

3차원 회전을 이용한 인조 번호판 생성기의 번호판 인식 성능 비교

*이유진 **김상준 ***박경무 박구만

*서울과학기술대학교 미디어IT공학과

**서울과학기술대학교 정보통신미디어공학전공

***서울과학기술대학교 전자IT미디어공학과

e-mail : youuuu00@naver.com

Comparison of number plate recognition performance of Synthetic number plate generator using 2D and 3D rotation

*Lee, Yu-Jin, **Kim, Sang-Joon, ***Park, Gyeong-Moo, Park, Goo-Man

*Dept. of Media IT Engineering

**Dept. of Information Technology and Media Engineering

***Dept. of Electronics and IT Media Engineering

Seoul National University of Science and Technology

요약

최근 딥러닝을 이용한 자동차 번호판 인식 알고리즘에 있어서 인조 번호판을 생성하여 데이터 수집과 라벨링 작업 시간을 줄이기 위한 연구가 진행되고 있다. 하지만 인조 번호판의 특성상 정면의 이미지로 구성되어 있기 때문에 자동차의 정면에서 촬영된 번호판의 인식률은 높지만 측면에서 촬영된 번호판의 경우 인식률이 낮아진다. 본 논문에서는 다양한 카메라 설치 위치에 따른 다각도로 촬영된 번호판 영상의 인식률을 보완하기 위해 이미지를 3차원으로 회전하여 데이터를 생성하는 인조 번호판 생성기 프로그램을 개발하였다. 3차원 회전을 하였을 때 번호판 인식 성능을 비교하기 위해 기존 방식으로 생성한 번호판과 제안 방식으로 생성한 번호판 각 600,000장씩 생성하여 총 1,200,000장을 생성하였으며, 데이터의 비율에 따라 10가지의 학습 데이터 셋을 구성하였다. 인조 번호판 데이터의 학습 결과를 평가하기 위해 실제 번호판 이미지 1789장으로 테스트 셋을 구성하였고, 기존의 인조 번호판 생성 방식과 인식 정확도를 비교 분석하였다.

1. 서론

주차 관리 시스템, 고속도로 통행료 확인, 과속 및 방법 차량 탐지와 같이 자동차 번호판 인식 시스템은 다양한 환경에서 사용되고 있다. 자동차 번호판 인식을 위한 딥러닝 모델 학습 성능을 위해 인조 번호판을 생성하여 데이터를 수집하고 라벨링하는 시간과 인력 소모를 줄이는 연구가 진행되고 있다. 기존에 제안되었던 인조 번호판 생성 시스템의 인조 이미지의 경우 데이터의 다양성을 위해 정면 번호판 이미지에서 x, y 축으로, 즉 2차원으로 회전하여 생성하였다[1]. 이렇게 생성된 이미지로 학습하여 검출할 경우 교차로나 차량 출입구 앞, 건물 등 자동차를 정면에서 촬영하여 번호판을 인식할 때 효과가 있을 수 있다. 하지만 드론으로 촬영해야 하는 상황이나 CCTV(Closed Circuit Television) 설치 환경 상의 문제로 측면에서 인식하는 상황에서는 번호판을 인식해내기 어렵다.

이에 본 논문에서는 다양한 카메라 설치 위치에 따른 다각도로 촬영된 번호판 영상의 인식률을 보완하기 위해 인조 번호판 생성 시 3차원으로 회전시켜 데이터를 생성하는 것을 제안한다. 아래 그림 1과 같이 생

성된 번호판을 x, y 축 뿐만 아니라 z축으로 회전시킴으로써 데이터 셋을 더 다양하게 구축할 수 있어 이를 통해 다양한 환경에 강인한 딥러닝 모델을 구현할 수 있다. 효용성을 증명하기 위해 2차원으로 회전하여 생성된 번호판 데이터와 3차원으로 회전하여 생성된 번호판 데이터 각각을 학습하여 번호판 인식 정확도를 측정하였다. 그림 1에서 왼쪽 이미지는 기존에 연구되었던 인조 번호판의 이미지로, x와 y축으로만 회전하였고, 오른쪽 이미지는 본 논문에서 제안하는 x, y, z축으로 회전된 이미지이다.

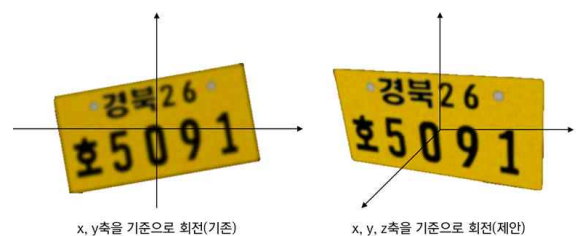


그림 1. 2차원 회전한 이미지와 3차원 회전한 이미지의 차이

2. 관련 연구

2.1. YOLOv3

CNN 기반의 딥러닝은 이미지와 이미지에 들어있는 객체의 좌표, 클래스 이름과 같은 객체 정보를 입력하여 해당 객체를 학습시키는 모델을 만들고, 만들어진 모델을 통해 이미지를 입력하였을 때 객체를 검출하는 프로세스를 가진다. 딥러닝 기반의 객체 검출 알고리즘 중 대표적인 알고리즘인 YOLOv3(You Only Look Once v3)[2]는 객체 위치의 후보 영역을 찾고 후보영역의 클래스를 분류하는 과정을 한 단계로 진행한다. 또한 객체 검출을 한 단계로 진행하기 때문에 속도가 빨라 실시간 처리 속도를 유지하고 이전 버전인 YOLO[3]의 작은 객체를 검출하지 못했던 약점을 보완하면서 성능이 향상된 버전이다. 본 논문에서는 자동차 번호판 인식의 실시간 처리와 번호판의 작은 문자들을 인식하기 위해 YOLOv3를 사용하였다.

2.2. openFrameworks

openFrameworks[4]는 그래픽, 이미지, 비디오 처리 등과 같이 다양한 라이브러리를 포함하는 C++ 기반 오픈 소스 라이브러리이다. 윈도우, 맥OS, 리눅스, iOS, 안드로이드와 같이 다양한 운영체제를 지원하고, 그에 맞는 개발환경 또한 선택할 수 있어 다양한 플랫폼에서 동작하도록 디자인 되어있다. 또한 “addon” 시스템을 통해 OpenCV나 OpenGL, FreeImage등을 포함한 이미 구현된 라이브러리를 추가하거나 직접 구현하여 확장할 수 있다. 본 논문에서는 인조 번호판을 생성하는데 openFrameworks를 이용하였다. openFrameworks의 OpenCV 라이브러리를 addon에 추가하여 이미지의 잡음, 블러, 밝기 조절과 같은 효과를 주었고, 2차원 회전 모듈을 구현하였다. 그리고 자체 내장되어 있는 OpenGL 라이브러리를 이용하여 3차원 벡터를 다루는 ofVec3f 메소드를 통해 이미지의 3차원 회전을 구현하였다.

3. 인조 번호판 생성

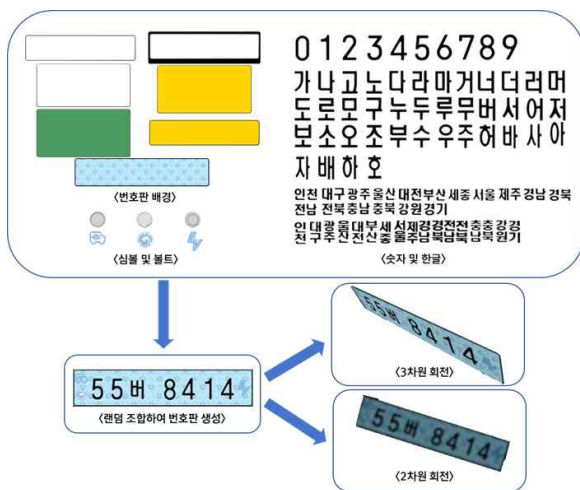


그림 2. 인조 번호판 생성 알고리즘

인조 번호판 데이터 생성 과정은 그림 2와 같다. 본 연구에서는 비사업용 자동차(흰색, 초록색) 번호판과 자동차운수사업용 자동차(노란색)

번호판, 친환경 자동차(파란색) 번호판 이미지를 생성했다. 먼저 번호판 배경 이미지가 랜덤으로 선택되고, 해당 번호판 배경의 규격에 맞게 숫자와 한글 이미지가 선택된다. 친환경 자동차일 경우 심볼 이미지가 선택되고, 그렇지 않은 경우엔 볼트 이미지가 선택된다. 선택된 배경, 숫자, 한글, 볼트 또는 심볼 이미지가 조합되어 번호판 이미지를 만든다.

이후 생성된 번호판 이미지를 회전하였는데 2차원 회전의 경우 이미지의 중심 좌표를 기준으로 openCV의 getRotationMatrix2D 함수를 통해 각 픽셀값들을 회전시키고, 그 후 warpAffine 함수를 통해 이미지를 재구성하였다. 3차원 회전의 경우 OpenGL의 회전 모듈을 이용하였다. 먼저 회전할 축(x, y, z)과 각도를 설정하여 getRotated 함수를 통해 입력 이미지를 회전시키고, 회전된 3차원 좌표를 2차원 이미지로 표출시키기 위해 2차원 좌표로 변환시킨다. 이후 YOLOv3를 통해 학습시키기 위해 숫자와 한글을 포함한 문자의 클래스 id와 바운딩 박스 좌표 및 너비, 높이를 텍스트 파일로 저장한다.

마지막으로 회전된 이미지에 가우시안 노이즈를 생성하고, 밝기를 조절한 뒤 가우시안 블러를 적용하였다. 잡음 생성은 먼저 입력된 이미지와 똑같은 크기의 이미지를 만들고 그곳에 미리 설정한 평균과 표준편차를 따르는 가우시안 분포의 노이즈를 랜덤으로 생성한다. 그리고 addWeighted 함수를 통해 입력된 이미지의 값에 노이즈 이미지의 값을 더했다. addWeighted 함수의 식은 다음의 식 (1)과 같으며, α 와 β 에 1을 입력하고 γ 에 0을 입력하였다. 그리고 src1은 입력 이미지, src2는 노이즈 이미지를 의미한다. 이미지의 밝기는 입력 이미지의 RGB값에 랜덤으로 생성된 스칼라 값을 차감하여 조절하였다. 블러 효과는 OpenCV에서 제공하는 가우시안 블러 기법을 적용하였다.

$$dst = \alpha * src1 + \beta * src2 + \gamma \quad \text{식 (1)}$$

최종적으로 생성된 이미지는 다음과 같다. 그림 3은 위와 같은 과정으로 생성된 번호판 데이터의 이미지로, 왼쪽 이미지는 2차원 회전하여 생성한 번호판 이미지이고, 오른쪽 이미지는 3차원 회전하여 생성한 번호판 이미지이다.



그림 3. 생성된 번호판 데이터 예(좌: 2차원 회전, 우: 3차원회전)

번호판의 경우 대부분 숫자로 이루어져 있어 한글 클래스(가, 나, 서울, 인천 등)는 숫자 클래스에 비해 데이터가 현저히 적다. 따라서 일반 번호판 이미지만으로 학습을 진행한다면 데이터가 많은 숫자 클래스로 학습이 압도되어 한글 클래스의 인식이 저하될 수 있다. 이러한 클래스 불균형 문제를 해결하기 위해 그림 4와 같이 한글 이미지를 무작위로 추가한 균형 데이터를 생성하였다. 그리고 일반 번호판 이미지와 균형 번호판 이미지를 조합하여 데이터 셋을 구성하였다. 그림 4는 한글과 지역 이미지를 무작위로 조합하여 생성된 균형 번호판 데이터의 예시이다.



그림 4. 균형 번호판 데이터 예

4. 실험 결과

2차원 회전과 3차원 회전을 하였을 때 번호판 인식 성능을 비교하기 위해 각각 2차원, 3차원 회전한 일반 번호판 이미지 300,000장과 균형 번호판 이미지 300,000장을 생성하여 총 1,200,000장을 생성하였다. 인조 번호판의 문자 인식 성능 비교를 위해 2차원 회전과 3차원 회전 여부, 일반 데이터와 균형 데이터의 비율에 따라 표 1과 같은 경우의 수를 나누어 총 10가지의 데이터 셋을 구축하였다. 데이터 셋 No.는 데이터 셋의 고유 번호이고, 1번부터 5번 데이터 셋은 2차원 회전한 번호판 데이터로 구성되어 있고 6번부터 10번 데이터 셋은 3차원 회전한 번호판 데이터로 구성되어 있다. 각 데이터 셋은 총 300,000장의 이미지로 구성되어 있으며 일반 번호판 이미지와 균형 번호판 이미지의 비율이 다르다.

표 1. 학습 데이터셋 구축

데이터 셋 No.	회전 구분	일반 번호판 이미지	균형 번호판 이미지	합계
1	2차원 회전	300,000	0	300,000
2		200,000	100,000	300,000
3		150,000	150,000	300,000
4		100,000	200,000	300,000
5		0	300,000	300,000
6	3차원 회전	300,000	0	300,000
7		200,000	100,000	300,000
8		150,000	150,000	300,000
9		100,000	200,000	300,000
10		0	300,000	300,000

구축한 데이터 셋을 바탕으로 Darknet-53으로 구성된 YOLOv3를 사용하여 총 10번의 학습을 진행하였다. 클래스는 그림 2의 글자 이미지와 같이 총 84개로 구성하였고, iteration은 class 수 * 2000 값인 168,000으로 설정하였다. 입력 이미지의 크기는 416 * 416 으로 설정하였다.

번호판 인식 성능을 검증하기 위해 자체 test set인 번호판 영상 image_set_1과 image_set_2를 준비하였다. image_set_1은 건물에 설치한 CCTV를 통해 촬영된 자동차의 번호판 이미지 모음이고, image_set_2는 스마트폰으로 촬영한 주차된 자동차의 번호판 이미지 모음이다. image_set_1의 경우 멀리서 자동차 정면을 찍은 영상이기 때문에 정면 이미지의 번호판이 대부분이지만 해상도가 낮다. 반대로 image_set_2의 경우 근접 거리에서 스마트폰을 움직이면서 촬영했기 때문에 다각도의 번호판 영상이 있고 해상도가 높다. 검출 과정의 입력 이미지의 크기는 416 * 416으로 설정하였다. 표 2는 번호판 인식 결과 이미지 모음 중 일부분으로 image_set_1과 image_set_2에 따른 2차원 회전한 데이터 셋의 번호판 인식 결과와 3차원 회전한 데이터 셋의

번호판 인식 결과이다. 표 1에서 볼 수 있듯, 2차원 회전한 4번 데이터 셋은 일반 번호판 이미지와 균형 번호판 이미지의 비율이 1대 2인 데이터 셋이고, 3차원 회전한 7번 데이터 셋은 일반 번호판 이미지와 균형 번호판 이미지의 비율이 2대 1인 데이터 셋이다. 번호판의 정면 이미지 위주로 구성된 image_set_1의 경우 2차원 회전하여 생성한 데이터 셋과 3차원 회전하여 생성한 데이터 셋 모두 제대로 인식한 것을 볼 수 있다. 또한 image_set_2의 인식 결과를 보면 3차원 회전하여 생성한 데이터 셋의 경우 다각도의 번호판 이미지도 제대로 인식하는 것을 볼 수 있다. 하지만 2차원 회전하여 생성한 데이터 셋의 경우 가장자리에 위치한 숫자는 인식뿐만 아니라 검출에 실패한 것을 볼 수 있다.

표 2. 번호판 인식 결과

구분	2차원 회전한 4번 데이터 셋의 번호판 인식 결과	3차원 회전한 7번 데이터 셋의 번호판 인식 결과
image_set_1		
	정답 : 79나 4314	
	인식 결과 : 79나 4314	인식 결과 : 79나 4314
	정답 : 85조 5874	
	인식 결과 : 85조 5874	인식 결과 : 85조 5874
image_set_2		
	정답 : 14노 9318	
	인식 결과 : 14노 9318	인식 결과 : 14노 9318
	정답 : 서울33바 8115	
	인식 결과 : 서울33바 8115	인식 결과 : 서울33바 8115
image_set_2		
	정답 : 21조 9670	
	인식 결과 : 1조 9670	인식 결과 : 21조 9670
	정답 : 03더 4387	
	인식 결과 : 033더 4387	인식 결과 : 03더 4387
image_set_2		
	정답 : 48라 3596	
	인식 결과 : 48라 359	인식 결과 : 48라 3596
	정답 : 32가 1487	
	인식 결과 : 32가 14	인식 결과 : 32가 1487

표 3은 10가지의 데이터 셋에 따른 번호판 인식 정확도 측정 결과로 전체 클래스의 평균값이다. 데이터 셋 No.는 표 1의 데이터 셋 No.와 같다. 1번부터 5번은 2차원 회전한 데이터 셋이고, 6번부터 10번은 3차원 회전한 데이터 셋이다. Ground-truth는 준비한 test set의 문자 개수이며 detection results는 ground-truth의 인식 결과이다. Mean log-average miss rate는 인식 결과 에러율의 전체 클래스 평균값이

다. mAP는 recall 값에 대응하는 precision값의 평균값을 전체 클래스 개수로 나눈 값이며, fps는 초당 처리되는 frame 개수이다. image_set_1의 인식 성능이 가장 높은 데이터 셋은 4번으로 2차원 회전하여 인조 번호판을 생성하고, 일반 번호판 데이터와 균형 번호판 데이터의 비율을 1대 2로 설정한 데이터 셋이다. 그리고 image_set_2의 인식 성능이 가장 높은 데이터 셋은 7번으로 3차원 회전하여 인조 번호판을 생성하고, 일반 번호판 데이터와 균형 번호판 데이터의 비율을 2대 1로 설정한 데이터 셋이다. 3차원으로 회전하여 생성한 데이터 셋의 경우 image_set_1의 인식 결과가 2차원으로 회전하여 생성한 데이터 셋과 비슷하거나 약간 떨어지는 것을 볼 수 있지만 image_set_2의 인식 결과는 2차원으로 회전하여 생성한 데이터 셋보다 성능이 뛰어난 것을 볼 수 있다. 또한 데이터 셋이 일반 번호판으로만 이루어진 경우 글자 클래스가 숫자 클래스 보다 인식률이 좋지 않았고, 균형 번호판으로만 이루어진 경우 숫자 클래스가 글자 클래스보다 인식률이 좋지 않았다. 2차원 회전하여 생성한 데이터 셋의 경우 일반 번호판 이미지와 균형 번호판 이미지의 비율이 1대 2인 4번 데이터 셋의 번호판 인식 성능이 가장 높았고, 3차원 회전하여 생성한 데이터 셋의 경우 일반 번호판 이미지와 균형 번호판 이미지의 비율이 2대 1인 7번 데이터 셋의 번호판 인식 성능이 가장 높았다.

5. 결론

본 논문에서는 다각도로 촬영된 자동차 번호판 영상의 인식률을 보완하기 위해 3차원으로 회전한 인조 번호판을 생성하여 딥러닝 학습한 모델을 구현하였다. 또한 2차원 회전한 인조 번호판을 학습한 모델과 실제 번호판의 인식 성능을 비교 분석하였다. 제안 방법으로 학습하여 번호판을 인식할 시, 성능 테스트를 위해 준비한 test set 중 번호판의 정면 이미지뿐만 아니라 측면 이미지 인식에 효과가 있다는 점을 증명하였다. 하지만 번호판 이미지의 해상도가 너무 낮은 경우 인식 성능이 저하되었으며, 한글 클래스는 숫자 클래스에 비해 전체적으로 인식률이 좋지 않았다.

번호판 인식 시스템의 인식률을 개선하기 위해 향후 연구계획으로는 먼저 데이터 셋의 크기를 백만 장 단위로 늘리는 것이다. 총 300,000장의 데이터 셋으로 학습을 진행하였는데, 일반 번호판 데이터와 균형 번호판 데이터의 비율에 따른 학습 성능 결과를 바탕으로 최적의 비율로 데이터 셋을 재구성할 계획이다. 또한 번호판 생성 시 진행되는 이미지 블러의 범위를 조정하여 해상도가 낮은 이미지도 인식할 수 있도록 보완할 것이다. 그리고 번호판을 회전시킨 후 각 문자의 파라미터를 텍스트 파일로 저장하는데 한글 이미지의 경우 숫자에 비해 면적이 넓어 회전 시 이웃해 있는 문자도 포함하는 경향이 있어 파라미터가 적절하게 적용되도록 조절할 계획이다. 마지막으로 최적의 상태로 번호판을 인식할 수 있도록 입력 이미지의 전처리 및 후처리 방법을 분석하여 번호판 인식 성능 향상 연구를 진행할 예정이다.

6. 참고문헌

[1] 이재현, 조성만, 이승주, 김청화, 박구만, “인조 데이터를 활용한 차량 번호판 인식 시스템,” 전자공학회논문지, 57(1), 107-115, January, 2020.
 [2] Joseph Redmon, Ali Farhadi, “YOLOv3: An Incremental

Improvement,” April, 2018, arXiv:1804.02767.
 [3] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi, “You Only Look Once: Unified, Real-Time Object Detection”, May, 2016, arXiv:1506.02640v5.
 [4] openFrameworks, *Wikipedia*, November, 2018

표 3. 번호판 인식 정확도 측정 결과

테스트 영상	데이터 셋 No.	회전 구분	Ground-truth	Detection results		Mean log-average miss rate	mAP	FPS
				True positive	False positive			
image_set_1	1	2차원 회전	8,689	7,434	505	0.18	79.21	48.85
	2			7,365	471	0.19	74	48.64
	3			8,102	776	0.12	84.24	47.82
	4			8,173	685	0.08	85.14	45.28
	5	3차원 회전		7,356	696	0.21	75.93	47.48
	6			7,252	673	0.21	73.65	46.38
	7			7,796	609	0.18	80.7	42.62
	8			7,368	475	0.19	75.01	45.91
	9			7,915	663	0.15	82.1	45.65
	10			6,574	445	0.23	72.55	43.99
image_set_2	1	2차원 회전	3,830	3,510	341	0.23	79.03	47.54
	2			3,594	328	0.21	80.42	47.53
	3			3,621	124	0.15	83.04	47.74
	4			3,647	99	0.12	85.32	48.03
	5	3차원 회전		3,476	445	0.25	76.65	47.93
	6			3,526	336	0.18	83.15	48.13
	7			3,750	109	0.08	92.42	47.76
	8			3,673	286	0.2	84.54	47.71
	9			3,619	107	0.11	90.66	46.78
	10			3,576	328	0.2	82.55	48.21