

## 딥 러닝 기법을 이용한 무인기 표적 분류 방법 연구

최순현<sup>1)</sup> · 조인철<sup>2)</sup> · 현준석<sup>2)</sup> · 최원준<sup>2)</sup> · 손성환<sup>2)</sup> · 최정우<sup>\*1)</sup><sup>1)</sup> 한국과학기술원 전기및전자공학부<sup>2)</sup> LIG넥스원(주) 레이더 연구소Research for Drone Target Classification Method  
Using Deep Learning TechniquesSoonhyeon Choi<sup>1)</sup> · Incheol Cho<sup>2)</sup> · Junseok Hyun<sup>2)</sup> · Wonjun Choi<sup>2)</sup> ·  
Sunghwan Sohn<sup>2)</sup> · Jung-Woo Choi<sup>\*1)</sup><sup>1)</sup> The School of Electrical Engineering of KAIST, Korea<sup>2)</sup> Radar R&D, LIG NEX1 Co., Ltd., Korea

(Received 5 December 2023 / Revised 4 January 2024 / Accepted 15 February 2024)

## Abstract

Classification of drones and birds is challenging due to diverse flight patterns and limited data availability. Previous research has focused on identifying the flight patterns of unmanned aerial vehicles by emphasizing dynamic features such as speed and heading. However, this approach tends to neglect crucial spatial information, making accurate discrimination of unmanned aerial vehicle characteristics challenging. Furthermore, training methods for situations with imbalanced data among classes have not been proposed by traditional machine learning techniques. In this paper, we propose a data processing method that preserves angle information while maintaining positional details, enabling the deep learning model to better comprehend positional information of drones. Additionally, we introduce a training technique to address the issue of data imbalance.

Key Words : Deep Learning(딥 러닝), Drone Target Classification(드론 표적 분류)

## 1. 서론

실제 무인기의 출몰로 인해 취득 가능한 연안의 해상 환경에서의 레이더 정보를 기반으로 드론과 같은 저속 소형 무인기의 판별이 중요한 문제로 대두되고

있다. 특히, 소형 무인기의 경우 새와 동적 특성이 매우 유사하여 구분하기 어려운 문제점이 존재한다. 아울러 레이더 수신 데이터(RD Map)의 경우 무인기를 제외한 데이터는 충분한 양을 확보할 수 있는 데 반해, 무인기 데이터의 경우 발생 빈도가 상대적으로 낮아 데이터 확보가 어려워 데이터 불균형 문제가 발생한다. 현재까지 감시 레이더 데이터를 기반으로 새와 무인기를 구분하는 연구는 활발하게 진행되어 왔다<sup>3,4)</sup>.

\* Corresponding author, E-mail: jwoo@kaist.ac.kr

Copyright © The Korea Institute of Military Science and Technology

이전 연구에서는 주로 새와 무인기의 상이한 비행 패턴을 기반으로 머신러닝 기법을 적용하여 분류하려 했지만, 사용된 데이터의 필드는 주로 속력과 방향으로 제한되어 있어 고도와 같은 중요한 정보를 반영하지 못하고 있다. 또한, 무인기 데이터가 새와 항공기 데이터에 비해 현저히 부족한 상황에서 효과적인 학습 방법론을 제시하지 못하고 있다.

이러한 한계를 극복하기 위해 본 논문에서는 속도 뿐만 아니라 위치 정보를 최대한 활용하여 무인기, 새, 항공기의 비행 패턴을 효과적으로 학습시킴과 동시에 각도 정보를 보존하는 정규화 방식을 제안한다. 아울러 무인기 데이터가 새와 식별 가능 항공기에 비해 현저히 부족한 상황에서 딥 러닝 모델을 효과적으로 학습시키기 위한 기법에 대해 연구한다. 구체적으로, 데이터 불균형을 해결하기 위해 Loss Weight, ArcFace<sup>[1]</sup>, 회전 증강을 도입한다. 또한, 제안된 기법으로 학습된 모델의 특정 데이터 셋에 대한 과적합 여부를 검증하기 위해 학습 데이터와 테스트 데이터가 서로 다른 4개의 Cross Validation Set을 이용해 Macro F1 score를 측정한다.

추가로, 레이더 데이터로부터 추출된 표적의 Track을 딥 러닝 모델의 입력으로 사용하기 위해 일정한 길이의 Segment로 나누는 과정이 수반되는데, 본 논문에서는 Segment로 나누는 단위(Segment Length, SL) 및 간격(Hop)을 달리하며 성능을 측정해 딥 러닝 모델이 학습하는 맥락 정보에 따른 성능 변화를 확인한다. 마지막으로, Grad-CAM<sup>[6]</sup>을 이용하여 클래스별 강조된 부분을 통해 학습된 분류기가 클래스의 어떤 특징을 기준으로 분류했는지 분석한다.

본 논문의 구성은 다음과 같다. 2장에서는 레이더 데이터에 대해 각도 정보를 보존하며 정규화를 가능하게 하는 전처리 방법 및 데이터 불균형이 있는 경우에 딥 러닝 모델을 효과적으로 학습시키는 방법론에 대해 설명하고, 3장에서는 본 논문에서 제안하는 방법들을 검증하기 위한 실험의 구성 및 실험 결과에 대한 분석을 진행한다. 마지막으로 4장에서는 종합적인 결론에 대해 설명한다.

## 2. 레이더 데이터 전처리 및 학습 방법론

### 2.1 레이더 데이터 전처리

레이더 데이터의 필드 중 심층 신경망의 입력으로 활용되는 필드 정보는 아래 Table 1과 같다.

Table 1. Field categories in radar data

Field	Description	Unit
Slant ( $R$ )	Slant Range between Radar and Target	m
Azimuth ( $\phi$ )	Azimuth of the Target from Radar	degree
Elevation ( $\theta$ )	Elevation of the Target from Radar	degree
Velocity ( $v$ )	Velocity of the Target	m/s
Heading ( $\theta_H$ )	Direction of Motion	degree

레이더 Track의 필드 정보가 서로 다른 도메인의 정보로 구성되어 있어, 이를 고려한 전처리 방법이 필요하다. 전처리는 데카르트 좌표계 변환, 데이터 분할, 데이터 정규화, 위치 특이성 제거 순서로 진행된다. 우선 데카르트 좌표계 변환은 Table 1에서 제시된 필드 정보를  $2\pi$  모호성을 없애기 위해 데카르트 좌표계로 변환한다. 구체적인 수식은 아래 (1~5)와 같다.

$$X = R \cos(\theta) \cos(\phi) \tag{1}$$

$$Y = R \cos(\theta) \sin(\phi) \tag{2}$$

$$Z = R \sin(\theta) \tag{3}$$

$$v_{\perp x} = v \cos(\theta_H) \tag{4}$$

$$v_{\perp y} = v \sin(\theta_H) \tag{5}$$

여기서  $R$ 은 Slant,  $\theta$ 는 Elevation,  $\phi$ 는 Azimuth,  $v$ 는 Velocity,  $\theta_H$ 는 Heading을 나타낸다. 위 수식에서  $v_{\perp x}$ ,  $v_{\perp y}$ 는 각각  $x$ ,  $y$ 축 방향의 속력이다.

아울러 심층 신경망의 입력되는 데이터의 길이를 동일하게 맞추고, 레이더 Track의 시계열 정보를 효과적으로 학습시키기 위해 Fig. 1에서 제시된 것처럼 Track을 Segment Length(SL)로 분할한다. 이때, Track 내의 데이터들을 SL 간격으로 분할하는 것이 아닌 SL보다 작은 Hop 간격  $H$ 로 분할하여 주변 Segment와 중첩되도록 입력 Segment를 생성한다. 이에 대한 구체적인 수식은 아래 수식 (6)과 같다.

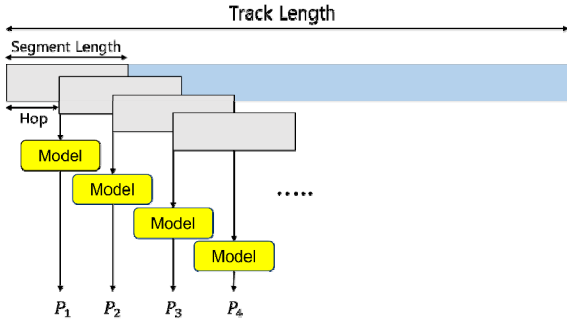


Fig. 1. Splitting the track into segments with segment length at hop intervals

$$\text{Segment}[s] = \text{Track}[sH : sH + SL] \quad (6)$$

이후 각 필드 정보는 딥러닝 모델의 효과적인 학습을 위해 정규화를 한다. 여기서  $(X, Y)$  및  $(v_{\perp x}, v_{\perp y})$ 의 경우, 각 필드를 개별적으로 정규화한다면 표적의 각도 정보가 소실될 수 있으므로 동일한 값으로 동시에 정규화하기 위해 극좌표계 거리의 RMS 값으로 아래 수식 (7~11)을 통해 정규화를 한다.

$$\hat{X}^{(t)}[n] = X^{(t)}[n] - \bar{X}^{(t)} \quad (7)$$

$$\hat{Y}^{(t)}[n] = Y^{(t)}[n] - \bar{Y}^{(t)} \quad (7)$$

$$\rho_{rms} = \sqrt{\frac{1}{TN} \sum_{t=1}^T \sum_{n=1}^N (\hat{X}^{(t)}[n]^2 + \hat{Y}^{(t)}[n]^2)} \quad (8)$$

$$[X^{r(s)}[n], Y^{r(s)}[n]] = \left[ \frac{\hat{X}^{(t)}[n]}{\rho_{rms}}, \frac{\hat{Y}^{(t)}[n]}{\rho_{rms}} \right] \quad (9)$$

$$v_{rms} = \sqrt{\frac{1}{TN} \sum_{t=1}^T \sum_{n=1}^N (v_{\perp x}^{(t)}[n]^2 + v_{\perp y}^{(t)}[n]^2)} \quad (10)$$

$$[v_{\perp x}^{r(t)}[n], v_{\perp y}^{r(t)}[n]] = \left[ \frac{v_{\perp x}^{(t)}[n]}{v_{rms}}, \frac{v_{\perp y}^{(t)}[n]}{v_{rms}} \right] \quad (11)$$

여기서  $T$ 는 Track의 개수이며,  $N$ 은  $t$ 번째 Track의 길이를 의미한다.  $(X, Y)$  필드에 대해서 수식 (7~9)을 적용하여 정규화를 하고,  $(v_{\perp x}, v_{\perp y})$  필드에 대해서는 수식 (10, 11)을 적용하여 정규화를 한다. 구체적으로 수식 (7)에서  $\bar{X}^{(t)}$ 는  $t$ 번째 Track의 평균을 나타내며,  $t$

번째 Track  $X^{(t)}[n]$ 에  $\bar{X}^{(t)}$ 를 빼주어  $\hat{X}^{(t)}$ 을 계산한다. 이 과정은  $Y$ 에 대해서도 동일하게 진행된다. 이렇게 계산된  $\hat{X}^{(t)}$ 와  $\hat{Y}^{(t)}$ 을 기반으로 RMS 값인  $\rho_{rms}$ 를 계산하고 수식 (8)과 같이 정규화한다. 속도에 대해서는 속도의 평균값을 빼는 과정이 없이 수식 (10)에서 RMS를 계산하고 수식 (11)과 같이 정규화를 한다.

이 과정으로 정규화를 진행한다면 각도 정보가 소실되지 않고 보존이 되는 장점이 있다. 고도  $Z$ 의 경우, 각도에 대한 의존성이 없으므로 아래 수식 (12)처럼 독립적으로 정규화를 진행한다.

$$Z' = \frac{Z - E(Z)}{\sigma(Z)} \quad (12)$$

여기서  $E$ 는 평균을 의미하며,  $\sigma$ 는 표준편차를 나타낸다. 위 정규화 과정을 거친 후, 학습 시 사용되는 Segment들의  $(X, Y)$  필드의 위치 특이성을 제거하여 절대 위치 대신 상대 위치를 사용한다. 이를 위해  $s$ 번째 Segment의  $(X, Y)$  필드 값에  $s$ 번째 Segment의 초기 값을 뺀다. 이 과정은 아래 수식 (13, 14)와 동일하다.

$$X^{r(s)}[n] = X^{r(s)}[n] - X^{r(s)}[0] \quad (13)$$

$$Y^{r(s)}[n] = Y^{r(s)}[n] - Y^{r(s)}[0] \quad (14)$$

## 2.2 데이터 불균형 해소를 위한 학습 방법론

새, 무인기, 항공기를 분류하기 위한 훈련을 할 때, 무인기의 데이터가 다른 클래스보다 현저히 적어 무인기 클래스에 대한 특징을 충분히 학습하지 못할 수 있다. 또한, 모델이 무인기에 대한 복잡한 패턴을 잡아내지 못하여 분류 성능의 하락으로 이어질 수 있으며, 특히 모델이 주로 다수 클래스의 예제에 노출되어 다수 클래스에 대해서는 잘 동작하지만, 소수 클래스에 대해서는 성능이 낮을 수 있다. 이러한 문제점을 해결하기 위해 아래 수식 (15)와 같이 클래스 가중치를 도입하여 클래스의 빈도에 반비례하도록 가중치를 설정하였다.

$$w = [w_1, w_2, \dots, w_C] \text{ where } w_i = \frac{1/n_i}{\sum_{k=1}^C (1/n_k)} \quad (15)$$

수식 (15)에서  $C$ 는 클래스의 총 개수이며,  $n_l$ 는  $l$ 번 째 클래스의 학습 데이터의 개수를 의미한다.

전통적인 Cross-Entropy를 통해 모델을 학습시킨다면 클래스 간의 거리를 최소화하는 방향으로 학습되는데, 단순히 클래스 간의 거리를 최소화하게 된다면 모델은 주로 다수 클래스에 대한 예측 성능은 향상되지만 소수 클래스에 대한 학습이 미흡해질 수 있다. 이러한 문제를 극복하고자 본 연구에서는 ArcFace<sup>[1]</sup>에서 제시된 방법과 같이 각도 마진 패널티를 도입한다. ArcFace는 특징이 유사한 데이터를 분류하는 태스크에 유용하게 사용되며, 특히 이상 소리 진단에서 STgram-MFN<sup>[5]</sup> 모델을 ArcFace를 사용하여 학습시킬 때 눈에 띄는 성능 향상을 보여주었다.

본 연구에서는 데이터 불균형 문제를 효과적으로 다루기 위해 앞서 제시한 ArcFace를 수식 (15)에서 제안한 Loss Weight과 결합하여 아래 수식 (17)을 도입한다.

$$\cos\theta_j = \frac{W_j^T x_i}{\|W_j\| \|x_i\|} \quad (16)$$

$$L = -w_{y_i} \log \frac{\exp(s \cos(\theta_{y_i} + m))}{\exp(s \cos(\theta_{y_i} + m)) + \sum_{j=1, j \neq y_i}^N \exp(s \cos\theta_j)} \quad (17)$$

수식 (16)은 수식 (17)에 사용되는 각도에 코사인을 취한 것을 나타내며, 가중치 행렬  $W$ 의  $j$ 번째 열의 가중치  $W_j$  및  $i$ 번째 Output Feature  $x_i$ 를 L2 정규화한 형태이다. 이는 가중치와 Feature를 정규화하여 가중치 사이의 각도에만 의존하도록 하여, 학습된 임베딩 Feature들은 반경이  $s$ 인 하이퍼 스피어 상에 분산 되도록 해주기 위함이다. 마지막으로 각도 마진  $m$ 을 추가하여  $i$ 번째 입력의 레이블  $y_i$ 과 일치하지 않는 클래스는 멀어지게 한다. 따라서 가중치를 반영한 ArcFace를 통해 클래스 간의 각도를 고려하여 클래스 간의 간격을 더 크게 만들어 데이터 불균형이 있는 상황에서 더 효과적인 학습을 할 수 있다.

추가로 새와 식별 가능 항공기에 비해 무인기의 데이터가 부족한 상황에서 데이터 불균형 문제를 해소함과 동시에 학습시키고자 하는 데이터의 일반적이고 다양한 특징을 인공지능 모델이 학습할 수 있도록 하

기 위해 본 논문에서는 회전 증강을 제시한다.  $Z$ 축을 기준으로 했을 때  $(X, Y)$  좌표 평면이 회전하여도 물리적 특성은 불변하므로 위치 정보를 아래 수식 (18)과 같이 변환하여 데이터를 증강할 수 있다.

$$\begin{pmatrix} X' \\ Y' \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} X \\ Y \end{pmatrix} \quad (18)$$

수식 (18)에서  $\theta \sim \text{Uniform}(0, 2\pi)$ 는 0과  $2\pi$  범위 내 uniform distribution에서 추출된다.

### 3. 실험 결과

#### 3.1 성능 지표

본 연구에서는 무인기 분류 모델의 평가 지표로 Macro F1 score를 사용한다. Macro F1 score는 다중 클래스 분류 문제에서 사용되는 평가 지표 중 하나로, 각 클래스에 대한 F1 score의 평균값을 계산한 것이다. 이에 대한 구체적인 수식은 다음과 같다.

$$\text{F1 score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (19)$$

$$\text{Macro F1 Score} = \frac{1}{N} \sum_{i=1}^N \text{F1 score}^{(i)} \quad (20)$$

F1 score는 수식 (19)와 같이 정밀도와 재현율의 조화 평균으로 계산되며, Macro F1 score는 수식 (20)과 같이 각 클래스에 대한 F1 score를 계산한 후, 그 값을 모든 클래스에 대해 평균을 계산하여 얻는다. Macro F1 score는 모든 클래스에 대한 성능을 동등하게 고려하므로 클래스 간의 불균형이 있을 때 유용한 지표이다.

#### 3.2 실험 설정

본 연구에서 사용된 데이터는 2초당 1번씩 스캔되어 생성되었으며, 클래스별 Track의 개수는 새 1,301개, 무인기 35개, 식별가능항공기 1,016개이다. Fig. 2에서는 클래스별 Track 길이의 분포를 보여준다. 아울러 SL(10, 30, 50, 70) 및 Hop(1, 5, 10, 30, 50, 70)을 달리하여 데이터 세트를 생성하였으며, 각 생성된 데이터 세트는 0.76:0.04:0.2 비율로 학습 데이터, 검증

데이터, 테스트 데이터로 분할된다. 또한, 과적합 유무를 판단하기 위하여 학습 및 테스트 데이터가 서로 다른 Cross Validation 4 Set를 생성한다.

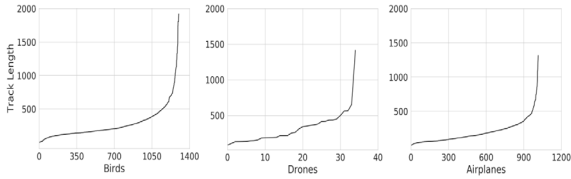


Fig. 2. The distribution of track lengths across classes in the drone detection dataset

데이터 불균형이 있는 조건에서 Loss Weight, ArcFace, 회전 증강의 방법을 도입하여 이에 따른 성능 변화를 확인한다. 실험에서 사용된 배치의 크기는 1024, Epoch 수는 500, 학습율은 0.0005로 설정하였다. 사용된 모델은 1D ResNet<sup>[2]</sup>이며, 모델의 입력으로 5개의 Field를 채널로 하는 1D 시퀀스가 입력된다. 사용된 필터의 수는 512이고, 2개의 Layer로 훈련한다. ArcFace를 손실함수로 사용할 때, 마진  $m$ 은 0.6이며 스케일링 파라미터( $s$ )는 30을 사용한다.

### 3.3 분류기 모델 성능 평가

Table 2는 SL 70, Hop 5인 경우 Loss Weight, 회전 증강, 그리고 ArcFace의 조합에 따라 모델의 성능 변화를 나타낸다. 1번과 5번에서는 Loss Weight의 유무에 따른 성능 변화를 나타내며, Loss Weight을 사용하는 경우, 데이터가 다른 클래스에 비해 적은 무인기 데이터의 가중치를 높게 주므로 무인기에 대한 비중이 크게 되어 True Positive Rate이 커지지만 반대로 다른 클래스에 대한 가중치는 낮아지므로 새가 무인기로 분류되는 데이터가 많아져 성능이 1번 대비 약간 하락한다. 1번과 3번에서는 ArcFace의 유무에 따른 성능 변화를 나타내며, ArcFace만 적용하는 3번의 경우에는 샘플에 대한 클래스 간 각도가 벌어지면서 결정 경계가 명확해지므로 새를 무인기로 분류하는 비율은 낮아진다. 하지만 새와 특성이 비슷한 무인기 데이터를 새로 분류하는 경향이 있어 무인기에 대한 True Positive Rate이 작아진다.

이러한 Trade off를 해결하기 위해 7번에서는 ArcFace와 Loss Weight을 결합한 Loss로 모델을 학습시켰으며, 결과는 Macro F1 score가 0.932로 1번과 비교하여

1 %가 향상되었다. 아울러 대다수 조건에서 Rotation Augmentation(회전 증강)을 적용하는 것이 적용하지 않은 것보다 낮은 성능을 보여주었는데, 이는 앞에서 설명한 Trade off를 해결하지 못한 상태에서 Rotation Augmentation을 적용하여 효과적으로 공간 정보를 학습시키지 못했기 때문이다. 반면 8번에서는 Trade off를 해결한 상태에서 Rotation Augmentation을 적용하여 효과적으로 공간 정보를 학습시킬 수 있었고, 결과적으로 Loss Weight, ArcFace, Rotation Augmentation을 모두 적용한 조건에서 가장 높은 성능을 달성할 수 있었다. 다음 Fig. 3은 Loss Weight, ArcFace, Rotation Augmentation을 적용한 모델의 confusion matrix를 보여준다.

Table 2. The average macro F1 score across 4 sets of cross validation for loss weight, rotation augmentation, and ArcFace(SL 70 and Hop 5)

	Loss Weight	Rotation Augmentation	ArcFace	Average
1	X	X	X	0.922
2	X	O	X	0.913
3	X	X	O	0.917
4	X	O	O	0.907
5	O	X	X	0.915
6	O	O	X	0.895
7	O	X	O	0.932
8	O	O	O	<b>0.937</b>

Fig. 3에서 제시된 confusion matrix를 살펴보면 항공기가 무인기로 분류되는 부분은 거의 0 %로 모델이 두 클래스를 명확하게 구별하고 있으며, 항공기의 특징이 명확하게 학습되었다는 것을 나타낸다. 그러나 새와 무인기에 대한 모델의 분류 성능이 낮으며 그 비율은 새이지만 무인기로 분류되는 경우가 1 %, 그리고 무인기이지만 새로 분류되는 경우가 약 14 %로 나타났다.

다음 실험으로 본 논문에서는 SL 및 Hop에 따른 모델의 성능을 살펴본다.

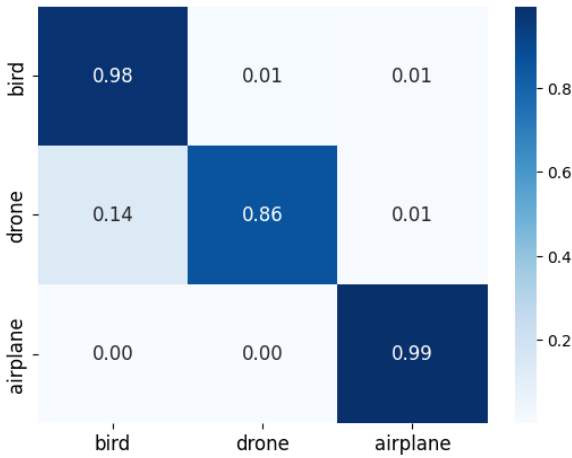


Fig. 3. The confusion matrix trained with loss weight, ArcFace, and rotation augmentation(SL 70 and Hop 5)

Table 3. Macro F1 score with the variation of SL and Hop

Hop	Segment Length (SL)			
	10	30	50	70
1	0.879	<b>0.922</b>	0.948	0.944
5	<b>0.888</b>	0.920	<b>0.952</b>	0.948
10	0.850	0.895	0.937	<b>0.959</b>
30	—	0.899	0.930	0.958
50	—	—	0.871	0.904
70	—	—	—	0.891

Table 3은 SL 및 Hop을 변화시키며 Loss Weight, 회전 증강, ArcFace로 학습시킨 모델의 Macro F1 score를 나타낸다. 우선 Table 3에서 같은 Hop인 경우 SL이 커지면 커질수록 Macro F1 score가 일관적으로 높아지는 것을 확인할 수 있다. 이는 모델이 데이터 길이(SL)의 데이터를 입력받아 데이터 분류를 진행할 때, 그 길이가 너무 작아 데이터 내에서 해당 클래스의 특징을 파악할 수 없는 경우, 모델이 판단할 수 있는 맥락 정보가 작아져 정확한 분류를 하는 것이 불가능하기 때문이다. 반면, SL이 크다면 모델이 판단할 수 있는 맥락 정보가 충분하여 SL이

커짐에 따라 Macro F1 score가 높아지는 것을 확인할 수 있다.

또한, Hop 간격에 따라서는 Hop 간격이 SL과 동일한 경우보다 Hop 간격이 SL과 작은 경우가 더 높은 성능을 보여준다. 특히, SL이 30인 경우 Hop 간격이 작아지면 작아질수록 성능이 더 높아지지만, SL이 10, 50, 70인 경우 각각 5, 5, 10의 Hop 간격에서 최고의 성능을 보였다. 이러한 결과를 통해 SL마다 최적의 성능을 보이는 Hop이 다르며, 그 이유는 Hop 간격이 작아질수록 생성되는 입력 데이터 사이 겹치는 영역이 많아져 모델의 과적합이 발생하기 때문이다. Fig. 4는 Table 3에서 가장 높은 성능을 보여준 SL 70, Hop 10인 경우 Confusion Matrix이다. Fig. 4의 Confusion Matrix를 살펴보면 식별 가능 항공기가 무인기로 분류되는 부분은 0%로 모델이 이 두 클래스를 명확하게 구별하고 있으며, 새와 무인기에 대한 모델의 분류 성능 또한 Fig. 3에서 제시된 분류 성능보다 높음을 확인할 수 있다.

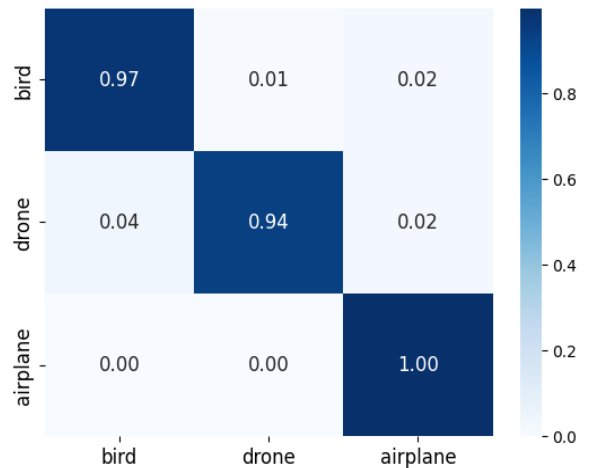


Fig. 4. The confusion matrix trained with loss weight, ArcFace, and rotation augmentation(SL 70, Hop 10)

### 3.4 학습된 분류기 모델 분석

학습된 분류기가 각 클래스 별로 어떤 특징을 기준으로 분류하였는지 분석하기 위해 Grad-CAM<sup>[6]</sup>을 적용한다. Fig. 5에 나타난 새 데이터의 경우 속도(1행)가 0에 가까운 낮은 값을 가지며, 속도 변동이 크게 없다. 아울러 각도(2행) 변동이 심하며, 낮고 일정한 고도

(3행)로 비행하는 특성을 보인다. 또한, Grad-CAM(4행)으로 강조된 영역을 살펴보면 각도의 변동이 심한 부분에서 공통으로 영역이 강조된 것으로 보아 각도의 변동이 새의 주요한 특징임을 확인할 수 있으며, 종합적으로 낮은 속도 및 낮은 고도인 상황에서 각도의 변동이 심한 경우 새로 판단하는 것을 확인할 수 있다. 반면 Fig. 6에 나타난 무인기 데이터를 살펴보면 강조된 부분은 각도 변화가 크게 없었으며 낮고 일정한 속도 및 낮고 일정한 고도에서 무인기로 판단하였다는 것을 확인할 수 있었다. 마지막으로, Fig. 7에 나타난 항공기 데이터의 강조된 영역의 특징은 각도의 변화가 새 보다 비교적 적으며, 속도가 새 혹은 무인기보다 큰 특징이 있었다. 또한, 고도가 다른 클래스보다 높은 것으로 보아 보조적인 정보로 활용되었음을 알 수 있다.

#### 4. 결론

본 논문에서는 기존의 연구에서 반영하고 있지 않던 위치 정보를 효과적으로 사용할 수 있는 전처리 방식을 통해 표적의 비행 패턴에 대한 특징을 추출할 수 있었으며, 데이터 불균형이 있는 상황에서 인공지능 기반 무인기 표적 분류기의 학습을 위해 Loss Weight, ArcFace, 회전 증강을 도입했다.

이 방법론의 효과를 검증하기 위해 Cross Validation 4 Set를 구성하여 성능을 검증하였으며, 실험 결과 이 Loss Weight, ArcFace, 회전 증강을 모두 적용한 경우가 가장 높은 성능을 보였다. 추가로, 데이터를 자르는 길이(SL)와 자르는 간격(Hop)을 변화시켜 Macro F1 score를 측정된 결과, 데이터를 자르는 길이가 크면 클수록 높은 성능을 나타냈지만, Hop의 경우 Hop이 작을수록 성능이 향상되었지만, 너무 작아지면 과적합의 영향으로 성능이 하락하는 경우도 있었다.

또한 Grad-CAM을 활용하여 학습된 분류기가 각 클래스 별로 어떤 특징을 기준으로 분류하는지에 대한 분석을 수행한 결과, 새로 분류하는 주요 특징은 속도와 고도가 낮고 일정하며 각도 변동이 큰 경우였다. 반면 무인기의 경우, 속도와 고도가 낮고 일정하며 각도 변동이 작은 경우였다. 항공기의 경우 다른 클래스의 속도보다 높은 경우가 많았으며, 이를 통해 항공기의 주요한 특징은 높은 속도에 기인하는 것으로 확인되었다.

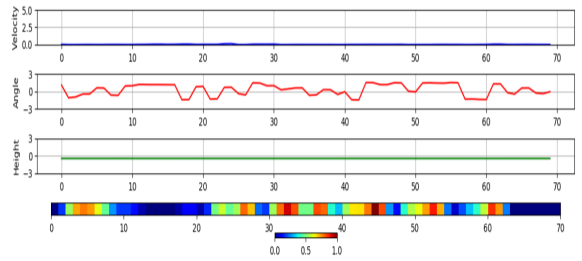


Fig. 5. The velocity, angle, height, and Grad-CAM results of instances correctly classified as birds

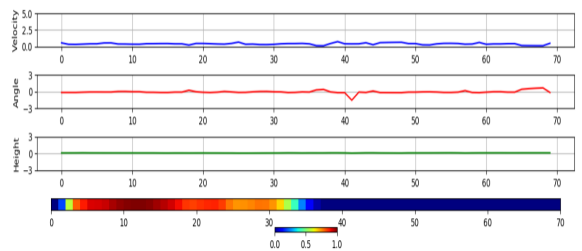


Fig. 6. The velocity, angle, height, and Grad-CAM results of instances correctly classified as drones

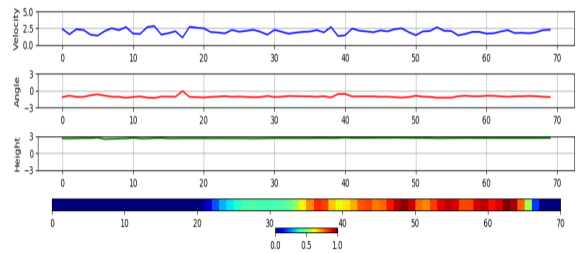


Fig. 7. The velocity, angle, height, and Grad-CAM results of instances correctly classified as airplanes

#### References

[1] Deng, J., Guo, J., Xue, N., & Zafeiriou, S., "Arcface: Additive angular margin loss for deep face recognition," In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 4690-4699, 2019.

- [2] He, K., Zhang, X., Ren, S., & Sun, J., "Deep residual learning for image recognition," In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770-778, 2016.
- [3] Kengeskanov, M., Seghrouchni, A. E. F., Zitar, R. A., & Barbaresco, F., "Drone/Bird Classification Based on Features of Tracks Trajectories," In Proceedings of 2023 IEEE Aerospace Conference, pp. 1-8, 2023.
- [4] Liu, J., Xu, Q. Y., & Chen, W. S., "Classification of bird and drone targets based on motion characteristics and random forest model using surveillance radar data," IEEE Access, 9, pp. 160135-160144, 2021.
- [5] Liu, Y., Guan, J., Zhu, Q., & Wang, W., "Anomalous sound detection using spectral-temporal information fusion," In Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 816-820, 2022.
- [6] Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D., "Grad-cam: Visual explanations from deep networks via gradient-based localization," In Proceedings of the IEEE international conference on computer vision, pp. 618-626, 2017.