

깊이정보를 이용한 영상주차유도시스템 카메라 자동 보정

김 대 진*

*동국대학교 영상문화콘텐츠연구원 조교수

The Automatic Calibration of a Video Parking Guidance System Using Depth Information

Dae-Jin Kim*

*Assistant Professor, Research Institute for Image & Cultural Contents, Dongguk University, Seoul 04626, Korea

[요 약]

최근 주차공간의 효율적 관리를 위해서 영상주차유도시스템이 점점 보급화 되고 있다. 단순히 주차장소를 찾기 위하여 사용되기도 하지만, 인식기술을 이용하여 차량 운전자가 직접 본인이 주차시킨 곳을 찾는 서비스까지 연동되기도 한다. 이 때 카메라 입력력을 통해서 주차여부인식 및 차량 번호 인식 등을 진행하기 때문에 원본 영상의 화질을 시간이 지나도 그대로 유지 하는 것이 중요하다. 그러나, 주차장의 경우 차량진입에 의한 진동이 많이 발생하고, 시간이 지남에 따라 장비가 비틀어지거나, 물리적인 충격에 의해서 카메라 초기 설정값이 틀어질 수 있기 때문에, 카메라 자동보정(Auto-Calibration)을 통해서 지속적인 성능 유지하는 것이 필요하다. 따라서 본 논문에서는 자동 보정 시 Monodepth를 이용하여 단일 영상에서 깊이(Depth)정보를 추출하고, 이를 통하여 일정 거리를 유지할 수 있으며, 카메라설정을 조정하여 객체인식과 번호인식의 정확도를 초기설정만큼 유지 하여 시간이 지남에 따라 유지보수가 발생 할 수 있는 사항들을 자동화 할 수 있다.

[Abstract]

Recently, video parking guidance systems have been becoming increasingly popular for the efficient management of parking spaces. Although they are simply used as a guide for finding a place to park, they are also linked to a parked vehicle search service to find a place where a vehicle driver is parked by using recognition technology. As parking occupation recognition and license plate recognition occur through camera input, it is important to maintain the quality of original camera input over time. However, in the case of parking lots, vibration occurs frequently and the equipment may become distorted over time, while the initial setting value of the camera may change due to physical impulse; thus, continuous performance maintenance is required through camera auto-calibration. Therefore, in this paper, Monodepth is used for automatic correction to extract depth information from a single image, which enables the maintenance of not only a certain distance between camera and object but also the accuracy of both object and number recognition by adjusting the camera settings. Hence, it was possible to maintain initial accuracy while automating maintenance.

색인어 : 영상주차유도시스템, 모노덱스, 자동보정, 주차여부인식, 객체인식**Keyword** : Video Parking Guidance System, Monodepth, Auto-Calibration, Parking Occupation Recognition, Object Detection<http://dx.doi.org/10.9728/dcs.2023.24.9.2139>

This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 30 June 2023; Revised 03 August 2023

Accepted 04 August 2023

*Corresponding Author; Dae-Jin Kim

Tel: +82-2264-0160

E-mail: sampoo00@dongguk.edu

I. 서론

차량이 증가하고 일반화됨에 따라, 효율적인 주차를 위한 서비스들이 생겨나고 있다. 가장 일반화된 서비스는 차량번호인식(LPR : License Plate Recognition)기술로 주차문화를 바꾸는 계기가 되고 있다. 또한 최근에는 주차공간의 효율적 관리를 위해서 주차유도시스템이 점점 보급화 되고 있다[1]. 주차유도시스템은 주차위치를 찾아서 차량 운전자가 빈 주차면에 편하고 빠르게 주차가능 하도록 유도 하고, 주차 위치를 안내함으로써 편리성을 극대화하며, 무인 정산기로 요금계산을 유도하여 출구정산을 하는 차량을 효율적으로 관리할 수 있게 하는 시스템이다[2]. 예전에는 초음파센서를 이용하여 빈 주차영역 찾기를 목적으로 한 시스템이었으나, 요즘은 그림 1과같이 주차장에서 주차된 내차의 위치까지 찾을 수 있는 서비스가 가능한 영상 기반의 주차유도시스템이 대중화 되고 있다.

영상주차유도시스템의 경우 카메라 입력을 통해서 주차여부인식 및 차량 번호 인식 등이 일어나기 때문에 원본 영상의 화질을 시간이 지나도 그대로 유지 하는 것이 중요하다. 그러나 시간이 지남에 따라 차량 진입에 따른 진동에 의해서 초점이 흐려지거나, 노후에 의해 장비가 비틀어지거나, 물리적인 충격에 의해서 카메라 초점이 틀어 질 수 있기 때문에 카메라의 입력상태를 초기와 같이 유지하는 것이 불가능하다. 따라서 본 논문에서는 카메라로부터 받은 단일 영상 입력을 Monodepth를 이용하여 깊이(Depth)정보를 추출하고, 이를 통해 카메라 자동보정(Auto-Calibration)을 이용하여 일정한 카메라 설정을 유지할 수 있는 방법에 대하여 연구하였다. 논문의 구성은 2장에서 영상주차유도시스템의 가장 핵심 부

분인 Edge WAS(Web Application Server) 및 객체인식(Object Detection) 알고리즘에 대해 분석하고, 시스템 운영 시 발생가능한 문제점에 대해 알아본다. 3장에서는 객체의 깊이 정보를 얻기 위해 Monodepth를 적용해 보고, 이를 객체인식과 차량번호인식에 연동하여 자동보정 할 수 있는 방법을 연구하였다. 4장에서는 제안한 알고리즘을 이용하여 구현해 보고, 주어진 환경에서의 성능비교를 하였으며, 5장에서 결론을 지었다.

II. 영상유도시스템 구현

그림 1과 같은 구성을 가지는 영상주차유도시스템은 크게 객체인식, 차량 모니터링, 카메라 관리, 차량번호인식, Parking Slot 영역설정, 만공등 제어, 차량 사진 저장 등이 기술적 요소이다. 이중 Edge WAS가 가장 핵심 기술요소이며, 그 역할은 아래와 같다.

- 1) Camera 그룹 & Parking Slot & 만공등 관리 제어
- 2) Camera로 부터 이미지 습득
- 3) 영상처리
 - 차량, 사람 객체인식
 - Parking Slot 재부재 판단
- 4) LPR 연동
- 5) 만공등 관리 및 push 제어
- 6) 카메라, 주차, 안내판 및 서버 정보 설정정보 업데이트
- 7) 서버 리소스 관리(CPU, Memory, Bandwidth):모니터링 틀에서 API(Application Programming Interface) 호출

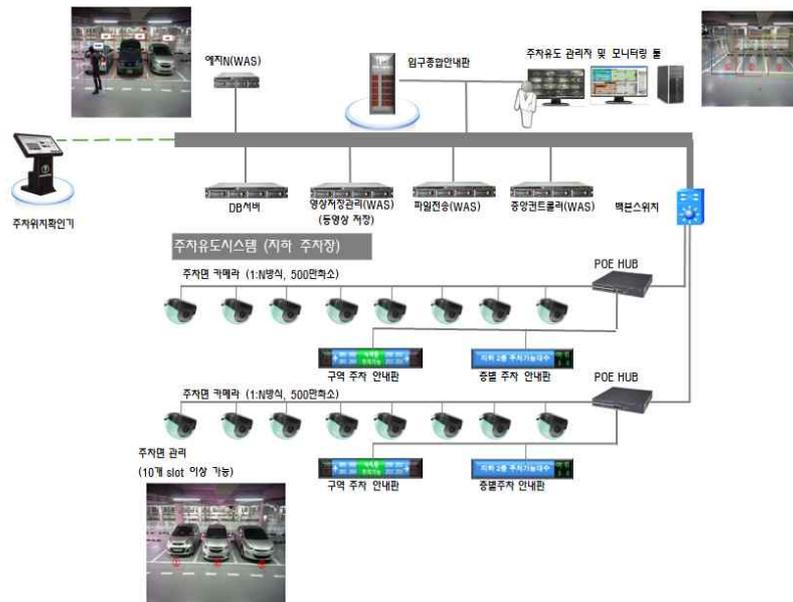


그림 1. 영상기반의 주차유도시스템 구성도
 Fig. 1. Parking guidance system architecture based camera

- 8) 중앙컨트롤러 WAS로 Parking Slot 상태정보 (Occupation, Plate Number) push 전달
- 9) 영상처리 + LPR 사용한 이미지 저장

이 중 핵심 기술 요소는 차량 재부재 판단을 위한 객체인식이며, 서비스 개발을 위해서 속도 및 정확도를 보장할 수 있는 YOLOv5(You Only Look Once) 알고리즘을 사용하여 자동차, 사람과 같은 객체 인식을 하였다.

2-1 YOLOv5를 이용한 객체

객체 인식률을 높이 가져가면서 동시에 검출 속도를 향상 시키려는 다양한 연구들이 시도되었다. 그 중 최근 가장 주목을 받은 방법인 One Stage Detector의 대표적인 모델인 YOLO알고리즘을 이용한 객체 인식으로, 네트워크는 최종 출력 단에서 BBox(Bounding Box)와 Class Object Index가 동시에 결과물로 나오도록 설계되었다[3],[4].

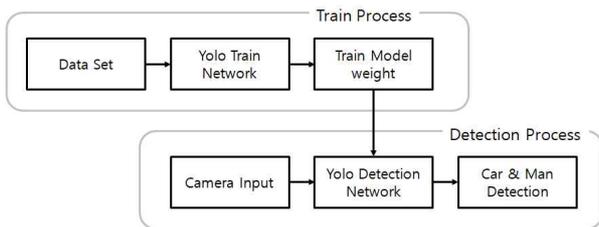


그림 2. YOLOv5을 이용한 객체 학습 및 인식 과정
Fig. 2. Object train&detection processing using YOLOv5

YOLOv5을 이용한 객체 학습 및 인식과정은 그림 3와 같다. 주차장내에 인식해야 될 객체를 자동차와 사람으로 2가지로 Class를 구성하였으며, YOLOv5의 Custom Object 학습 방법을 통해서 2가지(자동차, 사람) 객체로 학습하였다[5].

그림 3에서는 주차장내 YOLOv5을 이용한 객체인식 결과를 보여준다.

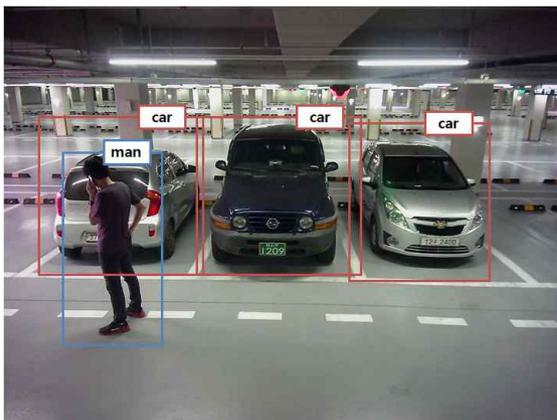


그림 3. 주차장내 객체인식
Fig. 3. Object detection using at parking slots

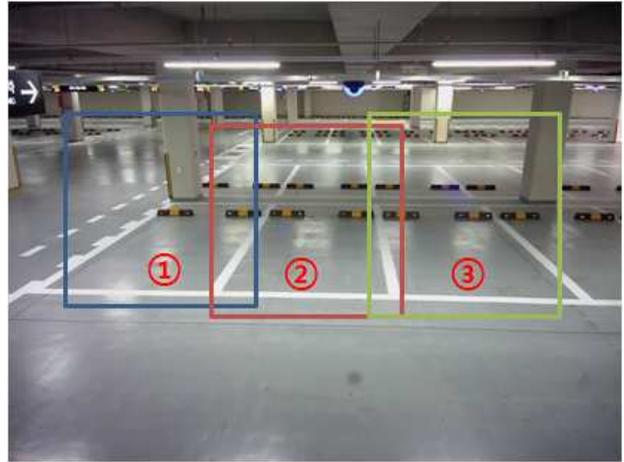


그림 4. 객체인식을 위한 주차 영역 설정
Fig. 4. Slots setting for object detection

주차면에 주차여부를 확인하기 위해서 그림 4처럼 각 주차 영역마다 객체탐지영역을 미리 설정하였다. 영역을 설정할 때 실제 주차된 차량이 카메라 각도에 따라서 주차면 일부를 차지할 수 있으므로 양 옆의 주차면 까지 충분히 겹치도록 객체 탐지 영역을 설정해야 정확한 주차여부인식 결과가 나올 수 있다.

주차여부판단은 설정된 인식영역대비 차량 객체 인식영역의 IOU(Intersection Over Union) 75%이상을 차지하면 주차가 된 것으로 본다. 여기서 IOU 75%는 실험을 통한 경험적인 값으로 주차장 환경에 따라 초기 설정값을 결정하면 된다[6].

2-2 기존 시스템의 문제점 제시

모든 주차장 Slot 위에 카메라를 설치하고 그림 1과 같은 시스템을 구성하여 서비스를 구성할 수 있다. 그런데, 주차장 내에는 시간이 지남에 따라 초기 카메라 설정들을 변화 시키는 많은 요소들이 발생한다. 진동에 의해서 초점이 흐려지거나, 장비가 비틀어질 수 있으며, 물리적인 충격에 의해서도 카메라 초기 설정값이 틀어질 수 있기 때문에, 시간이 지나면 카메라 자동보정은 필수적인 요소가 된다. 이를 위해 본 논문에서는 하나의 영상으로부터 깊이정보를 추출하기 위해 Monodepth를 이용한 깊이정보추출을 사용하였고, 이를 이용한 카메라 자동보정으로 시간이 지나도 일정한 카메라 설정을 유지 할 수 있도록 하였다.

III. 제안하는 영상유도시스템에서 카메라 자동보정 (Auto-Calibration)

여러 대의 카메라로부터 입력 받은 영상으로 부터 거리를 측정할 수 있으나, 최근에 들어서는 단일 영상에서도 딥러닝을 이용한 거리 측정이 가능하고, 성능 고도화가 되고 있는

상황이다. 본 논문에서는 Monodepth[7]을 이용하여 깊이 정보를 측정하고, 이를 이용하여 지속적으로 초기 설정된 카메라 설정값을 유지할 수 있다.

3-1 Monodepth

1) 이미지 복원을 통한 깊이 측정

단일 입력 이미지 I^l 가 주어 졌을 때 깊이를 나타내는 함수를 만들어 낼 수 있다. 양안 카메라가 센서로 부터 이미지를 입력 받는 것처럼 하나의 이미지로부터 보정된 다른 이미지를 재구성 할 수 있는 함수를 구할 수 있다면 이미지화된 3D 형태 정보를 가져올 수 있다. 깊이 정보를 예측하기에 앞서, 왼쪽 이미지로부터 밀집정합(Dense Correspondence)을 구하고, 이를 기반으로 오른쪽 이미지를 재구성 할 수 있다. 왼쪽 이미지도 마찬가지로 오른쪽 이미지로부터 재구성이 가능하다.

$$\tilde{I}^r = I^l(d^r) \tag{1}$$

$$\tilde{I}^l = I^r(d^l) \tag{2}$$

d : Disparity Map

2) 딥러닝을 통한 깊이예측

카메라의 양안시차의 정보를 이용하여 깊이 정보를 가져올 수 있지만, 왼쪽 이미지와 오른쪽 이미지의 왜곡의 차이를 통해서 깊이 정보를 예측할 수 있다. 이때 딥러닝 네트워크를 구성하여 Disparity map을 예측할 수 있고 이미지 재구성이 가능하다. 그림 5에서는 네트워크 모델을 통해서 이미지 재구성을 위한 과정을 보여준다.

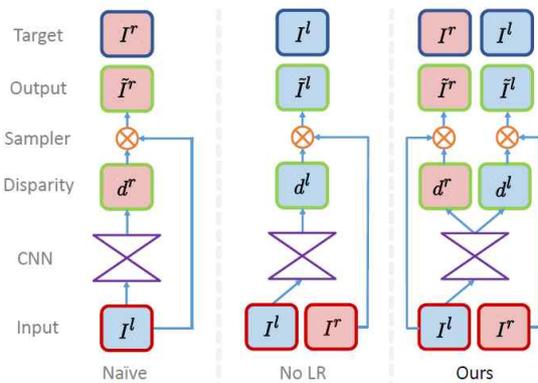


그림 5. 이미지 mapping을 위한 sampling 과정[7]
 Fig. 5. Sampling process for backward mapping[7]

Disparity map을 생성할 때 Convolution Layers들을 통해서 생성 가능한데 이때 사용한 Model Architecture는 그림 6과 같다. 네트워크는 크게 Encoder와 Decoder로 나뉘

고, conv1부터 conv7b는 Encoder파트 이고, upconv7부터 disp1은 Decoder 이다. 결과 값으로 4개의 다른 scale에 해당하는 Disparity (disp4~disp1)를 생성할 수 있고, 이때 네트워크 구성을 통해서 결과 값으로 입력 이미지의 2배의 해상도를 가지는 2개의 Disparity map을 구성할 수 있다(left to right, right to left).

"Encoder"						"Decoder"					
layer	k	s	chns	in	out	layer	k	s	chns	in	out
conv1	7	2	3/32	1	2	left					
conv1b	7	1	32/32	2	2	conv1	upconv7	3	2	512/512	128
conv2	5	2	32/64	2	4	conv1b	conv7	3	1	1024/512	64
conv2b	5	1	64/64	4	4	conv2	upconv6	3	2	512/512	64
conv3	3	2	64/128	4	8	conv2b	conv6	3	1	1024/512	32
conv3b	3	1	128/128	8	8	conv3	upconv5	3	2	512/256	32
conv4	3	2	128/256	8	16	conv3b	conv5	3	1	512/256	16
conv4b	3	1	256/256	16	16	conv4	upconv4	3	2	256/128	16
conv5	3	2	256/512	16	32	conv4b	conv4	3	1	128/128	8
conv5b	3	1	512/512	32	32	conv5	disp4	3	1	128/2	8
conv6	3	2	512/512	32	64	conv6	upconv3	3	2	128/64	8
conv6b	3	1	512/512	64	64	conv6b	conv3	3	1	130/64	4
conv7	3	2	512/512	64	128	conv7	disp3	3	1	64/2	4
conv7b	3	1	512/512	128	128	conv7b	upconv2	3	2	64/32	4
							conv2	3	1	66/32	2
							disp2	3	1	32/2	2
							upconv1	3	2	32/16	2
							conv1	3	1	18/16	1
							disp1	3	1	16/2	1
											conv1

k : kernel size
 s : stride
 $chns$: Number of input and output channels of each layer

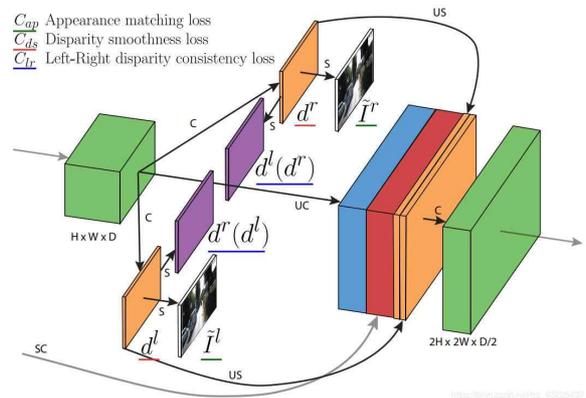
그림 6. Disparity map 생성을 위한 convolution layer(encoder, decoder) 모델 구성[7]

Fig. 6. Model architecture for creation disparity map by convolution layer (encoder, decoder)[7]

3) 손실함수

학습을 진행할 때 그림 7과 같이 3가지 기준의 손실 모듈을 계산할 수 있고, 이들 모듈의 합을 통해서 손실함수를 만들 수 있다.

손실함수 scale s 에 해당하는 손실 C_s 를 통해서 전체 손실값을 계산할 수 있다.



d^l : Left Disparity map
 d^r : Right Disparity map
 C : Convolution, UC: Up-Convolution,
 S : Bilinear Sampling, US: Up-Sampling, SC: Skip Connection.

그림 7. 평탄필터, 이미지 복원, 좌/우 disparity를 이용한 손실조합

Fig. 7. The loss combines (smoothness, reconstruction, and left-right disparity consistency)

$$C = \sum_{s=1}^4 C_s \quad (3)$$

$$C_s = \alpha_{sp}(C_{ap}^s + C_{ap}^r) + \alpha_{ds}(C_{ds}^l + C_{ds}^r) + \alpha_{lr}(C_{lr}^l + C_{lr}^r) \quad (4)$$

C_{ap} : 학습 이미지와 복원된 이미지 사이 손실값

C_{ds} : 평탄필터를 통한 Disparity 손실값

C_{lr} : 좌우 이미지의 Disparity 손실값

• 학습 이미지와 복원된 이미지 사이 손실값(Appearance Matching Loss)

Convolution Architecture를 이용한 학습으로 이미지복원(Image Reconstruction)이 이루어 졌고, 이 때 사용한 손실함수는 아래와 같다.

$$C_{ap}^l = \frac{1}{N} \sum_i \alpha \frac{1 - SSIM(I_i^l, \tilde{I}_i^l)}{2} + (1 - \alpha) \| I_i^l - \tilde{I}_i^l \| \quad (5)$$

I : 입력 이미지

\tilde{I} : 생성 이미지

이때 $SSIM$ (Structural Similarity Index)[8]을 이용하여, Photometric 픽셀의 유사정도를 측정할 수 있다. 또한, $\alpha = 0.85$ 로 설정하여 손실함수에 적용하였다. 이 값은 실험을 통한 경험적인 값을 설정한 것이다.

• 평탄필터를 통한 Disparity 손실값(Disparity Smoothness Loss)

Disparity의 Local Smooth를 위하여, Disparity 기울기(gradient) ∂d 와 이미지 기울기 ∂I 를 사용한 Edge와 관련된 손실함수 식 (6)을 사용하였다.

$$C_{ds}^l = \frac{1}{N} \sum_i | \partial_x d_i^l | e^{- \| \partial_x I_i^l \|} + | \partial_y d_i^l | e^{- \| \partial_y I_i^l \|} \quad (6)$$

• 좌우 이미지의 Disparity 손실값(Left-Right Disparity Consistency Loss)

학습 시에 Stereo 영상을 이용하기 때문에 식 (7)과 같이 왼쪽 영상의 Disparity인 d_i^l 와 오른쪽 Disparity인 d_{i+d}^r 의 값이 서로 같다는 정보를 활용하여 손실함수를 적용하였다.

$$C_{dc}^l = \frac{1}{N} \sum_i | d_i^l - d_{i+d}^r | \quad (7)$$

3-2 카메라 자동보정(Camera Auto-Calibration)

영상주차유도시스템은 하나의 IP(Internet Protocol)카메라가 주차면을 3~4면 정도를 바라보고 객체인식과 차량번호인식을 진행한다. 그림 8에서는 주차면에 카메라 설치 방법에

대해서 보여준다. 객체인식의 경우는 10개 이상의 주차면의 경우에도 인식이 가능하나, 차량번호인식은 번호판의 글자 및 숫자가 일정 크기가 되어야 인식이 가능하므로, 4K영상을 기준으로 했을 때 3~4면 정도를 인식하는 것이 일반적이다. IP 카메라 설치 시 사람이 일일이 초점을 조정하고, 영상을 확인하며, 재조정을 하게 된다. 영상주차유도가 설치되는 현장은 대부분 1000개 이상의 주차면에 적용되는 경우가 많기 때문에 사람이 직접 초점작업을 하게 되면, 인건비가 상승되고, 시간적인 낭비 및 정확도에 문제가 발생하게 된다. 또한 영상주차유도시스템이 설치된 현장은 차량이 많이 입/출입 한다. 실내 주차장의 경우 실외 보다는 환풍이 잘 안되기 때문에 1년 정도 지나면 매연에 의해서 오염물질이 카메라 렌즈에 묻고, 차량진동에 의해서 초점이 흔들리게 된다.

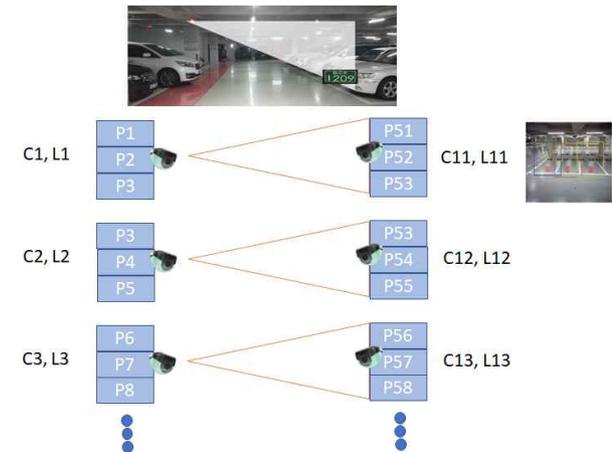


그림 8. 주차면 카메라 설치 방법
Fig. 8. Camera installation method in parking slots

따라서 매년 카메라를 세척하고, 흔들린 초점을 처음처럼 다시 보정작업을 해야 한다. 이때 또다시 인건비가 발생되고, 시간적인 낭비 및 정확도에 문제가 발생한다.

설치/유지보수의 문제점을 해결하기 위해서 초점 설정을 자동화 할 수 있는 연구가 필요하다. 이를 위한 제어 기술로 ONVIF(Open Network Video Interface Forum)[9]를 이용하여 IP카메라의 PTZ(Pan/Tilt/Zoom)제어가 필수적이다. 이때 사람이 하나씩 확인하면서 제어하기 보다는 자동보정을 통해서 이 부분에 대한 문제점 해결이 가능하다. 자동보정을 위해서 자동차 자율주행에서 사용되는 기술인 Monodepth 알고리즘을 이용하며, 객체깊이정보를 추출하여 카메라와 차량 간 거리를 유지할 수 있고, 해당 거리에서 초점 설정을 함으로써 차량 번호인식 성능을 일정하게 유지할 수 있다. 즉, IP카메라의 ONVIF제어 기술과 Monodepth 알고리즘을 이용하여 객체의 깊이를 측정하고 카메라의 자동보정을 적용할 수 있다. 그림 9에서는 Monodepth를 이용하여 객체 깊이를 측정된 영상을 보여준다. 이때 KITTI 2015[10] 데이터 셋을 사용하였다.



그림 9. Monodepth를 이용한 object depth 측정
 Fig. 9. Object depth prediction using monodepth

3-3 Object Detection 및 LPR 연동

Pre/Post-Processing[6]을 이용하여 객체인식 및 차량 번호의 인식률을 향상시킬 수 있다. 입력영상의 인식률 개선을 위하여 기존 Edge 추출 알고리즘(Sobel, Watershed, Gradient 등) 분석 및 AI 기반의 개선된 Edge 추출 알고리즘을 통하여 빛/객체/시설물 간섭에 적응적 처리를 할 수 있다. 또한 기존 Dehaze 인공지능 Model(DehazeNet, DCP, FAST, CLAHE 등) 분석 및 AI 기반의 개선된 Dehaze 알고리즘을 이용하여 빛/객체/시설물 간섭에 적응적 처리를 할 수 있다. 이는 환경에 따라 적응적으로 적용되어야하며, 본 논문의 테스트환경에서는 Pre-Processing으로 Sobel Filter와 DehazeNet을 사용하였다.

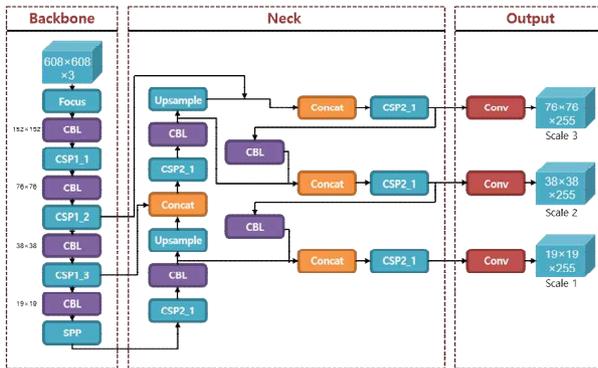


그림 10. YOLOv5 구성도
 Fig. 10. YOLOv5 architecture

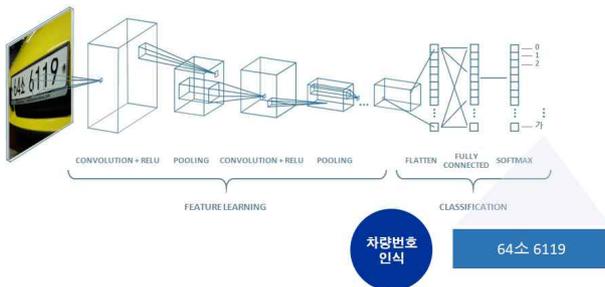


그림 11. CNN Model을 이용한 차량번호인식
 Fig. 11. LPR using CNN model

영상처리를 통해서 개선된 입력영상은 객체인식과 차량번호인식의 성능을 높이는 것을 목표로 한다. 그림 10에서는 YOLOv5 네트워크를 이용하여 객체인식을 진행하였다. 알고리즘을 살펴보면 크게 Backbone, Neck, Output 부분으로 구성된다. Backbone 역할은 입력 영상에서 Feature 정보를 추

출하고, Neck에서는 추출된 Feature 정보를 융합하고 특징 맵의 3개의 축적을 생성한다. 그리고 Output에서는 생성된 기능 맵에서 객체 탐지를 하도록 구성된다.

그림 11과 같은 CNN(Convolution Neural Network) 모델을 통해 차량 번호 인식을 진행하였다. Convolution과 Pooling Layer를 각각 2번씩 거쳐서 Flatten 된 정보를 기반으로 차량번호인식에 사용하였다[11].

IV. 제안 시스템 구현 결과

4-1 시스템 구현환경

영상주차유도 시스템을 구성해 놓고, 초기에 설정된 카메라 셋팅을 그대로 데이터베이스에 기록해 두었다. 진동 등의 변화에 의해서 카메라 설정이 변화 된 것으로 가정하고 테스트하기 위해 5대의 카메라에 대해서 임의로 초점을 흐리게 하고, Zoom In/Out을 통해서 기준 거리 변화를 주었다. 차량과 카메라의 기준거리는 약 5m로 초기에 설정되어 있으며, 변화된 환경은 Zoom In/Out에 의해서 기준 거리와 차이를 발생 시켰다. 카메라로부터는 4096x2016 해상도의 이미지를 습득하였으며, 1개의 카메라에서 3~4개의 주차면을 바라보고 있다. 초점의 기준을 설정할 때 3개의 주차면에서는 2번째 Parking Slot에 초점 기준을 정했고, 4개의 주차면인 경우 3번째 Parking Slot에 초점 기준을 두었다. 16개의 주차면을 3일 동안 테스트 진행했으며, 이들 주차면에 주차했던 차량은 총 278대가 사용되었다. 1일차에는 초기설정환경에서 테스트를 진행하였고, 2일차에는 초점변경 환경에서 테스트를 진행하였으며, 3일차에는 자동보정 후 환경에서 테스트를 진행하였다. 자동 보정 과정은 Zoom In/Out, 초기 설정된 Depth값에 의한 초점설정, LPR 인식(95% 이상)을 반복하면서 카메라 설정값을 보정하는 과정을 가졌다. Edge WAS 장비는 OS가 64bit Windows10 환경에서 Intel i5-7500 CPU@3.40GHz, 32Gbyte의 메모리를 사용하였으며, YOLOv5 객체인식시 GPU를 사용하였고, 그래픽카드를 RTX3080TI를 사용하였다. Edge WAS개발 시 WAS는 Visual Studio 2022 C++로 개발 하였다. 그림 12에서는 실제 테스트를 위해 개발한 프로그램의 GUI 및 적용 예를 보여준다.



그림 12. 영상유도시스템 GUI 적용 프로그램
 Fig. 12. Video parking guidance system GUI application

4-2 시스템 성능비교

표 1에서는 주어진 환경을 초기설정, 초점변경, 자동보정의 환경으로 구분하였을 때 차량번호판에 대한 번호인식률의 Accuracy와 차량번호판/차량 객체인식의 AP(Average Precision)를 조사하였고, 이때 IOU를 0.50로 하였다.

표 1. 성능비교

Table 1. Performance comparison

	Accuracy(%) (LPR)	AP@.50(%) (Plate)	AP@.50(%) (Car)
Initial focus	97.5	98.3	98.6
Changed focus	30.1	32.9	49.4
Auto-focus	97.1	98.8	98.6

기존의 초기설정에 의해서 서비스 시작 시 최적화된 값으로 설정 되어 있는 환경이기 때문에 초기설정 환경에서는 Accuracy와 AP가 높은 결과 값을 가진다. 그러나, 장비노후화, 차량진동 등 외부 환경에 의해 초점 변경되었다는 가정 환경아래에서는 Accuracy와 AP가 급격히 낮은 값을 가졌다. 그러나, Monodepth를 이용하여 깊이 정보를 추정하고 초기 설정값에 유사하게 예측하여 자동보정 후 Accuracy와 AP를 측정 하였을 때, 초기설정환경과 거의 유사한 결과를 얻을 수 있었다. 따라서 자동 보정기능을 이용하여 일정한 카메라 환경을 유지하면서, 인식률을 유지할 수 있었다.

V. 결 론

본 논문에서는 영상주차유도시스템에서 시간이 지남에 따라 진동에 의해서 초점이 흐려지고, 장비가 비틀어지거나, 물리적인 충격에 의해서 카메라 초점이 틀어지면서 카메라 입력영상의 질이 나빠질 수 있기 때문에, 카메라 자동 보정을 통한 지속적인 인식성능 유지하는 방법을 연구하였다. 영상유도시스템의 카메라 자동 보정을 위해서 위에서 제시한 실험 환경에서 테스트가 이루어 졌으며, Monodepth 알고리즘을 통해서 단일카메라에서 깊이 정보를 가져 왔으며, 계속해서 가져올 수 있었고, 차량이 주차되어 있는 지 알기위해서 YOLOv5 알고리즘을 이용하여 객체인식을 하였고, CNN 기반의 번호인식 엔진을 통해서 차량번호인식을 진행하였으며, 시간이 지나 초점이 지난 후에도 카메라의 일정 거리를 유지하고, 초점을 컨트롤을 하여 객체인식과 번호인식의 정확도를 초기설정 만큼 유지 할 수 있었다. 따라서, 시간이 지남에 따라 유지보수가 발생 할 수 있는 사항들을 자동화 하고, 인건비 및 시간적 비용 소비를 줄이면서 초기 정확도를 유지 할 수 있었다.

참고문헌

- [1] D.-J. Kim, "Implementation of Parking Management System Using Cloud Based License Plate Recognition Service," *Journal of Digital Contents Society*, Vol. 19, No. 1, pp. 173-179, January 2018. <https://doi.org/10.9728/dcs.2018.19.1.173>
- [2] L. Li, L. Zhang, X. Li, X. Liu, Y. Shen, and L. Xiong, "Vision-Based Parking-Slot Detection: A Benchmark and a Learning-Based Approach," in *Proceedings of 2017 IEEE International Conference on Multimedia and Expo (ICME)*, Hong Kong, China, pp. 649-654, July 2017. <https://doi.org/10.1109/ICME.2017.8019419>
- [3] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas: NV, pp. 779-788, June 2016. <https://doi.org/10.1109/CVPR.2016.91>
- [4] Y. Zhang, Z. Guo, J. Wu, Y. Tian, H. Tang, and X. Guo, "Real-Time Vehicle Detection Based on Improved YOLO v5," *Sustainability*, Vol. 14, No. 19, 12274, September 2022. <https://doi.org/10.3390/su141912274>
- [5] Github. Custom Object Train [Internet]. Available: <https://github.com/ultralytics/yolov5/wiki/Train-Custom-Da-ta>.
- [6] D.-J. Kim, C.-P. Yoon, and C.-G. Hwang, "Deep Learning Based Parking Occupation Detection Using Pre/Post-Processing," *Journal of Digital Contents Society*, Vol. 20, No. 10, pp. 2087-2096, October 2019. <https://doi.org/10.9728/dcs.2019.20.10.2087>
- [7] C. Godard, O. Mac Aodha, and G. J. Brostow, "Unsupervised Monocular Depth Estimation with Left-Right Consistency," in *Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu: HI, pp. 6602-6611, July 2017. <https://doi.org/10.1109/CVPR.2017.699>
- [8] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image Quality Assessment: From Error Visibility to Structural Similarity," *IEEE Transactions on Image Processing*, Vol. 13, No. 4, pp. 600-612, April 2004. <https://doi.org/10.1109/TIP.2003.819861>
- [9] ONVIF. Profile S [Internet]. Available: <https://www.onvif.org/profiles/profile-s>.
- [10] A. Geiger, P. Lenz, and R. Urtasun, "Are We Ready for Autonomous Driving? The KITTI Vision Benchmark Suite," in *Proceedings of 2012 IEEE Conference on Computer Vision and Pattern Recognition*, Providence: RI, pp. 3354-3361, June 2012.

<https://doi.org/10.1109/CVPR.2012.6248074>

- [11] R. Rattanakornphan, P. Keeratiwintakorn, and T. Ruttanapunyagorn, "Thai License Plate Recognition Based on YOLO Detector and CNN-Classifer," in *Proceedings of 2022 6th International Conference on Information Technology (InCIT)*, Nonthaburi, Thailand, pp. 331-336, November 2022.

<https://doi.org/10.1109/InCIT56086.2022.10067311>



김대진(Dae-Jin Kim)

1998년 : 대진대학교 전자공학과(공학사)

2000년 : 동국대학교 대학원 전자공학과(공학석사)

2010년 : 대진대학교 대학원 전자공학과(공학박사)

2017년~현재 : 동국대학교 영상문화콘텐츠연구원 조교수

※ 관심분야 : 동작인식, 이상탐지, 얼굴인식, 콘텐츠 DNA, 워터마크, 딥러닝, 번호인식, 자율주행 등