

YOLOv5에서 자동차 번호판 및 문자 정렬 알고리즘에 관한 연구

A Study on Vehicle License Plates and Character Sorting Algorithms in YOLOv5

장문석¹, 하상현², 정석찬^{2*}

Mun-Seok Jang¹, Sang-Hyun Ha², Seok-Chan Jeong^{2*}

〈Abstract〉

In this paper, we propose a sorting method for extracting accurate license plate information, which is currently used in Korea, after detecting objects using YOLO. We propose sorting methods for the five types of vehicle license plates managed by the Ministry of Land, Infrastructure and Transport by classifying the plates with the number of lines, Korean characters, and numbers. The results of experiments with 5 license plates show that the proposed algorithm identifies all license plate types and information by focusing on the object with high reliability score in the result label file presented by YOLO and deleting unnecessary object information. The proposed method will be applicable to all systems that recognize license plates.

Keywords : Vehicle License Plate, YOLO(You Only Look Once), LPR(License Plate Recognition), Character Recognition, Object Detection

1 1저자, 동의과학대학교 AI전자과
E-mail: msjang@dit.ac.kr

2 동의대학교 인공지능그랜드ICT연구센터

2* 교신저자, 동의대학교 e비즈니스학과, 인공지능그랜드
ICT연구센터 E-mail: scjeong@deu.ac.kr

1 Dept. of AI Electronic Engineering, Dong-Eui Institute of Technology
(First Author)

2 AI Grand ICT Research Center, Dong-Eui University

2* AI Grand ICT Research Center, Dept. of e-Business,
Dong-Eui University (Corresponding Author)

1. 서론

자동차 번호판 인식(License Plate Recognition)은 카메라와 제어장치, 조명 장치, 컴퓨터, 문자 인식 알고리즘 등으로 구성되어, 빠르고 정확하게 번호판을 인식하는 기술로 버스전용차로 위반이나 속도 위반, 불법 주정차 위반 단속 같은 단속용, 요금 정산을 목적으로 하는 주차 관제용, 주차서비스 질을 개선하기 위해 주차 유도 등에서 활용된다[1]. 이를 위해서 영상처리를 통해 선형 SVM(Support Vector Machine)을 이용해 번호판을 검출하고, CNN(Convolution Neural Network) 알고리즘을 사용하여 번호판의 글자와 숫자를 인식하거나[2], 광학 문자 인식(OCR(Optical Character Recognition))으로 영상에 포함된 글자를 문자로 변환하는 방법을 사용하고 있다[3]. 이러한 방법은 입력된 영상의 배경이 변하거나, 영상 왜곡 등에 의해서 번호판을 잘못 인식하는 경우가 많다.

최근에 YOLO(You Only Look Once)를 이용하여 객체를 검출(Object Detection)하는 많은 연구가 진행되고 있고[4-8], 최근에 v5까지 출시되어 있다[9]. 이것은 이미지 전체에 대해서 하나의 신경망이 한 번의 계산만으로 경계 박스(bounding box)와 클래스 확률(class probability)을 예측하는 시스템으로 R-CNN, Fast R-CNN, Faster R-CNN보다 정확도는 다소 떨어지나, 검출 속도가 빠른 장점이 있다[10-12]. YOLO를 이용하여 객체를 검출하기 위해서는 학습 데이터의 경로, 클래스 개수 및 종류를 관리하는 yaml 파일과 경계박스의 정보를 관리하는 라벨 파일을 제작해야 한다. 그리고 결과로 검출된 객체 클래스, 경계박스, 신뢰도 점수(confidence score) 등 다양한 정보를 얻을 수 있다.

한국의 자동차 번호판은 1972년형, 1973년형, 1996년형, 2004년형, 2005년형, 2006년형, 2018년형 등 1라인, 2라인과 같은 다양한 종류의 번호

판이 실제 사용되고 있고, 번호판의 종류 역시 비사업용(차종+용도+일련번호) 형식, 사업용(지역+차종+용도+일련번호) 형식 등 다양한 종류의 번호판이 사용되고 있다.

이러한 번호판 형식에 대해 YOLO를 이용하여 객체 검출 결과 데이터는 검출된 숫자와 한글 순서대로 출력이 되기 때문에 데이터를 순서에 맞게 정렬한다. 하지만 기존의 YOLO를 이용한 연구[5-7]들은 단순 배열만을 가지는 2006년 이후의 번호판에 대해서만 결과를 제시하고 있어 한국의 다양한 번호판 정보를 출력할 수 없다. 본 논문에서는 이러한 문제 해결하기 위해 YOLO의 객체 검출 결과를 이용하여 자동차 번호판 인식에 있어서 한국에서 사용되고 있는 다양한 자동차 번호판의 배열을 정렬하는 알고리즘을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 한국 자동차 번호판의 종류와 문자 배열의 구성에 대해 알아보고, 3장에서는 본 논문에서 제안하는 YOLO의 객체 검출 결과를 이용한 자동차 번호판 배열 알고리즘을 설명하고, 4장에서는 실험 결과를 분석하고, 5장에서는 결론을 맺는다.

2. 한국의 자동차 번호판의 문자 배열 규칙

한국의 자동차 번호판은 1973년 교통부 고시 제10호를 시작으로 2019년에 이르기까지 수많은 개정을 거쳐 현재의 번호판 체계에 이르고 있다[13]. Fig. 1은 한국자동차 번호판의 체계를 나타



Fig. 1 Structure of Korean license plate

Table 1. Classification of Korean license plates (Except for diplomatic use)

구분	분류		기호
차의종류	승용	비사업	100-699
		대여사업	
		일반사업	01-69
		승합	70-79
		화물	80-97
		특수	98, 99
차의용도	비사업	자가용	가, 나, 다, 라, 마, 거, 너, 더, 러, 머, 고, 노, 도, 로, 모, 구, 누, 두, 루, 무, 버, 서, 어, 저, 보, 소, 오, 조, 부, 수, 우, 주
			사업
		대여사업	허, 하, 호

낸다. 이것은 1995년 개정을 통해 지역, 차의 종류, 차량의 용도, 일련번호를 나타낸다. 차의 종류는 2018년 개정을 통해 차의 종류를 3자리로 확대하여 적용하고 있다.

한국자동차번호판의 분류는 Table 1과 같이 앞 숫자는 승용차, 승합차, 화물차, 특수차 등의 차의 종류를 나타내고, 중간에 한글은 비사업용, 사업용(운수, 렌트카, 택배) 등의 차의 용도를 나타내고, 마지막으로 일련번호를 나타낸다[13]. 사업용의 경우 지역을 할 수 있는 문자가 삽입된다.

한국자동차 번호판의 형태는 Fig. 2와 같이 연도별 크기와 문자 폰트는 조금씩 다르지만 5가지 형태로 분류할 수 있다. Fig. 2(a),(b),(c)는 비사업용 번호판으로 숫자로 표현된 자동차 종류(2-3자리), 한글로 표현된 자동차 용도(1자리) 그리고 숫자로 표현된 일련번호(4자리)로 표현되며, Fig. 2(d),(e)는 사업용 번호판으로 한글 지역명(2자리), 숫자로 표현된 자동차 종류(2자리), 한글로 표현된 자동차 용도(1자리) 그리고 숫자로 표현된 일련번호(4자리)로 표현된다.



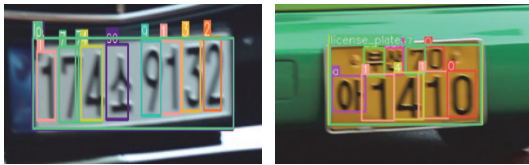
Fig. 2 Type of Korean license plate

호(4자리)로 표현된다. 여기에서 비사업용 번호판은 숫자는 6 또는 7글자, 한글은 1글자를 가지고, 1 또는 2줄로 표현됨을 알 수 있다. 사업용 번호판은 숫자는 6글자, 한글은 3글자를 가지고 1 또는 2줄로 표현됨을 알 수 있다.

3. YOLO의 객체 검출 결과를 이용한 자동차 번호판 배열 알고리즘

3.1 YOLO에서 객체 검출 결과 분석

YOLO를 이용하여 객체를 검출하기 위해서는 학습 데이터의 경로, 클래스 개수 및 종류를 관리하는 yaml 파일을 제작하고, 전체 이미지에서 경계박스의 중심점과 너비, 높이의 비율 정보를 관리하는 라벨 파일을 제작해야 한다. 그리고, YOLO를 이용하여 데이터를 학습시키고, 학습 결과로 검출된 객체 클래스, 경계박스, 신뢰도 점수(confidence score) 등 다양한 정보를 얻는다.



(a) 비사업용 번호판 인식 결과 (b) 사업용 번호판 인식 결과

Fig. 3 License plate recognition result using YOLO

Table 2. Fig. 3(a) recognition result

객체 번호	중심점 x위치	중심점 y위치	너비	높이	신뢰도 점수
7	0.344141	0.509028	0.0414063	0.115278	0.27
36	0.413281	0.508333	0.317187	0.138889	0.29
1	0.276953	0.511806	0.0289063	0.104167	0.33
1	0.473047	0.498611	0.0320312	0.111111	0.88
4	0.348828	0.507639	0.0335938	0.1125	0.89
9	0.442578	0.5	0.0304688	0.113889	0.89
7	0.312891	0.509722	0.0304688	0.113889	0.89
36	0.389062	0.505556	0.034375	0.113889	0.91
2	0.539453	0.493056	0.0273438	0.111111	0.94
3	0.505859	0.494444	0.0289063	0.111111	0.95
67	0.414453	0.507639	0.316406	0.1375	0.98

Table 3. Fig. 3(b)의 인식 결과

객체 번호	중심점 x위치	중심점 y위치	너비	높이	신뢰도 점수
4	0.530469	0.521528	0.226562	0.201389	0.43
59	0.513281	0.463194	0.028125	0.068056	0.46
0	0.572656	0.457639	0.028125	0.0625	0.71
7	0.545313	0.460417	0.028125	0.068056	0.76
45	0.483594	0.463889	0.0296875	0.069444	0.83
1	0.571484	0.547222	0.0429688	0.119444	0.87
4	0.535547	0.55	0.0429688	0.119444	0.88
0	0.613281	0.544444	0.0421875	0.116667	0.91
17	0.446875	0.544444	0.0484375	0.091667	0.91
1	0.489844	0.553472	0.0484375	0.118056	0.93
67	0.530859	0.521528	0.222656	0.201389	0.96

Fig. 3은 YOLO를 이용하여 자동차 번호판을 인식한 결과를 나타낸다. Fig. 3.(a)는 비사업용 번호판, Fig. 3. (b)는 사업용 번호판 인식 결과를 나타낸다. Fig. 3. (a)에서 인식에 경계박스에 대한 정보는 Table 2와 같이 표현되고, Fig. 3. (b)에서 인식에 경계박스에 대한 정보는 Table 3과 같이 표현된다. Table 2와 Table 3에서의 중심점 x 위치, 중심점 y 위치, 너비, 높이의 값은 전체 이미지 크기에 대한 비율의 값을 나타낸다. Table 2에서 첫 번째 줄 객체번호와 일곱 번째 줄 객체번호 ‘7’은 같은 객체를 나타내지만, 일곱 번째 줄의 신뢰도 점수가 0.89로 더 좋은 결과를 내기 때문에 이것을 선택한다. 그리고, 객체 정보 36는 ‘소’에 대한 결과이고, 67은 번호판 전체 영역을 나타낸다.

객체 정보의 신뢰도 점수를 고려하여 정렬 없이 인지된 순서대로 출력하면, Table 2는 “1497소23”을 나타내고, Table 3은 “산07부120아1”을 나타낸다. 이러한 정보를 올바르게 표현하기 위해서는 각 객체의 위치에 맞는 문자열 정렬이 필요하다.

3.2 자동차 번호판 위치정렬 알고리즘

Fig. 3의 자동차 번호판 인식 결과를 이용하여, Fig. 2의 자동차 번호판의 올바른 정보를 표현하기 위해서는 번호판의 형태별로 분류하고, 각 문자의 위치에 대한 정보를 위치 순서대로 정렬해야 한다.

본 논문에서 제안하는 자동차 번호판 위치정렬 알고리즘은 Table 2, Table 3과 같은 YOLO에서 객체 인식 결과를 이용하여 전처리 과정, 한글처리 과정, 숫자 처리 과정으로 나누어진다.

3.2.1 전처리

Fig. 4는 자동차 번호판 위치정렬을 위한 전처리 과정을 나타낸다. 이것은 자동차 번호판의 지역명 (FirstPosHan, LastPosHan)을 지정하고, YOLO에서 객

체 인식 결과(Table 2, Table 3의 리스트)를 입력으로 받고, 자동차 번호판의 체계를 저장하기 위한 변수 (Variable)를 지정한다. 그리고 신뢰도 점수가 적은 객체를 리스트에서 삭제한 후 숫자 정보만을 저장하는 NumList, 한글 정보만을 저장하는 HanList, 67의 객체 번호를 가지는 번호판으로 분류한다.

1. FirstPosHan = [서, 부, 대, 인, 광, 대, 을, 경, 강, 총, 홍, 전, 전, 경, 경, 제]
2. LastPosHan = [을, 산, 구, 천, 주, 전, 산, 기, 원, 북, 남, 북, 남, 북, 남, 주]
3. Input 번호판 객체 인식 리스트
4. Variable = 지역, 종류, 용도, 일련번호
5. 번호판 객체 인식 리스트를 좌표로 변환하여 같은 객체 번호, 위치를 계산하여 신뢰도 점수가 적은 객체 삭제
6. 번호판 객체 인식 리스트에서 숫자이면 숫자리스트 (NumList), 한글이면 한글리스트 (HanList), 번호판으로 분류

Fig. 4 Position alignment preprocessing

```

리스트의 객체 번호를 한글로 변환
IF HanList 갯수 >= 3 THEN
    index = 0
    FOR HanList의 모든 문자에 대해 DO
        IF HanList[index] == FristPosHan THEN
            지역[0] = HanList[index]
            HanList에서 index 문자 삭제
        ELSE
            Display "Error"
            index = index + 1
        index = 0
        FOR HanList의 모든 문자에 대해 DO
            IF HanList[index] == LastPosHan THEN
                지역[1] = HanList[index]
                HanList에서 index 문자 삭제
            ELSE
                Display "Error"
                index = index + 1
        index = 0
        용도 = HanList[index]
        Index = 0
        FOR 지역의 모든 문자에 대해 DO
            IF 지역[0] == FirstPosHan[index]
            and 지역[1] == LastPosHan[index] THEN
                Continue
            ELSE
                Display "Error"
                index = index + 1
        ELSE
            index = 0
            용도 = HanList[index]
    
```

Fig. 5 Korean processing process

3.2.2 한글처리

Fig. 5는 자동차 번호판 위치정렬을 위한 한글처리 과정을 나타낸다. 한글 정보는 지역명과 용도를 나타낸다. Table 3와 같이 한글의 개수가 3개일 경우에는 한글 리스트의 모든 문자를 조사해서 FristPosHan과 LastPosHan에 있는 문자의 위치 (index)를 찾은 다음, 지역과 FirstPosHan, LastPosHan의 같은 위치에 있는 지역명을 선택하고, 나머지 하나는 글자를 용도로 선택한다. Table 2와 같이 한글의 개수가 1개일 경우에는 용도로 선택한다.

3.2.3 숫자 처리

Fig. 6은 자동차 번호판 위치정렬을 위한 숫자 처리 과정을 나타낸다. 숫자 정보는 종류와 일련번호를 나타낸다. 이것은 Fig. 2의 (a), (c), (e)에서와 같이 1라인, Fig. 2의 (b), (d)와 같이 2라인

```

Variable = Firstline, Secondline, Sameline
i = 0, j = 0
FOR NumList의 리스트 전체 DO
    FOR NumList의 리스트 전체 DO
        IF i != j THEN
            degree = NumList[i], NumList[j]의 각도 계산
            IF degree <= 15 THEN
                Sameline에 NumList[i], NumList[j] 추가
            ELSE
                if (NumList[i].y - NumList[j].y) < 0 THEN
                    Firstline에 NumList[i] 추가
                    Secondline에 NumList[j] 추가
                ELSE
                    Firstline에 NumList[j] 추가
                    Secondline에 NumList[i] 추가
            j = j + 1
            i = i + 1
    Firstline, Secondline, Sameline의 중복된 객체 인덱스 삭제
    Firstline, Secondline, Sameline의 x좌표 기준으로 정렬
    IF Firstline 갯수 > 0 and Secondline 갯수 > 0 THEN
        종류 = Firstline
        일련번호 = Secondline
    ELSE
        IF Sameline의 개수 == 6 THEN
            종류 = Sameline[0], Sameline[1]
            일련번호 = Sameline[2], Sameline[3], Sameline[2], Sameline[3]
        IF Sameline의 개수 == 7 THEN
            종류 = Sameline[0], Sameline[1], Sameline[2]
            일련번호 = Sameline[3], Sameline[4], Sameline[5], Sameline[6]
    Display 지역 종류 용도 일련번호
    
```

Fig. 6 Number processing process

으로 배치되기 때문에 숫자를 저장하는 변수를 1라인의 숫자 정보를 저장하는 Sameline, 2라인에서 윗줄에 배치되는 Firstline, 아랫줄에 배치되는 Secondline으로 지정한다.

먼저 라인에 따른 숫자 배치를 확인하기 위해 숫자 리스트의 모든 객체의 x, y 좌표를 이용하여 객체 사이의 각도를 측정하여 15도 이내이면 Sameline에 저장한다. 그렇지 않고 y의 좌표가 낮으면 Firstline에 저장하고, 높으면 Secondline에 저장한 후 중복된 객체 인덱스는 삭제한다.

다음으로 종류와 일련번호 지정을 위해서 Firstline과 Secondline에 데이터가 저장되어 있으면 2라인으로 간주하고 Firstline은 종류로 지정하고, Secondline은 일련번호로 지정한다. Sameline의 개수가 6이면 처음 2문자는 종류, 나머지 4문자는 일련번호, 7이면 처음 3문자는 종류, 나머지 4문자는 일련번호로 지정한다.

이렇게 전처리, 한글처리, 숫자 처리를 통해 자동차 번호판이 가지는 지역, 종류, 용도, 일련번호 데이터를 최종 출력한다.

4. 실험 환경 및 고찰

본 논문에서 제안하는 자동차 번호판 위치 정렬 알고리즘의 성능을 확인하기 YOLOv5를 이용하여 Fig. 2에 적용된 번호판 형태에 국토교통부 자동차 번호판 등의 기준에 관한 고시[13]의 문자 이미지를 기준으로 랜덤한 숫자, 한글, 지역명으로 9,673개 차량 번호판 이미지를 만들었으며, Fig. 2(b)에 적용된 2006년식 자동차 번호판은 Aihub의 자동차 차종/연식/번호판 인식용 영상[14]에서 Fig. 2(b)와 관련된 번호판 이미지를 487개를 합성하여 전체 10,160개의 이미지를 학습하여 best.pt 파일을 추출하여 객체 검출에 사용하였다.

실험은 진입 차량의 영상 녹화에서 객체 검출한 결과

를 이용하여 검출된 번호판 정보의 배열을 출력하였다. Fig. 7은 Fig. 2(a)에서 제시한 1라인 1 한글 7 숫자, Fig. 8은 Fig. 2(b)에서 제시한 2라인 1 한글 6 숫자, Fig. 9는 Fig. 2(c)에서 제시한 1라인 1 한글 6 숫자, Fig. 10은 Fig. 2(d)에서 제시한 2라인 3 한글 6 숫자, Fig. 11은 Fig. 2(e)에서 제시한 1라인 3 한글 6 숫자의 출력 결과를 나타낸다.

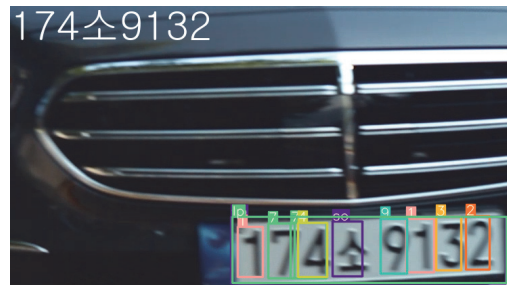


Fig. 7 1 line 7 characters Hangul 7 numeric license plate output

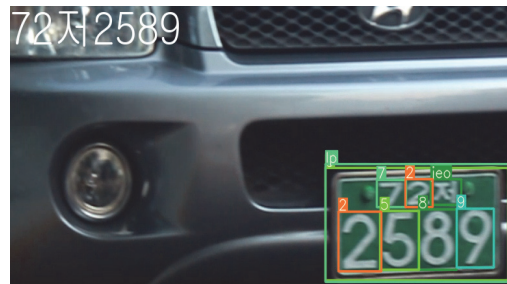


Fig. 8 2 lines, 6 characters Hangul 6 numeric license plate output



Fig. 9 1 line 1 character Hangul 6 numeric license plate output



Fig. 10 2 lines, 3 characters Hangul 6 numeric license plate output



Fig. 11 1 line 3 characters Hangul 6 numeric license plate output

본 논문의 연구결과는 YOLOv5에서 제시한 학습 결과 라벨 파일을 바탕으로 하여 자동차 번호판 배열이 제대로 되는 것과 관련된 결과이기 때문에 Fig. 7, 8, 9, 10, 11에서 제시한 것과 같이 형태별 문자 배열이 제대로 인식됨을 알 수 있었다.

5. 결론

본 논문은 한국에서 사용되고 있는 다양한 종류의 번호판 형식을 YOLO를 이용하여 객체를 검출한 후 정확한 자동차 번호판 정보를 추출하기 위한 정렬 방법에 대해 제안한다. 이를 위해 5가지 번호판 종류를 라인 수, 한글 수, 숫자 수로 분류하였고, 각각에 대한 정렬 방법을 제안하였다.

실험 결과 5가지의 번호판 출력에 있어서 YOLO에서 제시하는 결과 라벨 파일에서 신뢰도 점수가 높은 객체를 중심으로 하고, 불필요한 객체 정보는 삭제하여 모든 번호판의 종류와 번호판 정보를 올바르게 판독할 수 있었다.

본 연구를 통해 자동차 번호판을 인식하는 모든 시스템에 적용 가능할 것이다.

사 사

본 연구는 과학기술정보통신부 및 정보통신기획평가원의 지역지능화혁신인재양성(Grand ICT연구센터) 사업의 연구결과로 수행되었음(IITP-2021-2020-0-01791).

참고문헌

- [1] 한국정보통신기술협회 정보통신용어사전, 자동차번호판인식, (2021)
- [2] 김정환, 임준홍, “딥러닝을 이용한 번호판 검출과 인식 알고리즘”, 전기전자학회논문지, 제23권, 제2호, pp. 642-651, (2019).
- [3] 김남우, 허창우, “Emgu CV를 이용한 자동차 번호판 자동 인식 프로그램의 성능 평가에 관한 연구”, 한국정보통신학회 논문지, 제20권, 제6호, pp. 1209-1214, (2016).
- [4] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection", arXiv.org:1506.02640, (2016).
- [5] 이승주, 박구만, “가상 데이터를 활용한 번호판 문자 인식 및 차종 인식 시스템 제안”, 방송공학학회논문지, 제20권, 제6호, pp. 776-788, (2020).
- [6] 김재정, 김창복, “YOLO와 CNN을 이용한 강인한 차량 번호판 인식 시스템 구현”, 한국정보통신학회논문지, 제19권, 제4호, pp. 1-9, (2021).

- [7] 이재현, 조성만, 이승주, 김청화, 박구만, “인조 데이터를 활용한 차량 번호판 인식 시스템”, 전자공학회 논문지, 제57권, 제1호, pp. 107-115, (2020).
- [8] 당순정, 김응태, “YOLOv2기반의 영상위평을 이용한 강인한 오토바이 번호판 검출 및 인식”, 방송공학회논문지, 제25권, 제5호, pp. 713-725, (2019).
- [9] YOLOv5 Documentation, <https://docs.ultralytics.com>
- [10] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation", arXiv:1311.2524, (2014).
- [11] R. Girshick, "Fast R-CNN", arXiv:1504.08083v2, (2015).
- [12] Sh. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", arXiv:1506.01497, (2016).
- [13] 국토교통부, 자동차 등록번호판 등의 기준에 관한 고시, 국토교통부고시 제2020-344호, (2020).
- [14] 한국지능정보사회진흥원 AI 허브, <http://www.aihub.or.kr>

(접수: 2021.08.12. 수정: 2021.08.30. 게재확장: 2021.09.03.)