

## 컴퓨터 비전 소개: 과제 5

### 지침

- 이 숙제는 2025년 5월 28일 수요일 오후 3시까지 제출해야 합니다.
- 제출에는 두 부분이 포함됩니다.
  1. 실행된 모든 출력을 담은 ipynb 파일을 제출합니다.  
**코드로 뭔가를 해야 하는 질문은 빨간색으로 표시했습니다.**  
모든 셀의 출력(결과, 플롯 등)이 표시되는지 확인하고 파일을 저장합니다.  
이 코드는 Colab 환경에서 실행되도록 최적화되어 있으며, 디렉토리 경로를 변경하는 것과 같은 간단한 수정이 허용됩니다.
  2. 모든 질문에 대한 답변과 주요 선택 사항을 포함하여 작성한 PDF 파일을 제출하세요.  
**보고서에서 반드시 해야 할 일이 있는 질문은 파란색으로 표시했습니다.**  
여러 파일을 합쳐 제출할 수도 있습니다. 여러 PDF 파일을 합치는 온라인 링크 예시는 다음과 같습니다: <https://combinepdf.com/>.  
작성은 전자 파일이어야 합니다. 질문을 구성하는 것을 포함하여 손으로 쓴 글은 허용되지 않습니다.

### Google Colab 환경

- 이 과제를 위해 ipynb 파일과 과제 이미지를 개인 Google에 업로드하세요.  
운전하다.
- Colab에 드라이브를 마운트하고 Python을 사용하여 코드를 완성하세요. Google Colab은 <https://colab.research.google.com/>에서 이용할 수 있습니다. Google Colab에서 코드를 편집하고 실행하는 것을 권장하지만, 로컬 컴퓨터를 사용해도 됩니다.

## 객체 감지

이 문제 세트에서는 YOLO v1 기반의 단일 단계 객체 검출기를 구현합니다. 성능이 더 뛰어난 R-CNN 모델과 달리, 단일 단계 검출기는 이미지 또는 특징 맵에서 영역 제안을 명시적으로 잘라내지 않고 경계 상자와 클래스만 예측합니다. 따라서 실행 속도가 훨씬 빠르고 구현이 간단합니다.

객체 감지 시스템의 코드는 제공했지만, 몇 가지 핵심 기능은 구현되지 않은 상태로 남겨두었습니다. 여러분의 과제는 1) 모델과 코드를 이해하고, 2) 이러한 부족한 부분을 채우는 것입니다.

따라서 이 문제 세트는 이전 문제 세트보다 훨씬 더 많은 코드를 읽어야 합니다. 하지만 실제로 작성하는 코드의 양은 이전 문제 세트와 비슷할 것입니다.

객체 탐지 작업의 표준 데이터셋인 PASCAL VOC 데이터셋을 사용하여 탐지기를 훈련하고 평가할 것입니다. 전체 데이터셋에는 총 11,000개의 훈련/검증 데이터 이미지와 27,000개의 레이블이 지정된 객체가 포함되어 있으며, 이는 20개 클래스에 걸쳐 있습니다(그림 1).

아래에서 객체 감지 파이프라인의 단계와 구현할 모듈을 간략하게 설명합니다. 여기에 제시된 지침은 모든 내용을 담고 있지 않으므로, 자세한 구현 내용은 제공된 노트북의 주석을 참조하시기 바랍니다. 또한, 코드 작성이 필요한 부분으로 바로 넘어가지 말고, 제공된 시스템의 주석을 먼저 읽고, 각 함수의 입력과 출력을 이해하여 그 유용성을 파악하는 것이 좋습니다.



그림 1: PASCAL VOC 데이터 세트의 예시 이미지와 탐지 결과.

## (a) 검출기 백본 네트워크

[1점]

이미지 특징 추출을 위한 백본 네트워크로 MobileNetV2를 사용할 것입니다. 이는 효율적인 계산을 위한 간단한 합성곱 신경망입니다. 학습 속도를 높이기 위해 ImageNet 분류 문제를 해결하도록 네트워크를 사전 학습시켰습니다. 이는 이미 구현되어 있습니다.

보고서에서는 MobileNetV2의 주요 특징과 장점을 설명하고, 사전 학습된 가중치로 백본을 초기화하는 주요 이점을 설명하세요.

## (b) 활성화 및 제안

[3점]

(2점) 입력 이미지를 백엔드 네트워크에 통과시키면 (D, 7, 7) 모양의 합성곱 특징 맵이 생성되는데, 이를 D차원 특징들로 구성된 7x7 격자로 해석합니다. 이 격자의 각 셀에서 A개의 경계 상자 집합을 예측합니다.

이러한 경계 상자의 형식은 다음과 같습니다. 중심이  $x$ 인 격자를 고려하십시오.  $g_{c,y}^{g,c,y}$ . 이것과 관련하여,  $t_w, t_h$  ) 나중 입력할 예측 네트워크는 오프셋( $t_x$  중심)을 예측합니다. 이 변환을 적용하면 중심, 너비, 높이( $x_{hp}$ )가 있는 경계 상자 또는 제안을 얻습니다. 오프셋을 실제 경계 상자 매개변수로 변환하려면 노트북의 지침을 읽고 GenerateProposal 함수를 구현하세요.

보고서에 샘플 입력에 중첩된 7x7 활성화 그리드를 보여주는 이미지를 삽입하세요. 그런 다음, 코드가 오프셋 텐서의 차원에 맞게 그리드를 확장하고, 예측된 오프셋을 추출하고, 이를 바운딩 박스 매개변수로 변환하고, 그 값을 최종 제안 텐서로 조립하는 방법을 간략하게 설명하세요.

## (c) 예측 네트워크

[4점]

(3점) 다음 과제는 두 경계 상자의 교집합(IoU)을 추정하는 IoU 함수를 구현하는 것입니다. (자카르 유사도라고도 함) 두 윈도우  $W1$  과  $W2$  에 대한 IoU는 다음 과 같습니다.

$$IoU(W1, W2) = \frac{|W1 \cap W2|}{|W1 \cup W2|}, \quad (1)$$

여기서  $\cap$ 와  $\cup$ 는 각각 교집합과 합집합 연산입니다. 이 함수는 나중에 객체 감지 모듈에서 예측된 경계 상자과 실제 경계 상자 사이의 IoU를 계산하는 데 사용됩니다.

보고서에서 Intersection-over-Union이 무엇을 측정하는지, 즉 예측된 상자와 기준 진실 상자 사이의 중첩을 어떻게 정량화하는지, 그리고 이 지표가 객체 감지에서 현지화 정확도를 판단하는 데 왜 중요한지 간단히 설명하세요.

## (d) 경계 상자 및 클래스 예측 헤드

[4점]

(3점) 예측 네트워크는 백본 네트워크의 기능을 입력으로 받아 출력합니다.

분류 점수를 넣고 각 경계 상자를 오프셋합니다.

완전 합성 네트워크의 7 X 7 그리드의 각 위치에 대해 예측 네트워크는 C개의 숫자를 출력하는데, 이 숫자는 그리드 셀에 중심이 있는 경계 상자에 대한 C개의 객체 범주에 대한 분류 점수로 해석됩니다.

또한, 각 위치의 A개 경계 상자 각각에 대해 예측 네트워크는 오프셋(경계 상자를 나타내는 4개 숫자)과 신뢰도 점수(큰 양수 값은 경계 상자에 객체가 포함될 확률이 높음을 나타내고, 큰 음수 값은 경계 상자에 객체가 포함될 확률이 낮음을 나타냄)를 출력합니다(그림 2 참조).

자세한 지침은 노트북을 참조하고 PredictionNetwork 클래스의 forward 함수를 구현하세요.

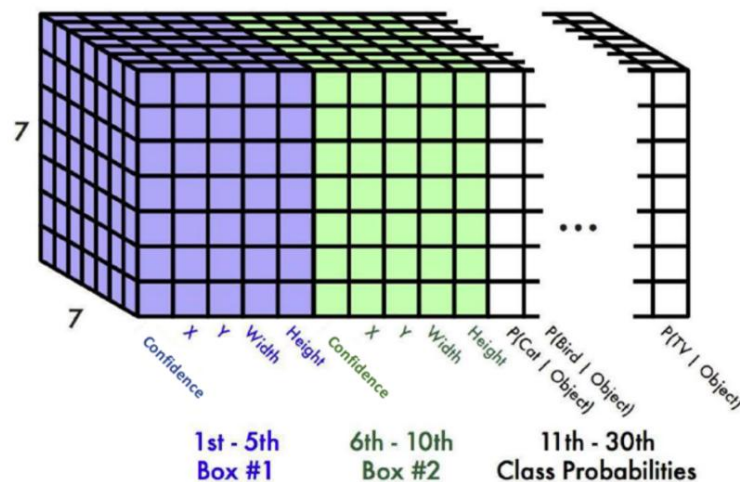


그림 2: 예측 네트워크의 출력.

보고서에서는 PredictionNetwork의 순방향 패스를 구현한 방법을 설명하고, 사용자 정의 레이어를 통해 백본 피쳐 맵을 공급하여 통합 출력 텐서를 생성한 방법과 해당 텐서를 세 가지 출력(bbox 오프셋, 신뢰 점수, 클래스 점수)으로 분할한 방법을 설명하세요.

## (e) 손실 함수

[1점]

신뢰도 점수, 경계 상자 오프셋, 클래스 점수, 이 세 가지 값을 회귀 분석해 보겠습니다. 각 손실을 개별적으로 계산하는 코드는 제공했지만, 전체 손실에 대한 가중 합을 계산해야 합니다.

보고서에서는 신뢰 점수 회귀, 경계 상자 회귀, 객체 분류라는 세 가지 손실 구성 요소를 간략하게 설명하세요.

## (f) 객체 감지 코드

[2점]

(1점) 순방향 패스는 이전에 작성하신 모든 모듈과 저희가 미리 정의해 두었던 모듈들을 모두 캡슐화합니다. 객체 감지기의 순방향 패스 대부분을 구현했습니다.

당신을 위해, 하지만 마지막 단계는 채워지지 않았습니다. 손실을 계산해 주세요 conf loss, cls loss, - - 정규 손실, 그 합계, 총 손실. -

보고서에서 손실 구성요소 3개의 가중치와 합계를 계산하여 손실 구성요소를 생성하는 방법을 설명하십시오. 최종 총 감지 손실.

## (g) 객체 탐지기 훈련

[1점]

모든 것이 예상대로 작동하는지 확인하려면 감지기를 약간 과적합하게 만들 수 있습니다. 데이터 하위 집합입니다. 이 과적합 실험은 실행하는 데 몇 분 정도 걸립니다.

보고서에서 실험 설정(배치 크기, 학습률, 최적화 프로그램, 숫자)을 간략하게 설명하십시오. 에포크, 서브셋 크기, 하드웨어)를 입력하고 실험 코드에서 생성된 플롯을 삽입합니다. 간략하게 설명해 주세요.

## (h) 비최대 억제(NMS)

[3점]

(2점) 마지막으로, 중복된 바운딩 박스 예측을 제거합니다. 특히, 비최대 억제(NMS)를 사용하여 바운딩 박스 예측과 크게 겹치는 부분을 제거합니다.

신뢰도 점수가 더 높은 또 다른 경계 상자 예측입니다.

보고서에서 중복된 핑 탐지를 제거하기 위해 가장 높은 점수를 받은 상자를 반복적으로 선택하고 나머지 상자를 버림으로써 비최대 억제를 적용한다고 설명하십시오.

해당 상자와의 IoU가 선택한 임계값을 초과하는 경우, 이를 계속하여 상자가 하나도 남지 않을 때까지 반복합니다.

## (i) 추론

[1점]

추론 단계에서는 검증 이미지를 입력으로 사용하여 이미지 내 각 객체에 대한 경계 상자, 신뢰도 점수, 클래스 레이블을 예측합니다. 낮은 신뢰도와 반복적인 경계 상자는 임계값과 NMS에 의해 필터링됩니다.

보고서에 추론 예측 결과 이미지를 삽입하고 간략한 전체 요약 제공하십시오. 성능, 일반적인 오류 및 추세에 대한 정보입니다.

## (j) 평가

[1점]

탐지기 성능을 평가하기 위해 평균 평균 정밀도(mAP)를 사용합니다. 메트릭. mAP는 데이터 세트의 모든 클래스에 대한 정밀도의 평균을 구하여 계산됩니다. 귀하의 편의를 위해 mAP를 계산하는 코드도 제공했으므로 실행만 하면 됩니다. 추론 코드를 확인하고 감지기가 약 12% mAP 이상을 달성하는지 확인하세요. 검증 데이터 세트.

보고서에서 평균 정밀도(AP)가 어떻게 계산되는지 설명하고 클래스별 AP 막대를 삽입하세요. 차트를 작성하고 해당 차트가 전달하는 내용을 설명한 다음 전체 mAP 값의 의미를 논의합니다. 모든 계층에 걸쳐.