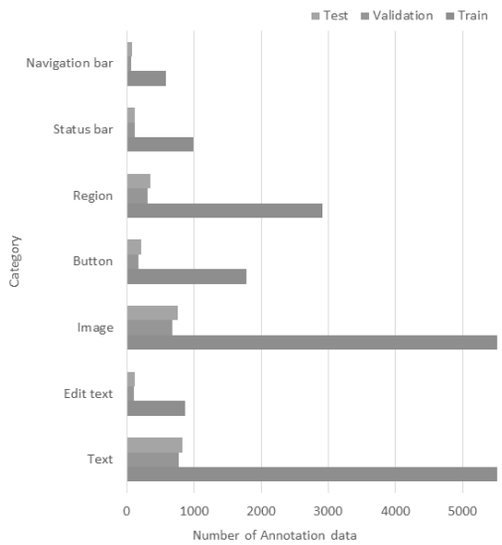

Fig. 3. Annotation for Each Category

이미지에 모여있고 위치가 확실히 구분되어 있을 때로 정의하였다. status bar는 모바일 디바이스의 가장 상단의 상태 창으로써 현재 모바일 디바이스의 옵션 상태가 주로 기록되어 있다. navigation bar는 가장 하단의 시스템 버튼이 있는 구역으로 정의하였다. edit text의 경우 수정 가능한 text 영역으로써 클릭하여 수정 가능한 상태가 되는 text로 정의하여 수집하였다.

기존 앵커 베이스의 모델들은 추출된 특징에 앵커를 투영하여 객체의 위치와 크기를 찾는다. 이때 앵커는 사전에 데이터셋을 기반으로 그 크기와 비율을 정의해 놓는다. 따라서 각 annotation data의 크기와 비율을 아는 것은 성능 향상에 중요한 사항이 될 수 있다. 본 논문에서 구축한 데이터셋에서 annotation data의 area와 비율의 상관관계는 Fig. 4에서 볼 수 있다. 데이터셋의 annotation data는 전체적으로 가로가 더 긴 형태인 aspect ratio 값이 1이 넘는 형태가 가장 많다. 데이터셋 중에서 text, status bar, navigation bar 카테고리의 영향으로 가로 비가 큰 경향을 보였다. 영역이 작은 경우 aspect ratio 값이 더 큰 경향을 보였으며, 반대로 객체의 영역이 커질수록 aspect ratio는 1에 수렴하는 경향을 보였다.

카테고리별 area information은 Fig. 5와 같다. area는 region이 가장 크고 navigation bar, status bar, button, edit text, image, text 순으로 컸다. region의 경우 여러 다른 객체를 포함하는 큰 범위로 정의하였기 때문에 평균적으로 가장 큰 area를 가지고 있다. navigation bar와 status bar의 경우 모든 모바일 디바이스 화면에 대해 가장 상단과 가장 하단에 전체 가로 폭만큼 위치한다. 이때, 일반적으로 navigation bar의 폭이 status bar의 폭 보다 약 2배 가까이 두꺼워 area 역시 약 2배 큰 경향을 보였다. edit text의 경우 평균적으로 text의 약 2배의 area를 가지고 있다. image의 경우 이모티콘과 같이 작은 이미지들과 이미지의 대부분을 차지할 정도로 크기가 큰 경우가 모두 포함되어 편차가 가장 크고, 평균적으로는 약 30000 pixel의 area를 가지고 있다. text의 경우 폭이 가장 얇아서 area 역시 가장 작다.

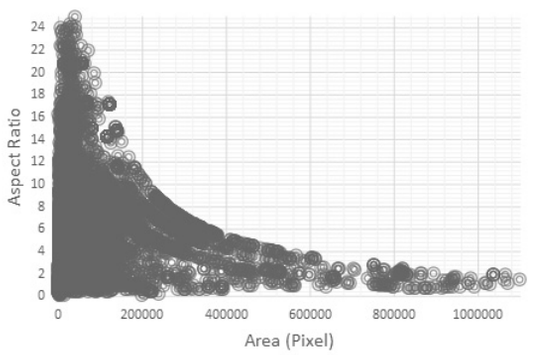

Fig 4. Bounding Box Area Against Aspect Ratio

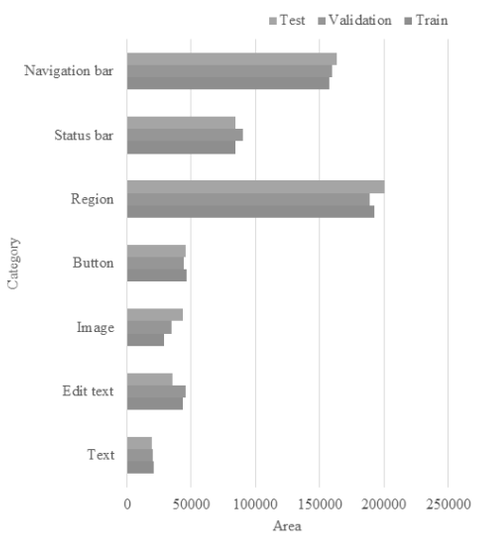

Fig 5. Area Means for Each Category

5. 실 험

5.1 트레이닝

구축한 데이터셋 중에서 train set을 학습하였다. Multi-scale training은 사용하지 않았다. 구축한 데이터셋의 카테고리별 annotation data의 개수가 달라 발생하는 Class imbalance 문제와 그로 인해 발생하는 False negative 문제를 해결하기 위해 Hard negative mining을 적용하였다. 한 대의 V100 GPU를 사용하여 학습을 진행하였다. loss 값이 0.01 이하로 떨어질 때까지 학습을 진행하고, 그중에서 validation set에 대하여 가장 낮은 loss 값을 가진 epoch를 선택하였다. initial learning rate는 0.001, Momentum은 0.9로 설정하여 학습을 진행하였다. Weight decay는 0.9로 설정하여 모델이 과적합 되는 것을 방지하였다.

5.2 실험 결과

실험 결과는 Table 2에서 볼 수 있다. 기준 모델인 DSSD-512는 ResNet-101을 backbone network로 사용하였다. input resolution은 512 x 512이며 1:1의 비율을 가지고 있다. 실험 결과 mAP는 43.0이며 FPS는 12.9이다. 본 논문에

Table 2. Experiment Result

	DSSD-512	Our-512	Our-1080
Backbone	ResNet-101	SENet-101	SENet-101
mAP	43.0 (Base)	68.4 (+59.0%)	86.5 (+101.1%)
FPS	12.9 (Base)	10.2 (-2.7)	9.5 (-3.4)
Input resolution	512 x 512	512 x 1024	1080 x 1920

Fig. 6. Inference Result

서 제안한 모델인 Our-512는 backbone network로 SENet-101을 사용하였다. 모델의 input resolution은 512 x 1024이며 1:2의 비율을 가진다. 실험 결과 mAP는 68.4로 DSSD-512에 비하여 59% 향상된 정확도를 보였다. FPS는 10.2로 DSSD-512와 비교하여 초당 2.7장 느린 추론 시간을 보였다. Our-1080은 Our-512와 마찬가지로 SENet-101을 backbone network로 사용하였다. input resolution은 Our-512보다 큰 1080 x 1920을 가지고 있다. mAP는 86.5로 DSSD-512과 비교하여 101%, Our-512과 비교하여 26.4% 정도 정확도가 증가하였다. FPS는 DSSD-512와 비교하여 3.4, Our-512와 비교하여 0.7 정도 느린 추론 시간을 보였다.

Our-512에서는 2배 커진 input resolution의 area와 변경된 backbone network로 기존 베이스라인인 DSSD-512에 대비하여 초당 10장 이상의 처리 속도를 유지하면서 mAP는 59%가량 증가하였다. Our-1080의 경우 Our-512에 비해 input resolution의 area가 4배 가까이 증가한 반면, FPS는 1 이하로 감소하였고 mAP는 DSSD-512 대비 101%, Our-512 대비 26%가량 증가하여 더욱 정확하게 클릭 가능한 객체를 탐지하였다.

추론 결과 사진은 Fig. 6에서 볼 수 있다. 이미지의 가장 상단에 가로로 긴 모양의 status bar, 좌측 그림의 가장 하단에 navigation bar를 확인할 수 있다. status bar를 제외한 영역에서 나머지 다른 Category에 해당하는 이미지를 확인할 수 있다. image의 경우 크기가 작은 객체부터 가로 혹은 세로로 긴 형태의 다른 비율을 가진 객체까지 모두 탐지되는 것을 확인하였다.

6. 결 론

본 논문에서는 모바일 디바이스 화면을 대상으로 클릭 가능한 객체를 찾기 위해, 데이터셋을 구축하고 객체 탐지 모델을 설계하였다. DSSD의 backbone netowrk인 ResNet을 SENet으로 변경하여 추가 오버헤드 없이 특징을 추출하였다. SSD layers와 Deconvolution module를 FPN 구조로 쌓아 올리고, 이를 통해 특징의 크기를 같게 하여 header로 연결하였다. header에서 최종적인 객체의 위치와 카테고리를 추출하였다. 데이터셋은 서로 다른 해상도를 가진 모바일 디바이스를 통해, 다양한 애플리케이션을 대상으로 수집하였다. 수집한 이미지들을 text, edit text, image, button, region, status bar, navigation bar로 구성하여 7개의 카테고리로 구분하고 annotation 하였다. 최종적으로 24,937개의 annotation data를 수집하였다. 제안한 모델과 구축한 데이터셋을 이용하여 실험을 진행하였고 기존 DSSD 모델과 비교하여 최대 약 101% 향상된 성능을 입증하였다.

References

[1] Y. Baek and D. Bae, "Automated model-based android GUI testing using multi-level GUI comparison criteria," in *Proceedings of the IEEE/ACM International Conference on Automated Software Engineering*, Singapore, pp.238-249, 2016.

[2] I. A. Salihu, R. Ibrahim, B. S. Ahmed, K. Z. Zamli, and A. Usman, "AMOGA: A static-dynamic model generation strategy for mobile apps testing," *IEEE Access*, Vol.7, pp.17158-17173, 2019.

[3] A. Usman, N. Ibrahim, and I. A. Salihu, "TEGDroid: Test case generation approach for android apps considering context and GUI events," *International Journal on Advanced Science, Engineering and Information Technology*, Vol.10, No.1, pp.16-23, 2020.

[4] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, pp.580-587, 2014.

[5] J. R. R. Uijlings, K. E. A. Van De Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," *International Journal of Computer Vision*, Vol.104, No.2, pp.154-171, 2013.

[6] R. Girshick, "Fast R-CNN," in *Proceedings of the IEEE International Conference on Computer Vision*, Santiago, pp.1440-1448, 2015.
[7] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proceedings of the Advances in Neural Information Processing Systems*, Quebec, pp.91-99, 2015.
[8] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Proceedings of the European Conference on Computer Vision*, Amsterdam, pp.21-37, 2016.
[9] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv:1409.1556v6 [cs.CV] 10 Apr. 2015.
[10] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Nevada, pp.779-788, 2016.
[11] C.-Y. Fu, Wei Liu, Ananth Ranga, Ambrish Tyagi and Alexander C. Berg, "DSSD: Deconvolutional Single Shot Detector," arXiv:1701.06659v1 [cs.CV] 23 Jan. 2017.
[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep resudual learning for image recognition," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Nevada, pp.770-778, 2016.
[13] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition,* Utah, pp.7132-7141, 2018.
[14] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *Proceedings of the European Conference on Computer Vision,* Amsterdam, pp.483-499, 2016.
[15] Y. Gao, O. Beijbom, N. Zhang, and T. Darrell, "Compact bilinear pooling," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Nevada, pp.317-326, 2016
[16] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature Pyramid Networks for Object Detection," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition,* Honolulu, pp.2117-2125, 2017.
[17] M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *International Journal of Computer Vision,* Vol.88, No.2, pp.303-338, 2010.

조 민 석
https://orcid.org/0000-0002-7483-4142
e-mail : jms0923@korea.ac.kr
2020년 충남대학교 컴퓨터공학과(학사)
2020년 ~ 현 재 고려대학교 전기전자공학과 석사과정
관심분야 : 컴퓨터 비전, 딥러닝, 분산 시스템

전 혜 원
https://orcid.org/0000-0002-6761-5865
e-mail : uclacarol@korea.ac.kr
2020년 한국외국어대학교 컴퓨터공학과 (학사)
2020년 ~ 현 재 고려대학교 전기전자공학과 석사과정
관심분야 : 컴퓨터 비전, 딥러닝, 분산 시스템

한 성 수
https://orcid.org//0000-0002-4915-6247
e-mail : sshan1@kangwon.ac.kr
1998년 경상대학교 정보통신공학과(학사)
2005년 순천향대학교 정보통신공학과(석사)
2019년 고려대학교 영상정보처리협동과정 (박사)
2015년 ~ 2016년 오리온 테크놀리지 이사
2018년 ~ 2019년 순천향대학교 조교수
2019년 ~ 현 재 강원대학교 자유전공학부 조교수
관심분야 : 컴퓨터 교육, 영상정보처리, 병렬처리, 딥러닝

정 창 성
https://orcid.org/0000-0001-9654-8406
e-mail : csjeong@korea.ac.kr
1981년 서울대학교 전기공학과(학사)
1984년 Northwestern University 전자계산학과(석사)
1987년 Northwestern Unviersity 전자계산학과(박사)
1987년 ~ 1992년 포항공과대학교 전자계산학과 조교수
1992년 ~ 1998년 고려대학교 전자공학과 부교수
1998년 ~ 현 재 고려대학교 전자공학과 정교수
관심분야 : 컴퓨터 비전, 유비쿼터스 컴퓨팅, 네트워크 가상 컴퓨팅, 클라우드 컴퓨팅