

# Automatic identification of Spread-F from Ionograms of Bac Lieu observatory by Using Convolutional Neural Network

LUU Viet Hung<sup>1</sup>, PHAM Thi Thu Hong<sup>2</sup>

<sup>1</sup> Faculty of Applied Sciences, Ho Chi Minh City University of Technology and Education, 1 Vo Van Ngan Street, Linh Chieu Ward, Thu Duc District, Ho Chi Minh City, Vietnam

<sup>2</sup> Institute of Geophysics, Vietnam Academy of Science and Technology, A8 Building, 18 Hoang Quoc Viet Street, Nghia Do Ward, Cau Giay District, Hanoi, Vietnam

\* Corresponding email: [hungluu@hcmute.edu.vn](mailto:hungluu@hcmute.edu.vn)

**Abstract:** *Spread-F caused by night time ionospheric disturbances is common at equatorial and low-latitudes. Spread-F causes changes in the amplitude and phase of electromagnetic waves passing through the ionosphere and affects the propagation of radio waves in space. Spread-F are classified into 4-types: Frequency Spread-F (FSF), Range Spread-F (RSF), Mixed Spread-F (MSF), and Branch Spread-F (BSF). Accurate identification and classification of Spread-F types helps to statistically analyze the physical characteristics of Spread-F and ionospheric disturbance mechanisms. The detection and classification of Spread-F types from ionograms at Bac Lieu Observatory were carried out based on visual inspection. Currently, modern ionosondes were installed at Bac Lieu Observatory and collect a large amount of ionogram. The manual identification of Spread-F types is very laborious and costly. Therefore, automatic identification and classification of Spread-F is an urgent requirement. In this study, we tested convolutional neural networks to automatically identify and classify Spread-F types from the ionograms of Bac Lieu observatory. The initial results show that the identification and classification of Spread-F types at Bac Lieu have high accuracy, and it is possible to apply artificial intelligence to automatically identify and classify the Spread-F types of this observatory.*

**Keywords:** *Spread-F; Ionogram; Convolutional neural network; Detection; Classification*

## 1. Introduction

The Spread-F (SF) phenomenon was first observed in 1938 by Booker and Wells (Booker and Wells 1938) using an ionospheric probe to map the electron density line at Huancayo, Peru. They noticed that at night the ionospheric trace was not sharp but spread over a large altitude range of tens kilometers. Based on Rayleigh's scattering theory, they assumed that it was the formation of electron clouds with a size of about 25 m. With the research achievements made during many decades of measurements with ground-based radar and satellites, people have a more complete understanding of the formation and development of the F-layer spreading phenomenon. Nowadays, the F-layer spreading phenomenon is known as the heterogeneous structures of electron density in the F layer of the ionosphere that occur at night in equatorial and low-latitude regions. The heterogeneous structures of the ionospheric F layer vary in size from decimeters to hundreds of kilometers. Woodman and La Hoz (Woodman and La Hoz 1976), using observations of Jicamarca in Peru, classified SF into four types: Frequency Spread F (FSF), Range Spread F (RSF), Mixed Spread F (MSF), and Branch Spread F (BSF). SF phenomena vary strongly seasonally, meridionally, and even daily.

Electromagnetic signals transmitted from Earth to space or from space satellites to Earth as they pass through the ionosphere can be attenuated and scattered due to inhomogeneity in the F layer of the ionosphere. This has negative effects on satellite-to-Earth communications, global positioning systems, ballistic missiles and power transmission systems, etc. Therefore, the construction of a unified physical model to explain the mechanisms and occurrence of SF, in order to prevent negative impacts they cause, is the main problem to be solved. The precise identification and classification of Spread-F types from ionograms allows a statistical study of SF characteristics as well as SF formation and occurrence mechanisms. In Vietnam, SFs are observed at the Hanoi and Bac Lieu observatories (Lan and Cường 2013; Thu et al. 2024). At Bac Lieu observatory, due to its location near the equator, the four types of FS are observed, whereas at Hanoi, three types of FS are observed: FSF, RSF, and MSF. The BSF is not observed in Hanoi.

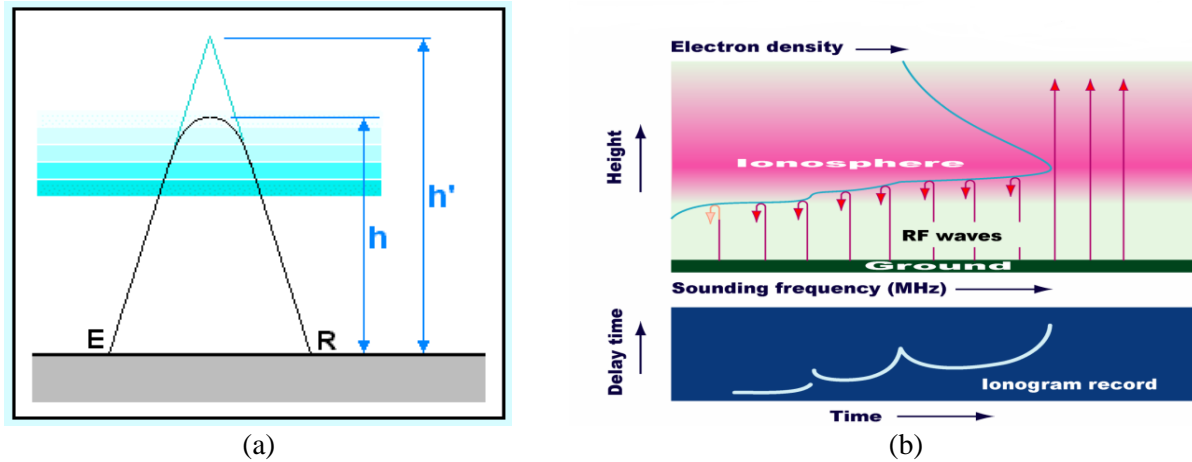
The use of artificial intelligence in ionogram identification has been published in some works (Abadi et al. 2022; Lan et al. 2020; Lan et al. 2018; Pillat, Fagundes, and Guimarães 2015). In (Lan et al.

2020), Ting Lan made a comparison of the use of 3 methods in recognizing and classifying ionograms: Convolutional Neural Network (CNN), Decision Tree, and Random Forest. The results showed that CNN is the best for classifying ionograms. Therefore, in this work, we built machine learning programs using CNN to identify ionograms of Bac Lieu observatory.

## 2. Dataset

In this study we use 872 ionograms from the Bac Lieu observatory (9.28 north latitude, 105.73 east longitude) measured overnight (from 6:00 pm to 6:45 am the following morning), including 190 FSF images, 194 MSF images, 103 BSF images, 191 RSF images and 194 images without SF.

The types of SF appearing on the ionogram are described in detail in the "Handbook of Ionogram Interpretation and reduction" by Piggott and Rawer (Piggott & Rawer, 1972) and can be summarized as follows: FSF is the type for which the ionogram trace spreads over frequency, MSF is the type for which the ionogram trace spreads over both frequency and altitude, RSF is the type for which the ionogram trace spreads over altitude, and BSF is the type for which the ionogram trace is not clearly defined.



**Fig. 1.** Schematic illustration: (a) Vertical sounding of the ionosphere (b) Principle of wave reflection on the ionosphere and traces obtained on the ionogram.

The ionograms were obtained by vertically probing ionosphere at frequencies ranging from 1 MHz to 30 MHz. The transmitter emits short radio pulses vertically into space, which are reflected back to the receiver when the frequency of the emitted wave is equal to the plasma frequency (Fig. 1a). The reflected frequency is proportional to the electron density according to the following formula:

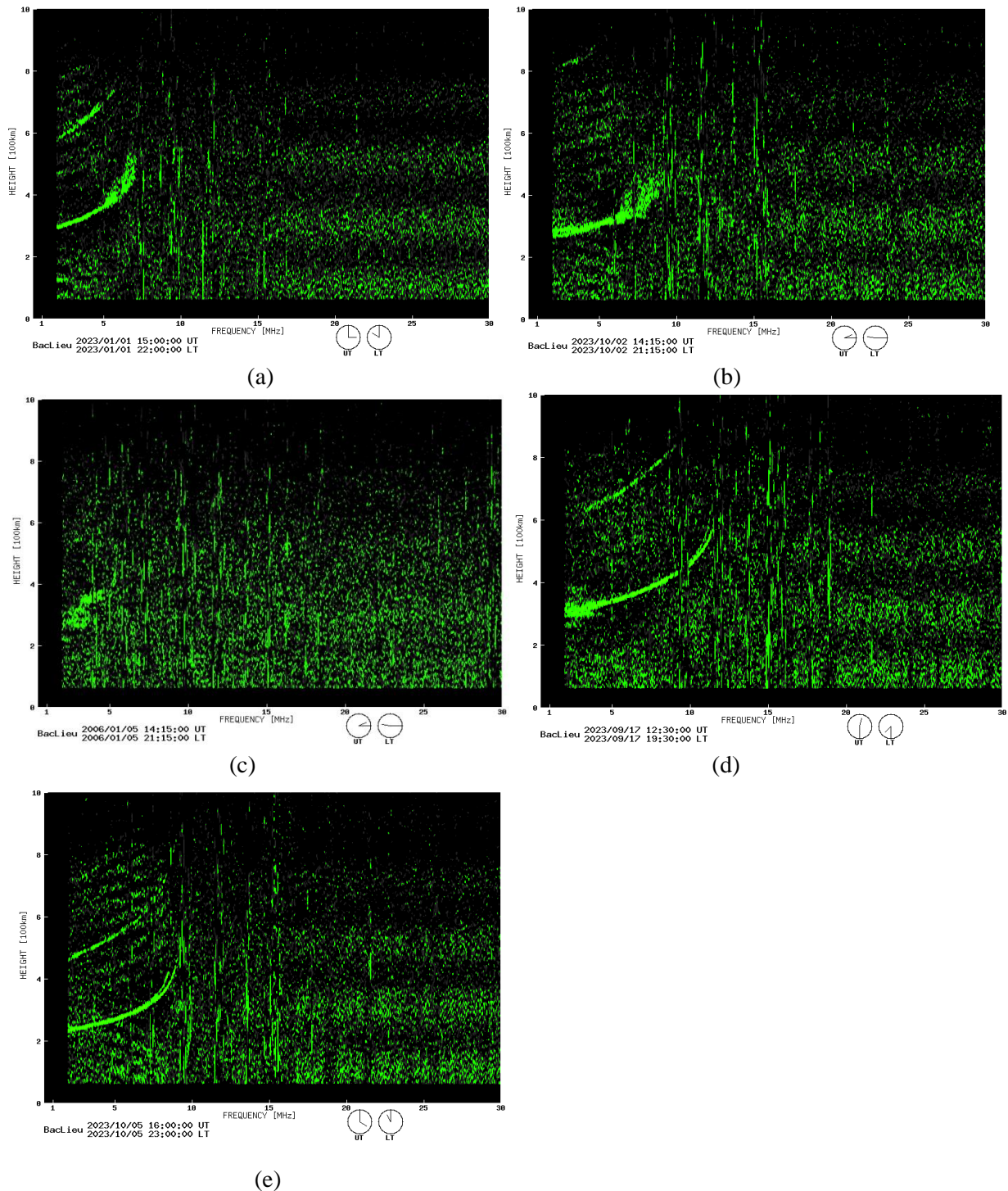
$$f = 9\sqrt{N_e} \quad (1)$$

The maximum frequency at which radio pulses can still be reflected by the ionosphere is called the critical frequency and is the maximum value of the electron density reached. The height determined by the measurement is called the apparent height:

$$h' = c \cdot t / 2 \quad (2)$$

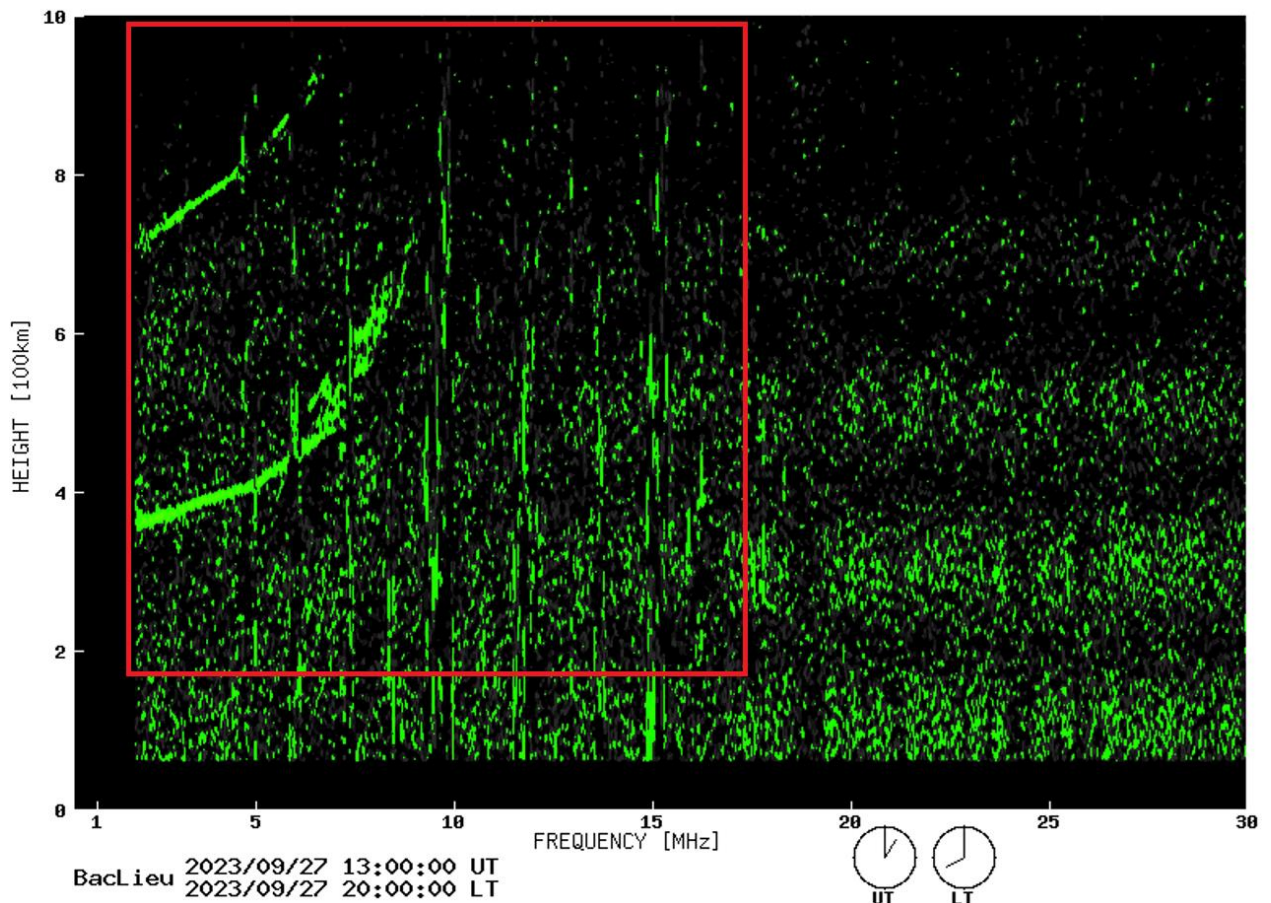
Where  $c$  is the speed of light and  $t$  is the time delay. The resulting ionograms, which represent the reflected signals from the ionosphere layers, are the  $h(f)$  curves shown in Fig. 1b.

Examples of actual ionograms without SF and with each type of SF obtained at the Bac Lieu observatory are shown in Fig. 2.



**Fig. 2.** Types of Spread-F ionograms; (a) Frequency Spread-F (FSF), or F – type; (b) Mixed Spread-F (MSF), or L – type; (c) Branch Spread-F (BSF), or P – type; (d) Range Spread-F (RSF), or Q - type, (e) ionogram without Spread-F.

In the ionograms, the ionospheric F-trace only appears at altitudes ranging from 180 to 1000 km and for a frequency range below 15 MHz (Lan et al. 2020). From the original ionograms, we cut out the part of the image that does not contain the F trace, and keep the part bounded by the red square in Fig. 3 to carry out the classification.

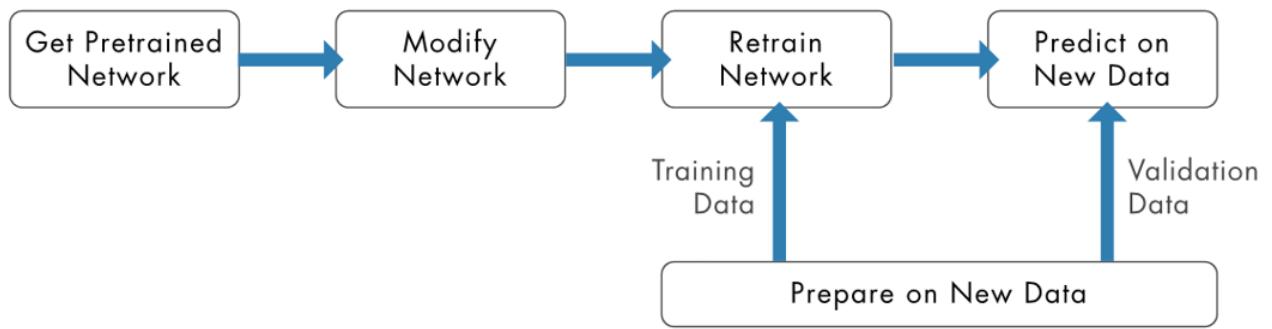


**Fig. 3.** Selected part of ionogram (within the red square) used for identification and classification.

### 3. Methodology

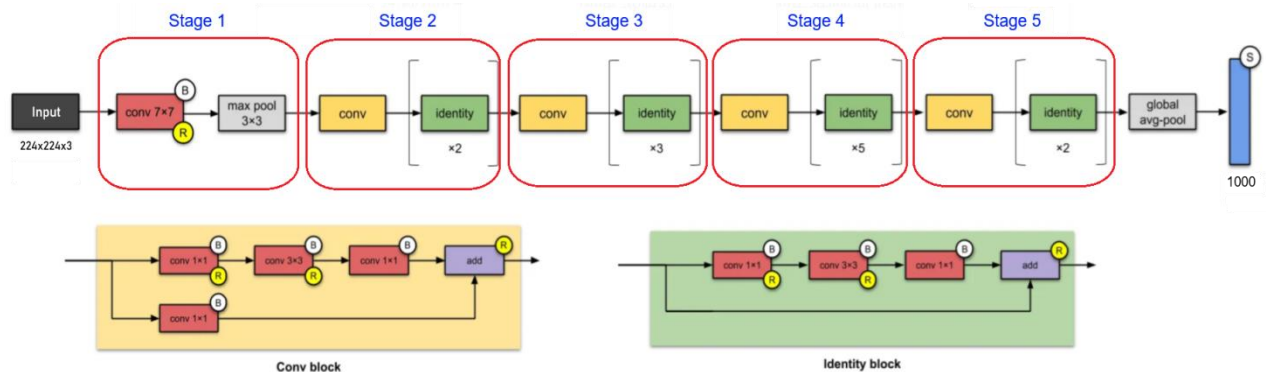
In this study, we use the transfer learning method to train the model with the Bac Lieu ionogram dataset. Transfer learning is a method widely applied to image classification. The essence of transfer learning is to train a new model by refining the pre-trained model to use the knowledge that the model has learned from a huge image database and train a new model to solve another related task. The transfer learning method makes it possible to work with a smaller set of databases and reduce training time compared with training from scratch (see, for example, (MathWorks, 2025)). The transfer learning process is described in Figure 4.

The calculation for network training in this work is realised by using Matlab software (MATLAB, 2022). First, we choose a pre-trained network from the Deep Network Designer App of Matlab. Currently, there are many pre-trained networks that are highly effective and widely used, such as VGG-Visual Geometry Group, GoogLeNet, ResNets-Residual Networks, etc.). Here, we choose to use residual networks. The ResNet residual network was published in 2015 by Kaiming He et al. (He et al. 2016). This network has been trained on more than one million images from the ImageNetRes database, and it can classify up to 1000 objects. Currently, ResNet has many versions, such as ResNet-18, ResNet-34, ResNet-50, ResNet-101, ResNet-152, etc., in which ResNet is the name of the network architecture, and the number following it is the number of layers in the network. With hundreds of layers, ResNets are very deep learning models. The deeper the network, the more features it can extract from the image. However, if we increase the number of layers too much, we come up against the problem of ‘gradient vanishing’, and the network’s performance is saturated or even degraded due to the loss of original information when it passes through too many deep learning layers. By using a bypass connection to go through one or more layers of the network, ResNets can overcome the problem of gradient vanishing during network training and thus become a very effective deep learning network.



**Fig. 4.** Schematic diagram of training a new network from a pre-trained network using transfer learning (image taken from Matlab software).

We have chosen ResNet-50 and ResNet-101 pre-trained networks for transfer training. The structure of the ResNet-50 network is depicted in Fig. 5 (Karim, 2019). Image classification using the ResNet-50 network is implemented through five main stages as summarized below ( Fig. 5). Stage 1: The input image of size (MathWorks) is let to pass through a [7x7] convolution layer which extracts features from it. The resulting convolution is passed through a [3x3] max-pooling layer to reduce the size. Stages 2 to 5: the information is passed through a series of residual blocks to be processed and transformed. The residual blocks can be either convolution blocks or identity blocks. If the residual blocks are convolution blocks, the information from one branch is convolved and then added to the other branch. For identity blocks, the value of one branch is added directly to the other branch without applying convolution. The results obtained from the residual blocks are passed through a fully connected layer to make the classification.



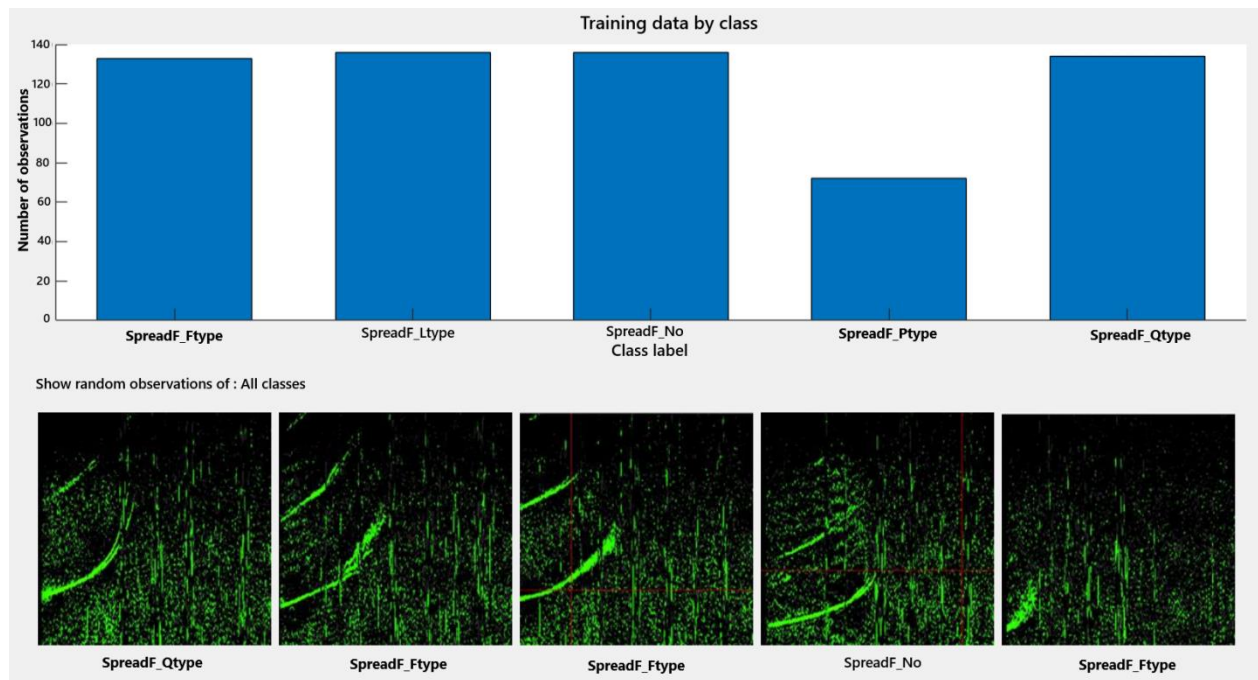
**Fig. 5.** ResNet-50 neural network configuration modified from (Karim, 2019) .

In the ResNet-101 network, the number of layers of Stage 4 in Figure 5 is increased to 23 instead of 6. ResNet-101 has more parameters than ResNet-50, so it is deeper and therefore can capture more complicated features and representations of the input image than ResNet-50.

Next, we need to make some changes to the selected network by changing the fully connected layer and the final image classification layer of the network. In the pre-trained network, the number of image object classes of these layers is set to 1000. In the transfer learning network, we replace the number of classes of these layers with 5, which is the number of image classes in our classification problem. Then, we input the ionograms into the pre-trained network to train the new network.

The data set of 872 images is converted into 224x224 images according to the input requirements of the pre-trained network and they are placed in folders named “SpreadF\_ Ftype”, “SpreadF\_ Ltype”, “SpreadF\_ Ptype”, “SpreadF\_ Qtype” and “SpreadF\_ No”. These folders contain the FSF, MSF, BSF, RSF images, and the images without SF, respectively. Number of images in each folder are 190, 190, 103, 191 and 194, respectively. The percentages of the FSF, MSF, BSF, and no SF images are approximately 22%, and the percentage of the RSF images is approximately 12%. The names of these folders are used as labels for each type of ionogram image used for recognition and classification. The set of all images is divided into 3 sets, including the training set (70%), testing set (20%), and validation set (10%). The images used for training after being imported into the network are shown in the Figure below. We can see that the Branch Spread-F (BSF) has the smallest number with 72 images. The other types have almost the same number

and are about 136 images. The bottom part of this Figure randomly displays some images from the training set.



**Fig. 6.** The number of images of each label type of the training dataset (top part of the figure) and 5 randomly selected images out of 611 of the training set.

Next, we choose the optimization algorithm and set the parameters to train the new network. The ResNet architecture uses 3 default optimization algorithms to train the model: SGD-Stochastic gradient descent with Momentum, Adam, and Rmsprop. After several training experiments with these algorithms, we chose the Adam optimization algorithm to train the new networks.

Adam algorithm is first introduced by Diederik P. Kingma and Jimmy Lei Ba (Kingma, 2014). Adam stands for Adaptive Moment Estimation, an algorithm for first-order gradient-based optimization of stochastic objective functions. This algorithm is congruent with training deep neural networks since it computes individual adaptive learning rates for different parameters.

Next, we need to reset the value of the initial learning rate parameter by assigning it a small value. If the initial learning rate is set to 0, the weights of the pre-trained model are kept the same (frozen), and the knowledge that the model has learned will be used up. Then, we retrain the pre-trained network on the new image dataset (here, the ionogram dataset). Retraining the network on the new image dataset is carried out in an iterative process that takes place in many epochs, each epoch consists of several iterations. The number of epochs chosen is 10 for the ResNet-101 model and 15 for the ResNet-50 model since using this number of epochs, the parameters such as accuracy and loss function (cross-entropy) of the training process are stable and change little. On the other hand, with this number of epochs, the training time for each model is about 35-80 minutes, which is acceptable.

Detailed descriptions of CNN can be found in the literature , (Purwono et al., 2022; Taye, 2023).

In our image classification problem, the network training process using CNN can be summarized as follows. First, the input image is passed through a convolutional layer. The convolutional output is passed through a max-pooling layer to reduce the dimensionality, and the information is then passed through a series of 16 residual blocks for ResNet-50 architecture or 33 residual blocks for ResNet-101 architecture. The residual blocks can be convolution blocks or identify blocks. The ResNet architecture incorporates “skip” connections that allow information from previous layers to pass directly through them to later layers. The skip connections bypass one or more intermediate layers while preserving information and allowing parameter gradients to propagate more easily from the output layer to previous layers of the network, allowing deeper networks to be trained. The output from the residual blocks is passed through a fully connected layer. The number of outputs of this layer is equal to the number of image classes to be classified. The results are passed through a softmax layer (exponential mean function) to calculate the probability of

each class of image data being classified. Finally, the image classification layer performs image classification, identifying which class the input image belongs to among the labeled classes.

During training, the accuracy and loss values for the training set and validation set are calculated and displayed on the graph. These quantities are calculated as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3)$$

Where:

TP: True Positive: occurs when predicting an image as belonging to a class and it is a precise prediction;  
 TN: True Negative: occurs when predicting an image not to belong to a class and this is a correct prediction;  
 FP: False Positive: occurs when predicting an image as belonging to a class when it actually does not; and  
 FN: False Negative: occurs when predicting an image not to belong to a class when it actually does belong to that class.

Accuracy as defined above is the number of correct predictions out of the total number of predictions.

$$Loss = - \sum_{i=1}^n \sum_j^m y_{i,j} \log_e(p_{i,j}) \quad (4)$$

Where:

n: number of images, m: number of classes,  $y_{i,j}$  is the true label of the image,  $y_{i,j} = 1$  if image  $i$  belongs to class  $j$ ,  $y_{i,j} = 0$  if image  $i$  does not belong to class  $j$ ;  $p_{i,j}$  is the label predicted by the model,  $p_{i,j}$  is the probability that image  $i$  belongs to class  $j$ .

The value of the loss function indicates the difference between the network's predicted label and the true label of the data.

Since in our ionogram classification problem, the number of images of the classes is different, we calculate the Precision, Recall and F1-score quantities.

$$Precision = \frac{TP}{TP + FP} \quad (5)$$

is the ratio of true positive images (TP) among the images classified as positive (TP + FP).

$$Recall = \frac{TP}{TP + FN} \quad (6)$$

is the ratio of true positive images (TP) among the images that are actually positive (TP + FN).

$$F_1 - score = 2 \frac{Precision \cdot Recall}{Precision + Recall} \quad (7)$$

is the harmonic mean of Precision and Recall.

### 3. Results

In this section, 14 models named Model\_1 to Model\_14 obtained by transfer training are presented. Model\_1 to Model\_9 were retrained from the ResNet-50 network, and Model\_10 to Model\_14 were retrained using the ResNet-101 network. All of these models use the Adam optimization algorithm and are trained over 15 epochs. The computer used is an Intel CPU Core i5-11400F- 2.6 GHz.

The accuracy and loss plots against the number of iterations during training of models 1, 3, and 6 are shown in Fig. 7. Models 1, 3, and 6 have InitialLearnRate parameters of 0.001 (the default value). 0.01 and 0.0001 are values that are increased or decreased by 10 times compared to the default value. For the other models, the accuracy and loss plots are similar to those of model 6, there is nothing special so they are not shown here.

The model training parameters (MiniBatchSize, InitialLearnRate, LearnRateDropFactor), accuracies (Test Accuracy, Evaluation Accuracy, F1- Score), and training time of each model are shown in the tables.

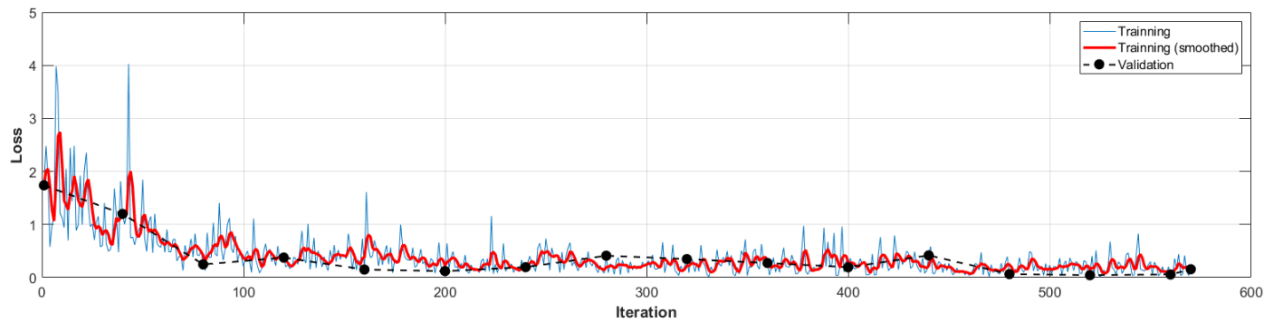
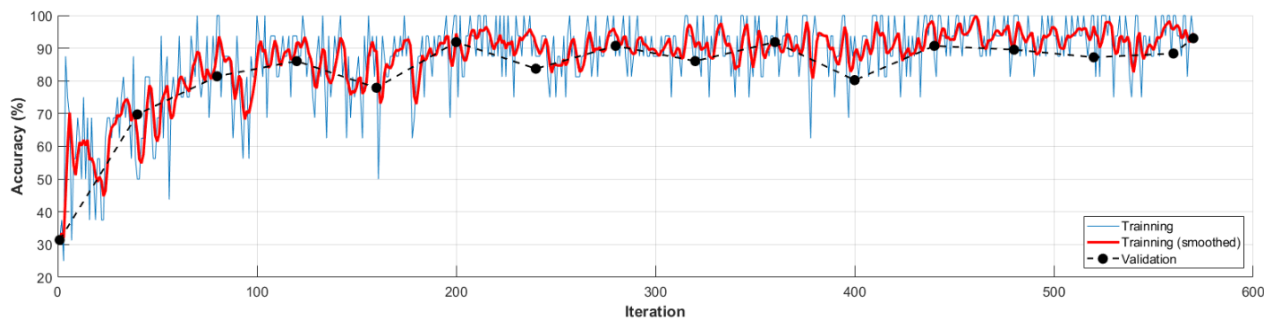
Table 1 shows the models Model\_1 - Model\_5 trained with the parameter MiniBatchSize = 16. The parameters InitialLearnRate and LearnRateDropFactor of Model\_1 are set to default, with values of 0.001 and 0.1, respectively. In the models from Model\_2 to Model\_5, these two parameters are increased and decreased by 10 times.

Table 2 shows the models Model\_5 - Model\_9. They are trained with different values of the parameter MiniBatchSize, while the parameters InitialLearnRate and LearnRateDropFactor are set to the same, with values of 0.0001 and 0.01, respectively.

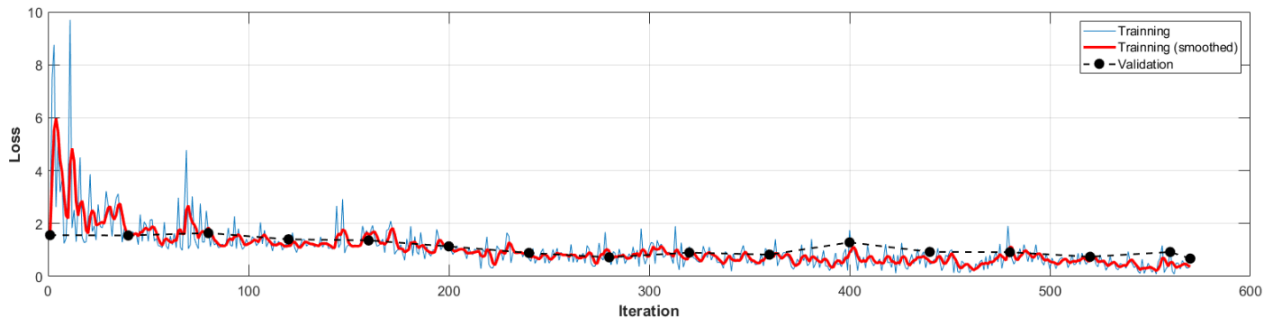
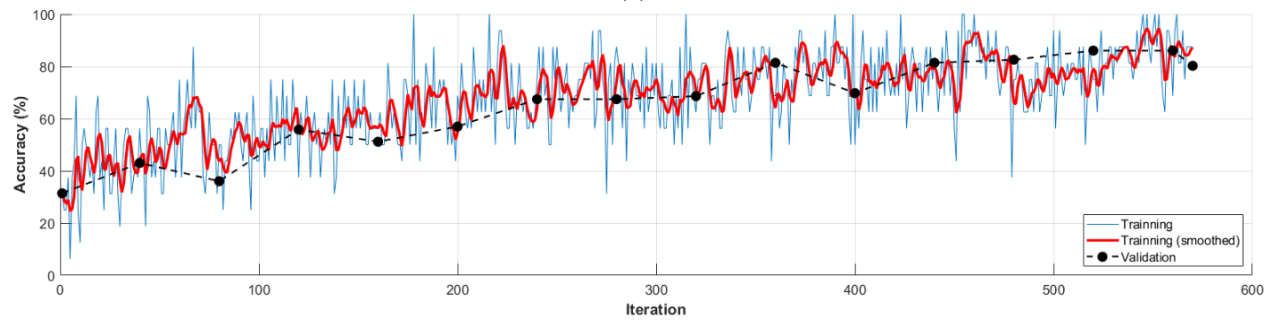
Table 3 shows the models Model\_10 - Model\_14 trained from the ResNet-101 network. They were trained with different values of the MiniBatchSize parameter, while the InitialLearnRate and LearnRateDropFactor parameters were set to the same, with values of 0.0001 and 0.01, respectively.

The accuracy of all 14 models when used to recognize the test set of ionogram images is shown in Figs 7 - 9. For each model, the number of images correctly identified is indicated by the blue numbers. The light brown numbers indicate the number of images incorrectly recognized.

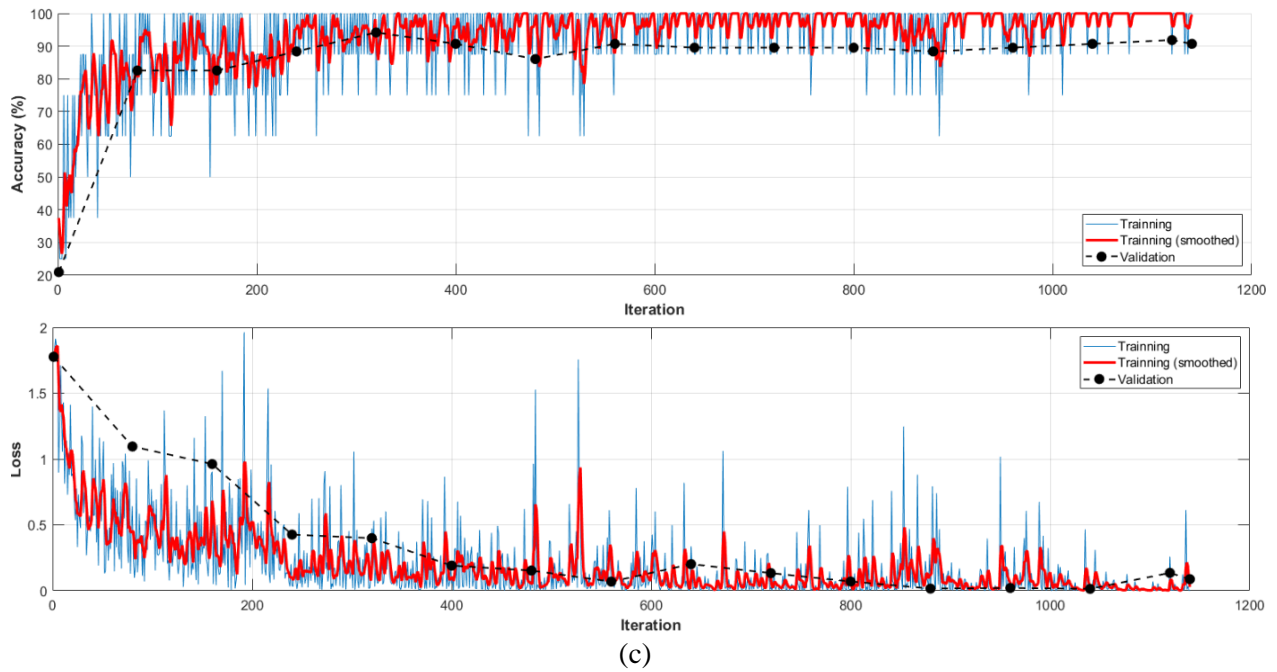
Table 4 presents the results of using the 14 trained models to recognize 10 ionogram images of 5 types of ionograms, each with 2 images. These images were not used in the dataset used to train and test the 14 deep learning networks mentioned above.



(a)



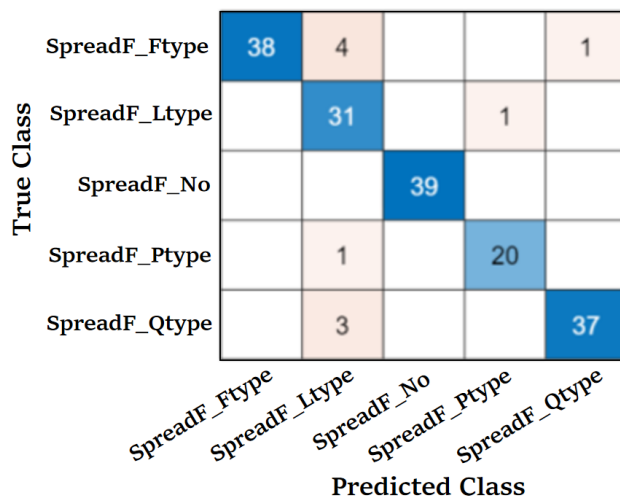
(b)



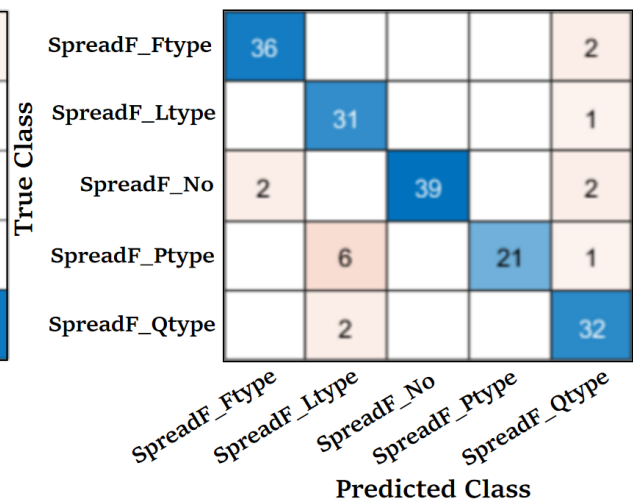
**Fig. 7.** Accuracy and loss versus number of iterations graph during training of models; (a) Model\_1 (b) – Model\_3, (c) Model\_6.

**Tab. 1.** Model\_1 to Model\_5 trained from ResNet-50 with MiniBatchSize = 16 parameters, EarlyLearnRate and LearnRateDropFactor parameters set to different values

ResNet-50							
Model	MiniBatch Size	InitialLearn Rate	LearnRate Drop Factor	Test accuracy (%)	Validation accuracy (%)	F1-Score (%)	Training time (minute)
Model_1	16	0.001	0.1	94,29	89,53	94,51	42,40
Model_2	16	0.001	0.01	90,86	83,72	90,98	42,52
Model_3	16	0.01	0.1	84,57	88,37	84,20	42,58
Model_4	16	0.0001	0.1	91,43	91,86	92,01	42,53
Model_5	16	0.0001	0.001	99,43	89,53	99,49	43,11



**Model\_1: Test Accuracy = 94,24%**



**Model\_2: Test Accuracy =90,86%**

True Class	SpreadF_Ftype	37	1			2
	SpreadF_Ltype		31		1	9
	SpreadF_No			36	2	
	SpreadF_Ptype		7		18	1
	SpreadF_Qtype	1		3		26
		SpreadF_Ftype	SpreadF_Ltype	SpreadF_No	SpreadF_Ptype	SpreadF_Qtype

Predicted Class

**Model\_3: Test Accuracy = 84,57%**

True Class	SpreadF_Ftype	37	5			4
	SpreadF_Ltype		30			
	SpreadF_No			39		1
	SpreadF_Ptype	1	3		21	
	SpreadF_Qtype		1			33
		SpreadF_Ftype	SpreadF_Ltype	SpreadF_No	SpreadF_Ptype	SpreadF_Qtype

Predicted Class

**Model\_4: Test Accuracy = 91,43%**

**Fig. 8.** Accuracy of the models (Model\_1 – Model\_4) in identifying the ionograms used for the test. The blue figures are the correct results, the light brown figures are the number of images that the model misidentified.

True Class	SpreadF_Ftype	38	1			
	SpreadF_Ltype		38			
	SpreadF_No			39		
	SpreadF_Ptype				21	
	SpreadF_Qtype					38
		SpreadF_Ftype	SpreadF_Ltype	SpreadF_No	SpreadF_Ptype	SpreadF_Qtype

Predicted Class

**Model\_5: Test Accuracy = 99,43%**

True Class	SpreadF_Ftype	37				
	SpreadF_Ltype	1	38		4	1
	SpreadF_No			39		
	SpreadF_Ptype		1		17	
	SpreadF_Qtype					37
		SpreadF_Ftype	SpreadF_Ltype	SpreadF_No	SpreadF_Ptype	SpreadF_Qtype

Predicted Class

**Model\_6: Test Accuracy = 96,00%**

True Class	SpreadF_Ftype	36	1			
	SpreadF_Ltype	1	37			
	SpreadF_No			38		
	SpreadF_Ptype		1		21	
	SpreadF_Qtype	1		1		38
		SpreadF_Ftype	SpreadF_Ltype	SpreadF_No	SpreadF_Ptype	SpreadF_Qtype

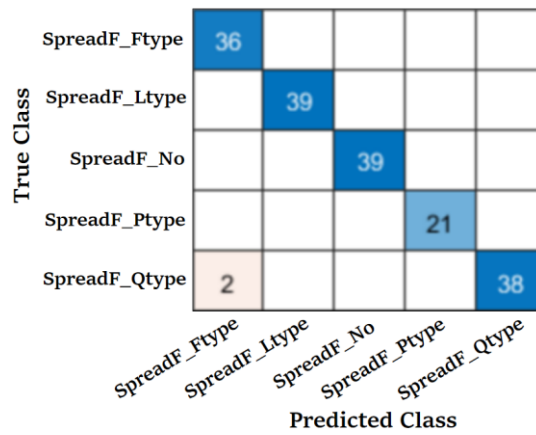
Predicted Class

**Model\_7: Test Accuracy = 91,14%**

True Class	SpreadF_Ftype	36				
	SpreadF_Ltype	1	39			
	SpreadF_No			39		
	SpreadF_Ptype				21	
	SpreadF_Qtype	1				38
		SpreadF_Ftype	SpreadF_Ltype	SpreadF_No	SpreadF_Ptype	SpreadF_Qtype

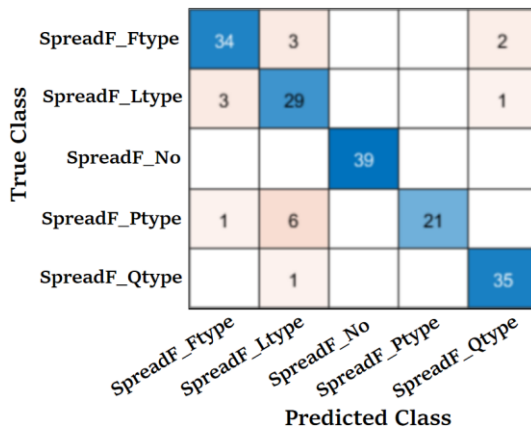
Predicted Class

**Model\_8: Test Accuracy = 98,86%**

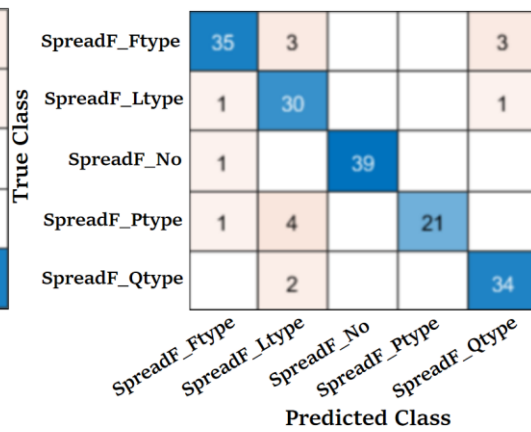


**Model\_9: Test Accuracy = 98,86%**

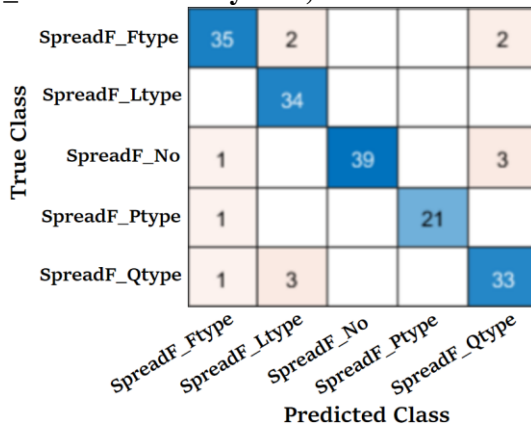
**Fig. 9.** Accuracy of the models (Model\_5 – Model\_9) in identifying the ionograms used for the test. The blue figures correspond to the correct results, the light brown figures to the number of images that the model misidentified.



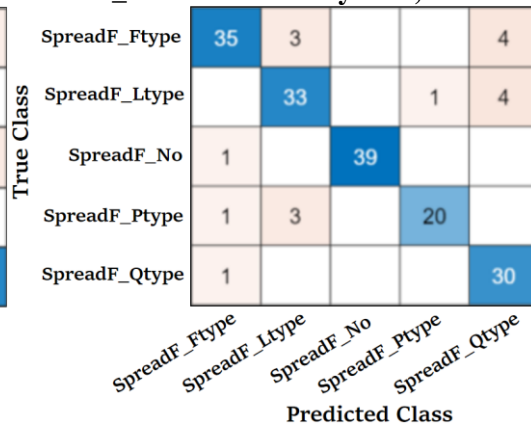
**Model\_10: Test Accuracy = 90,29%**



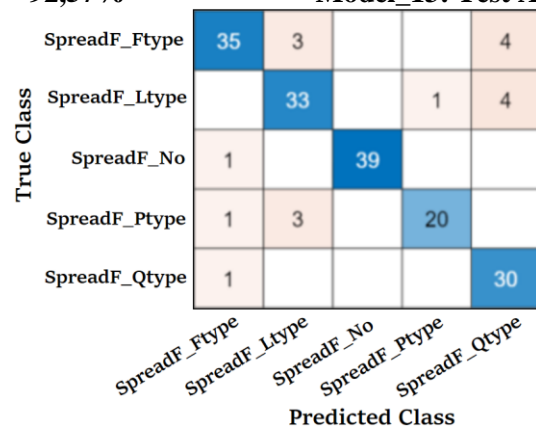
**Model\_11: Test Accuracy = 90,86%**



**Model\_12: Test Accuracy = 92,57%**



**Model\_13: Test Accuracy = 89,71%**



**Model\_14: Test Accuracy = 92,00%**

**Fig. 10.** Accuracy of the models (Model\_10 – Model\_14) in identifying the ionograms used for the test. The blue figures correspond to the correct results, the light brown figures to the number of images that the model misidentified.

**Tab. 2.** Model\_5 to Model\_9 trained from ResNet-50. The MiniBatchSize parameter has different values, the EarlyLearnRate and LearnRateDropFactor parameters of the models are set equal, with values of 0.0001 and 0.01, respectively.

ResNet-50					
Model	MiniBatchsize	Test accuracy (%)	Validation accuracy (%)	F1- Score (%)	Training time (minute)
Model_5	16	99.43	89.53	99.49	43.11
Model_6	8	96.00	91.86	95.39	49.14
Model_7	32	97.14	89.53	97.22	39.41
Model_8	48	98.86	91.86	98.97	36.28
Model_9	64	98.86	90.70	98.97	36.07

**Tab. 3.** Model\_10 to Model\_14 trained from ResNet-101 network. The MiniBatchSize parameter of the models has different values, the EarlyLearnRate and LearnRateDropFactor parameters are set equal, with values of 0.0001 and 0.01, respectively.

ResNet-101					
Model	MiniBatchsize	Test accuracy (%)	Validation accuracy (%)	F1- Score (%)	Training time (minute)
Model_10	8	90.29	91.86	90.31	80.50
Model_11	16	90.86	91.86	91.03	69.10
Model_12	32	92.57	89.53	93.12	62.28
Model_13	48	89.71	89.53	89.87	57.53
Model_14	64	92.00	91.86	91.89	57.21

**Tab. 4.** Results of using 14 models to recognize 10 ionograms of the 5 types, each type has 2 images. Grey cells are correctly recognized, white cells are incorrectly recognized.

Ionogram's name	ResNet -50									ResNet -101				
	Model_1	Model_2	Model_3	Model_4	Model_5	Model_6	Model_7	Model_8	Model_9	Model_10	Model_11	Model_12	Model_13	Model_14
0403061445 P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
0501061415 P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
0014040920 23Q	Q	Q	Q	Q	P	Q	Q	Q	Q	Q	Q	Q	Q	Q
0015111020 23L	L	L	L	L	L	L	P	P	L	L	P	L	L	L
1518100820 23Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q
3014131020 23L	P	P	P	L	P	L	P	P	L	P	P	L	L	P
3016100120	F	F	F	F	F	F	F	F	F	F	F	F	F	F

23F														
4520151020 23F	P	F	Q	F	P	F	F	F	F	F	P	F	F	F
4518091020 23K	K	K	Q	K	K	K	K	K	K	K	K	K	K	K
4521241020 23K	Q	Q	P	K	P	K	K	P	P	K	K	K	K	K

#### 4. Discussion

The image classification recognition models were trained with different MiniBatchsize, InitialLearnRate and LearnRate Drop Factor parameters. Table 1 shows that the value of InitialLearnRate strongly affects the convergence speed of the learning process and the classification results. If the initial learning rate is high (model 3), the learning process with 15 epochs is not sufficient (Fig. 7), but must be trained by many epochs with a longer learning time. Table 1 shows that the LearnRate Drop Factor parameter (0.01 by default) is increased or decreased 10 times and that the results are not very different.

When using the ResNet-50 architecture (models 5 to 9 in Table 2) and the ResNet-101 architecture (models 10 to 14 in Table 3), it can be seen that the MiniBatchsize value set from 8 to 64 does not clearly affect the image classification results.

Most of the models have test accuracy and F1-score greater than 90% (except model 3), and the validation accuracy is above or below 90%. These accuracy values are quite high, so the image recognition and classification results of the built models are reliable.

When using all 14 models to recognize 10 new ionograms, we see that 4 models correctly recognize all 10 images, which are models 4, 6, 12, and 13; 4 and 6 are 2 ResNet-50 models, and 12 and 13 are 2 ResNet-101 models. The figure also shows that it seems that the ResNet-101 models recognize more accurately than the ResNet-50 models.

Model 5 has the highest accuracy for the test data (it correctly recognizes 174 out of 175 test images) but now only correctly recognizes 6 out of 10 images. Of these 10 images, 4 are correctly recognized by all 14 models, and 2 are correctly recognized by 13 out of 14 models. There are 3 images with the number of correctly identified models being 11, 10, and 8 respectively. Image 301413102023L is a Mixed Spread-F (MSF) image, but only 5 models correctly identified it as Spread-F (MSF) image, and the remaining 9 models incorrectly identified it as Branch Spread-F (BSF). When observed by the eye, this image looks a bit like Branch Spread-F (BSF), so it was mistakenly identified by many models.

Ionograms are taken continuously every 15 minutes, so when an image is classified into different types, we can compare it with the identification results of the images taken immediately before and after to decide which type it belongs to. To be more certain, these images can be visually checked by experienced interpreters.

The results of using 14 models to recognize 10 ionograms in Table 4 show that there are 116 correct results out of a total of 140 results. Of the total 24 incorrect results, 20 are of type P (Branch Spread-F), indicating that it may be due to the difference in the number of ionograms used when training the models: the number of type P (Branch Spread-F) images used when training is only about 50% of the number of other types of images. Therefore, when training, increasing the number of type P ionogram to the same number as the number of other types of ionograms can reduce this error.

#### 5. Conclusion

Fine-tuning pre-trained deep learning networks such as ResNet-50 and ResNet-101 enables working with small ionogram dataset and saves time of model training. The results of applying the 14 constructed networks to recognize and classify ionograms show an average accuracy of approximately 90%. To obtain classification results with higher accuracy, a dataset with a uniform number of ion image types is required. On the other hand, since the networks have been pre-trained on natural image datasets and have learned their high-order features, which may limit the transfer learning process on the ionogram dataset, it may be necessary to intervene more deeply in the network training process, such as (Ji et al. 2019) removing some convolutional layers of the pre-trained network to train on the ionogram dataset. This will be our work in the next research.

#### Acknowledgements

The authors thank Dr. Truong Ngoc Son for helpful advice on training the CNN models, Dr. Christine Amory-Mazaudier for fruitful scientific discussions, M.Sc. Nguyen Van Thuan for helping us revise the manuscript, and anonymous reviewers for improving the work.

### Declarations

*Competing interests:* The authors declare no competing interests.

### Author Contribution

Phạm Thi Thu Hong prepared the ionogram dataset for recognition and classification. Luu Viet Hung trained the CNN models and wrote the manuscript.

### Data availability statement

The data used in this research can be made available by the corresponding author upon reasonable request.

### Literature - References

1. Abadi, Prayitno, Umar Ali Ahmad, Yuichi Otsuka, Punyawati Jamjareegulgarn, Dyah Rahayu Martiningrum, Agri Faturahman, Septi Perwitasari, Randy Erfa Saputra, and Reza Rendian Septiawan. 2022. "Modeling post-sunset equatorial spread-F occurrence as a function of evening upward plasma drift using logistic regression, deduced from ionosondes in Southeast Asia", *Remote Sensing*, 14: 1896.
2. Booker, HG, and Wells HW. 1938. "Scattering of radio waves by the F-region of the ionosphere", *Terrestrial Magnetism Atmospheric Electricity*, 43: 249-56.
3. He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. "Deep residual learning for image recognition." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770-78.
4. Ji, Qingge, Jie Huang, Wenjie He, and Yankui Sun. 2019. "Optimized deep convolutional neural networks for identification of macular diseases from optical coherence tomography images", *Algorithms*, 12: 51.
5. Karim, R. (2019). Illustrated: 10 cnn architectures. *Towards data science*.
6. Kingma, D. P. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:*
7. Lan, Ting, Hui Hu, Chunhua Jiang, Guobin Yang, and Zhengyu Zhao. 2020. "A comparative study of decision tree, random forest, and convolutional neural network for spread-F identification", *Advances in Space Research*, 65: 2052-61.
8. Lan, Ting, Yuannong Zhang, Chunhua Jiang, Guobin Yang, and Zhengyu Zhao. 2018. "Automatic identification of Spread F using decision trees", *Journal of Atmospheric Solar-Terrestrial Physics*, 179: 389-95.
9. Lan, Trần Thị, and Đào Thế Cường. 2013. "Some characteristic of equatorial spread F at Phu Thuy over a solar cycle", *Vietnam Journal of Earth Sciences*, 35: 265-71.
10. MathWorks. 2025. Get Started with Transfer Learning. Retrieved from <https://www.mathworks.com/help/deeplearning/gs/get-started-with-transfer-learning.html>
11. MATLAB, V. 2022. 9.12. 0 (R2022a). *The MathWorks Inc.: Natick, MA, USA*.
12. Pillat, V G, Fagundes PR, and Lamartine Nogueira Frutuoso Guimarães. 2015. "Automatically identification of Equatorial Spread-F occurrence on ionograms", *Journal of Atmospheric Solar-Terrestrial Physics*, 135: 118-25.
13. Piggott, W. R., & Rawer, K. (1972). *URSI handbook of ionogram interpretation and reduction*.
14. Purwono, P., Ma'arif, A., Rahmani, W., Fathurrahman, H. I. K., Frisky, A. Z. K., & ul Haq, Q. M. 2022. Understanding of convolutional neural network (cnn): A review. *International Journal of Robotics Control Systems*, 2(4), 739-748.
15. Taye, M. M. 2023. Theoretical understanding of convolutional neural network: Concepts, architectures, applications, future directions. *Computation*, 11(3), 52.
16. Thu, Hong Phạm Thi, Amory MC, Minh Le Huy, Susumu Saito, Dung Nguyen Thanh, Ngoc Luong Thi, Hung Luu Viet, Thang Nguyen Chien, Thanh Nguyen Ha, and Nishioka Michi. 2024. "Occurrence rate of equatorial Spread F and GPS ROTI in the ionospheric anomaly region over Vietnam", *Vietnam Journal of Earth Sciences*, 46: 553-69.
17. Woodman P., Francisco R., La Hoz C. 1976. Radar observations of F region equatorial irregularities, *Journal of Geophysical Research*, 81: 5447-66.