

심층 컨벌루션 신경망 기반의 실시간 드론 탐지 알고리즘

Convolutional Neural Network-based Real-Time Drone Detection Algorithm

이 동 현[†]

Dong-Hyun Lee[†]

Abstract: As drones gain more popularity these days, drone detection becomes more important part of the drone systems for safety, privacy, crime prevention and etc. However, existing drone detection systems are expensive and heavy so that they are only suitable for industrial or military purpose. This paper proposes a novel approach for training Convolutional Neural Networks to detect drones from images that can be used in embedded systems. Unlike previous works that consider the class probability of the image areas where the class object exists, the proposed approach takes account of all areas in the image for robust classification and object detection. Moreover, a novel loss function is proposed for the CNN to learn more effectively from limited amount of training data. The experimental results with various drone images show that the proposed approach performs efficiently in real drone detection scenarios.

Keywords: Convolutional Neural Networks, Drone Detection

1. 서 론

드론은 과거 군사적 용도를 위해 개발되었으나 현재 상업 용으로 크게 성장하여 다양한 분야로 영역을 넓히고 있다. 민간 시장으로는 재난 감시, 농업, 물류, 방송 등에 활용되고 있으며, 또한 취미, 레저용으로도 대중화, 보편화되고 있는 추세이다. 미국 방위 시장 조사에 따르면 드론의 전 세계 시장규모는 2010년 약 52억 달러에서 2022년에는 약 114억 달러까지 성장할 것으로 전망되고 있다. 특히 4차 산업혁명의 시대에서 드론 산업은 제조, 서비스 융합 모델로 주목받고 있으며 세계 각국은 드론 산업 선점을 위해 치열한 경쟁구도를 구축하고 있다. 산업뿐 아니라 학계에서도 드론의 이동 경로 생성, 위치 인식, 인공 지능 기반 자율 주행, 군집 비행 등 다양한 연구가 진행되고 있다.

하지만 급격한 드론 시장의 성장 추세에 드론을 악의적으로 사용하는 등의 부작용에 대한 우려 또한 비례해 커지고 있

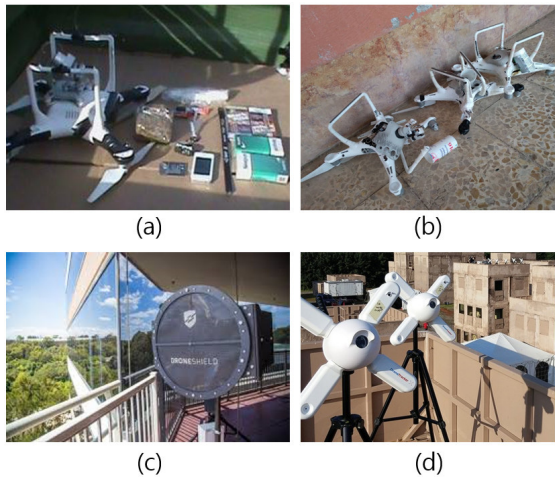
다. 드론은 온, 오프라인에서 쉽게 구매가 가능하고 조작성이 간단하며 운송 능력과 정밀제어, 빠른 이동 속도 등의 특징으로 인해 [Fig. 1(a), (b)]와 같이 테러, 산업 스파이, 사생활 침해, 불법 밀반입 등에 악용되는 사례가 증가되고 있다. 또한 조종자의 미숙으로 인해 공항이나 변전소, 발전소 중요 시설에 추락, 항공기와 충돌, 그리고 레저용 헬기라이더나 패러글라이더와 충돌하는 사례 등이 발생하고 있으며, 목적이나 소유주가 다른 다수의 드론이 공중에서 운용될 경우 드론 간의 충돌로 인한 사고가 빈번히 발생하고 있다.

이와 같은 드론으로 인한 피해를 예방하기 위하여 [Fig. 1(c), (d)]와 같이 초고해상도 레이더, 마이크, 카메라, RF 탐지 등 다양한 센서를 융합한 드론 탐지 시스템이 개발되고 있다. 하지만 이러한 시스템은 일반적으로 고비용이며 시스템의 복잡성으로 인해 전문 운용자가 필요하며 가정, 또는 개인용으로 설비가 용이하지 않고 휴대가 불가능하다는 단점이 있다. 또한 기존의 드론 탐지 시스템은 장비의 규모와 무게 등의 문제로 항공기나 레저용 헬기라이더 및 패러글라이딩에 장착하여 운용하기에는 어려움이 있다. 이 뿐 아니라 드론 간의 충돌을 예방하기 위한 드론 탐지를 할 경우 드론 내부에 장착된 배터리 용량의 한계와 임베디드 보드의 연산 능력 한계로 인해

Received : Jun. 23. 2017; Revised : Aug. 15. 2017; Accepted : Sep. 13. 2017

* This paper was supported by Kumoh National Institute of Technology.

† Corresponding author: Electrical Engineering, Kumoh National Institute of Technology, 61 Daehak-ro, Gumi, Gyeongbuk, Korea (donglee@kumoh.ac.kr)



[Fig. 1] Drones being used to commit crime (a), (b), and various drone detection systems (c) DroneShield, (d) DEDrone

다수의 센서를 융합한 드론 탐지 시스템의 적용에 어려움이 있다. 따라서 개인이나 일반 가정, 소규모의 건물, 비행체 등에 장착하여 활용할 수 있으며 저비용으로 복잡한 설비 없이 용이하게 운용될 수 있는 드론 탐지 및 추적 시스템이 필요하다.

본 연구에서는 컨벌루션 신경망(Convolutional Neural Network (CNN))을 이용한 드론 탐지에 대한 연구를 수행한다. CNN은 높은 물체 인식 성능을 가지는 딥 러닝 알고리즘 중 하나로서 최근 영상처리에서 물체 인식에 활용되고 있다^[1]. CNN의 구조는 다수의 컨벌루션 층과 일반적인 인공 신경망 층의 결합으로 이루어져 있으며, 컨벌루션 층에서 컨벌루션 연산을 통해 특징점을 추출함으로써 학습효율을 높이고 국소 최적치 문제를 해결할 수 있다. 컨벌루션 신경망은 물체 분류^[2], 물체 탐지^[3] 및 영상 분할^[4] 등 다양한 분야에 응용되고 있다. 본 논문에서 중점을 두고 있는 드론 탐지는 object detection의 카테고리로 분류될 수 있다.

Object detection에서는 학습을 위한 이미지에 탐지하고자 하는 물체를 경계 상자(bounding box)로 표현함으로써 이미지 상에서의 물체의 위치, 넓이 및 폭을 레이블링 하여 심층망 학습 시에 ground truth 정보로 사용한다. 기존의 object detection과 관련된 연구로는 R-CNN, Fast R-CNN, YOLO 및 YOLOv2 등이 있다. R-CNN의 경우 여러 단계의 학습 과정을 필요로 하며 이로 인하여 물체 탐지 속도가 느리다는 단점이 있다^[5]. 이를 개선한 Fast R-CNN은 물체 후보 영역을 한번에 입력 받고 처리하게 된다^[6]. 하지만 물체 탐지를 위한 후보 영역을 계산하는 과정에서 상당한 시간이 소비된다. Faster R-CNN의 경우 물체 분류를 위한 네트워크에 후보 영역을 구하기 위한 망을 추가함으로써 계산 시간을 줄이게 된다^[7]. 하지만 위의 방식들은 기본적으로 GPU 기반의 높은 연산량을 필요로 하므로 임

베디드 시스템과 같이 연산 능력이 제한된 시스템에서 실시간으로 사용하기에 어려움이 있다.

YOLO와 이를 개선시킨 YOLOv2의 경우 빠른 연산 속도로 다양한 물체를 인식하고 분류 및 영상 내에서의 위치를 파악할 수 있는 알고리즘이다^[8,9]. 본 논문에서 제안하는 알고리즘은 YOLOv2를 개선하여 빠른 연산 속도와 함께 정밀한 object detection이 가능한 심층망 구조 및 학습법을 제안한다. 따라서 2장에서 YOLOv2의 특성과 문제점에 대해 자세히 다루고 3장에서 이를 개선한 제안한 알고리즘에 대하여 설명하고자 한다.

2. YOLOv2의 특징과 한계점

YOLOv2에서는 입력 이미지를 $S \times S$ 의 그리드로 나눈 후 레이블링 된 물체의 경계 상자의 중심이 위치한 그리드 셀에 속한 다수의 앵커 박스가 물체에 대한 클래스 확률($p(\text{class}|\text{object}) = p(\text{clo})$), 물체 존재 확률($p(\text{object}) = p(o)$), 그리고 물체 중심의 위치(x, y)와 경계 상자의 폭과 높이(w, h)를 추정한다.

하지만 YOLOv2의 경우 오직 물체가 존재하는 셀에 대해서만 학습이 수행되므로 그 외의 신경망 부분, 즉 해당 클래스에 속하지 않는 영역에 대한 학습은 수행되지 않는다. 이로 인해 물체가 존재하지 않은 영역에 대해서도 높은 클래스 확률을 가지는 상황이 발생하며, 트레이닝 이미지에서 물체의 중심이 위치한 셀에 속한 앵커 박스들만 클래스 확률 학습용으로 사용하기 때문에 학습 효율성이 낮아지는 단점이 있다. 특히 드론 탐지와 같이 하나의 특정 물체만을 탐지할 경우 YOLOv2는 softmax 알고리즘으로 하나의 클래스에 대해 $p(\text{clo})$ 를 계산하게 되어 모든 앵커 박스에 대해 1의 클래스 확률값을 가지는 문제가 발생한다. Softmax 대신 sigmoid를 사용하더라도 물체가 위치하지 않아 $p(\text{clo})$ 가 1의 값으로 설정된 적이 없는, 즉 학습이 수행되지 않은 이미지 영역에 대한 클래스 확률값은 초기에 설정한 확률값을 유지하게 되므로 해당 클래스의 물체가 존재하지 않더라도 초기값이 높은 경우 클래스 확률값이 높게 추정되는 문제가 발생한다. 비록 하나의 클래스만 존재할 경우 $p(o)$ 만을 사용하여 탐지가 가능하지만 $p(\text{clo}) = 0$ 인 경우에 대한 고려가 되지 않으며, 심층망이 가진 분류 기능을 활용하지 못한다는 단점이 있다.

YOLOv2에서 물체의 경계 상자의 폭과 높이를 추정하기 위하여 학습을 수행하기 이전에 5개의 서로 다른 폭과 높이를 가지는 앵커 박스를 미리 정의하여 사용한다. 앵커 박스의 폭과 높이의 추정값은 학습된 네트워크에서 출력된 값을 지수함수에 대입하고 이를 미리 정의한 앵커 박스의 폭과 높이에 곱하여 구한다. 하지만 이 경우 앵커 박스의 크기를 제한할 수 없으

며, 초기 앵커 박스의 크기가 실제 물체의 크기와 상이할 경우 수렴 시간이 증가하며, 이를 보완하고자 앵커 박스의 개수를 증가시킬 경우 학습 및 추정 과정에서의 연산량이 증가한다는 단점이 있다.

본 연구에서는 효율적으로 드론을 추정하는 심층망 기반 알고리즘과 그 학습법을 제안한다. 제안하는 방법은 물체의 중심이 속하지 않는 앵커 박스에 대해서도 클래스 확률을 학습할 수 있도록 *loss* 함수를 개선하였으며, 하나의 앵커 박스만을 사용하는 대신 앵커 박스가 추정하는 경계 상자의 폭과 높이를 이미지 크기에 맞추어 범위 제한이 가능하고, 학습 과정에서 초기 앵커 박스의 크기와 타겟 경계 상자의 크기 차이가 있더라도 신속히 수렴이 가능할 수 있도록 모델링 하였다. 또한 타겟 경계 상자와 앵커 박스 간의 IOU (intersection over union) 를 활용하여 물체의 중심 위치 추정 및 경계 상자 추정의 안정된 수렴이 가능하도록 하였다.

본 논문은 다음과 같이 구성된다. 3장에서는 제안하는 심층망의 네트워크 구조와 학습 방법에 대해 설명하고 4장에서는 제안하는 구조의 성능을 평가한다. 5장에서는 결론과 추후 연구방향에 대해 설명한다.

3. 심층망 구조 및 학습

3.1 심층망 구조

제안하는 심층망 구조는 [Table 1]에 나타내었으며 기본적으로 YOLOv2를 간략화한 구조이다. 각 층은 Convolutional layer와 Maxpool layer로 구성되어 있으며 leaky rectified linear

[Table 1] Full Description of the Proposed CNN

Layer	Type	Filters	Size/Stride	Output
1	Convolutional Maxpool	16	3 × 3 2 × 2 / 2	416 × 416 × 16
2	Convolutional Maxpool	32	3 × 3 2 × 2 / 2	208 × 208 × 32
3	Convolutional Maxpool	64	3 × 3 2 × 2 / 2	104 × 104 × 64
4	Convolutional Maxpool	128	3 × 3 2 × 2 / 2	52 × 52 × 128
5	Convolutional Maxpool	256	3 × 3 2 × 2 / 2	26 × 26 × 256
6	Convolutional Maxpool	512	3 × 3 2 × 2 / 1	13 × 13 × 512
7	Convolutional	1024	3 × 3	13 × 13 × 1024
8	Convolutional	6	3 × 3	13 × 13 × 6

activation 함수를 사용하였다.

기존의 YOLOv2가 5개의 앵커 박스를 사용하는 반면, 제안하는 구조는 1개의 앵커 박스만을 사용하며, 이를 보완하기 위한 앵커 박스 크기 및 위치 추정법을 제안하였다. 또한 본 연구에서는 심층망을 이용한 드론의 추정이므로 물체의 클래스는 1개로 정의되었으며, YOLOv2와 같이 이미지를 최종적으로 13 × 13 그리드로 나누어 각 그리드 셀에 속한 앵커 박스를 통하여 드론의 위치와 크기를 추정한다. 따라서 심층망의 최종 출력은 13 × 13의 그리드 셀, 각 셀 당 하나의 앵커 박스, 각 앵커 박스 당 6개의 출력(2차원 중심 좌표값(2개), 폭과 높이(2개), 클래스 확률(1개), 하나의 confidence (1개)) 을 포함하므로 [Table 1]의 Layer 8에서와 같이 13 × 13 × 6의 출력 크기를 가진다.

본 논문에서는 하나의 클래스에 대해 다루지만 클래스의 개수는 Layer 8에서 필터 크기를 클래스 개수에 맞추어 수정할 경우 다수의 클래스로 확장 가능하다. 하지만 효과적인 물체 탐지를 위해서 클래스 개수의 증가에 따라 심층망의 깊이를 증가시킬 필요가 있다. 본 논문에서 심층망의 깊이가 8개의 층으로 기존의 다른 심층망 구조들에 비해 비교적 얇은 이유는 첫째로 본 연구에서 추정하고자 하는 클래스가 1개이기 때문에 다수의 클래스를 탐지하는 심층망에 비해 상대적으로 얇더라도 충분한 탐지가 가능하기 때문이며, 둘째로 추후 드론이나 휴대용 장비와 같은 제한된 에너지와 컴퓨팅 자원을 가진 휴대용 임베디드 시스템에 장착된 카메라를 통해 실시간으로 드론 추정 및 추적을 수행하기 위함이다.

3.2 심층망의 클래스 확률 및 confidence 학습

심층망의 출력은 13 × 13인 그리드 셀의 각 셀에 속한 앵커 박스가 가진 6개의 값, 즉 중심의 좌표, 박스의 폭과 높이, 클래스 확률 $p(o)$, 그리고 confidence ($C = p(o) \cdot IOU$)이다. Confidence는 물체 존재 확률 $p(o)$ 에 IOU를 곱한 값이다. YOLOv2에서는 클래스 확률과 물체 존재 확률 및 confidence를 학습하기 위하여 학습 데이터에서 해당 클래스 물체의 중심이 속한 셀에 $p(o) = 1$ 의 값을 설정하고 물체의 중심이 속하지 않은 나머지 셀에 대해서는 $p(o) = 0$ 을 설정한 후, 클래스 확률은 오직 확률이 1인 셀에 대해서만 학습을 하고 물체 존재 확률은 모든 셀에 대해서 학습을 수행한다. 하지만 이와 같은 방식의 경우 $p(o) = 0$ 인 셀에 대한 클래스 확률 학습이 진행되지 않는다. 이 경우 물체 존재 확률이 높지만 실제로 어떠한 클래스에도 속하지 않는 물체를 추정할 상황 및 거짓-탐(false-positive)으로 인해 물체가 존재하지 않는 이미지 영역이 물체로 인식된 상황이 고려되지 않는다. YOLOv2는 위와 같은 상황이 발생할 경우 실제 물체로 인식된 이미지

영역이 심층망에 정의된 클래스에 속하지 않더라도 softmax 함수로 인해 정의된 클래스 내에서 상대적으로 높은 확률을 가지는 클래스를 해당 물체의 최종 클래스로 결정하게 된다. 특히 추정하고자 하는 클래스가 오직 1개인 경우 클래스 비교 대상이 없으므로 softmax 함수에 의해 모든 경우에서 클래스 확률이 1이 되므로 심층망이 분류기 기능을 수행하지 못하게 된다.

본 논문에서는 이를 보완하기 위하여 클래스 확률을 계산하는 과정에서 softmax 함수 적용 전에 1차적으로 sigmoid 함수를 먼저 적용하여 그 결과가 임계값보다 높은지 확인하고, 임계값보다 더 높은 값을 가지는 클래스들에 대해서 2차적으로 softmax 함수를 적용하여 최종 클래스 함수를 계산하도록 하였다. 이를 위하여 클래스 확률은 물체의 중심이 속한 셀 뿐 아니라 그 외의 셀에도 클래스 확률이 $p(clo) = 0$ 으로 수렴할 수 있도록 학습시킨다. 심층망은 위에서 설정한 목표 클래스 확률값과 confidence를 추정하기 위하여 YOLOv2와 동일하게 식 (1)을 통하여 추정 클래스 확률값과 confidence로 변환된다.

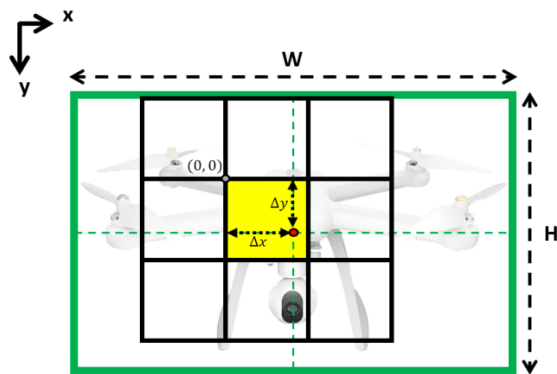
$$\hat{p}(clo) = \text{sig}(t_c), \hat{C} = \text{sig}(t_o) \quad (1)$$

위 식에서 t_c 와 t_o 는 클래스 확률과 confidence를 추정하기 위한 심층망의 출력이다.

3.3 앵커 박스의 중심 좌표 및 크기 학습

앵커 박스의 중심 좌표는 [Fig. 2]와 같이 물체의 중심이 위치한 셀의 왼쪽 위의 꼭지점을 원점으로 하여 오프셋이 정의된다. 이는 YOLOv2와도 동일한 방식이며, 각 셀의 변의 길이는 1이므로 x 와 y 오프셋의 범위는 0에서 1 사이의 값을 가진다. 따라서 sigmoid 함수를 사용하여 오프셋을 추정한다.

YOLOv2의 경우 식 (2)를 사용하여 앵커 박스의 추정 폭 \hat{w}



[Fig. 2] The x and y offsets of a bounding box are defined from the top-left corner of the grid cell where the center of the object is located

과 추정 높이 \hat{h} 를 계산한다.

$$\hat{w} = p_w \exp(t_w), \hat{h} = p_h \exp(t_h) \quad (2)$$

위 식에서 t_w 와 t_h 는 심층망에서 앵커 박스의 폭과 높이를 추정하기 위한 출력값이며, p_w 와 p_h 는 앵커 박스의 초기 추정 폭과 높이를 나타낸다. 심층망의 출력에서 이미지를 $W \times H$ 의 크기로 변환하므로 식 (2)에서 추정 폭과 추정 높이의 범위는 각각 $0 \leq \hat{w} \leq W$ 및 $0 \leq \hat{h} \leq H$ 로 제한되어야 한다. 하지만 위 식은 수식적으로 이를 만족시키지 않으며, 또한 exp 함수의 특성으로 인해 t_w 와 t_h 가 음수의 값을 가질 경우의 변화율이 양수를 가질 경우의 변화율보다 작기 때문에 학습 과정에서 초기 추정 폭과 높이의 값이 실제 물체의 추정 폭과 높이의 값에 비해 상대적으로 클 경우 수렴 시간이 증가한다는 단점이 있다. 본 논문에서는 이를 개선하기 위하여 식 (3)을 제안하였다.

$$\hat{w} = W \text{sig}(t_w - a_w), \hat{h} = H \text{sig}(t_h - a_h) \quad (3)$$

$$a_w = \log\left(\frac{W - p_w}{p_w}\right), a_h = \log\left(\frac{H - p_h}{p_h}\right)$$

위 식을 통하여 추정 폭과 추정 높이의 범위를 각각 $0 \leq \hat{w} \leq W$ 및 $0 \leq \hat{h} \leq H$ 로 제한할 수 있으며, 또한 sigmoid 함수의 특성으로 인해 p_w 와 p_h 가 $W/2$ 와 $H/2$ 를 기준으로 서로 다른 변화율을 가지며, 이를 통하여 초기 설정값인 p_w 및 p_h 과 타겟 경계 상자의 폭과 높이와의 차이가 크더라도 신속하게 수렴할 수 있다는 장점을 가진다.

3.4 최적화 학습을 위한 loss의 설정

본 연구에서는 심층망의 학습을 위해 기울기 강하 기반 학습법에 속하는 AdaGrad (Adaptive gradient) 알고리즘을 사용하였으며, 해당 알고리즘이 최소화하고자 하는 loss 함수, L_{tot} ,는 식 (4)와 같이 4가지 loss 함수의 가중된 합으로 정의하였다.

$$L_{tot} = w_{coord}L_{coord} + w_{class}L_{class} + w_{conf}L_{conf}$$

$$L_{coord} = (L_{xy} + L_{wh})$$

$$L_{xy} = \sum_{i=0}^{S^2} I_i^{obj} \exp(-IOU_i) [(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2] \quad (4)$$

$$L_{wh} = \sum_{i=0}^{S^2} I_i^{obj} \exp(-IOU_i) \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right]$$

$$L_{class} = \sum_i^{S^2} \sum_{c \in class} [\lambda_{class}^{obj} I_i^{obj} (1 - \hat{p}_i(c|o))^2 + \lambda_{class}^{noob} I_i^{obj} \hat{p}_i(c|o)^2] \quad (4)$$

$$L_{conf} = \sum_i^{S^2} [\lambda_{conf}^{obj} I_i^{obj} (1 - \hat{C}_i)^2 + \lambda_{conf}^{noob} I_i^{noob} \hat{C}_i^2]$$

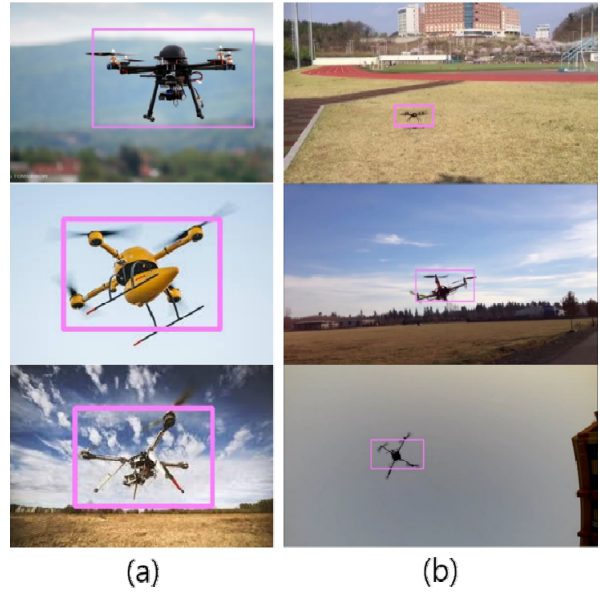
위 식에서 L_{xy} , L_{wh} , L_{conf} 및 L_{class} 는 각각 목표 경계 상자 와 앵커 박스 간의 중심 좌표, 폭과 높이, confidence, 및 클래스 확률의 에러의 제곱에 대한 합을 나타낸다. I_i^{obj} 는 i 번째 셀에 경계 상자의 중심이 위치한 경우 1의 값, 그 외엔 0의 값을 가지며, I_i^{noob} 는 그 반대의 값을 가진다. L_{xy} 와 L_{wh} 의 식에서 IOU_i 는 i 번째 셀에 속한 앵커 박스와 목표 경계 상자 간의 IOU 를 의미한다. 학습이 진행되어 전체적인 loss가 줄어들어 따라 목표 경계 상자의 중심이 위치한 셀에 속한 앵커 박스의 IOU 의 값은 감소하게 된다. 본 연구에서는 이에 착안하여 $\exp(-IOU_i)$ 를 L_{xy} 와 L_{wh} 의 내부적인 가중치로 적용함으로써 식 (4)와 L_{coord} 에 대한 가중치 w_{coord} 가 큰 값을 갖더라도 loss가 줄어들어 따라 L_{coord} 자체의 내부적인 가중치가 줄어들 수 있도록 하였다.

L_{class} 의 경우 물체가 존재하지 않은 셀에 속한 앵커 박스에 대해서도 0의 클래스 확률을 학습하게 하였으며, 또한 $p_i(c)$ 가 아닌, $p_i(c|o)$ 의 확률을 학습하도록 함으로써 confidence와 클래스 확률을 분리시켜 각각이 명확히 학습될 수 있도록 하였다. L_{conf} 의 경우 YOLOv2의 loss 함수와 동일하게 모든 셀에 대해서 학습할 수 있도록 설정하였다. L_{tot} 에서는 각각의 loss 값 간의 스케일이 다르므로 탐지하고자 하는 물체의 특성에 따라 각 loss에 대한 비중, 즉, w_{coord} 및 w_{conf} 의 w_{class} 값을 적절하게 조절할 수 있도록 하였다.

드론의 경우 상공을 비행하는 특성상 이미지 상에서 차지하는 크기의 비율이 대체적으로 작은 편이므로 클래스 확률값이 0이 아닌 셀만을 고려하는 L_{xy} 와 L_{wh} 에 비해 모든 셀을 고려하는 L_{conf} 와 L_{class} 의 값이 큰 경향을 보인다. 이를 고려하여 w_{coord} 를 w_{conf} 와 w_{class} 에 비해 상대적으로 큰 값으로 설정함으로써 L_{conf} 와 L_{class} 간의 상대적인 스케일을 조절할 수 있다. 또한 L_{conf} 와 L_{class} 의 경우 작은 물체에 대한 클래스 확률이 0인 셀의 개수가 그렇지 않은 셀의 개수가 많으므로 λ_{conf}^{obj} 와 λ_{class}^{obj} 를 λ_{conf}^{noob} 와 λ_{class}^{noob} 에 비해 큰 값으로 설정함으로써 상대적 중요도를 조절할 수 있도록 하였다.

4. 실험 및 결과

본 연구에서 제안한 드론 탐지 알고리즘의 성능을 평가하

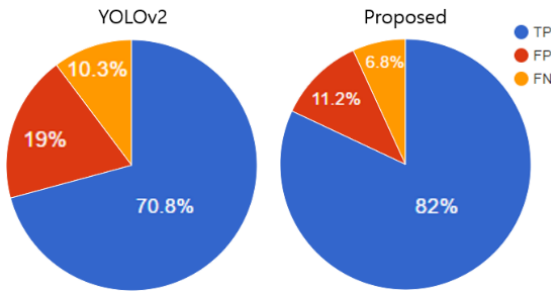


[Fig. 3] Large size drone dataset (dataset 1) (a) and small size drone dataset (dataset 2) (b). The targeting ground truth bounding boxes are visualized as rectangle boxes

기 위하여 다양한 드론 이미지 데이터로 알고리즘을 학습시켰으며, 동일한 트레이닝 데이터로 학습한 두 가지 다른 알고리즘과의 성능을 비교하였다. 데이터셋은 2가지로 구분하였으며, [Fig. 3(a)]와 같이 이미지 상에서 비교적 큰 크기를 차지하는 다양한 종류의 드론 이미지로 구성된 데이터셋 1과 [Fig. 3(b)]와 같이 이미지 상에서 비교적 작은 크기의 비율을 갖는 데이터셋 2로 구성된다. 실험에서 학습용 데이터셋 1은 209개, 테스트용 데이터셋 1은 70개, 학습용 데이터셋 2는 153개, 테스트용 데이터셋 2는 36개를 각각 제작하여 사용하였다.

본 논문에서 사용한 YOLOv2 버전은 구글의 오픈소스 기계 학습 라이브러리인 텐서플로우(Tensorflow)로 구현된 소스코드인 Darkflow^[10]를 사용하였으며 제안하는 알고리즘 또한 텐서플로우를 바탕으로 구현하였다. 실험에 사용한 컴퓨팅 환경으로는 Intel(R) Core(TM) i7-6700 CPU @ 3.40GHz 의 CPU를 사용하였으며, NVIDIA Geforce GTX 1080의 GPU를 사용하였다. 학습은 YOLOv2와 제안한 알고리즘 각각에 대하여 동일하게 10000 epoch 동안 학습을 수행하였다.

[Fig. 4]와 [Table 2]는 각 알고리즘을 테스트한 결과를 보여준다. 각 알고리즘은 각 데이터셋의 테스트 이미지에 대해 드론 탐지를 수행하였으며, 하나의 이미지당 최대 5개까지 선택할 수 있도록 하였다. 따라서 하나의 테스트 이미지에서 추정 가능한 드론의 대수는 최소 0대에서 최대 5대이다. 임계값의 설정은 각 알고리즘의 각 테스트 데이터에 대해 가장 높은 정밀도(precision), 재현율(recall), 및 평균 정밀도(mAP: mean average precision)를 가지는 값으로 설정하였다.



[Fig. 4] True-positive (TP), false-positive (FP), and false-negative (FN) rates of YOLOv2 and the proposed algorithm in drone detection

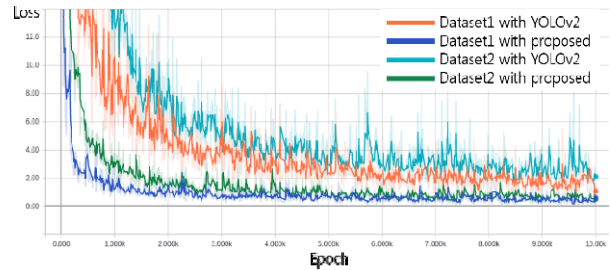
[Table 2] Results on Dataset 1 and Dataset 2

Algorithms	Dataset 1			Dataset 2		
	Precision	Recall	mAP	Precision	Recall	mAP
YOLOv2	0.79	0.96	0.87	0.78	0.79	0.75
Proposed	0.89	0.94	0.92	0.86	0.90	0.87

[Fig. 4]는 두 알고리즘 간의 참-긍정률(TP: true-positive), 거짓-긍정률(FP: false-positive) 및 거짓-부정률(FN: false-negative)에 대한 데이터셋 1과 데이터셋 2의 결과의 평균을 원 그래프로 나타낸다. 원 그래프 상의 비율에서 확인할 수 있듯이 제안하는 알고리즘은 학습 과정에서 드론이 존재하지 않은 이미지 영역에 대해서도 클래스 확률을 학습함으로써 기존의 YOLOv2의 방식에 비해 높은 참-긍정률과 낮은 거짓-긍정률과 거짓-부정률을 가짐을 확인할 수 있다. 하나의 이미지당 검출 속도의 경우 YOLOv2가 평균 0.0413초의 시간으로 처리하였으며, 제안하는 알고리즘은 평균 0.0188 초의 시간으로 처리하였다. 이는 YOLOv2가 5개의 앵커박스를 사용하는 반면 제안하는 알고리즘은 식 (3)과 같이 앵커박스의 크기를 추정하는 방법을 개선함으로써 1개의 앵커박스로 드론 탐지를 수행하기 때문이다.

[Table 2]에서 확인할 수 있듯이 두 알고리즘 모두 데이터셋 1에 대한 결과가 데이터셋 2의 결과보다 우수함을 보였다. 이는 데이터셋 1에 속하는 트레이닝 이미지들의 경우 드론이 큰 비율을 차지하고 드론의 중심 좌표 및 크기가 서로 유사하기 때문이다. 또한 제안한 알고리즘이 YOLOv2에 비하여 높은 정밀도, 재현율 및 평균 정밀도를 보였다.

[Fig. 5]는 두 알고리즘의 10000 epoch 동안 진행된 학습과정에서의 loss의 변화 그래프를 보여준다. 그래프에서 확인할 수 있듯이 제안하는 방법의 loss가 YOLOv2에 비해 상대적으로 빠르게 수렴하며, 또한 각 epoch간의 loss의 편차가 적음을 확인할 수 있다. 이는 제안하는 방식이 YOLOv2에 비해 안정적이고 신속하게 진행됨을 보여준다. 각 알고리즘 모두 데이



[Fig. 5] Loss graphs of YOLOv2 and the proposed algorithm during the training of Dataset 1 and Dataset 2

터셋 2에 대한 loss가 데이터셋 1의 loss보다 높은 값을 가짐을 확인할 수 있는데 이는 데이터셋 2에서의 드론의 크기가 상대적으로 작기 때문에 드론에 대한 추정 위치와 크기의 에러 및 드론 탐지에 대한 난이도가 높기 때문이다.

5. 결론

본 연구에서는 심층망 학습 기법을 적용하여 효율적으로 드론을 탐지하는 알고리즘을 제안하였다. 제안하는 방법은 기존의 YOLOv2의 방식에서 클래스 확률 학습과 좌표계의 회귀학습을 개선하여 제한된 학습 데이터에서도 효율적으로 학습이 가능하도록 하였으며, 이를 통하여 심층 신경망 기반 물체 인지 및 위치 인식에서 정밀도와 재현율을 높임으로써 드론 탐지에 대한 성능을 향상시켰으며, 이를 실험을 통하여 검증하였다. 추후 연구로는 기존의 칼만 필터나 회귀 신경망(recurrent neural network) 또는 심층 특징점 흐름(deep feature flow)와 같은 알고리즘을 적용하여 카메라 영상에서 실시간으로 드론을 탐지하고 추적하는 알고리즘을 개발하고자 한다.

References

- [1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Proceedings of International Conference on neural information processing systems*, pp. 1097-1105, 2012.
- [2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770-778, 2016.
- [3] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, "SSD: Single Shot Multibox Detector," *European Conference on Computer Vision*, pp. 21-37, 2016.
- [4] A. Kolesnikov and C. H. Lampert, "Seed, expand and constrain: three principles for weakly-supervised image segmentation," *European Conference on Computer Vision*, pp. 695-711, 2016.
- [5] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature

hierarchies for accurate object detection and semantic segmentation,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580-587, 2014.

- [6] R. Girshick, “Fast R-CNN,” *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1440-1448, 2015.
- [7] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” *Advances in neural information processing systems*, pp. 91-99, 2015.
- [8] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779-788, 2016.
- [9] J. Redmon and A. Farhadi, “YOLO9000: Better, Faster, Stronger,” *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7263-7271, 2017.
- [10] *Darkflow*, [Online], <https://github.com/thtrieu/darkflow> Accessed: March 13, 2017.



관심분야: 로보틱스

이 동 현

2007 경북대학교 전기 및 전자공학부 학사
 2009 KAIST 전기 및 전자공학부 석사
 2015 KAIST 전기 및 전자공학부 박사
 2016~현재 금오공과대학교 전자공학부
 조교수