M.S. THESIS

Convolutional Neural Network-based UAV Classification using Radar Spectrogram

레이더 스펙트로그램을 사용한
컨볼루션 신경망 기반 무인항공기 분류

BY

DONGSUK PARK

FEBRUARY 2021

Intelligent Systems
Department of Transdisciplinary Studies
Graduate School of Convergence Science and Technology
SEOUL NATIONAL UNIVERSITY

M.S. THESIS

Convolutional Neural Network-based UAV Classification using Radar Spectrogram

레이더 스펙트로그램을 사용한 컨볼루션 신경망 기반 무인항공기 분류

BY

DONGSUK PARK

FEBRUARY 2021

Intelligent Systems
Department of Transdisciplinary Studies
Graduate School of Convergence Science and Technology
SEOUL NATIONAL UNIVERSITY

Convolutional Neural Network-based UAV Classification using Radar Spectrogram

지도 교수 곽노준

이 논문을 공학석사 학위논문으로 제출함
2020년 12월

서울대학교 대학원
융합과학부 지능형융합시스템전공
박동석

박동석의 공학석사 학위논문을 인준함
2021년 01월

위원장 박재홍 (인)

부위원장 곽노준 (인)

위원 안정호 (인)

Abstract

With the upsurge in using Unmanned Aerial Vehicles (UAVs) in various fields, identifying them in real-time is becoming an important issue. However, the identification of UAVs is difficult due to their characteristics such as Low altitude, Slow speed and Small radar cross-section (LSS). To identify UAVs with existing deterministic systems, the algorithm becomes more complex and requires large computations, making it unsuitable for real-time systems. Hence, we need a new approach to these threats. Deep learning models extract features from a large amount of data by themselves and have shown outstanding performance in various tasks. Using these advantages, deep learning-based UAV classification models using various sensors are being studied recently.

In this paper, we propose a deep learning-based classification model that learns the micro-Doppler signatures (MDS) of targets represented on radar spectrogram images. To enable this, first, we recorded five LSS targets (three types of UAVs and two different types of human activities) with a frequency modulated continuous wave (FMCW) radar in various scenarios. Then, we converted signals into spectrograms in the form of images by Short time Fourier transform (STFT). After the data refinement and augmentation, we made our own radar spectrogram dataset. Secondly, we analyzed characteristics of the radar spectrogram dataset using the ResNet-18 model and designed the lightweight ResNet-SP model for the real-time system. The results show that the proposed ResNet-SP has a training time of 242 seconds and an accuracy of 83.39 %, which is superior to the ResNet-18 that takes 640 seconds for training with an accuracy of 79.88 %.

keywords: CNN, Classification, UAV, FMCW radar, STFT, Spectrogram, MDS

student number: 2019-29945

Contents

Abstract **i**

Contents **ii**

List of Tables **iv**

List of Figures **v**

1 Introduction **1**

2 Related Works **5**

2.1 Micro Doppler Signature (MDS) 5

2.2 Classification of UAVs using MDS 6

3 Dataset Generation **9**

3.1 Measurement 10

3.2 Pre-processing 12

4 Models **21**

4.1 ResNet-18 22

4.2 ResNet-SP 27

5 Experiment **32**

5.1 Experiment Result 32

5.2 Training Details 33

6 Conclusion **34**

Abstract (In Korean) **38**

List of Tables

3.1 The movements for each target and the settings for recording. . . 12

3.2 Spectrogram width of two targets before and after the data refinement. 18

3.3 Data augmentation method applied to generating training data . 20

3.4 Number of samples for each class in radar spectrogram dataset . 20

4.1 Accuracy of the ResNet-18 according to the signal form of the radar spectrogram 24

4.2 Accuracy of the ResNet-18 on two different noises. 25

4.3 Accuracy of the ResNet-18 according to conv. groups and layers. 26

4.4 Accuracy of the ResNet-18 according to the dilated kernel. . . . 29

5.1 Average test accuracy of five runs and their standard deviation. . 32

5.2 Computation time; Training time and inference time. 33

List of Figures

2.1 Radar spectrogram of walking: approaching and turning away. . 6

2.2 Spectrogram of UAVs; wing-flap (top), quad-copter (middle), and fixed-wing (bottom) 7

3.1 X-band FMCW radar (Ancortek's SDR-KIT 980AD2) image and Specification 10

3.2 Target images; three types of UAVs and two different human activities 11

3.3 Sequential video frames of a specific movement for each target: (a) Metafly flight from right to left, (b) Mavic Air 2 flight from front to back, (c) Disco flight coming forward, (d) Walking from right to left, (e) Sit-walking coming forward 13

3.4 Spectrogram resolution of Walking according to window sizes; as the window size increases, the frequency resolution increases. 14

3.5 Spectrogram resolutions of Metafly flight according to window overlap ratios: as the window overlap ratio increases, wing-flaps of Metafly appear more clearly 15

3.6 Spectrogram of background clutter (left) and Mavic Air 2 (right). The red box is the non-recorded section of the target. 17

3.7 Spectrograms of Mavic Air 2 before the refinement process (left) and after the refinement process(right). red box: chopped images with an average intensity below the threshold, blue box: chopped images with an average intensity above threshold . . . 18

3.8 Training loss curves; before the refinement process(left), after the refinement process (right) 19

4.1 Plane CNN layers (left), Residual block (right) 23

4.2 ResNet-SP Architecture 27

4.3 The receptive fields : 3x3 kernel (left), 5x5 kernel (center), 2-dilated 3x3 kernel (right) 28

4.4 Training loss curves; before the gradient clipping (left), after the gradient clipping (right) 30

4.5 Softmax output distributions; normal data (left), outlier data (right) 31

Chapter 1

Introduction

In recent years, the rapid development of UAV technology has increased the usage of UAVs in various fields such as agriculture, industry, and military fields. Even though the use of UAVs brings convenience to life, it poses severe threats if abused by enemies or terrorists. There have been reports of attempts to assassinate key figures or to attack oil facilities using UAVs loaded with small bombs. If UAVs are used to attack with biochemical weapons, damages will be more severe. Therefore, the real-time early detection and identification of UAVs are essential. However, it is difficult to identify UAVs due to their Low altitude, Slow speed and Small radar cross-section (LSS) characteristics. In the case of deterministic rule-based model, due to LSS characteristics of UAVs, the algorithm becomes more complex, which increases computations. Hence, effective alternatives enabling real-time identification of these new threats are required. Recently, deep learning-based classification models are being applied to various tasks. These models learn features from a large amount of data themselves and have especially shown outstanding performance in image classification tasks. In the UAV classification task, deep learning-based classification models using various sensors are being actively studied. Saqib et al. [15] performed the UAV identification task using pre-trained models of ZF-Net [22] and VGG-16 [17] on the bird-vs-UAV dataset with 2,727 video frames and showed the highest mean average precision of 0.66 at VGG-16. Seo et al. [16] recorded acoustic signals of UAVs and non-UAVs outdoors, obtained a 2D image by applying STFT, and tested the dataset with a self-designed CNN model and showed a detection rate of 98.97% and a false alarm rate of 1.28%. Radar poses several advantages over other sensors. Namely, radar is less affected by weather and low visibility environments than optical sensors. And unlike acoustic sensors, it is not vulnerable to ambient noise. Because of these advantages, deep learning-based UAV classification studies that birds vs. UAVs, UAVs vs. UAVs, and etc. are being conducted using radar sensors [20]. Among them, there are deep learning-based classification models that learn micro-Doppler signatures of UAVs represented on the radar spectrogram. A moving target generates the micro-Doppler effect by partial movements like the pendulum, rotation and vibration along with a constant Doppler shift induced from the main body. This MDS is a unique characteristic of the target and is well represented visually in the radar spectrogram, which is the STFT result of the radar signal. It is possible to train the radar signal with the deep learning-based image classification model because it is converted into the optical image format. Related studies showed high accuracy in UAV classification, but most of them measured only limited flight dynamics such as hovering, so it does not contain the various UAV flights in reality. And in the process of transforming the UAV radar signal into the image form such as RGB and grayscale, data distortion or loss may occur.

In this study, we propose a deep learning-based UAV classification model that learns MDS that is suitable for real-time systems by increasing the data diversity and designing a lightweight and stable model while considering the characteristics of the spectrogram dataset. To diversify the dataset, radar signals were recorded by diversifying the target type and the movement assuming a real-life scenario. We recorded five LSS targets (3 types of UAVs and 2 different human activities) each with an FMCW radar. UAV targets were selected according to flight type (multicopter, fixed wing, wing flap) and walking and sit-walking were chosen as ground moving targets. UAV signals were recorded by changing altitude, speed, and direction, and human signals were recorded by changing direction and range at a constant walking speed. The signals were converted into spectrogram images through STFT. Then, through the data refinement and augmentation, we generated own the radar spectrogram dataset. Then, we analyzed characteristics of radar spectrogram data using the ResNet-18 model [7], which is a popular image classification model. With this model, we analyzed the performance according to the radar spectrogram data type and the model structure. Based on this, we designed a lightweight ResNet-SP model which is more suitable for real-time systems. Additionally, we improved model's stability by applying anomaly detection and gradient clip methods to reduce learning instability caused by abnormal data. The results show that ResNet-SP has 83.39% of accuracy which is higher than 79.88% from the ResNet-18. Also, the training time is 242 seconds with our proposed model, which is faster than 640 seconds of ResNet-18. Furthermore, the ResNet-SP model is more stable through out the training process.

Our main contributions are summarized as follows:

(1) Generated radar spectrogram dataset covering various movements of targets.

(2) Analyzed radar spectrogram dataset characteristics with the ResNet-18 model.

(3) Designed lightweight ResNet-SP model suitable for real-time system.

The paper is organized as follows. Chapter 2 describes the micro-Doppler signature and introduce related works. Chapter 3 describes the generation process of the radar spectrogram dataset through radar measurement and pre-processing. Chapters 4 analyze data characteristics using the ResNet-18 model and design a lightweight ResNet-SP model for the real-time system. Chapter 5 shows experimental results and finally, Chapter 6 concludes the paper.

Chapter 2

Related Works

This chapter describes the micro-Doppler signature which is the main feature of the radar spectrogram dataset. And we introduce deep learning-based UAV classification studies that learn the MDS represented on the radar signals of moving targets.

2.1 Micro Doppler Signature (MDS)

The Doppler effect of a radar is a frequency shift or wavelength change generated from the reflected radar signal when a target moves or changes in a relative distance to an observer. Radar signal interacts with the target in motion and the returned signal changes its characteristics. While the Doppler effect is generated by a bulky motion of the body of the target, its micro-movements from the part of the body can generate such micro-frequency shifts, which is called the micro-Doppler effect [2]. This micro-Doppler signal is created by all subtle movements of a target, such as vibration, rotation, pendulum, etc., unique patterns or characteristics occur depending on the object type or different movements of the same

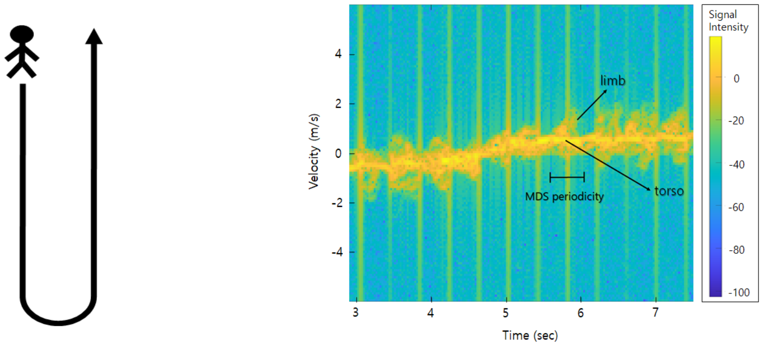


Figure 2.1: Radar spectrogram of walking: approaching and turning away.

object. Figure 2.1 shows the micro-Doppler signature of walking represented on the radar spectrogram. On this spectrogram, we can see the MDS shape generated by swinging limbs around the torso signal. And the shape of this MDS is represented differently depending on the length of the limb, the swinging period, and the angle. Hence, we can use this MDS shape as the main feature of the classification task.

In UAV, MDS appears differently according to the flight types, and even within the same flight type, it appears differently depending on the number of rotors, blade length, etc. Figure 2.2 is a radar spectrogram for UAVs of different flight types. The second column shows the spectrograms of the UAVs with the fuselage fixed at short-range, and the third column is the spectrogram of free-flight. We can see that each UAV spectrogram appears differently and using this characteristic we can further perform deep learning-based UAV classification.

2.2 Classification of UAVs using MDS

There are several deep learning-based UAV classification studies that learn the MDS represented on the radar spectrogram of moving target.

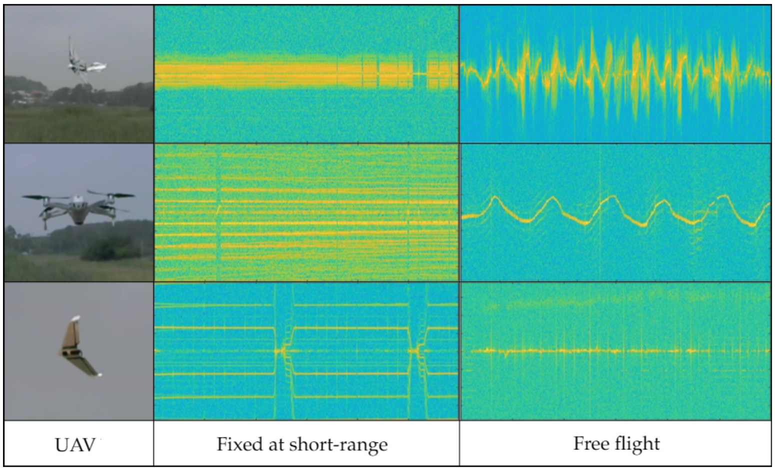


Figure 2.2: Spectrogram of UAVs; wing-flap (top), quad-copter (middle), and fixed-wing (bottom)

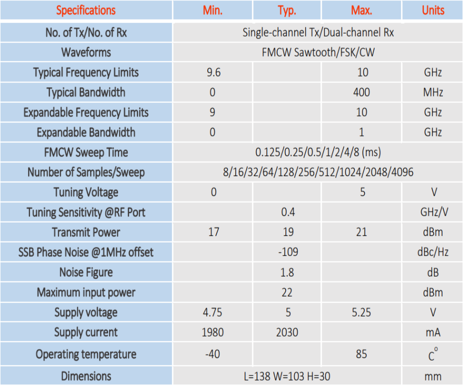
Choi et al. [3] suggested a deep-learning model which classifies three types of UAVs (Vario helicopter, DJI Phantom 2, and DJI S1000+) based on the micro-Doppler signatures in the spectrogram and confirmed the feasibility of the application of deep learning-based models in the UAVs classification. Raman et al.[13] proposed a radar spectrogram-based deep learning model that classifies birds and UAVs. They applied the following methods to mitigate the lack of diversity and quantity of UAV radar spectrogram data. First, they added the flying dynamics of UAVs for diversify the dataset. They added more flight dynamics such as radial traversing, pointing out that other previous studies such as [3] only utilized the hovering data of UAVs. Second, they applied the transfer learning [10] commonly used in the optical image classification to solve the lack of radar spectrogram data. Transfer learning is a method that can improve performance by transferring well-trained parameters of a network trained with a large dataset to the network with a small amount of data. To apply this method, the authors transformed the radar signal into an RGB spectrogram of the same color scale as the optical image and trained the dataset with the modified GoogleNet [19]. They showed high performance of over 99%.

However, the datasets still do not cover various flight movements of UAVs. And radar signal has very different characteristics from the optical image, so the data characteristics may be distorted or omitted in the process of transforming the color scale of radar spectrogram data to suit the optical image classification network.

Chapter 3

Dataset Generation

This chapter covers the whole process of generating a radar spectrogram dataset. UAV flight data is mainly generated by radar-related companies, agencies, or the military for particular purposes. There are no publicly released datasets and references so it is difficult to study. In particular, the dataset recorded by radar sensors is even rarer. Due to these characteristics of the research field, many researchers generate their own datasets and carry out research. However, most of the datasets have a lack of diversity, such as measuring only at short-range or limited movements of UAVs to obtain a clear signals. Considering the real-life scenarios, we measured various movements of targets with radar and pre-processed the measured signal to generate our radar spectrogram dataset. Section 3.1 describes the radar measurement process for targets, and section 3.2 describes the pre-processing process of the measured signals.



Specifications	Min.	Typ.	Max.	Units
No. of Tx/No. of Rx	Single-channel Tx/Dual-channel Rx			
Waveforms	FMCW Sawtooth/FSK/CW			
Typical Frequency Limits	9.6		10	GHz
Typical Bandwidth	0		400	MHz
Expandable Frequency Limits	9		10	GHz
Expandable Bandwidth	0		1	GHz
FMCW Sweep Time	0.125/0.25/0.5/1/2/4/8 (ms)			
Number of Samples/Sweep	8/16/32/64/128/256/512/1024/2048/4096			
Tuning Voltage	0		5	V
Tuning Sensitivity @RF Port		0.4		GHz/V
Transmit Power	17	19	21	dBm
SSB Phase Noise @1MHz offset		-109		dBc/Hz
Noise Figure		1.8		dB
Maximum input power		22		dBm
Supply voltage	4.75	5	5.25	V
Supply current	1980	2030		mA
Operating temperature	-40		85	C°
Dimensions	L=138 W=103 H=30			mm

Figure 3.1: X-band FMCW radar (Ancortek's SDR-KIT 980AD2) image and Specification

3.1 Measurement

Radar signals are less vulnerable to low visibility and weather conditions than video signals and have fewer restrictions on the line of sight (LOS), which indicates a straight line between the target and the sensor. These radars are divided into two types by the principle of radio wave emission; (1) 'pulse radar,' which transmits pulse signals and receives signals reflected from objects and (2) 'continuous wave (CW) radar', which continuously transmits and receives signals without a pause. To detect time-varying changes for low radar cross section (RCS) targets, the continuous wave radar is suitable and we decided to use an FMCW radar that continuously emits a frequency modulated signal at regular intervals to obtain time information. Our model is Ancortek's SDR KIT 980AD2 and the specifications are described in Figure 3.1. Additionally, to select well-recorded files, we installed a video camera synchronized with the radar and double-checked video files and radar spectrograms.

We recorded five different LSS targets with the FMCW radar. Assuming the

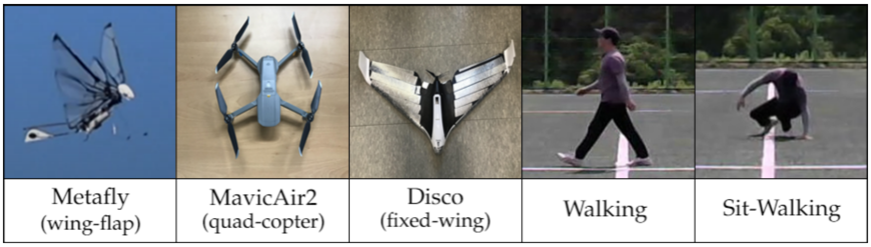


Figure 3.2: Target images; three types of UAVs and two different human activities

enemy approaching the local area, we selected three types of UAVs as aerial moving targets and two different human activities as ground moving targets. The three flight types of UAVs are ‘Metafly’, a wing-flapping drone that mimics wings of a bird and ‘Disco’, a fixed-wing, and ‘Mavic Air 2’, a quad-copter (4 rotors). ‘Walking’ and ‘Sit-walking’ are data of the same person. Figure 3.2 shows the images of the five targets.

We recorded various movements of targets within the 100m range. UAVs were recorded while changing altitude, speed, and direction freely, and humans were recorded while changing the distance and the direction at a constant pace. Only two UAVs (Metafly and Disco) were given some restrictions for the proper recording. Metafly was recorded within the 10m range because of its low signal intensity. Disco was recorded only in the left and right, front and rear, and concentric circular flight at an altitude of 10m with low-velocity settings because of its high-speed and wide turning radius. Disco is equipped with a single rotor at the rear of the fuselage so that thrust acts only forward and changes direction gradually by changing the Angle of the Attack (AoA) of the aileron at the wing-tips. So it requires a wide turning radius and often be placed outside of the radar’s detection range. Besides, because it moves at high speed, it quickly

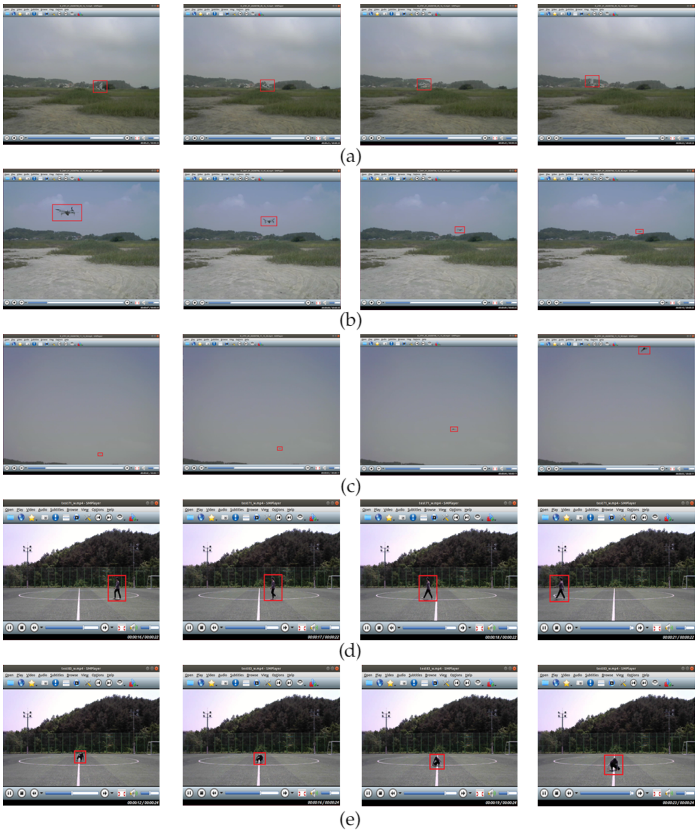
Parameter	Metafly	Mavic Air 2	Disco	Walking	Sit-walking
Alt. / Range (m)	0 - 10 / 0 - 10	0 - 10 / 0 - 100	10 / 0 - 100	0 / 0 - 100	0 / 0 - 100
Radar angle(vertical)	0	0	+ 20 º	0	0
Movement	Free flight	Free flight	Circular-flight	Free	Free

Table 3.1: The movements for each target and the settings for recording.

leaves the radar's detection range. Table 3.1 shows the movements for each target and the settings for recording. And Figure 3.3 is sequential video frames of a specific movement for targets. We operated Metafly and Mavic Air 2 manually, and Disco operated automatically by entering flight plans through the 'Free Flight Pro' mobile application. We recorded many times for each target and removed abnormal files such as overly noisy files or files intruded by other objects by cross-checking video files and spectrograms. Basically, we selected 10 well-recorded files for each target and divided the training dataset and the test dataset by a ratio of 8:2. (The exception is for Disco; 25 files were used because the recorded section was too short)

3.2 Pre-processing

In the pre-processing step, the recorded radar signals are transformed into spectrogram images through STFT and after the data refinement and augmentation, the radar spectrogram dataset is generated. The data refinement is the step for removing the spectrogram section in which the target is not recorded. To do this, we cut the spectrogram into short time intervals and removed cut images with an average intensity below a threshold. To increase the amount of data, we applied three data augmentation methods, keeping the format of the spectrogram: the x-axis represents time, the y-axis represents frequency and the color at each


(a)

(b)

(c)

(d)

(e)

Figure 3.3: Sequential video frames of a specific movement for each target: (a) Metafly flight from right to left, (b) Mavic Air 2 flight from front to back, (c) Disco flight coming forward, (d) Walking from right to left, (e) Sit-walking coming forward

point represents the amplitude of a specific frequency at a specific time.

In the signal processing of STFT, we applied different window sizes (128, 256, and 512) and the window overlap ratios (50%, 70%, and 85%) to get spectrograms of different resolutions. In addition, we applied the vertical flip after the data refinement to obtain spectrograms with reversed radial velocity sign.

A spectrogram [6] reveals the instantaneous spectral content of the time-domain signal and the spectral content variations over time. A spectrogram is

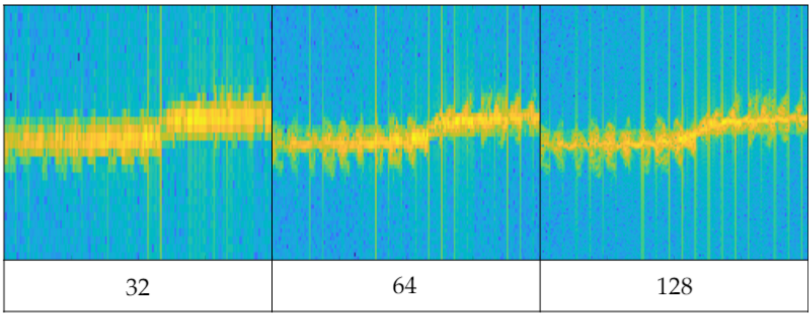


Figure 3.4: Spectrogram resolution of Walking according to window sizes; as the window size increases, the frequency resolution increases.

obtained by the squared magnitude of the STFT of a discrete signal. With the spectrogram, we can visually observe the spectrum of frequency changing over time. But when converting the spectrogram, finite-size sampling in a recorded signal may result in a truncated waveform from the original continuous-time signal, introducing discontinuities into the recorded signal. These discontinuities are represented in the FFT as high-frequency components, even though not present in the original signal. This appears as a blurry form, rather than a clear form on the spectrogram. This is called 'spectral leakage' because it looks as if energy is leaking from one frequency to another. In order to mitigate the spectral leakage, window functions are generally applied. The spectrogram resolution is determined by the window size and there is a trade-off between time and frequency resolution [4]. Figure 3.4 shows the differences in the spectrogram resolution according to window sizes.

If a narrow window size is applied, a fine time resolution can be obtained due to a short time interval, but the frequency resolution is degraded due to the wide frequency bandwidth. Conversely, if wide window size is applied, a fine frequency resolution is obtained due to a wide time interval and a narrow

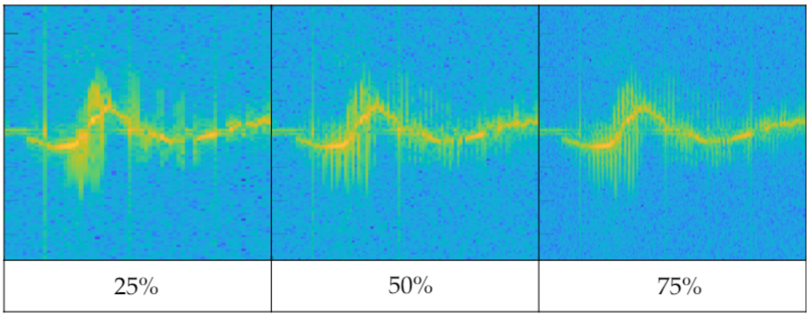


Figure 3.5: Spectrogram resolutions of Metafly flight according to window overlap ratios: as the window overlap ratio increases, wing-flaps of Metafly appear more clearly

frequency bandwidth, but the time resolution is degraded. The higher the resolution, the more detailed the object's MDS waveform is represented. We generated spectrogram images with different resolutions by applying three window sizes (128, 256, 512) to the original signal.

Even when the window size is determined, if several different frequencies are included in a window, they may not be distinguishable. One can use a window overlap that applies for redundancy when applying the next window in the STFT process to reduce this effect. The higher the overlap ratio is applied, the higher the resolution, but it requires more computations. Figure 3.5 shows the differences in the spectrogram resolution of Metafly (wing flapping UAV) according to different window overlap ratios. The higher the overlap ratio in the given window size, the more detailed the MDS shape is. And the trajectory of radial velocity by the entire body of the target is also precisely expressed.

In the time-velocity spectrogram, the height represents the target's radial velocity relative to the radar; the radial velocity component that appears on the upside (positive velocity) from the center represents the target is moving away from the radar, the downward (negative velocity) from the center represents that the target is moving toward the radar. The continuous waveform of the target over time generates a trajectory representing the movement characteristics according to the type of target on the spectrogram. For example, the difference in trajectory due to flight dynamics between fixed-wing aircraft and multiple helicopters is explained below. First, in fixed-wing UAVs, the propeller is fixed in the front or rear, so the thrust works only in one direction. Accordingly, the direction changes gradually by three factors; the inclination of the aileron at the rear of the main wing, the elevator of the horizontal tail wing, and the rudder of the vertical tail wing. In contrast, in a multi-copter, several rotors are distributed over the top of the fuselage. When changing the directions, it uses fuselage-tilting caused by the difference at each rotor rotation rate, so not only a gradual change of the direction but also a drastic change of the directions is possible. These distinctive flight characteristics appear as time-varying trajectories on the spectrogram; in the former case, it is gradual and curved and in the latter case, it appears in a sharp and vertical form. This trajectory will be trained with the target's characteristics along with the spectrogram shape and the spectrogram with a high overlap ratio will represent the radial velocity change in more detail. We applied three different window overlap ratios for each window size to obtain spectrograms with different resolutions for MDS and trajectory.

We performed data refinement process after STFT. UAV signal has low intensity due to its small size and material such as plastic or reinforcement styrofoam. So, as the distance increases, the signal intensity drops sharply or is not detected at all. So there are many unrecorded sections like background clutter in the spectrogram. Figure 3.6 shows the spectrogram for the background clutter and Mavic Air 2.

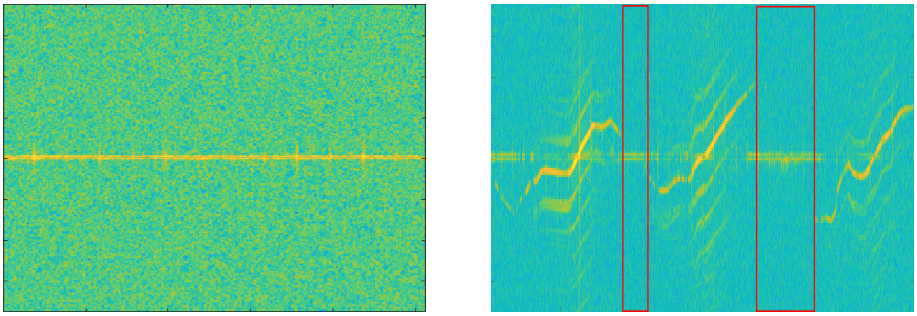


Figure 3.6: Spectrogram of background clutter (left) and Mavic Air 2 (right). The red box is the non-recorded section of the target.

In the spectrogram of Mavic Air 2 on the right, the red box is non-recorded sections because of the target's low signal intensity. When these non-recorded sections are trained with data, it is hard to expect the correct performance of the deep learning model. So we applied the following data refinement process to remove abnormal data. If the target is well captured, the clear spectrogram shape with strong intensity appears around a specific velocity component on the spectrogram and harmonic components are represented parallel around it. Based on this property, we first chopped the image at a time interval, which is the MDS periodicity of the target. Then, we removed chopped images with an average intensity below the threshold and stitched chopped images with an average intensity above the threshold. If the threshold is too high, only high-intensity signals recorded at a short-range would be retained and low-intensity signals at a long-range could be removed even though the MDS shape is represented. Conversely, if the threshold is too low, non-recorded sections of the target cannot be removed. So we determined the threshold by referring to the average intensity values of the background clutter and non-recorded sections of UAV spectrograms. Figure 3.7 represents the data refinement process for the

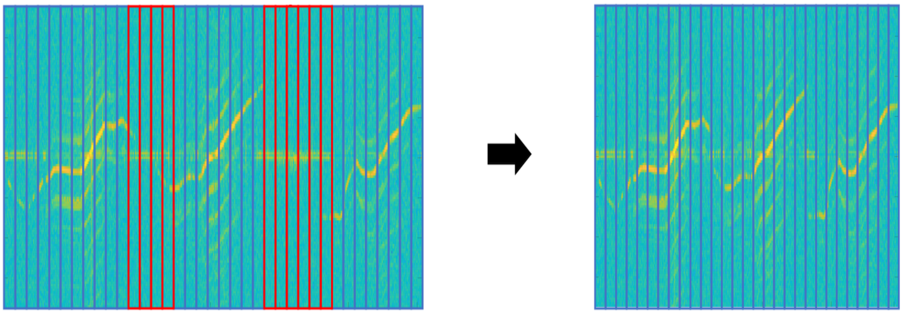


Figure 3.7: Spectrograms of Mavic Air 2 before the refinement process (left) and after the refinement process(right). red box: chopped images with an average intensity below the threshold, blue box: chopped images with an average intensity above threshold

Mavic Air 2 spectrogram. The spectrogram is cut at the same time interval, and the cut images with average intensity below threshold are removed. And images above the threshold are stitched together to generate a refined spectrogram.

※ MDS periodicity : Walking / Sit-walking(1/2sec), Metafly (1/24sec), Mavic Air 2 (1/92sec), Disco (1/183sec)

The data refinement process was applied to only two targets (MavicAir 2 and Disco), which have many non-recorded sections on the spectrogram. Table 3.2 shows the change in the spectrogram width of these two targets before and after the refinement.

Application of Refinement	Mavic Air 2	Disco
Before	995	339
After	747	168
Removal ratio	25%	50%

Table 3.2: Spectrogram width of two targets before and after the data refinement.

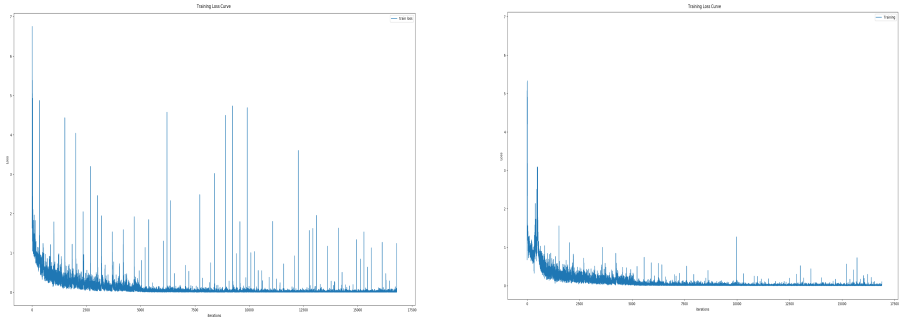


Figure 3.8: Training loss curves; before the refinement process(left), after the refinement process (right)

After refinement, the spectrogram size of Mavic Air 2 was reduced by about 25 % and the Disco by about 50 %. In particular, the spectrogram size of the Disco was significantly reduced due to the flight characteristics of fixed-wing UAV. Disco has a single rotor mounted at the rear of the fuselage, so the thrust acts only forward, and the direction changes gradually by the ailerons at the wing-tips. Hence, it requires a large turning radius and is often outside the radar detection range. Besides, its RCS is very low because of the fuselage material which is reinforced styrofoam. In other words, it was difficult to record due to the low RCS, and due to flight characteristics such as high-speed and wide turning radius, the recording time was too short within the radar detection range.

Figure 3.8 is the training loss curve before and after data refinement. In training with unrefined data, accuracy often fell significantly during training and the test accuracy had a large variation. You can see this by the number and size of spikes in the training loss curve on the left. Conversely, with refined spectrogram data, the phenomenon of drastic accuracy drop during training and the variation of test accuracy were reduced. And this can be seen by the decrease in spike size and quantity in the training loss curve on the right. Through the data

Category (Specification)	Window size (128, 256, 512)	Window overlap (50%, 70%, 85%)	Vertical Flip (O, X)	Total
Original signal	x 3	x 3	x 2	x 18

Table 3.3: Data augmentation method applied to generating training data

refinement process, the stability of the model has been improved.

After the refinement process, we applied the vertical flip to spectrograms. By using the vertical flip, we got additional spectrograms with reversed radial velocity sign. Totally we could generate 18 different spectrograms from original radar signal by applying three window sizes, three window overlap ratios and vertical flip. Table 3.3 shows applied data augmentation methods when generating the training data. The test set was generated with only one window size (128) and one overlapping ratio (70 %), and data augmentation was not applied.

For the training data, after pre-processing, the height of each spectrogram is resized to 128 and then cut into a 128x128 spectrogram image by applying a 50% overlap ratio. The test data is cut into a 128x128 spectrogram image by applying a 75% overlap ratio after the pre-processing process. To prevent the class imbalance, the number of classes of the trainset and the testset were balanced. The number of examples for each class was set to about 2000 in the trainset and about 200 in the testset. Table 3.4 shows the number of samples for each class in our dataset.

Class	Metafly	Mavic Air 2	Disco	Walking	Sit-walking	Total
Trainset	2142	2176	2196	2136	2112	10762
Testset	219	218	206	198	195	1096

Table 3.4: Number of samples for each class in radar spectrogram dataset

Chapter 4

Models

This section introduces the deep learning model and analyzes characteristics of the radar spectrogram dataset using the ResNet-18 model, a popular image classification model. By checking model performances depending on the data type of radar spectrogram and the noise, we confirm the optimal data type and a major feature of the spectrogram dataset. In addition, we check the performance by changing the structure of the model. And we design a lightweight and stable ResNet-SP model which is suitable for real-time systems by modifying the model, based on these characteristics.

The rule-based classifier is based on ’if-then’ rules designed by engineers. This method often complicates the model and lacks scalability, because rules must be specified every time to classify the new data. On the other hand, a CNN-based classifier extracts features from large amounts of data automatically. In addition, the robustness of CNN to shift and distortion [8] resulted in an outstanding performance in image classification tasks; in the 2014 ImageNet Large Scale Visual Recognition Challenge (ILSVRC), GoogLeNet and VGG-Net respectively ranked first and second with top-5 error rates of 6.67% and

7.3% and In the 2105 ILSVRC, ResNet recorded a recognition top-5 error rate of 3.57% which was less than the human recognition error rate.

CNN is a deep-learning model that uses convolutional operation, and is composed of several convolution layers and pooling layers. A model learns the different features of an image using various sizes and numbers of kernels. In the shallow layers, low-level features are learned, which can be basic shapes like lines and edges. In the deep layers, high-level features are learned, which contain more specific information for classifying objects. The model is designed to perform well by learning the various characteristics of the data. CNN can extract high-level features as layers are stacked deeper, but simply stacking layers deeper does not increase the performance. The reason is known to be the gradient vanishing problem [1] that occurs due to the multiplications of gradients in the parameter update stage as the layer gets deeper. As a result, training cannot be progressed and in some cases, the performance even degrades.

4.1 ResNet-18

He et al. [7] proposed a residual network that applies residual concepts to the CNN model. ResNet showed that as the layer gets deeper, the gradient vanishing problem can be reduced using the residual block and hence bring performance gain. This architecture has shown superior performance in various image processing tasks than previous CNN models. In the CNN model, the receptive field of the unit in the deeper layer is larger than in the shallow layer. This is because as the layer deepens, the output unit is indirectly connected to a broader area of the input image [5]. However, as the layer deepens, structural problems such as gradient vanishing and over-fitting can easily occur. As shown on the right

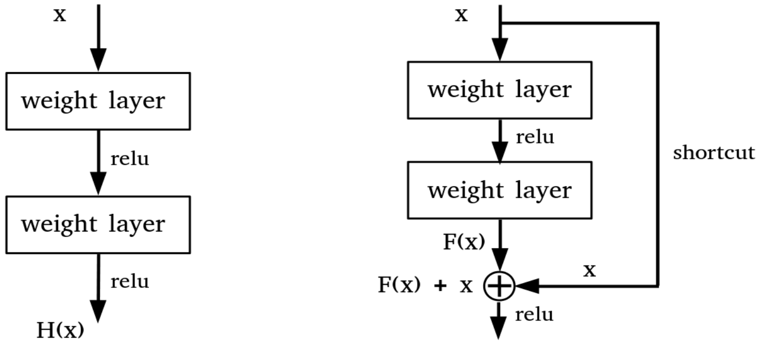


Figure 4.1: Plane CNN layers (left), Residual block (right)

side of Figure 4.1, ResNet solves the aforementioned problems by connecting a detour path called identity shortcut connection between intermediate layers.

In the residual block, the input ‘x’ goes through the first convolution layer, the activation function (Relu) and the second convolution layer, outputting F(x). The output F(x) passes through the activation function after the addition with the initial input ‘x’. Due to this shortcut connection, even if F(x) has passed through the two layers and the parameter approaches 0 due to gradient vanishing, the added initial input ‘x’ remains and is transferred to the next layer. Therefore, even if the layer is deepened, the representation power does not fall short of the layer before the identity function and the performance is improved by training. In the ImageNet dataset, the ResNet model showed higher performance than the vanilla CNN model, and achieved better performance as the layer deepens. ResNet is a well-balanced CNN model widely used in recent computer vision tasks. As radar signals were transformed into a format of an image, we can train the detection problem on the dataset with these ResNet models. Through this, we can analyze the radar spectrogram data with the ResNet model and design

Number of channels	Signal form	Accuracy (%)
1	Magnitude	75.98
2	Real, Imaginary	**79.88**
2	Magnitude, Phase	54.53

Table 4.1: Accuracy of the ResNet-18 according to the signal form of the radar spectrogram

an enhanced model.

First, we analyzed the performance of the model according to the information type of the spectrogram data. The radar signal is composed of complex numbers, containing the signal intensity and phase information, etc. In a related study, to utilize a deep learning-based optical image classification model, the signal value of the radar spectrogram was transformed into the color scale of the optical image. We assumed that the transformation without preserving radar data characteristics could distort or miss out on information and tested the performances according to the three different information form of the radar signals. The three types of radar information are represented as channel information: 1 channel of magnitude, 2 channels of real & imaginary and 2 channels of magnitude & phase. The magnitude is the square root of the sum of the squared real-value and the squared imaginary-value. The phase is obtained by taking the inverse tangent of the value obtained by dividing the imaginary value by the real value. The height and width sizes of the three spectrogram data were the same, and the accuracy is the average of five times measurements. Table 4.1 shows the classification accuracy according to the signal information form of radar spectrogram. The result shows the highest accuracy when the real and imaginary values of the radar signal are paired as two channels. Through this,

	Classification Accuracy (%)	
Noise level	Gaussian Noise	Uniform Noise
0.01	76.20	80.80
0.03	66.30	80.90
0.05	40.25	75.71

Table 4.2: Accuracy of the ResNet-18 on two different noises.

we confirmed that most features were maintained at the original form of the radar signal and that the change in signal form could lead to loss of information.

Next, we investigate the main features of the data that the model learns by adding two different noises to the dataset. One is the Gaussian noise that adds random value and the other is the uniform noise that adds the same value. The Gaussian noise image was created by generating a random variable following a standard normal distribution as the input size, multiplying it by a noise level representing noise intensity, and adding it to the original normalized image. The uniform noise image was created by setting the value of 1 to the input size, multiplying the noise level, and adding it to the original normalized image. Each value of the image does not exceed 1. Through Gaussian noise, we checked the model performance in the condition of where an arbitrary shape is added to the entire image, and in uniform noise, we checked the model performance when the sharpness of the MDS is reduced compared to the surrounding area. Each noise level was specified as a hyper-parameter by identifying the point at which the model's performance begins to deteriorate significantly. Table 4.2 shows the performance of the model for two types of noises.

The performance of the model decreases in both data as the noise level increases, but we see that the performance decreases sharply in the Gaussian noise, com-

Conv. Groups	Numbers of Layers	Feature-map size	Accuracy (%)
5	18	4 x 4	79.88
4	14	8 x 8	**81.43**
3	10	16 x 16	75.38

Table 4.3: Accuracy of the ResNet-18 according to conv. groups and layers.

pared to the uniform noise. This result shows that low-level features in the our dataset are the most important features to the model when classifying UAVs.

The ResNet-18 model consists of a 5 convolution group and a $2-5$ group, which consists of a few basic blocks. The feature map size is halved after going through each convolution group. We analyzed the performance by changing the convolution group of the model and the basic block(layer)s within the groups. We checked the performance by sequentially removing the convolution groups from the output of the model. Table 4.3 shows the accuracy of the model by the change of the convolution groups and layers. The results showed the highest accuracy in the 8x8 feature map size with the 5th convolution group removed, and the performance continued to decline after that. This is the highest model performance in the optimal feature map size that reflects the characteristics of the data, and this feature is applied when designing a new model. And we tested the performance by changing the number of basic blocks within the convolution group, but there was no significant trend. Through this experiment, we confirmed that the final feature map size is significant in learning the spectrogram data of the deep learning model, and that the additional depth of the layer does not significantly affect the performance improvement. Based on the above analysis, we design a lightweight model more suitable for real-time systems.

4.2 ResNet-SP

In the analyses with the ResNet-18 model, we checked that 1) the model mainly learns the low-level features of the radar spectrogram, 2) the size of the final feature map significantly affects the performance and 3) the depth of layers do not significantly affect the performance. Furthermore, we designed the ResNet-SP model for real-time systems by applying optimal settings based on these analyses and applying additional compression and stabilization methods.

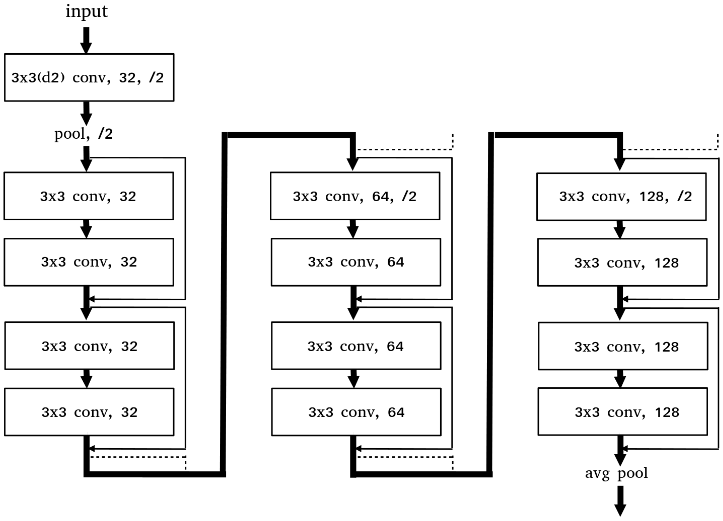


Figure 4.2: ResNet-SP Architecture

Figure 4.2 is the architecture of ResNet-SP. We removed the 5th convolution group from the ResNet-18 model and kept the number of basic blocks the same. And we reduced the channels of each group by half. By applying a dilated kernel method, the computations were reduced because the smaller parameters are used in the model while keeping the receptive fields. And to increase the learning stability of the model, we applied an outlier detection method that re-

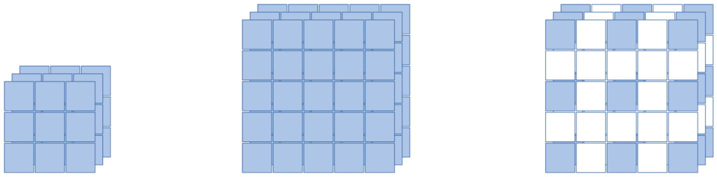


Figure 4.3: The receptive fields : 3x3 kernel (left), 5x5 kernel (center), 2-dilated 3x3 kernel (right)

moves abnormal data in the learning process through the distribution of the data and the gradient clip method that reduces the influence of the anomalous data by limiting the norm of the gradient.

In a kernel, the receptive field signifies the area of the input image where the kernel attends to. The size of the receptive field is the same as the kernel size and the larger the size, the more the overall characteristics of the image can be obtained. However, if the kernel size is increased to obtain a wider receptive field, the number of parameters increases, which increases the computation.

Dilated convolution is a method of adding zero-padding to the convolution kernel, which allows a wider receptive field while using the same number of parameters [21]. Figure 4.3 shows the receptive field images when 3x3, 5x5, and 3x3 kernels with dilation are applied. The 2-dilated 3x3 kernel has the same receptive field as the 5x5 kernel, with the same number of parameters as the 3x3 kernel. Through this method, the global feature can be extracted without increasing the number of parameters.

In the radar spectrogram, the MDS shape is formed around the main velocity by the main body, and the harmonic components appear parallel to around. Whereas the target is located locally in the visual image, the moving target sig-

Kernel	7 x 7	3 x 3	3 x 3
Dilation	1	2	3
Receptive field	7 x 7	5 x 5	7 x 7
Accuracy (%)	79.88	**80.26**	79.33

Table 4.4: Accuracy of the ResNet-18 according to the dilated kernel.

nal of the radar spectrogram is time-varying, and the local characteristics of the MDS formed in specific areas of the entire image and the global characteristics of the harmonic component formed at various frequencies coexist. We tried to reduce the computations without missing the global feature. Therefore, we applied a dilated convolution kernel within the range that does not significantly degrade the performance. Table 4.4 shows the performance when the size of the 7x7 kernel of the model is reduced to a 3x3 kernel with the dilation. When dilation of 2 was applied to the 3x3 kernel, the performance was maintained. However, when dilations of 3 or more were applied, the performance was gradually decreased. This means that applying large dilation will miss out on the MDS, the main feature that the model is supposed to learn. Therefore, in the end, we applied a dilation of 2 to the 3x3 kernel, instead of the 7x7 kernel.

Although many abnormal data were removed in the refinement process, abnormal data interfering training still remained in our dataset. In a deep neural network using multiple layers, when such abnormal data comes in a specific batch, large weights are successively multiplied in the parameter update process, causing a gradient exploding problem. As a result, previous well-trained parameters change rapidly, which causes a dramatic accuracy drop. We applied a gradient norm clipping method [12] that constrains the maximum norm of gradients to reduce the impact of anomalous data interfering with the learning

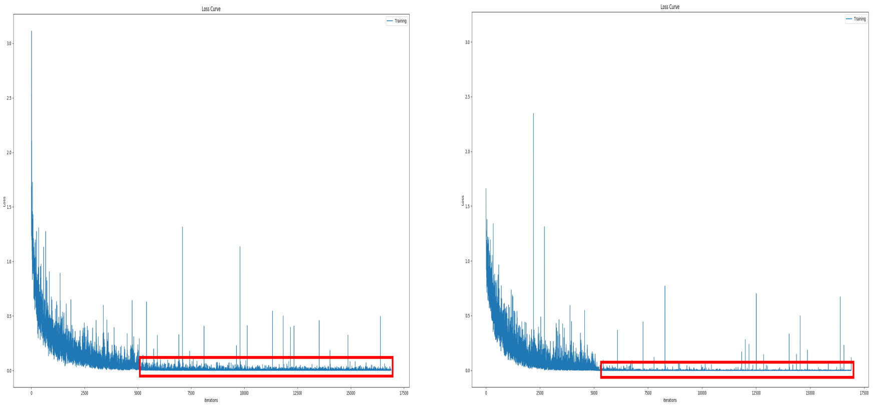


Figure 4.4: Training loss curves; before the gradient clipping (left), after the gradient clipping (right)

process. Figure 4.4 is the training loss curve before and after applying the gradient clipping method. In the figure, the red box represents the training loss after 5000 iterations, and we can see that the variation has decreased after applying the gradient clipping. And the variation of test accuracy was also decreased.

We additionally applied a method of removing abnormal data from the training process to increase the learning stability of the model. In our dataset, abnormal data are non-recorded or contaminated sections of the spectrogram. Most of these abnormal data were removed through the data refinement step, but some remain, interfering with the stable learning of the model. These abnormal data exist in every class. Anomaly detection is a research field that identifies outliers that deviate from the majority of normal data. There are various anomaly detection methods, but we applied the concept of a softmax model of end-to-end anomaly score learning. [11] This approach assumes that normal data appears at a relatively high frequency compared to anomalous data, and anomalous data appears at a lower frequency. When data of a specific class is input, the softmax

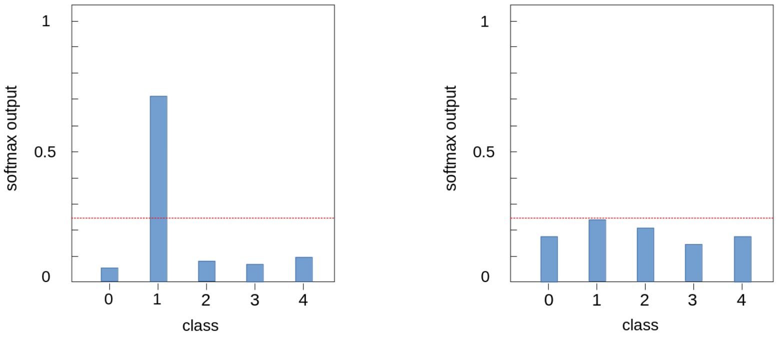


Figure 4.5: Softmax output distributions; normal data (left), outlier data (right)

value is much higher in that class than in other classes, and most of the normal data have this probability distribution. When normal data is entered into the model, the softmax value is significantly higher in one class. However, when abnormal data is entered into the model, the softmax value appears similar in several classes and the largest value is much smaller than the softmax value in normal data. Figure 4.5 is an example of the softmax distribution of normal and abnormal data. Using this characteristic, we excluded data in the training process if the highest softmax value of the input data does not exceed a certain threshold.

Chapter 5

Experiment

5.1 Experiment Result

This section shows performances with the ResNet-18 and the ResNet-SP models on the radar spectrogram dataset. The performances are the accuracy and computation time. The accuracy is the percentage of the correctly predicted samples out of the total samples. We used the average test accuracy of five runs and recorded their standard deviation. Table 5.1 shows the test accuracy and standard deviation of five runs. The ResNet-SP model shows higher accuracy and smaller standard deviation than the ResNet-18 model, which indicates better performance in accuracy and stability.

Models	Accuracy (%)	Standard deviation
ResNet-18	79.88	0.0204
ResNet-SP	**83.39**	**0.0115**

Table 5.1: Average test accuracy of five runs and their standard deviation.

The computation time is measuring the computational cost of the model. We recorded the training time and inference time of models, excluding the dataset generation process. Training time is the time to train the model using training data, and inference time is the time to predict new data with the trained model. Table 5.2 shows the training time and inference time for the radar spectrogram dataset of both models. The ResNet-SP model shows faster computation time than the ResNet-18 model in both training time and inference time.

Models	Training time (sec)	Inference time (ms)
ResNet-18	640.39	2.68
ResNet-SP	**242.22**	**1.98**

Table 5.2: Computation time; Training time and inference time.

5.2 Training Details

This section explains the details of our experiments. The model training was performed in Ubuntu with NVIDIA GeForce GTX Titan X edition GPU and a 3.6 GHz Intel Core i7-9700K CPU. We used the stochastic gradient descent (SGD) [14] with momentum [18] as an optimizer. The momentum coefficient was set to 0.9 which means 90% of the cumulated gradient from the previous step will be transmitted to the current step. The initial learning rate was 0.1 and the weight decay [9] coefficient for regularization was set to $1.0e-4$. We trained for 100 epochs in total, with a batch size of 64. We used the cross-entropy loss [23] for the final loss function.

Chapter 6

Conclusion

In this study, we recorded three different types of UAV signals and two different types of human activity signals in various scenarios using FMCW radar. Furthermore, we generated the radar spectrogram dataset with high diversity through STFT, the data refinement method, and the data augmentation method. Then, we analyzed the characteristics of the radar spectrogram dataset using the ResNet-18 and checked the optimal data form and model structure. In addition, we designed the ResNet-SP model, which is more suitable for real-time systems by compressing and stabilizing the ResNet-18 model. As a result of experimenting with both models with the same radar spectrogram dataset, the ResNet-SP showed higher stability, accuracy, and faster computational time than the ResNet-18 model. In future works, we hope to expand this study to a model that classifies UAV types by adding several additional UAVs and to improve the performance of the model using the acoustic spectrogram of the target along with the radar spectrogram. We also hope to improve the performance of the ResNet-SP model.

Bibliography

[1] Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166, 1994.

[2] V. C. Chen, F. Li, S.-S. Ho, and H. Wechsler. Micro-doppler effect in radar: phenomenon, model, and simulation study. *IEEE Transactions on Aerospace and electronic systems*, 42(1):2–21, 2006.

[3] B. Choi and D. Oh. Classification of drone type using deep convolutional neural networks based on micro- doppler simulation. In *2018 International Symposium on Antennas and Propagation (ISAP)*, pages 1–2, 2018.

[4] L. Cohen. *Time-frequency analysis*, volume 778. Prentice hall, 1995.

[5] I. J. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, Cambridge, MA, USA, 2016. http://www.deeplearningbook.org.

[6] R. Harmanny, J. De Wit, and G. P. Cabic. Radar micro-doppler feature extraction using the spectrogram and the cepstrogram. In *2014 11th European Radar Conference*, pages 165–168. IEEE, 2014.

[7] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[8] S. Hijazi, R. Kumar, and C. Rowen. Using convolutional neural networks for image recognition. *Cadence Design Systems Inc.: San Jose, CA, USA*, pages 1–12, 2015.

[9] A. Krogh and J. A. Hertz. A simple weight decay can improve generalization. In *Advances in neural information processing systems*, pages 950–957, 1992.

[10] S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.

[11] G. Pang, C. Shen, L. Cao, and A. van den Hengel. Deep learning for anomaly detection: A review, 2020.

[12] R. Pascanu, T. Mikolov, and Y. Bengio. On the difficulty of training recurrent neural networks, 2013.

[13] S. Rahman and D. A. Robertson. Classification of drones and birds using convolutional neural networks applied to radar micro-doppler spectrogram images. *IET Radar, Sonar Navigation*, 14(5):653–661, 2020.

[14] H. Robbins and S. Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.

[15] M. Saqib, S. Daud Khan, N. Sharma, and M. Blumenstein. A study on detecting drones using deep convolutional neural networks. In *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–5, 2017.

[16] Y. Seo, B. Jang, and S. Im. Drone detection using convolutional neural networks with acoustic stft features. In *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–6, 2018.

[17] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[18] I. Sutskever, J. Martens, G. Dahl, and G. Hinton. On the importance of initialization and momentum in deep learning. In *International conference on machine learning*, pages 1139–1147, 2013.

[19] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions, 2014.

[20] B. Taha and A. Shoufan. Machine learning-based drone detection and classification: State-of-the-art in research. *IEEE Access*, 7:138669–138682, 2019.

[21] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions, 2016.

[22] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[23] Z. Zhang and M. Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. In *Advances in neural information processing systems*, pages 8778–8788, 2018.

초 록

본 논문에서는, 레이더 스펙트로그램 상에 형성된 서로 다른 이동표적의 고유한 마이크로 도플러신호를 학습하는 딥러닝 기반 분류모델을 제안한다. 이를위해 우리는 다섯가지 소형 이동표적(무인항공기 3종과 사람행동 2종)을 선정하여 주파수변조 연속파레이더로 표적들의 다양한 움직임을 측정하고 측정한 신호에 단시간 푸리에 변환의 신호처리과정과 데이터 정제 및 증강의 전처리과정을 적용하여 자체 레이더 스펙트로그램 데이터셋을 생성한다.
이후 광학이미지 분류모델인 ResNet-18을 사용하여 레이더 스펙트로그램 데이터셋의 특성을 분석한다. 레이더신호를 광학이미지로 변형하는 과정에서의 정보왜곡 및 손실을 가정하여 세가지 레이더 신호형태에 따른 성능을 비교하고 최적의 데이터형태를 확인한다. 노이즈 시험 및 구조에 따른 성능변화를 통해 모델이 학습하는 주요한 데이터 특징과 이상적인 모델구조를 확인한다.
마지막으로 레이더 스펙트로그램 데이터셋 특성분석을 기반으로 추가적인 경량화 및 안정화 기법을 적용하여 실시간 시스템을 위한 ResNet-SP 모델을 설계하고 ResNet-18모델과의 성능비교를 통하여 연산속도 증가와 안정성 및 정확성 향상 등의 성능개선을 확인한다.

주요어: 컨볼루션 신경망, 분류, 무인항공기, 주파수변조 연속파레이더, 단시간 푸리에변환, 스펙트로그램

학번: 2019-29945