머신러닝에 기반을 둔 사진 속 개인정보 검출 및 블러링 클라우드 서비스

김민정, 이수영, 이지영, 함나연
*서울여자대학교 정보보호학과

hamny888@naver.com
alison7104@naver.com
bestjy1202@nate.com
dltndud100@gmail.com

Personal Information Detection and Blurring Cloud Services Based on Machine Learning

Na-youn Ham*, Min-jeong Kim*, Jiyoung Lee*,Soo-young Lee*
**Dept of Information Security, 1)Seoul Women University

요 약

클라우드가 대중화되어 많은 모바일 유저들이 자동 백업 기능을 사용하면서 민감한 개인정보가 포함된 사진들이 무분별하게 클라우드에 업로드 되고 있다. 개인정보를 포함한 클라우드가 악의적으로 해킹 될 시, 사진에 포함된 지문, 자동차 번호판, 카드 번호 등이 유출됨에 따라 대량의 개인정보가 유출될 가능성이 크다. 이에 따라 적절한 기준에 맞게 사진 속 개인 정보 유출을 막을 수 있는 기술의 필요성이 대두되고 있다.

현재의 클라우드 시스템의 문제를 해결하고자 본 연구는 모바일 기기에서 클라우드 서버로 사진을 백업하는 과정에서 영역 검출과 블러링의 과정을 제안하고 있다. 클라우드 업로드 과정에서 사진 속의 개인 정보를 검출한 뒤 이를 블러링하여 클라우드에 저장함으로써 악의적인 접근이 행해지더라도 개인정보의 유출을 방지할 수 있다.

머신러닝과 computer vision library등을 이용하여 이미지 내에 민감한 정보를 포함하고 있는 영역을 학습된 모델을 통해 검출한 뒤, OpenCV를 이용하여 블러링처리를 진행한다 사진 속에 포함될 수 있는 생체정보인 지문은 손 영역을 검출한 뒤, 해당 영역을 블러링을 하여 업로드하고 카드번호나 자동차 번호판이 포함된 사진은 영역을 블러링한 뒤, 암호화하여 업로드 된다. 후에 필요에 따라 본인인증을 거친 후 일정기간 열람을 허용하지만 사용되지 않을 경우 삭제되도록 한다.

개인정보 유출로 인한 피해가 꾸준히 증가하고 있는 지금, 사진 속의 개인 정보를 보호하는 기술은 안전한 통신과 더불어 클라우드의 사용을 더 편리하게 할 수 있을 것으로 기대된다.

1. 서론

4차 산업혁명으로 빠르게 변화하는 지금 백업과 용량에 대한 수요가 증가하면서 일상생활에서 클라우드는 없어서는 안되는 서비스로 자리를 잡고 있다. 이러한 클라우드에 지문, 자동차 번호판, 카드 번호 등 민감한 개인 정보가 업로드 되는데 만약 악의적으로 해킹될 시 개인 정보가 유출 될 가능성이 크다. 업로드 된 사진은 디바이스에서 지운다 해도 클라우드에 남아있게 된다. 타인의 정보를 탈취해 범죄를 저지르는 사람들은 이를 악용하기도 한다. 개인정보 유출로 인한 피해가 증가하고 클라우드를 악용하는 사람들이 많아짐에 따라 현 시스템에 문제의식을 갖게 되었다. 이를 계기로 개인정보 유출 방지를 위한 클라우드 서비스를 고안하였다.

1) "본 연구는 과학기술정보통신부 및 정보통신기술진흥센터의 SW중심대학지원사업의 연구결과로 수행되었음"(2016-0-00022)

2. 연구배경

웹 하드의 형태에서 클라우드 스토리지나, 웹과 모바일 기기를 이용한 소프트웨어 사용 등 다양한 형태의 서비스들이 제공되고 있다. 이 서비스를 사용하는 사용자와 제공자의 입장에서는 상호 신뢰성이 보장되어야 한다[1]. 때문에 클라우드 서비스에 대한 보안은 이전 하드웨어 형태의 저장소보다 보다 높은 수준의 보안을 필요로 한다. 클라우드의 사용

이 급격히 늘어나면서 클라우드의 보안 위협에 대한 많은 연구가 진행되고 있다. 이와 같은 연구들은 무선 네트워크의 위협, 공유기술의 취약성, 데이터 손실 및 누출, 대규모 분산 시스템의 버그 등 클라우드 컴퓨팅 서비스 자체의 보안 문제와 데스트톱에 비해 신뢰성이 낮은 모바일 단말의 취약점에 대해 다루고 있다[1][2].

본 연구에서는 기존의 클라우드 자체의 보안에 대한 연구와는 다르게 접근하였다. 클라우드 서비스와 모바일 단말의 취약성이 아닌 클라우드에 업로드 되는 사진 자체의 보안에 초점을 맞추었다. 머신러닝을 통해 취약한 물체를 학습시키고 클라우드에 파일이 업로드 되기 전 사진 내의 취약한 정보를 검출하고 블러링 하여 클라우드에 대한 해킹이 이루어져도 개인정보를 보호할 수 있는 기대를 할 수 있다.

3. 연구내용

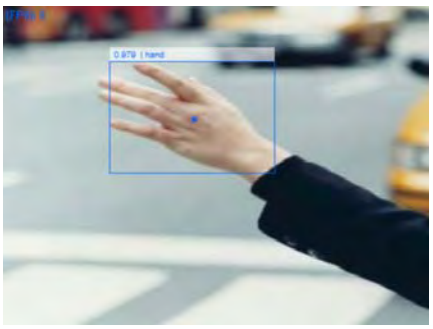
3.1 지문 영역 검출 및 블러링

지문을 인식하기 위해서 우선 손 부분을 먼저 검출할 수 있도록 해야 한다. 초반에는 OpenCV라는 컴퓨터 비젼 프로그램과 Matlab을 이용해서 손 영역을 검출한 뒤, 손가락 부분을 블러링 처리하려고 했으나, 다음 사진과 같이 정확도가 일정하지 못하다는 문제점이 발생하였다.

[그림 3-1] OpenCV를 이용한 오류

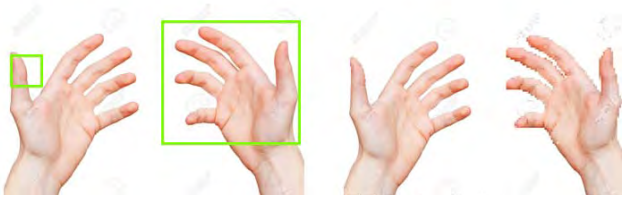
이에 Matlab을 이용하여 convexhull로 정확도를 개선시켜 보려 하였으나, 손이 포개진 경우나 많은 경우에는 정확도가 떨어진다는 문제점이 발생하였다.
기계학습으로 훈련된 모델을 사용하기 위해 Tensorflow로 기술을 변경하였고, Github의 Hand detection model을 사용하여 cocoapi Object detection을 사용하였다. 다음 사진은 모델을 이용한 사례이고, 이 모델과 coco API를 같이 사용하는 것으로 개선방안을 수립하였다[3].
훈련시킨 해당 모델을 사용하면 다음과 같이 정확도를 개선시킬 수 있다.

[그림 3-2] Tensorflow Hand detection Mode , CoCoAPI



[그림 3-3] Github 모델을 사용한 예시

이 모델을 이용하여 손을 검출한 뒤, 그 영역을 블러링하여 다시 클라우드에 재 업로드 하는 방식을 구현하였다.



[그림 3-4] 개선 결과물 (인식- 블러링)

3.2 자동차 번호판 검출 및 블러링

opencv를 활용한 자동차번호판의 인식 및 블러링 처리 결과이다. 먼저 아래 예시의 경우, opencv library를 이용한 것으로 이를 이용한 전처리 과정을 통해 자동차 번호판을 추출하게 된다. cvtColor 함수를 이용하여 gray 컬러로 이미지를 변경하여 이미지의 특징점을 추출한 다음, canny 알고리즘 등을 이용해 윤곽선을 찾아내고 번호판의 후보를 선정한다. 선정된 번호판 후보 중 가장 정확성이 높은 번호판 영역이 추출되면 따로 cvSetImageROI를 이용하여 관심영역을 선정한다. 관심영역이 선정되면 cvSmooth함수를 이용하여 관심영역에 블러링을 하게 된다. 이때 원본 이미지에 cvAddweighted함수로 블러링 이미지를 입혀 씌우는 방식을 통해 최종이미지를 출력한다.

[그림 3-6] OpenCV liberary를 이용한 자동차 번호판 추출 및 블러링

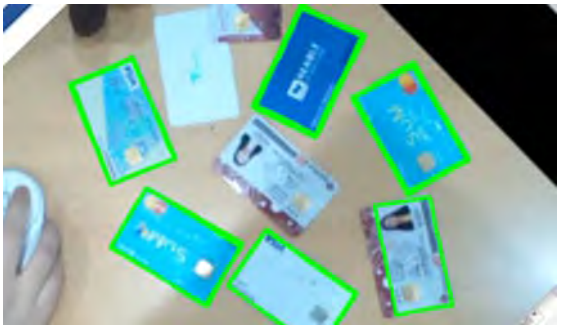
하지만 사선에서 찍힌 이미지, 혹은 너무 어두운 곳에서 찍힌 이미지는 정확도가 떨어짐을 확인 할 수 있다. 따라서 Computer vision Library인 opencv에 더불어 머신러닝의 물체인식 알고리즘은 yolo를 활용하여 인식율을 높이고자 한다. YOLO란 (You Only Look Once)로 물체를 인식하여 bounding box로 표현해주는 알고리즘이다[4]. YOLO 알고리즘은 속도는 굉장히 빠르지만 다른 알고리즘에 비하여 정확도는 조금 떨어지는 단점을 가지고 있다. 지도학습에 해당하는 object detection은 많은 양의 정답 데이터가 필요하기 때문에 직접 데이터를 수집하였고, 이를 바탕으로 lebelimg프로그램을 이용하여 사진 속 번호판 영역을 특정 지어 주었다. 이 이미지를 GNU general Public license v3.0인 darkfow라는 darknet의 tensorflow버전을 이용하여 yolo를 사용할 수 있다. 본 프로젝트에 알맞게 코드를 수정한 뒤 학습을 진행하고 새로운 이미지를 이용하여 detection을 진행할 수 있다.

3.3 카드 번호판 검출 및 블러링

카드는 OpenCV를 이용해 각 꼭짓점을 찾아 연결해 사각형을 찾는 방법으로 검출을 진행하였다. 이미지에서 형상의 점들의 시퀀스를 연결해 4개의 꼭짓점을 연결해 사각형의 카드 형태를 추출하였지만 모서리가 둥근 카드의 꼭짓점을 정확하게 찾지 못하는 문제점이 발생하였다.

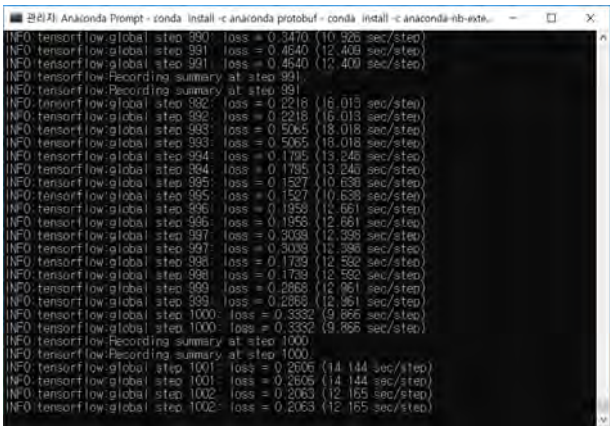
[그림 3-7] 4개의 꼭짓점으로 찾은 카드

정확도를 높이기 위해 find_squared()라는 함수를 이용해 사각형 검출을 하였다.



[그림 3-8] find_squares()로 검출한 카드

이 과정에서도 몇몇 카드를 검출하지 못하는 문제가 있었기에 더 정확한 인식을 위해 머신러닝의 오픈소스인 Tensorflow를 활용하여 물체의 인식률을 높이고자 한다. 트럼프 카드 검출 프로그램을 활용하여 신용카드에 대한 학습을 시킨 후 OCR로 카드 내의 문자 검출을 한다[5].



[그림 3-9] Tensorflow로 학습을 시키는 과정

약 40,000번 정도의 학습을 시키고, loss가 0.05 이하로 떨어지면 정확도가 높아진 것을 확인할 수 있다.

검출된 문자는 각 카드가 가지고 있는 고유한 문자열들과 비교를 해 일치하는 문자열이 검출되면 그 문자열을 가진 카드는 개인정보가 들어있는 카드로 인식을 할 수 있다. 최종적으로 검출된 개인정보가 들어있는 카드는 cvSmooth()를 이용하여 카드 내 민감한 정보의 블러링을 진행할 수 있다.

3.4 어플리케이션 동작

어플리케이션은 크게 두 가지로 동작을 하게 되는데, 사용자의 기기에 있는 사진을 클라우드 서버로 전송하는 동작과 클라우드를 편리하게 관리할 수 있는 UX/UI 기능을 하게 된다. 핸드폰의 기존 어플리케이션은 클라우드 서버와 추가적인 동작을 할 수 없기 때문에, 우리 팀만의 별도 갤러리 어플리케이션을 만들어 설치한다. 그 후 어플리케이션에서 기존 갤러리의 사진들에 접근할 수 있는 권리를 승인받아 사진들을 불러올 수 있도록 하는데 브릿지를 통해 서버와 통신한다.

3.5 서버구축

손, 자동차 번호판, 카드를 검출 블러링 하는 코드는 서버에서 하나로 합쳐진다. 서버는 브릿지를 통해 어플리케이션과 통신을 하고 사진을 받아온다. 받아온 사진은 서버에서 민감 정보를 검출 및 블러링 한 후 다시 사용자의 어플리케이션과 클라우드 서버에 보내게 된다.

서버는 CentOS로 네트워크를 브릿지 어댑터로 설정하여 외부와 연결가능하게 한다. 고정 IP를 설정해두어 재부팅 시에도 저장한 IP가 남아있을 수 있게 한다. 네트워크 설정 후에는 소켓을 열어 이미지를 주고받을 수 있게 한다[6]..

4. 결론

현재 점점 많은 유저들이 클라우드를 사용하고, 디바이스 분실 등을 우려해 클라우드의 자동백업 기능을 사용하고 있다. 업로드 된 사진은 디바이스에서 지운다 하더라도 클라우드에는 남아있게 된다. 이런 상태에서 누군가 클라우드를 악의적으로 해킹한다면 대량의 개인정보가 유출될 위험이 있다. 본 연구는 클라우드에 사진이 업로드 되기 전 사진에서 민감한 정보를 블러링 해주어 예기치 못한 해킹에도 개인정보 유출을 방지할 수 있다. 또한, 사용자들이 정보보호에 대해 인식을 제고할 수 있으며 정보유출로 인한 사회적 비용도 감소시킬 수 있다.

참고문헌

[1] Jeong-hoon Jeon “A study on the vulnerability of the Cloud computing security”

[2] Eun-Young Jang, Hyung-Jong Kim, Choon-Sik Park, Joo-Young Kim, Jae-il Lee “The study on a threat countermeasure of mobile cloud services”

[3] https://github.com/victordibia/handtracking

[4] https://github.com/Park-Ju-hyeong/yolo_darkflow

[5] https://github.com/EdjeElectronics/TensorFlow-Object-Detection-API-Tutorial-Train-Multiple-Objects-Windows-10/blob/master/README.md

[6] https://gist.github.com/kittinan/e7ecefddda5616eab2765fdb2affed1b