TROPICAL CYCLONE INTENSITY ESTIMATION USING SATELLITE IMAGERY

TEAM MEMBERS

1. Mohammed Saad Belgi UID: 2021700005
2. Aditi Bhutada UID: 2021700009
3. Janhavi Deshmukh UID: 2021700017

- Guided by Prof. Jignesh Sisodia

INTRODUCTION

- Accurately estimating tropical cyclone (TC) intensity is one of the most critical steps in TC forecasting and disaster warning/management.
- Accurate estimation of cyclone intensity is crucial for effective disaster preparedness, response, and mitigation efforts.

TRAITIONAL APPROACH: DVORAK TECHNIQUE

- For over 40 years, the Dvorak technique has been applied for estimating TC intensity by forecasters worldwide.
- It is an empirical technique that relies on the analysis of cloud patterns, including the organization and symmetry of the cloud system.
- Meteorologists look for specific cloud features, such as the arrangement of spiral bands, the development of an eye, and the cloud top temperatures.
- Subjectivity and uncertainty: Different analysts may provide slightly different estimates as subjective interpretation is involved.

LIMITATIONS OF TRADITIONAL APPROACHES

Subjectivity:

- Reliance on human judgment over empirical data.
- Can introduce biases and inconsistencies.
- Addressed by data-driven and automated approaches.

Reliance on Cloud Patterns:

- Reliance on human judgment over empirical data.
- Can introduce biases and inconsistencies.
- Addressed by data-driven and automated approaches.

Comparison accross Centers:

- Dvorak technique intensity estimates may differ among meteorological centers due to varying interpretation criteria, training, and analysis methods.

Limited Timeliness:

- Manual application of the Dvorak technique may not always provide timely intensity estimates, particularly during rapidly evolving cyclone situations.
- Delays in analysis and reporting could impact the effectiveness of early warning systems and disaster response efforts.

Sensitivity to T-Number:

- Minor variations in T-number, a measure of intensity in the Dvorak technique, can cause significant discrepancies in estimated maximum wind speed.
- Sometimes exceeding 12 knots, especially at hurricane intensities, highlighting sensitivity to cloud feature changes.

DATASET

×

	Digital Typhoon dataset
Temporal coverage	1978-2022 (present)
Temporal resolution	one hour
Target satellites	Himawari
Spatial coverage	Western North Pacific basin
Spatial resolution	5km
Image coverage	512×512 pixels (1250km from the center)
Spectral coverage	infrared (others on the Website)
Map projection	Azimuthal equal-area projection
Calibration	Recalibration
Data format	HDF5
Best track	Japan Meteorological Agency
Dataset browsing	Digital Typhoon website

- The "**Digital Typhoon Dataset**" is a dataset of meteorological satellite images and the best track of typhoons in the Northwest Pacific basin .
- There are two main sources of data:
  i) satellite imagery   ii) best track data
- The meteorological satellite imagery is created from infrared imagery (11 micrometers) from geostationary meteorological satellites **Himawari 1 to 9** by a map projection (Lambert azimuthal equal-area) of a 1250 km radius centered on the typhoon.

- The best track is created from the best track data published by the Japan Meteorological Agency by interpolating it into **hourly data**. As of October 2023, it consists of **189364 images** for **1099 typhoons** from 1978 to 2022.
- Best Track data contains different parameters from typhoon sequences. Some of the parameters included in this dataset are the **wind speed, centre pressure, latitude and longitude, grade**

REGRESSION :
PREDICTION OF CENTRAL PRESSURE

WHY PREDICT CENTRAL PRESSURE?

- Central pressure deficit is an intensity measure that combines maximum wind speed, storm size, and background rotation rate. and directly relates to the storm's overall strength and potential impacts
- Previous studies have shown that with the accurate knowledge of the surface pressure field, uncertainty in the predicted landfall positions and forecast tracks can be reduced significantly.
- TC structure and dynamic characteristics (e.g. vortex characteristics and horizontal circulation characteristics) analyses are also strongly dependent on the accurate estimation of TC surface pressure fields.

IMPLEMENTATION

- Task: Predicting central pressure of a tropical cyclone using a single channel infrared satellite image of the cyclone.
- Extracted data of tropical cyclones from years 2016-2022 (181 cyclone sequences)
- Split the dataset into train, validation and test sets with 70:10:20 ratio
- Modified deep CNN models ResNet18, ResNet34 and ResNet50 to use single channel images as input and changing output size to 1 (for regression instead of classification)
- Initialized Resnet models with pre-trained imagenet weights (on imagenet database) and trained last few layers on our dataset
- Trained models with simple shallow CNN architectures from stratch on our dataset
- Experimented with model performance without and with image augmentation
- Loss function used: Mean Squared Error
- Optimization algorithm used: Adaptive Moment Estimation (Adam)

```
print('No of images in train set:', len(train_set))
print('No of images in validation set:', len(val_set))
print('No of images in test set:', len(test_set))
```

```
No of images in train set: 23535
No of images in validation set: 3362
No of images in test set: 6724
```

RESNET

ResNet was developed to address the degradation problem encountered when training very deep neural networks. As the network depth increases, accuracy typically saturates and then degrades rapidly. ResNet aims to overcome this issue by introducing skip connections

What are Skip Connections?

- The key innovation in ResNet is the use of skip connections, which allow the gradient to bypass one or more layers in the network. These skip connections add the original input of a layer to its output, creating shortcut paths that facilitate the flow of gradients during training.
- By introducing skip connections, ResNet enables the training of very deep networks (with hundreds or even thousands of layers) without suffering from vanishing gradients or degradation in accuracy.

Basic Building Block: Residual Block

- The core building block of ResNet is the residual block, which contains two convolutional layers with batch normalization and ReLU activation functions, along with a skip connection.
- The skip connection directly adds the input of the block to its output, creating a residual mapping that the network learns to adjust rather than learning the entire mapping from scratch.

RESNET

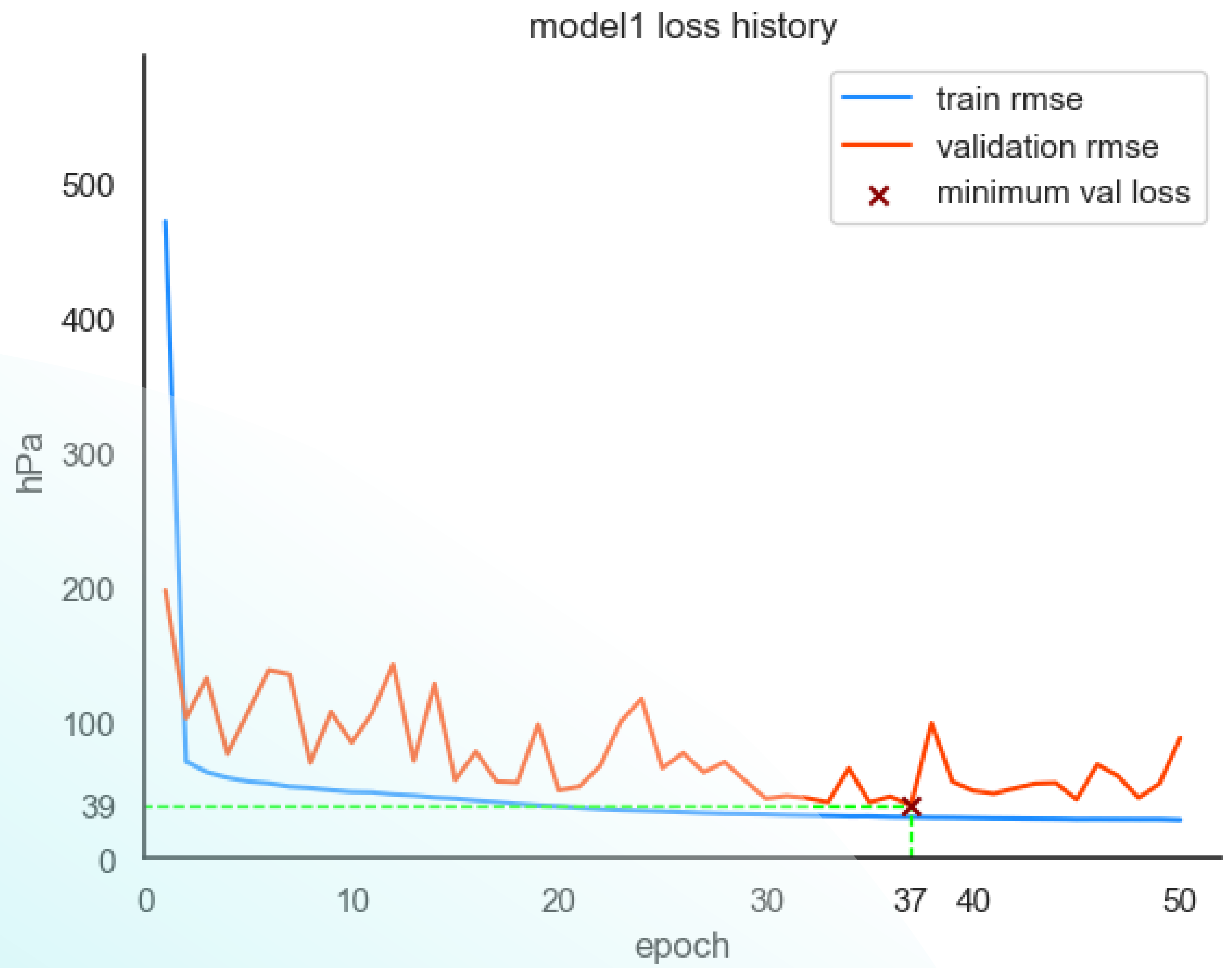
Architecture Variants:

- ResNet comes in several variants with different depths, including ResNet-18, ResNet-34, ResNet-50, ResNet-101, and ResNet-152.
- Deeper variants like ResNet-50, ResNet-101, and ResNet-152 achieve higher accuracy but require more computational resources for training and inference.

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix}3\times3, 64\\3\times3, 64\end{bmatrix}\times2$	$\begin{bmatrix}3\times3, 64\\3\times3, 64\end{bmatrix}\times3$	$\begin{bmatrix}1\times1, 64\\3\times3, 64\\1\times1, 256\end{bmatrix}\times3$	$\begin{bmatrix}1\times1, 64\\3\times3, 64\\1\times1, 256\end{bmatrix}\times3$	$\begin{bmatrix}1\times1, 64\\3\times3, 64\\1\times1, 256\end{bmatrix}\times3$
conv3_x	28×28	$\begin{bmatrix}3\times3, 128\\3\times3, 128\end{bmatrix}\times2$	$\begin{bmatrix}3\times3, 128\\3\times3, 128\end{bmatrix}\times4$	$\begin{bmatrix}1\times1, 128\\3\times3, 128\\1\times1, 512\end{bmatrix}\times4$	$\begin{bmatrix}1\times1, 128\\3\times3, 128\\1\times1, 512\end{bmatrix}\times4$	$\begin{bmatrix}1\times1, 128\\3\times3, 128\\1\times1, 512\end{bmatrix}\times8$
conv4_x	14×14	$\begin{bmatrix}3\times3, 256\\3\times3, 256\end{bmatrix}\times2$	$\begin{bmatrix}3\times3, 256\\3\times3, 256\end{bmatrix}\times6$	$\begin{bmatrix}1\times1, 256\\3\times3, 256\\1\times1, 1024\end{bmatrix}\times6$	$\begin{bmatrix}1\times1, 256\\3\times3, 256\\1\times1, 1024\end{bmatrix}\times23$	$\begin{bmatrix}1\times1, 256\\3\times3, 256\\1\times1, 1024\end{bmatrix}\times36$
conv5_x	7×7	$\begin{bmatrix}3\times3, 512\\3\times3, 512\end{bmatrix}\times2$	$\begin{bmatrix}3\times3, 512\\3\times3, 512\end{bmatrix}\times3$	$\begin{bmatrix}1\times1, 512\\3\times3, 512\\1\times1, 2048\end{bmatrix}\times3$	$\begin{bmatrix}1\times1, 512\\3\times3, 512\\1\times1, 2048\end{bmatrix}\times3$	$\begin{bmatrix}1\times1, 512\\3\times3, 512\\1\times1, 2048\end{bmatrix}\times3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

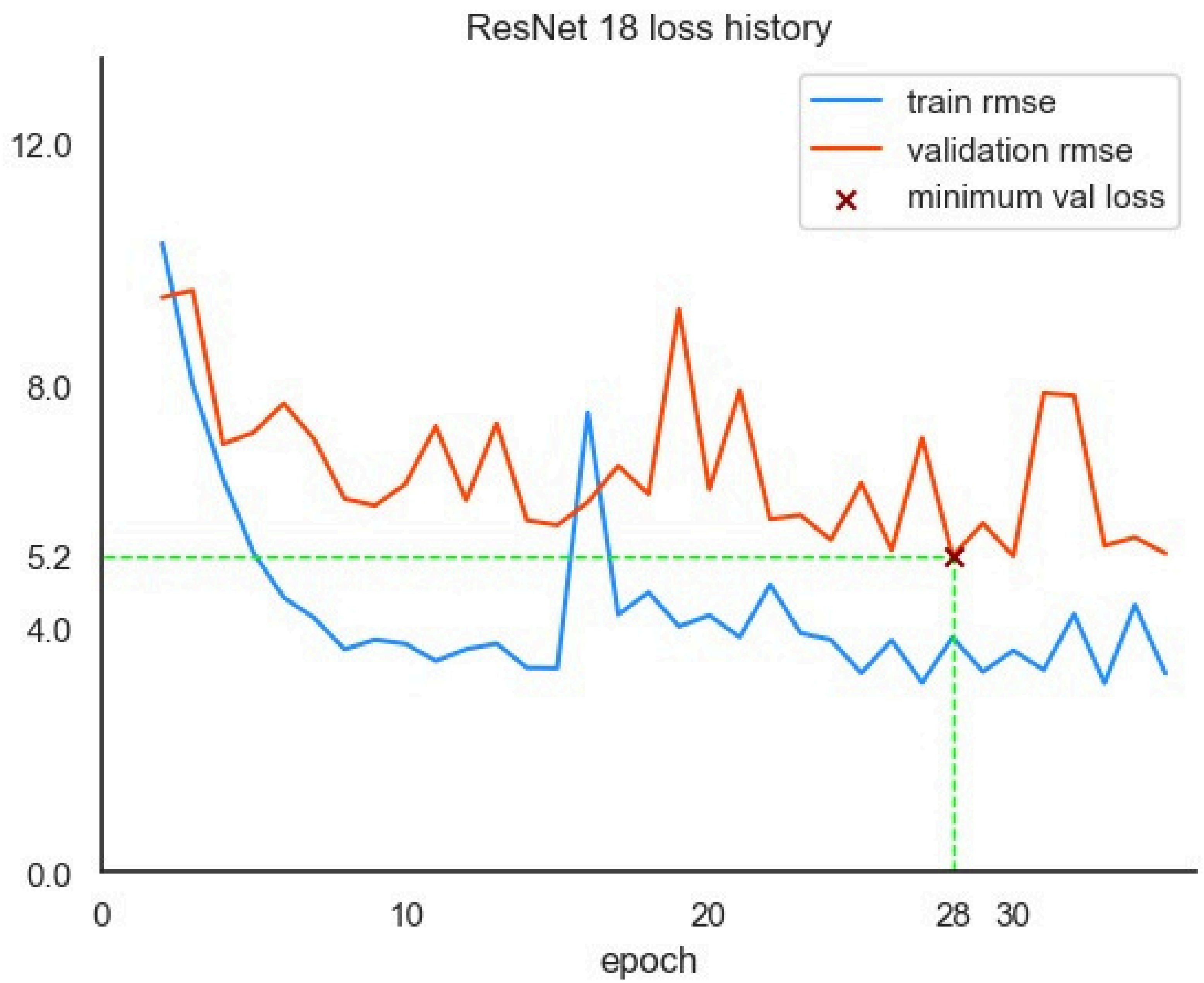
RESULTS:

Resnet 18
(Without augmentation & last block frozen)



- Best Val RMSE: 38.9537
- Best Test RMSE: 39.2766

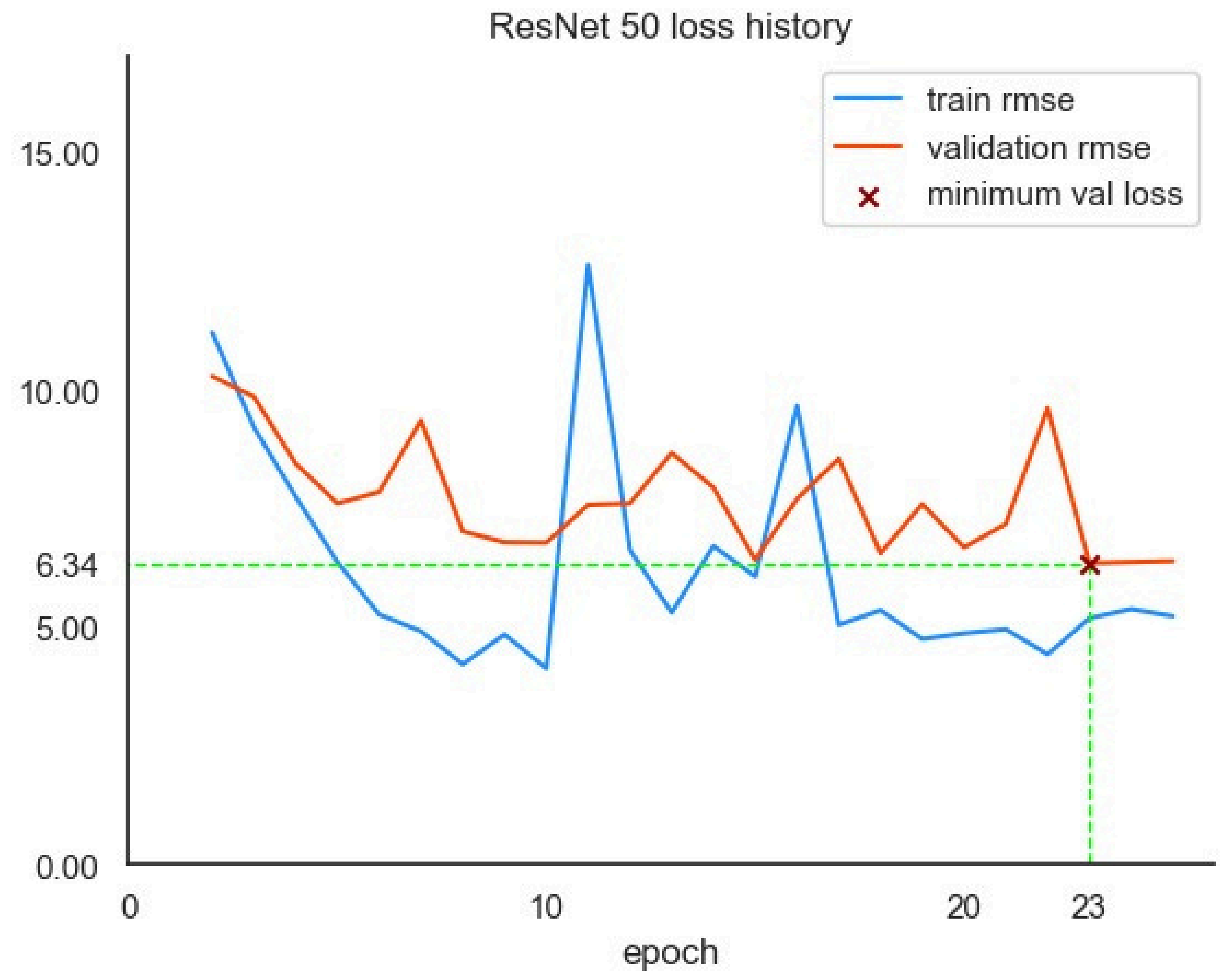
Resnet 18
(With augmentation & last block trainable)



- Best Val RMSE: 5.2072
- Best Test RMSE: 5.0847

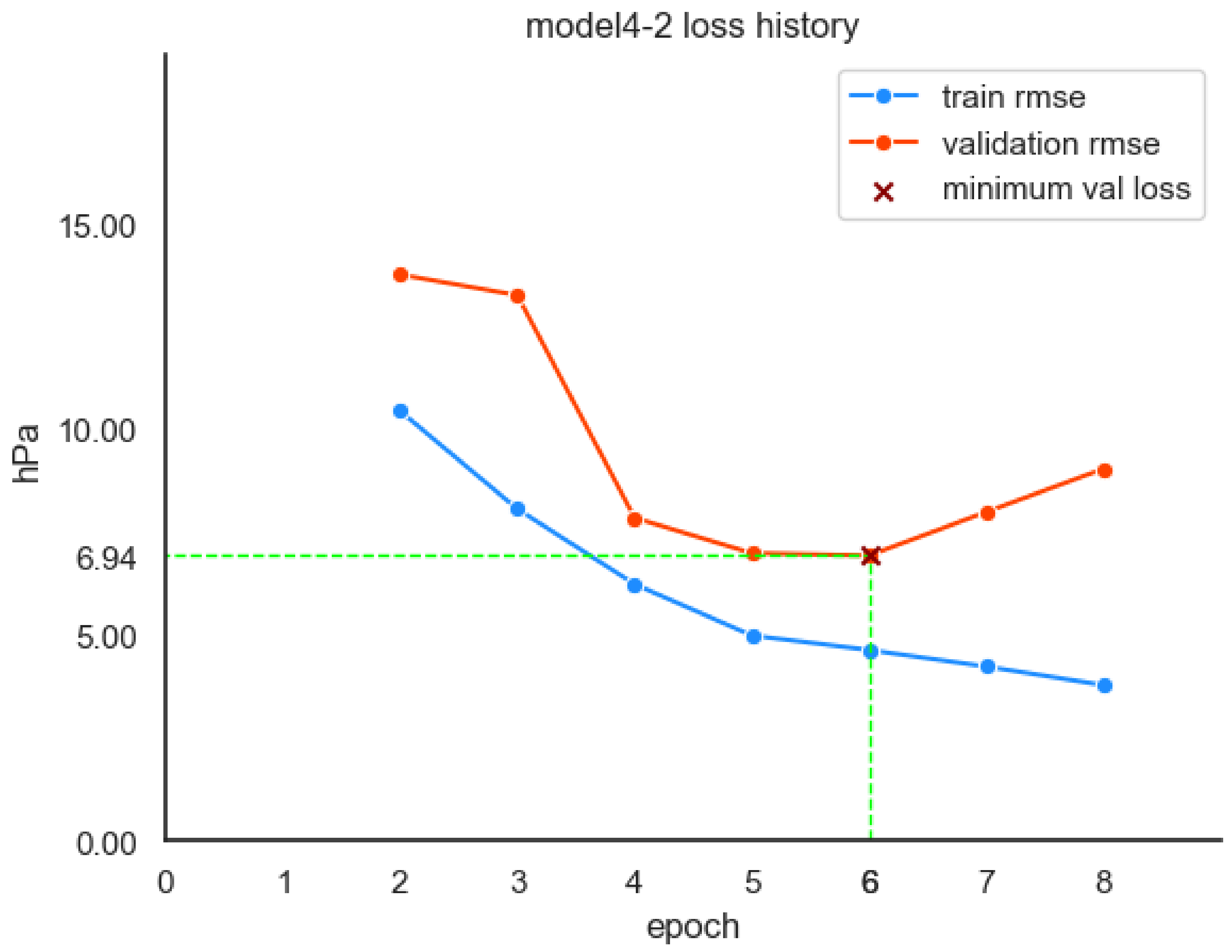
RESULTS:

Resnet 50
(With augmentation)



- Best Val RMSE: 6.3401
- Best Test RMSE: 6.1761

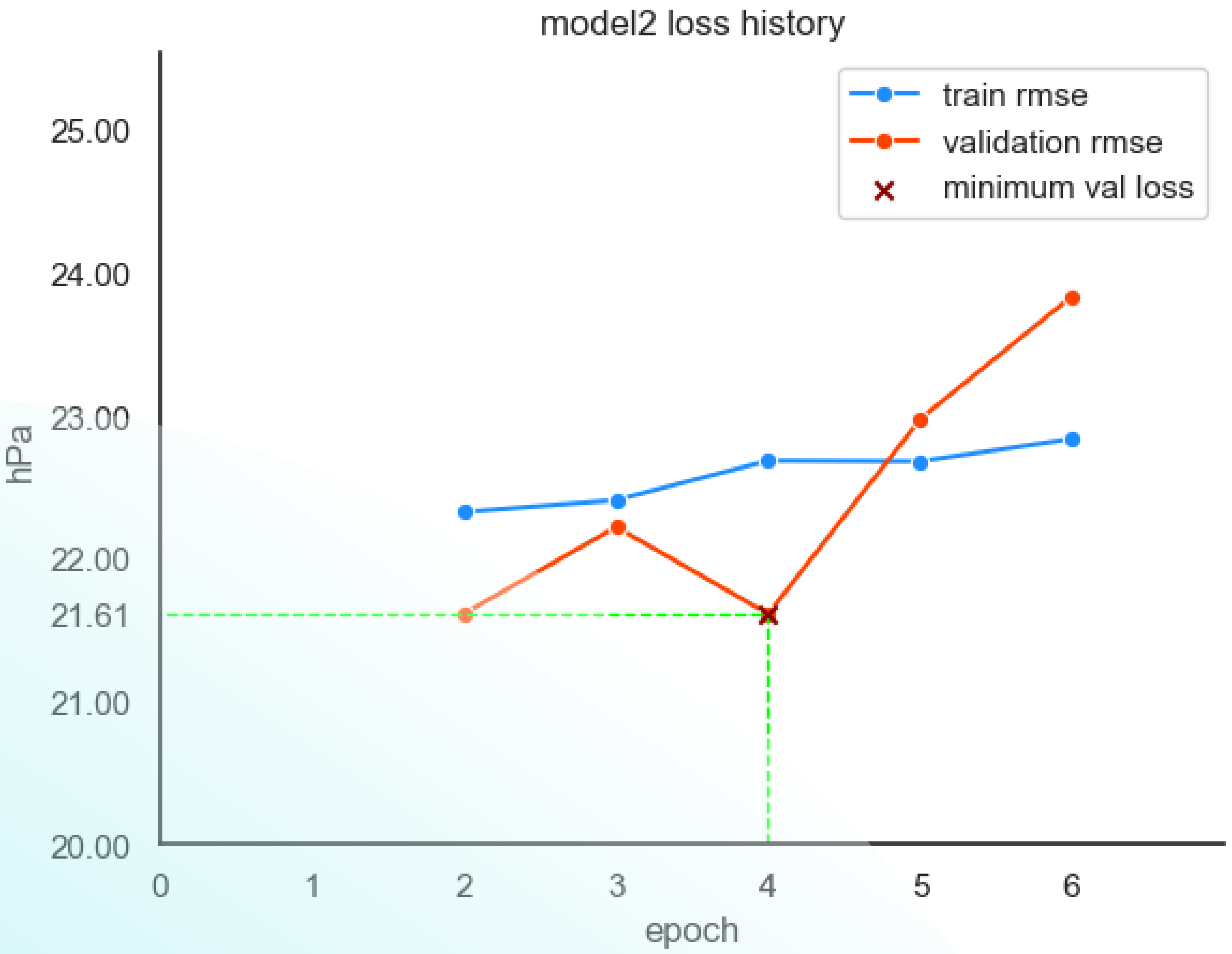
Resnet 34
(With augmentation)



- Best Val RMSE: 6.9387
- Best Test RMSE: 6.7541

RESULTS:

Custom CNN
(without augmentation)



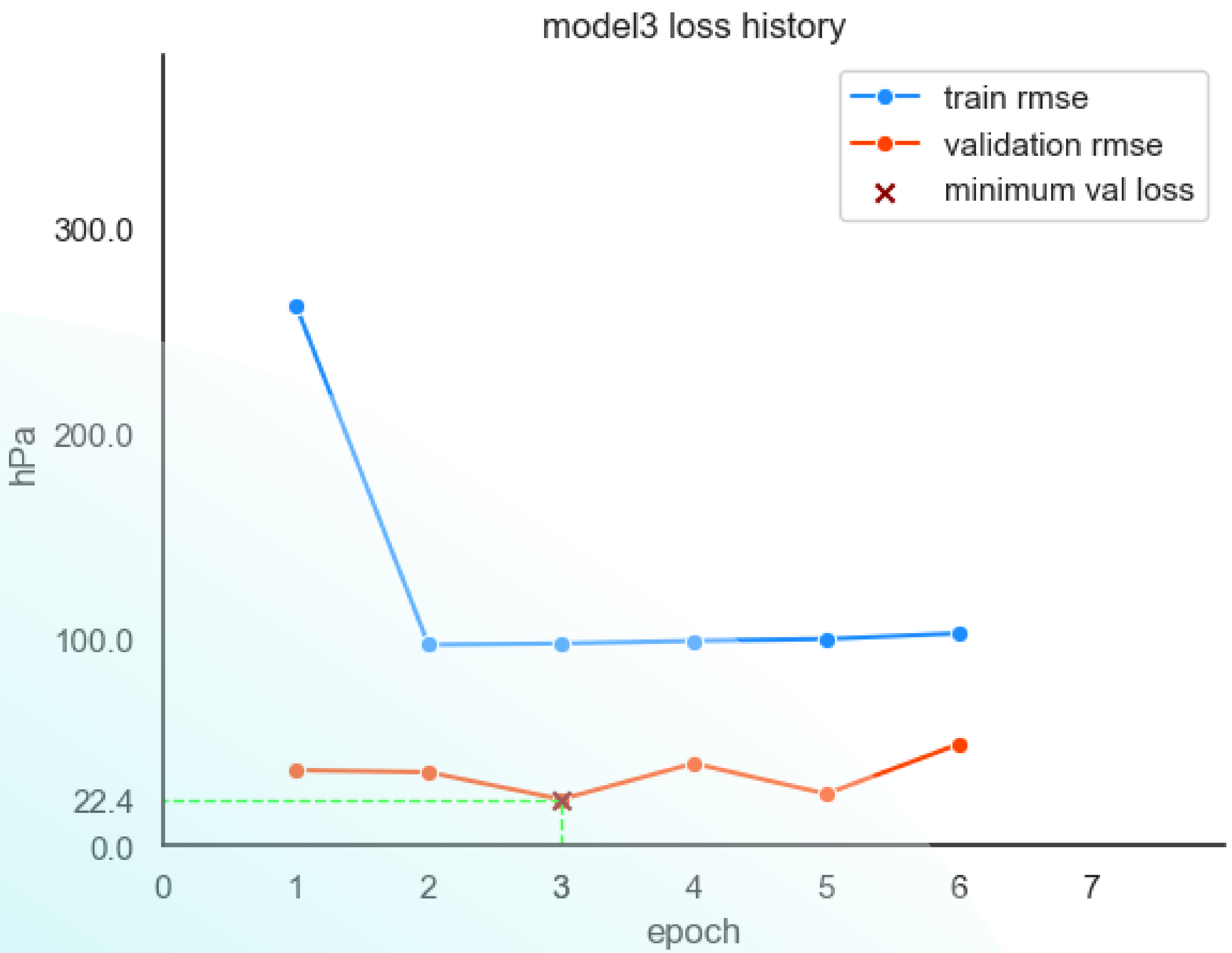
- BEST VAL RMSE: 21.6105
- BEST TEST RMSE: 22.0908

Architecture used:

SNo.	Type	Input channel of filters	Output channel filter	Filter/Pool size	Stride	Activation
1	Convolutional	1	32	4,4	2	ReLU
2	Convolutional	32	64	3,3	2	ReLU
3	Max pooling	-	-	2,2		ReLU
4	Convolutional	64	128	3,3	2	ReLU
5	Max pooling	-	-	2,2	-	-
6	Convolutional	128	256	3,3	2	ReLU
7	Max pooling	-	-	2,2	-	-
8	Fully connected layer	2304	64	-	-	ReLU
9	Fully connected layer	64	32	-	-	ReLU
10	Fully connected layer	32	1	-	-	Linear

RESULTS:

Custom CNN
(With augmentation)



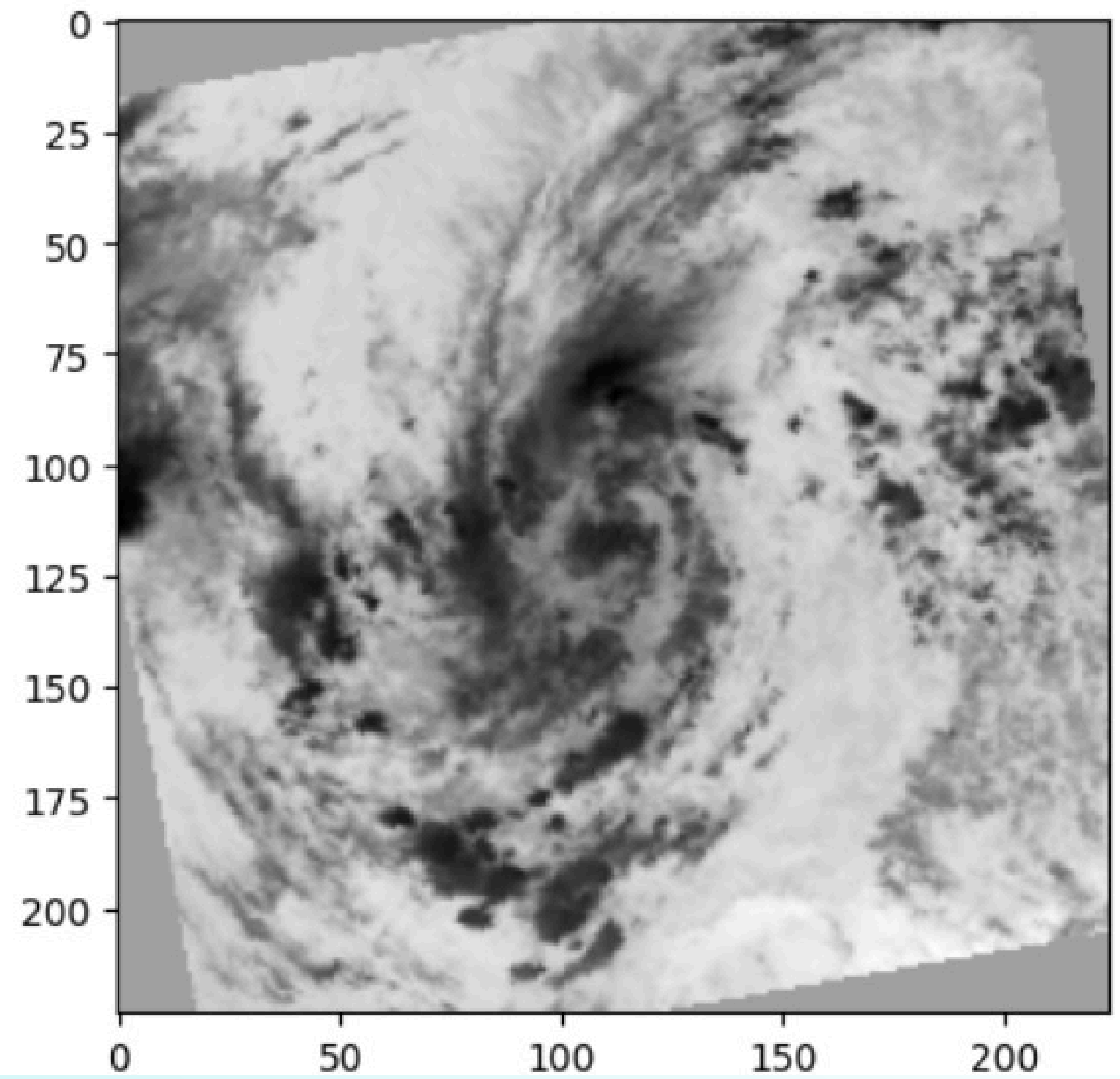
- Best Val RMSE: 22.445
- Best Test RMSE: 22.7868

Architecture used:

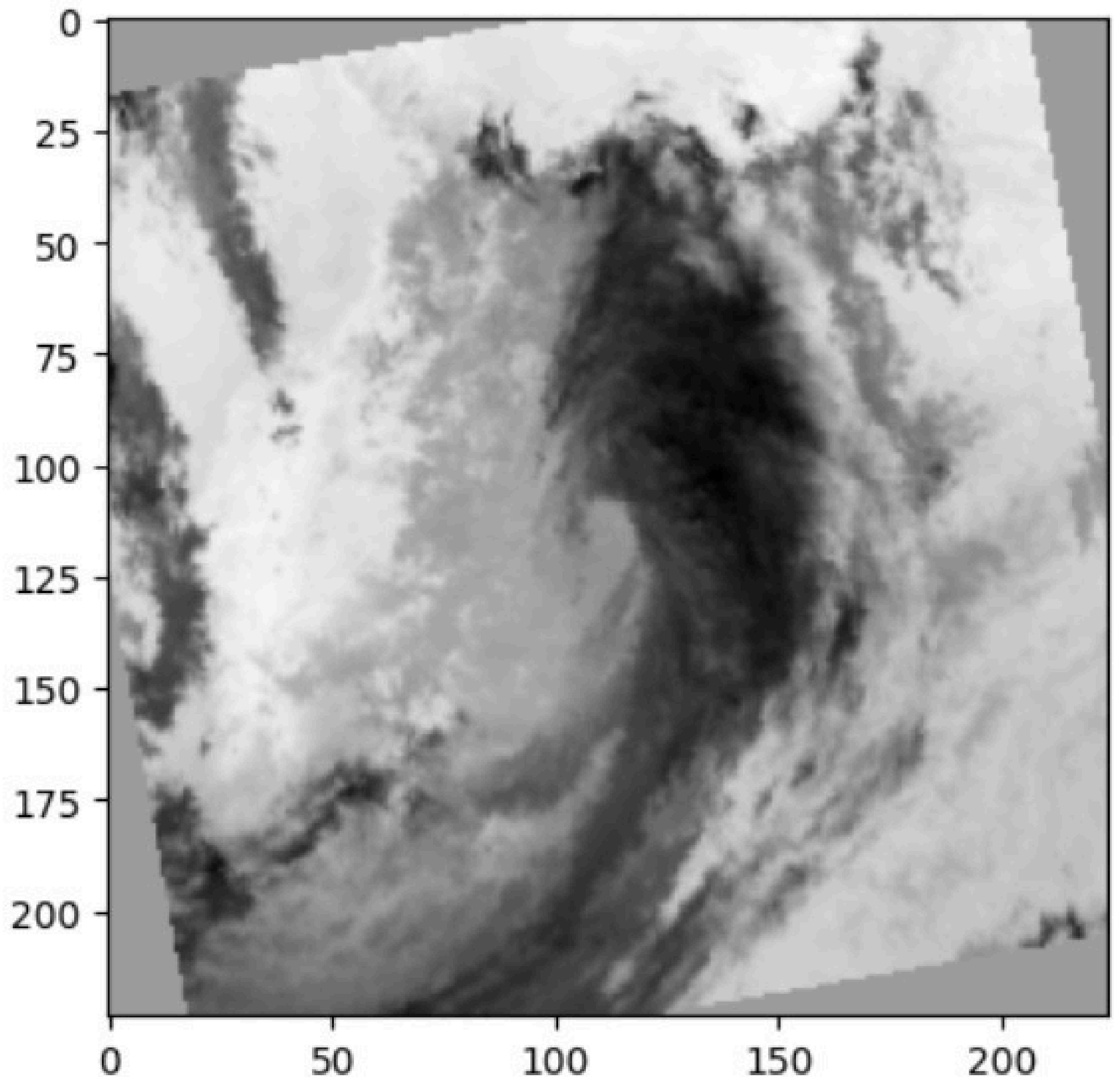
SNo.	Type	Input channel of filters	Output channel filter	Filter/Pool size	Stride	Activation
1	Convolutional	1	32	3,3	2	ReLU
2	Convolutional	32	32	3,3	2	ReLU
3	Max Pooling	-	-	2,2	-	-
4	Convolutional	64	64	3,3	2	ReLU
5	Max Pooling	-	-	2,2	2	-
6	Fully Connected Layer	576	128	-	-	Linear
7	Dropout	0.3	-	-	-	-

RESULTS (USING RESNET 18):

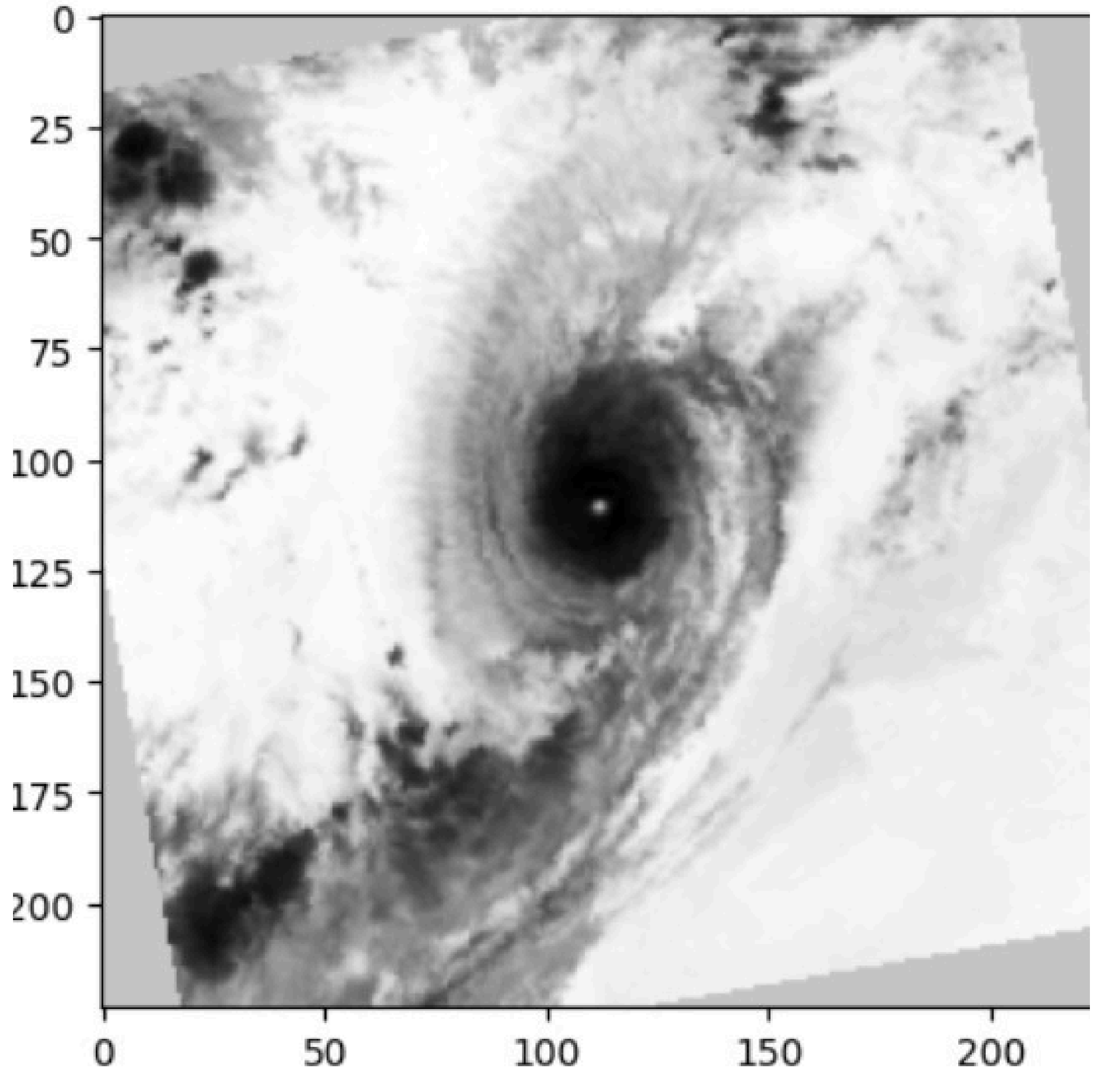
Correct central pressure: 994.0
Predicted central presssure: 994.2625122070312
RMSE: 0.26251219993578884



Correct central pressure: 976.7000122070312
Predicted central presssure: 979.734130859375
RMSE: 3.03411871066424



orrect central pressure: 905.0
redicted central presssure: 914.0277099609375
MSE: 9.027710111967632



CLASSIFICATION: PREDICTING INTENSITY GRADE

IMPLEMENTATION

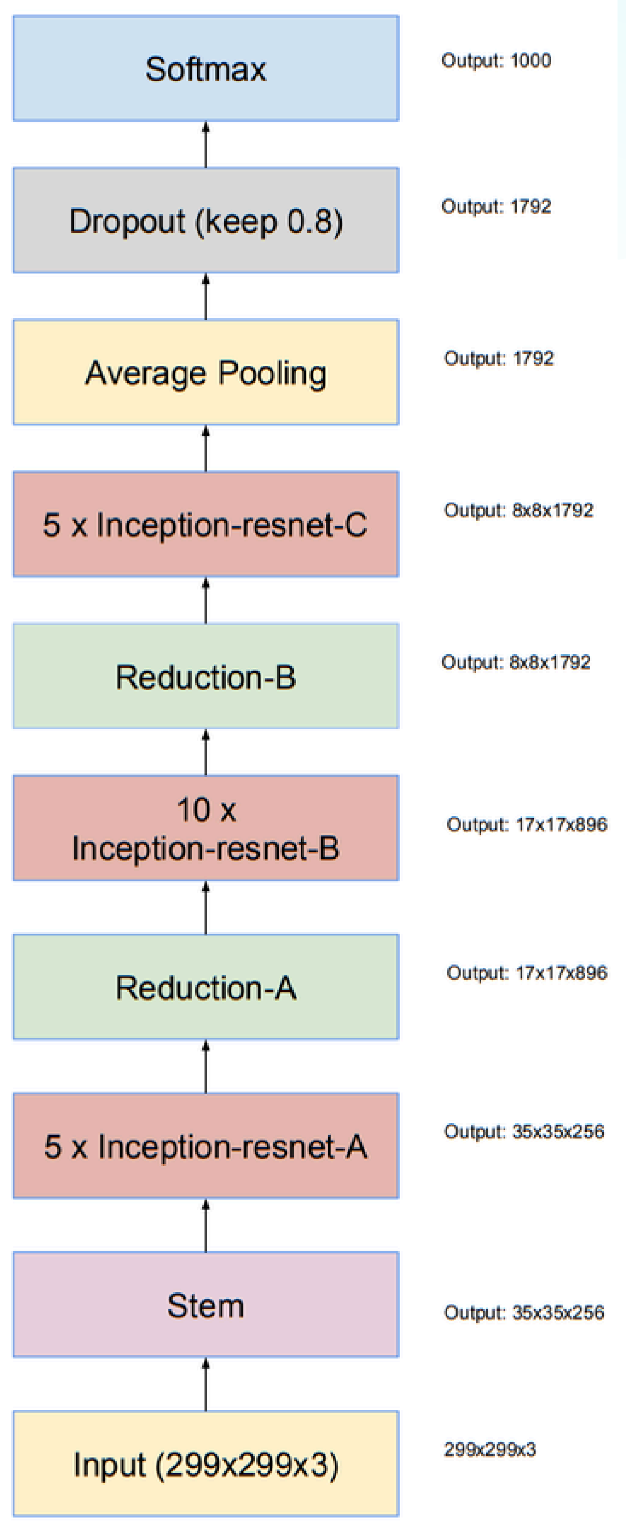
- Task: Classifying tropical cyclones into different grades, based on their intensity
- There are 5 output classes (grade 2 to 6). There were no images of grade 7 and 9 cyclones available in the dataset
- As the dataset was unbalanced, weighted random sampling was used to load images during training. Weights of a grade are inversely proportional to the proportion of the images of that grade in the training dataset
- Resnet 18 and 50 architectures (with pretrained weights of imagenet database) were tried with first layer modified to take single channel images and last layer modified for 5 output classes, with last few layers unfrozen
- Experimented with pretrained Inception V3 architecture with last few layers unfrozen
- Loss function used: Categorical Cross Entropy
- Optimization algorithm used: Adaptive Moment Estimation (Adam)

TC intensity grades according to JMA (Japan Meterological Intensity) best track:	
Grade	Classification and brief
1	Not used
2	Tropical Depression (TD)
3	Tropical Storm (TS)
4	Severe Tropical Storm (STS)
5	Typhoon (TY)
6	Extra-Tropical Cyclone (L)
7	Just entering into the responsible area of RSMC Tokyo-Typhoon Center
8	Not used
9	Tropical Cyclone of TS intensity or higher

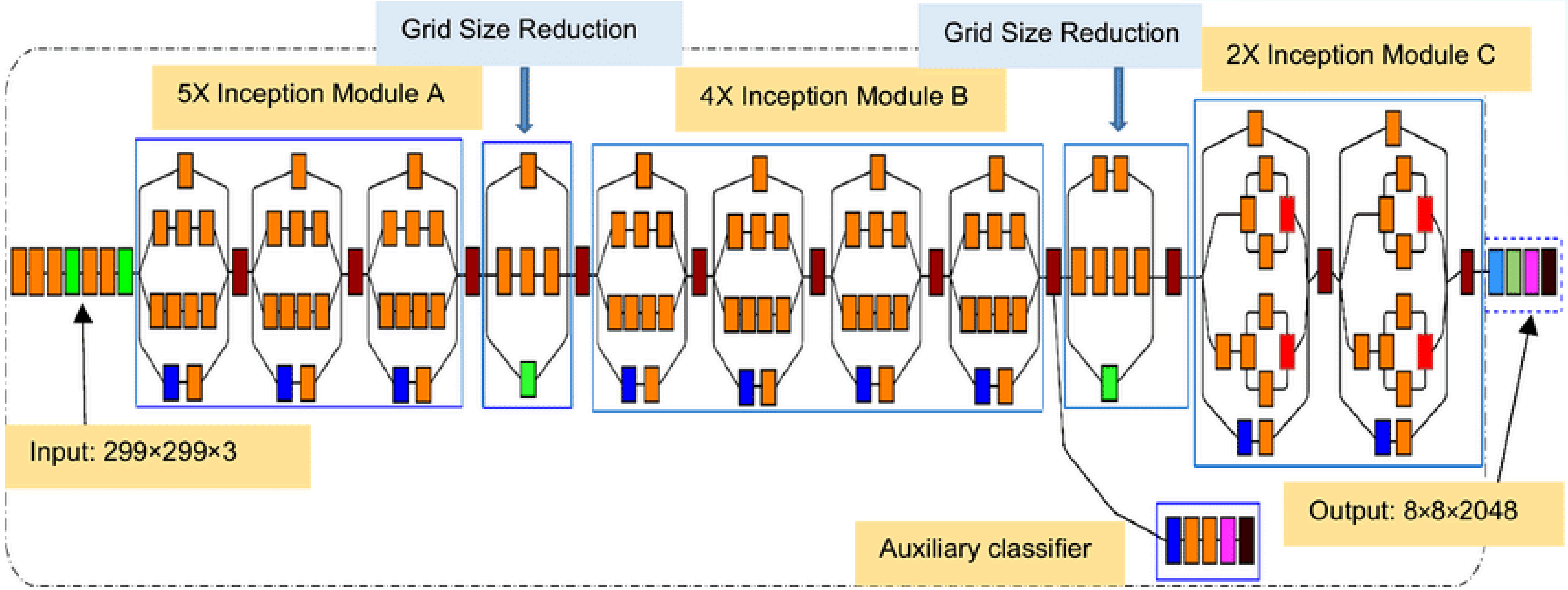
Grade	No. of Training samples - Initially	No. of Training samples - After weighted sampling
2	6837	4689
3	6034	4675
4	2764	4668
5	4795	4805
6	3097	4690

INCEPTION RESNET V2

- **Inception-ResNet-v2** is a convolutional neural architecture that builds on the Inception family of architectures but incorporates residual connections (replacing the filter concatenation stage of the Inception architecture).
- The network is 164 layers deep and has an image input size of 299-by-299
- Factorization Techniques: Employs factorization methods like separable convolutions to reduce computational complexity while maintaining representation capacity.
- **Batch Normalization**: Applies batch normalization layers throughout the network to stabilize and accelerate training by normalizing activations.
- **Auxiliary Classifiers**: Includes auxiliary classifiers at intermediate layers during training to provide additional supervision and combat vanishing gradients.
- **Global Average Pooling**: Concludes with global average pooling followed by fully connected layers and softmax activation for classification tasks.



INCEPTION V3



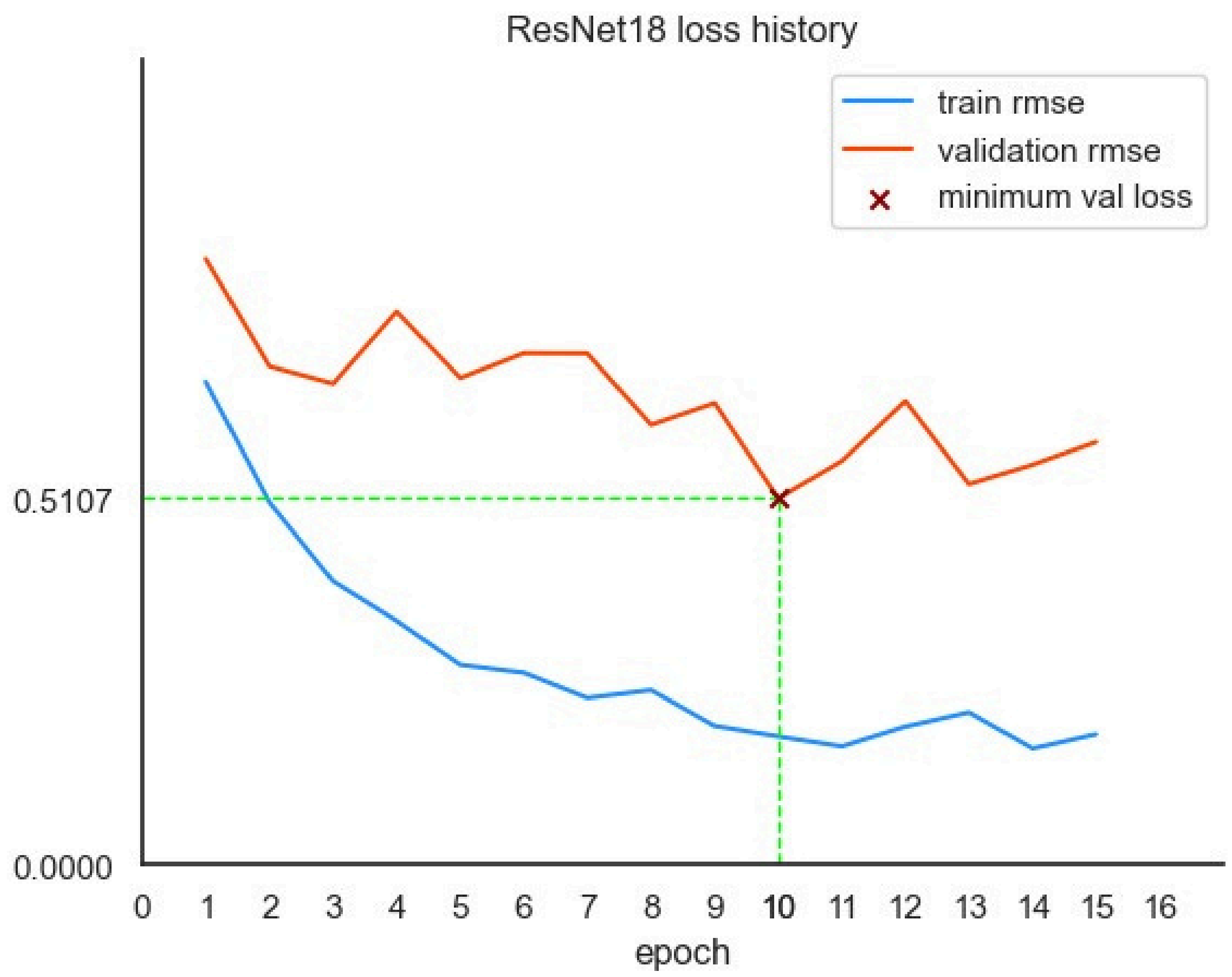
- The Inception V3 is a deep learning model based on CNNs, which is used for image classification, a superior version of Inception V1 which was introduced as GoogLeNet in 2014.
- It has a total of 42 layers and a lower error rate than its predecessors.

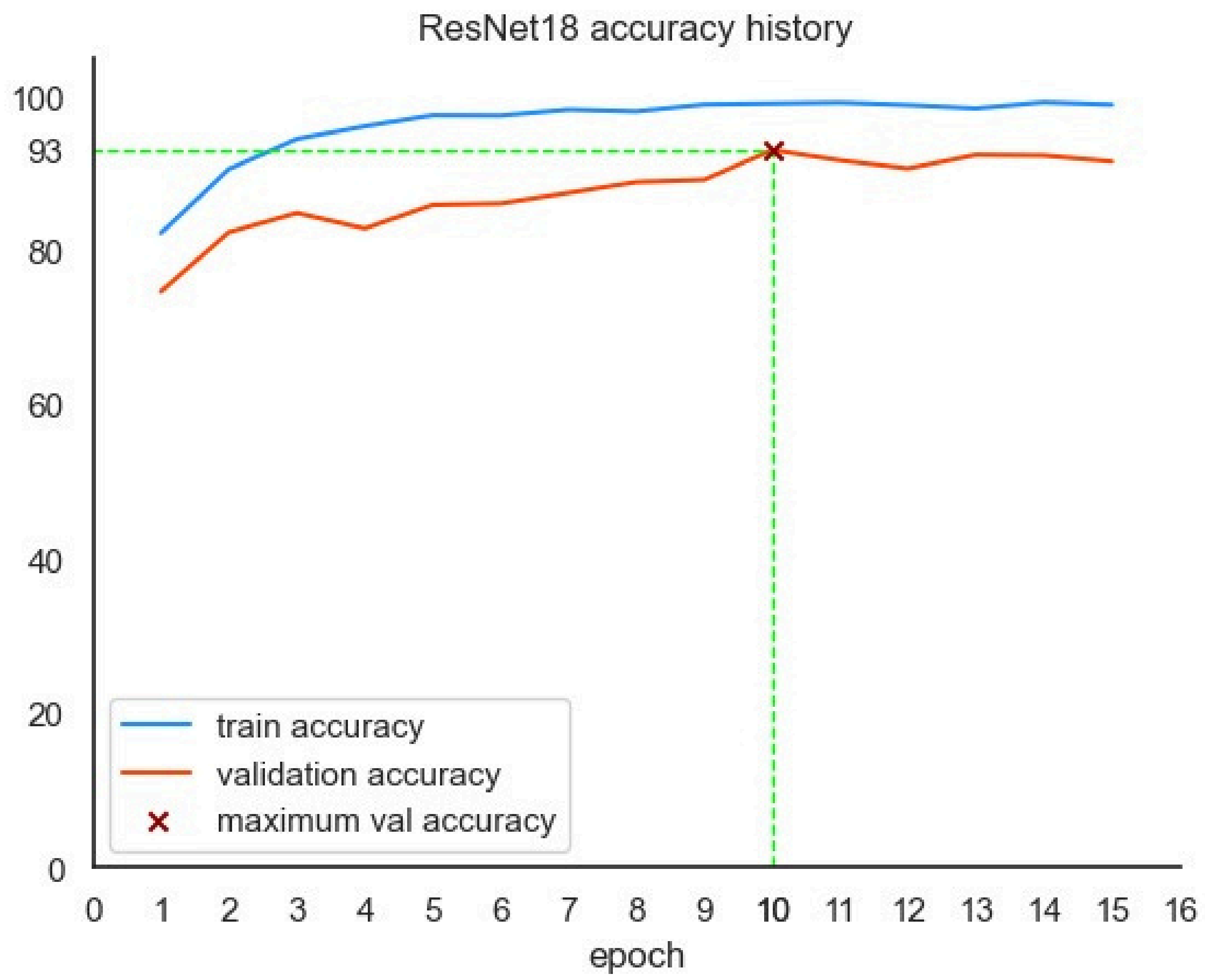
- The major modifications done on the Inception V3 model are: Factorization into Smaller Convolutions, Spatial Factorization into Asymmetric Convolutions, Utility of Auxiliary Classifiers, Efficient Grid Size Reduction

Why Inception V3 is better than V1 and V2?

- It has higher efficiency, a deeper network compared to the Inception V1 and V2 models, but its speed isn't compromised, It is computationally less expensive and uses auxiliary Classifiers as regularizes.

RESULTS:

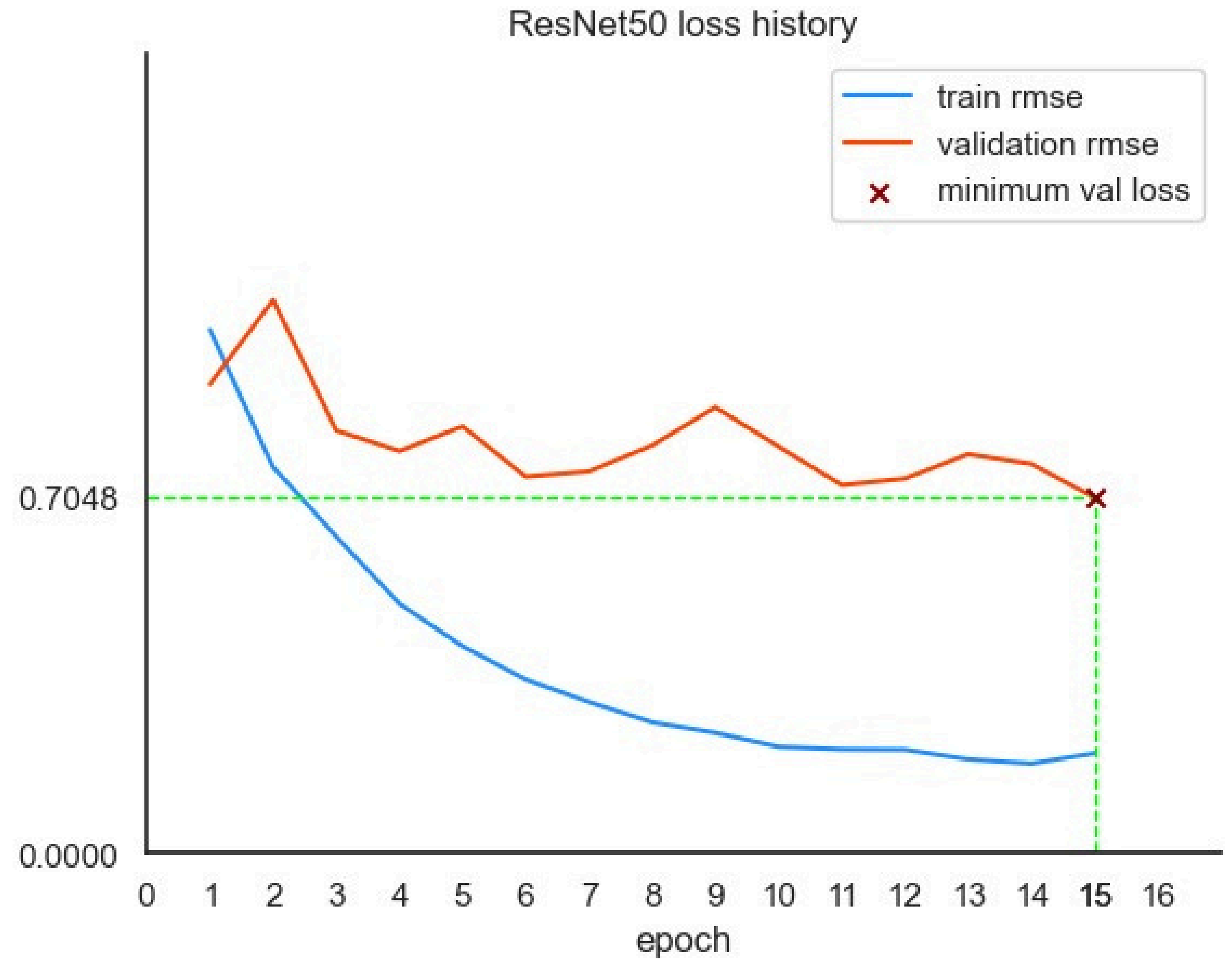


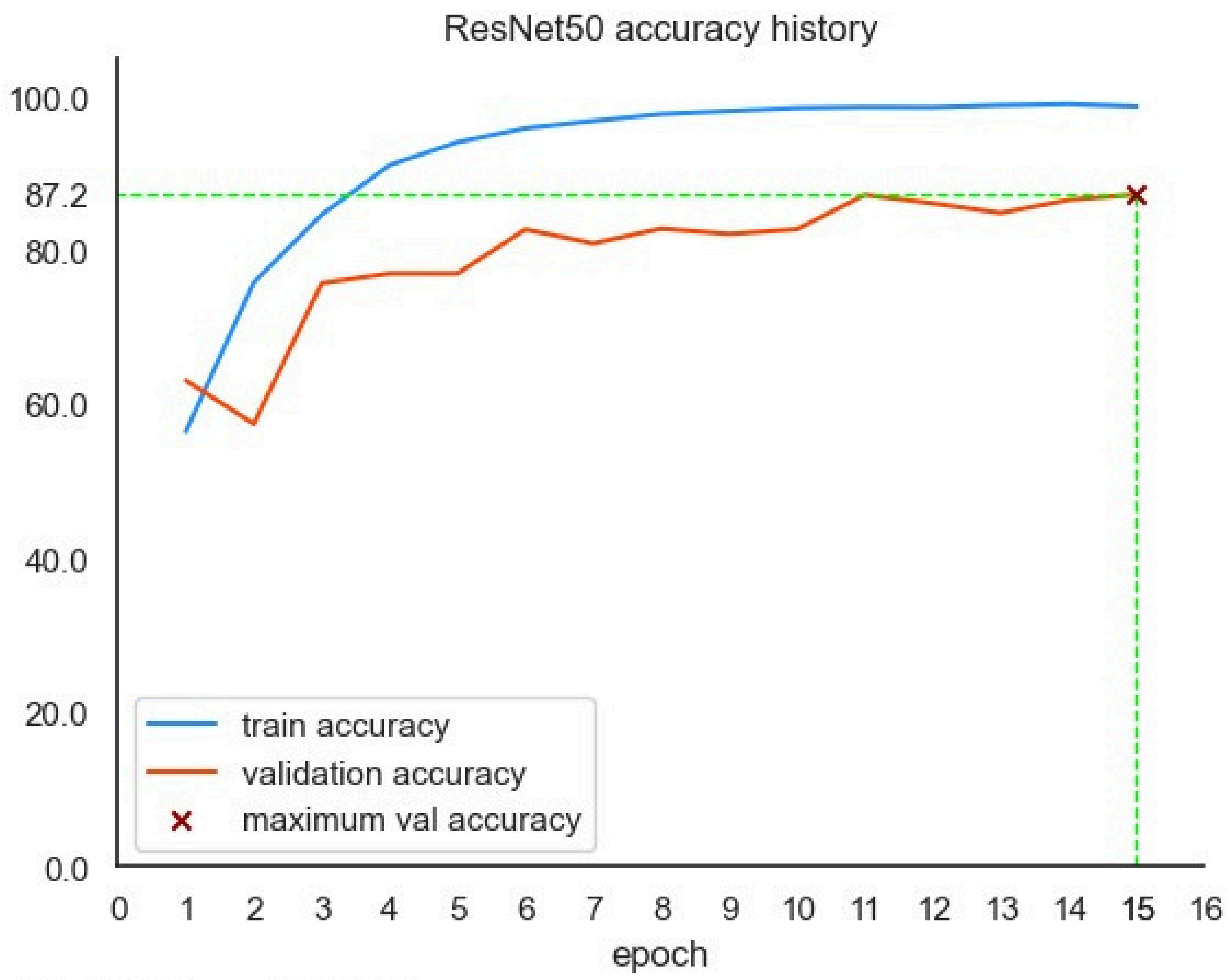


Test Classification Report :

	precision	recall	f1-score	support
2	0.94	0.94	0.94	1994
3	0.88	0.90	0.89	1664
4	0.88	0.84	0.86	773
5	0.96	0.94	0.95	1416
6	0.94	0.98	0.96	874
accuracy			0.92	6721
macro avg	0.92	0.92	0.92	6721
weighted avg	0.92	0.92	0.92	6721

RESULTS:

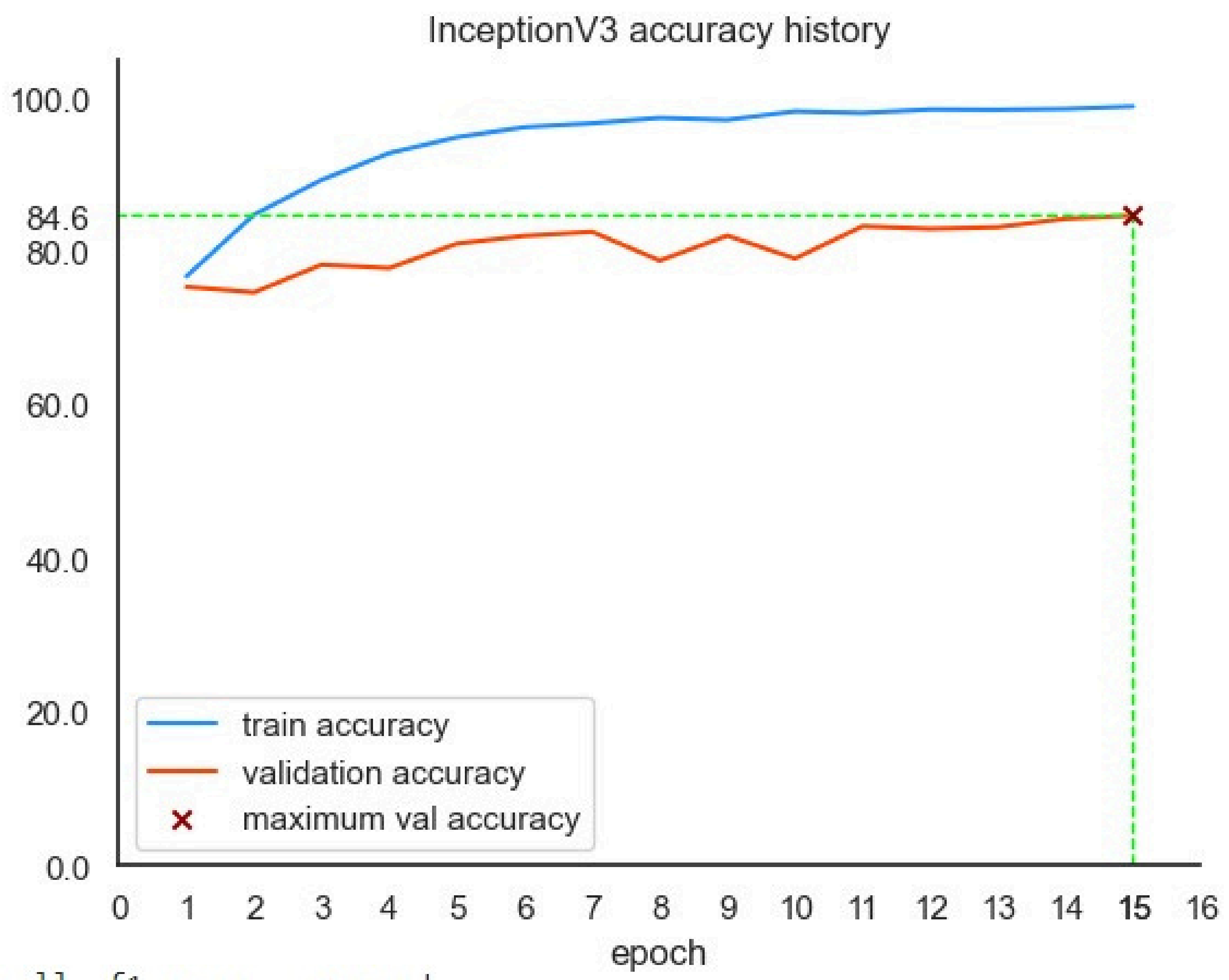




Test Classification Report :

	precision	recall	f1-score	support
2	0.89	0.93	0.91	1994
3	0.84	0.81	0.82	1664
4	0.78	0.77	0.78	773
5	0.91	0.93	0.92	1416
6	0.97	0.93	0.95	874
accuracy			0.88	6721
macro avg	0.88	0.87	0.88	6721
weighted avg	0.88	0.88	0.88	6721

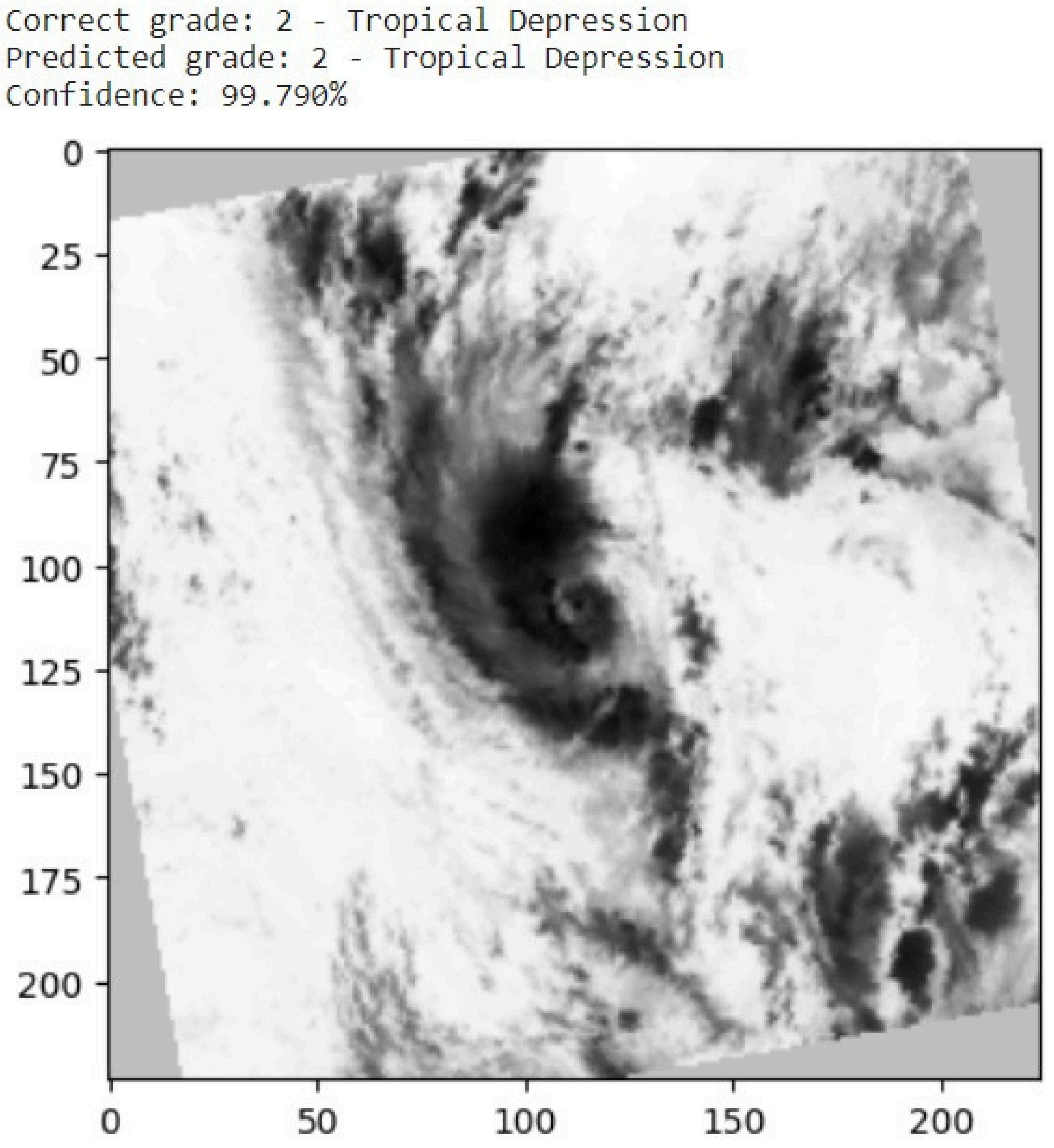
RESULTS:

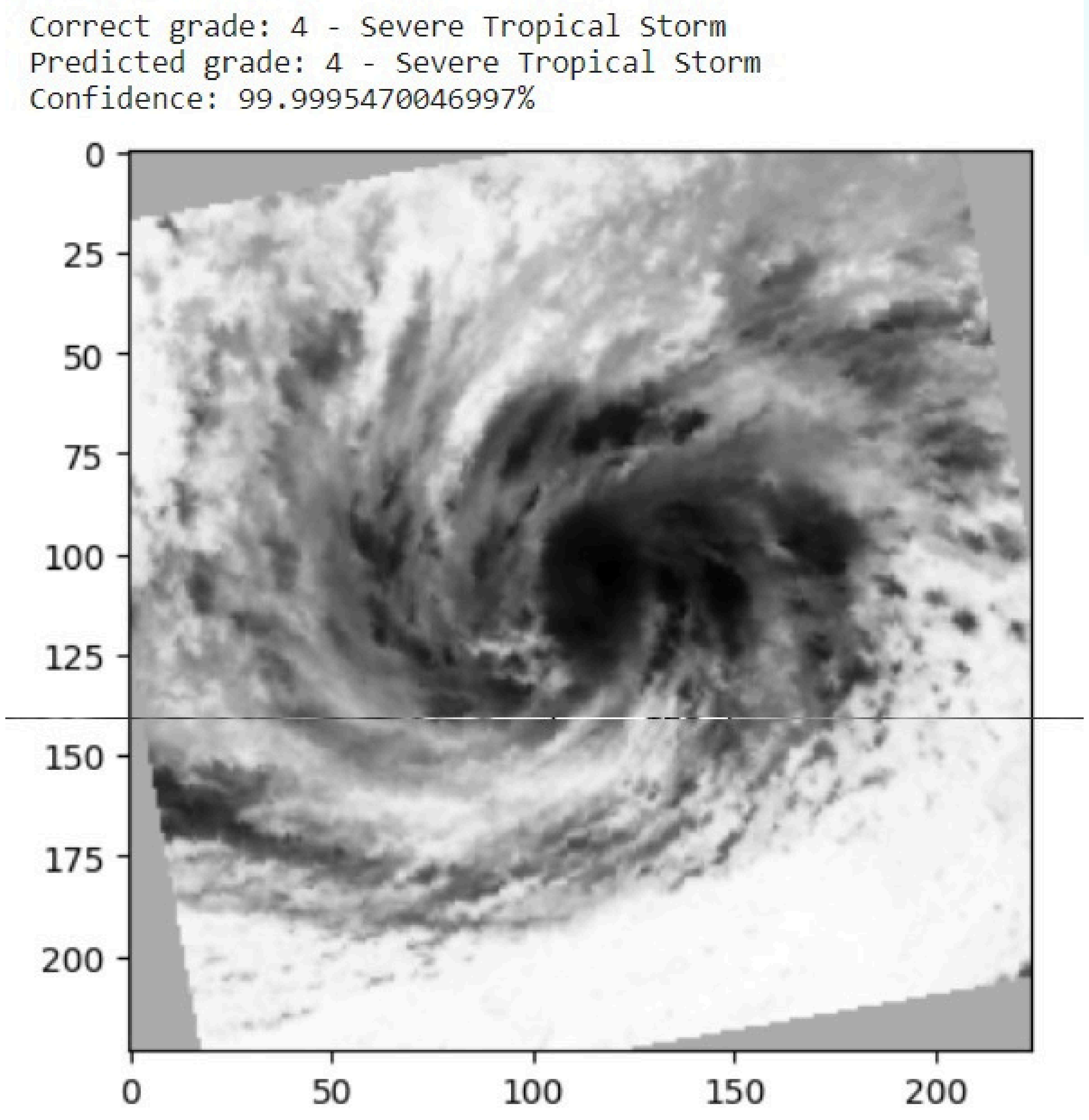


Test Classification Report :

	precision	recall	f1-score	support
2	0.89	0.83	0.86	1994
3	0.77	0.77	0.77	1664
4	0.68	0.79	0.73	773
5	0.92	0.90	0.91	1416
6	0.92	0.97	0.94	874
accuracy			0.84	6721
macro avg	0.84	0.85	0.84	6721
weighted avg	0.85	0.84	0.84	6721

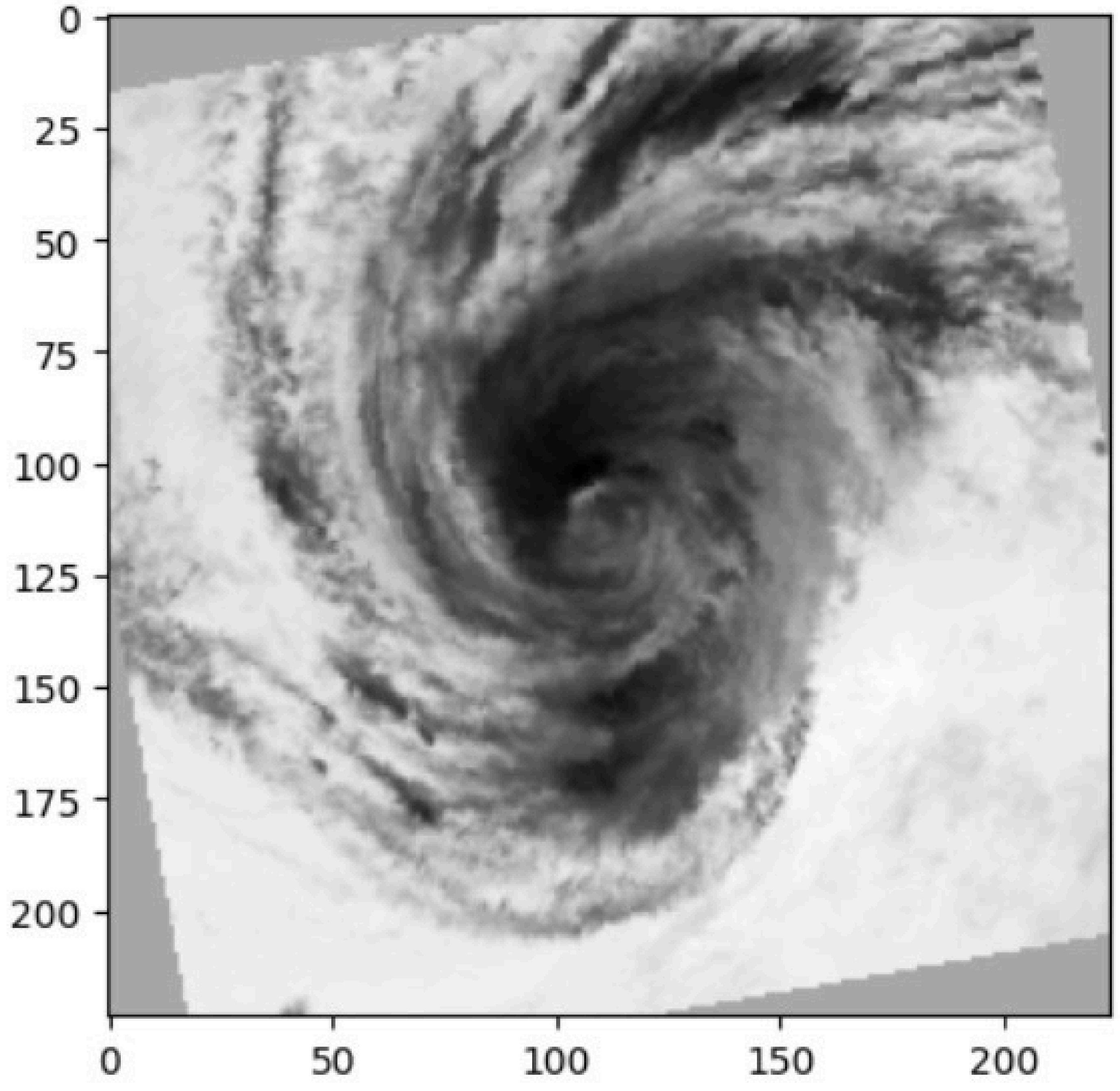
RESULTS : CLASSIFICATION USING RESNET 18



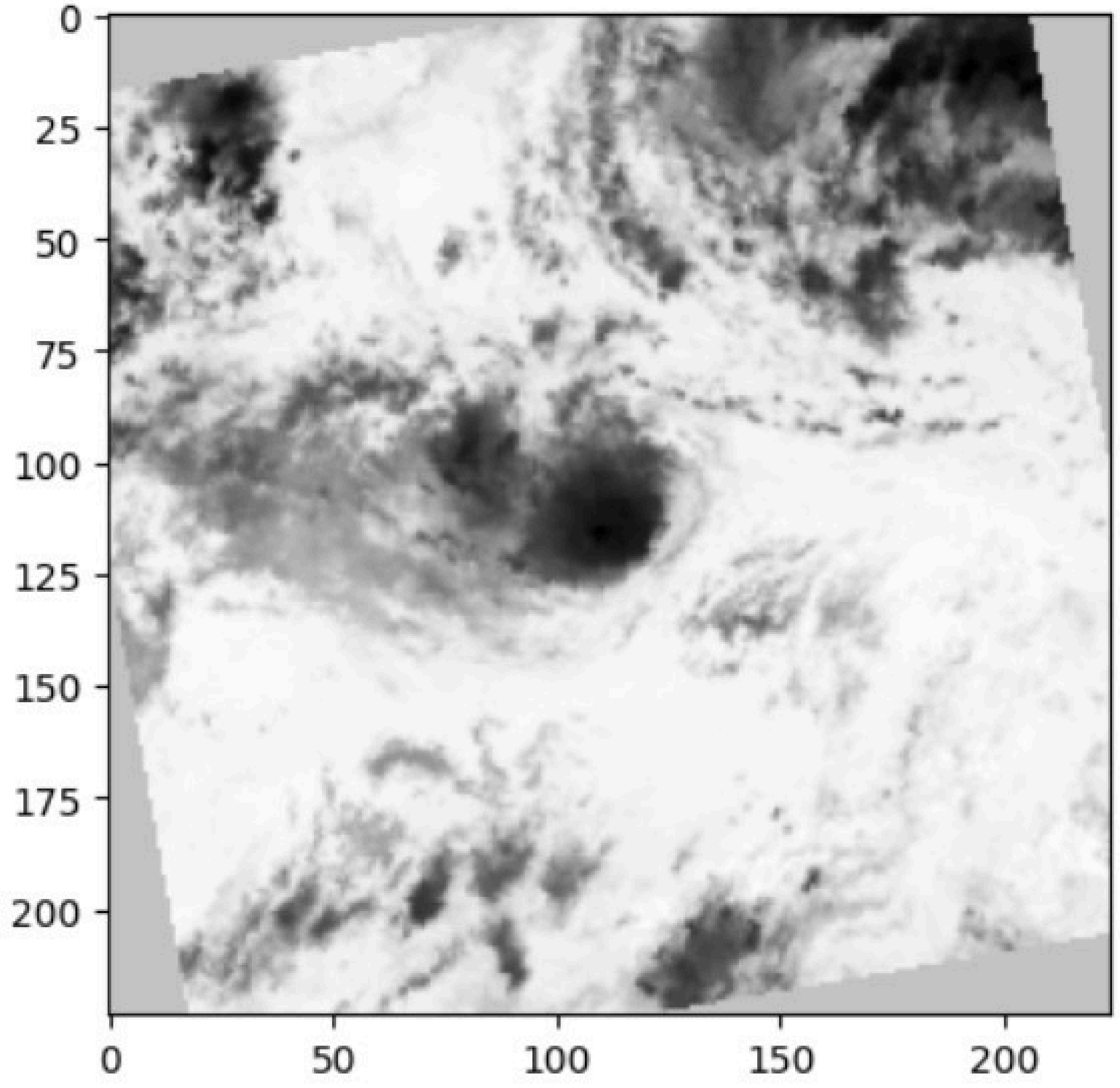


RESULTS : CLASSIFICATION USING RESNET 18

Correct grade: 5 - Typhoon
Predicted grade: 5 - Typhoon
Confidence: 100.0%



Correct grade: 3 - Tropical Storm
Predicted grade: 3 - Tropical Storm
Confidence: 99.69710111618042%



OVERALL MODELS SUMMARY:

Regression (Central Pressure intensity):

Model Name	Architecture	Batch Size	Input Image Size	Epochs	Augmentation Applied	Trainable Parameters	Total Parameters	Best Validation RMSE	Best Test RMSE
model1	**Resnet18**	64	224 x 224	50	no	3,713	11,170,817	38.9537	39.2766
model2	**Custom**	4	512 x 512	6	no	537,697	537,697	21.6105	22.0908
model3	**Custom**	128	256 x 256	6	yes	138,977	138,977	22.445	22.7868
model4	**Resnet34**	128	224 x 224	8	yes	13,118,081	21,278,977	6.9387	6.7541
model5	**Resnet50**	128	224 x 224	25	yes	14,969,985	23,503,873	6.3401	6.1761
model6	**Resnet18**	128	224 x 224	35	yes	8,397,441	11,170,817	5.2072	5.0847

Classification (Grades):

Model Name	Architecture	Batch Size	Input Image Size	Epochs	Augmentation Applied	Trainable Parameters	Total Parameters	Best Validation Accuracy	Best Test Accuracy
model1	**Resnet18**	128	224 x 224	15	yes	8,399,493	11,172,869	92.978	92.471
model2	**Resnet50**	128	224 x 224	15	yes	14,978,181	23,512,069	87.236	87.993
model3	**Inception V3**	128	299 x 299	15	yes	12,829,957	25,122,509	84.618	84.258

Comparison with digital typhoon dataset benchmark:

RMSE (hPa)	Full (512×512)	Resized (224×224)	Cropped (224×224)
ResNet18	10.51 ($\pm$0.11)	10.47($\pm$0.20)	**10.06** ($\pm$0.09)
ResNet50	11.12 ($\pm$0.41)	11.63 ($\pm$0.35)	10.09 ($\pm$0.04)

The result of the pressure regression task for two architectures and three types of input.

THANK YOU!