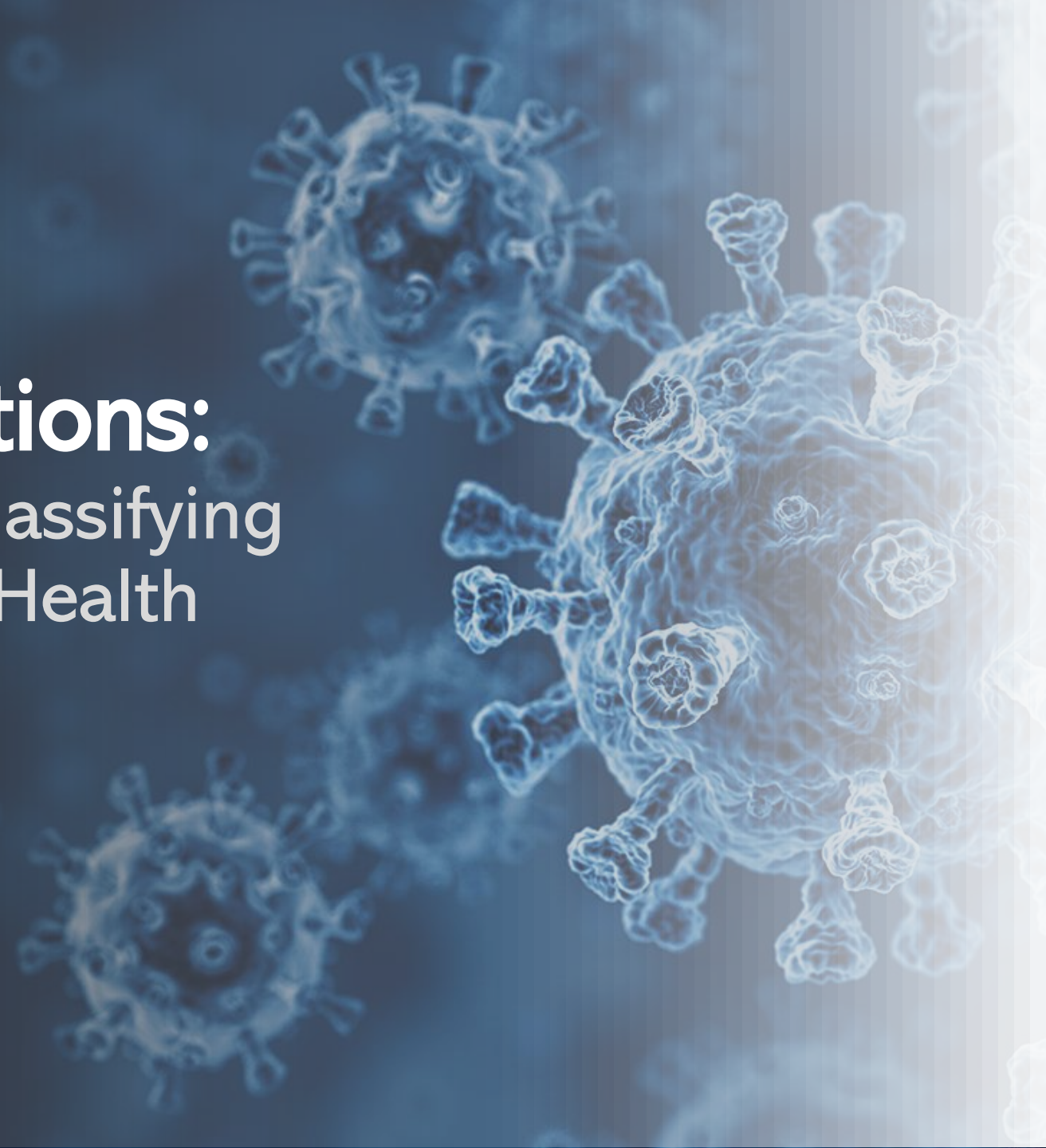


From Pixels to Predictions:

Exploring CNN Models for Classifying X-ray Images in Respiratory Health

John Pauline Pineda

December 25, 2023



OUTLINE

- Executive Summary
- Introduction
- Methodology
- Results Summary
 - Data Gathering and Description
 - Data Quality Assessment
 - Data Preprocessing
 - Data Exploration
 - Model Development, Validation, and Selection
- Detailed Findings
- Conclusion
- Appendix

EXECUTIVE SUMMARY

- **Study Objective**

- Conduct data analysis and modeling of x-ray images (categorized as Normal, Viral Pneumonia, and COVID-19) by developing multiple convolutional neural network classification models that could automatically learn hierarchical features directly from raw pixel data. Upon comparison and evaluation of candidate models, the final classification model will be selected based on robust performance estimates.

- **Methodology and Tools**

- Data Quality Assessment using Python OpenCV, Pandas, and NumPy APIs
- Data Preprocessing using OpenCV, Tensorflow, and Keras APIs
- Data Exploration using Python, Matplotlib, and Seaborn APIs
- Model Development and Validation using Python Scikit-Learn, Tensorflow, and Keras APIs

- **Overall Findings**

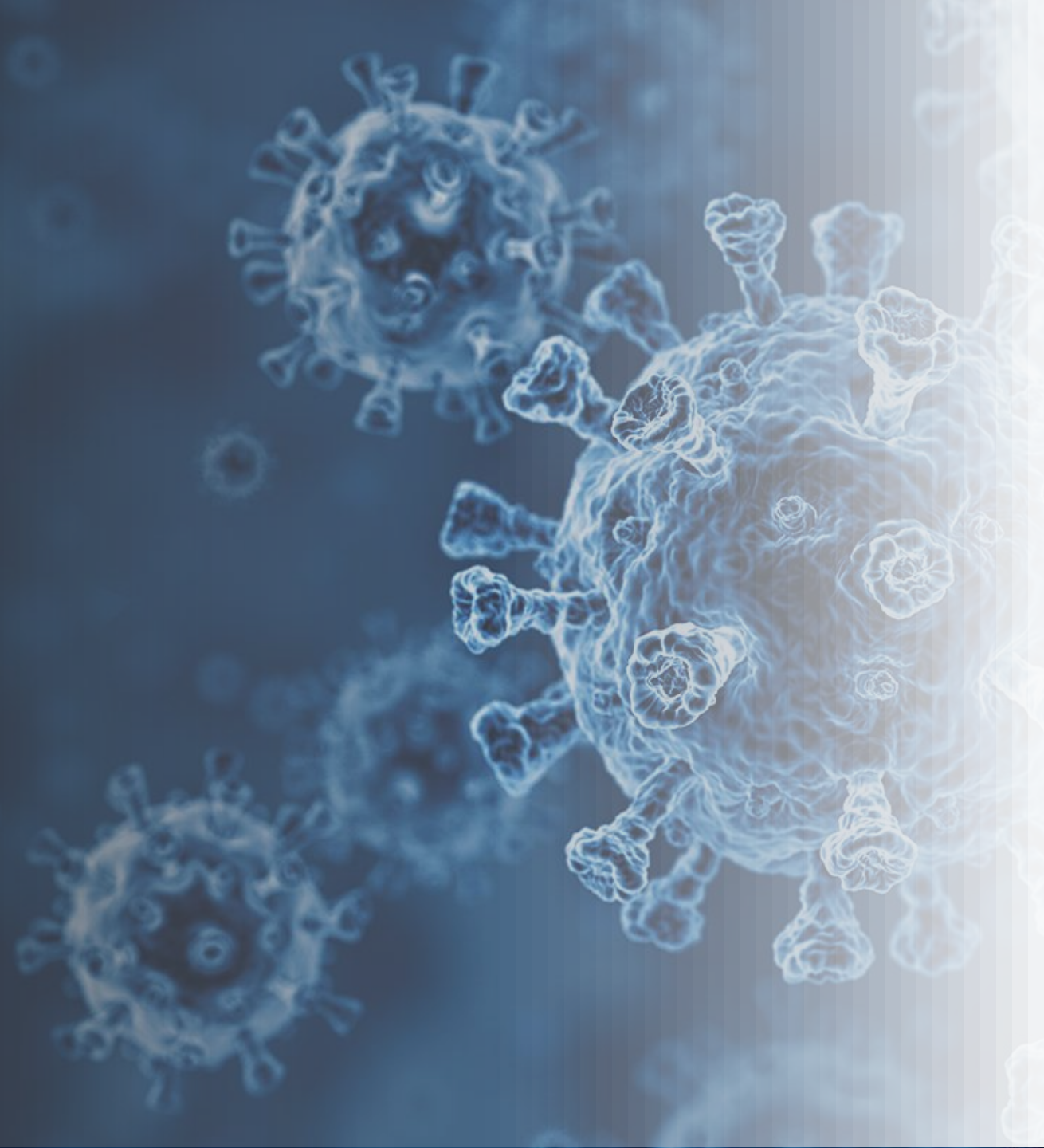
- Results were consolidated using the following GitHub URLs: [Python Notebook](#) | [Markdown Presentation](#)
- Differences in image pixel characteristics were observed between the x-ray image categories.
- X-ray image categories were modeled using different variations of the convolutional neural network model.
- Final prediction model was selected among candidates using robust classification performance.

INTRODUCTION

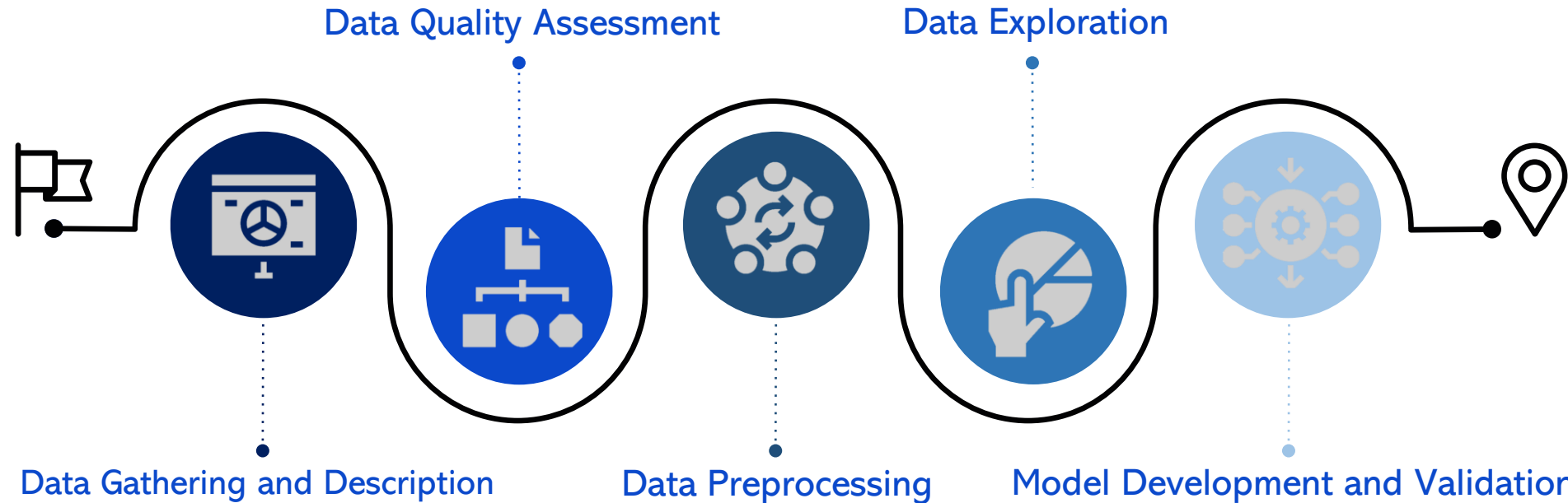
- In recent years, the integration of artificial intelligence (AI) into healthcare has emerged as a transformative force, revolutionizing the way we approach diagnostics and treatment. The urgency of the COVID-19 pandemic has underscored the **critical need for rapid and accurate diagnostic tools**. One such innovation that holds immense promise is the development of AI prediction models for classifying X-ray images in respiratory health.
- **This capstone project generally aims to develop a prediction model that could provide robust and reliable predictions of x-ray images differentiating normal, viral pneumonia, and COVID-19 respiratory health conditions.**
 - In particular, multiple convolutional neural network classification models will be formulated to automatically learn hierarchical features directly from raw pixel data. The final classification model will be selected among candidates based on robust performance estimates. The final model performance and generalization ability will be evaluated through internal validation.

Section 1

Methodology



METHODOLOGY



- Data download
- Image characteristics
- Image dimensions

- Duplicate images
- Null images

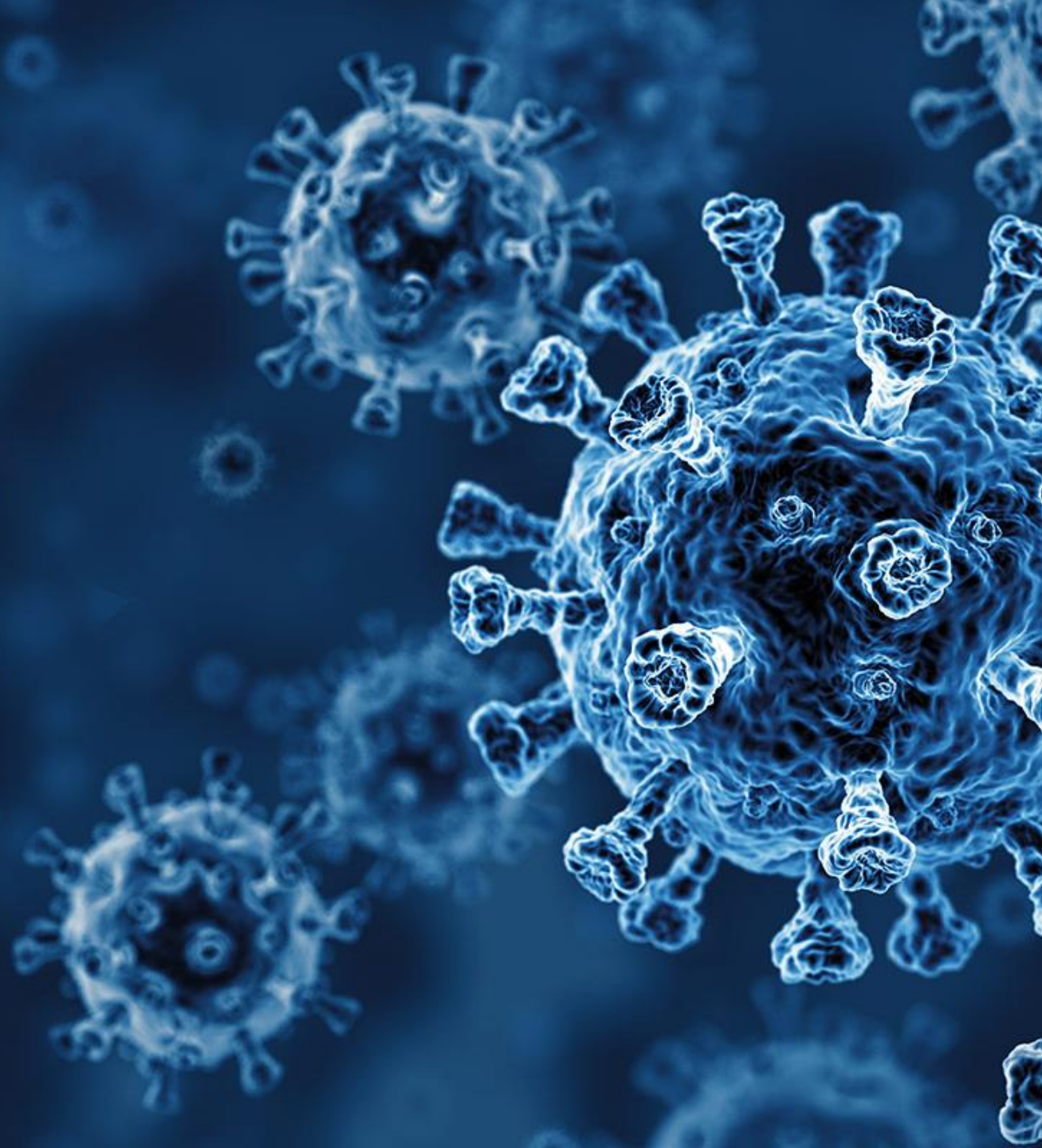
- Rescaling
- Rotation
- Height and width shift
- Horizontal flip
- Vertical flip
- Shear transformation
- Zooming

- Pixel statistics
- Pixel distribution

- Model development
- Model regularization
- Model validation
- Model selection

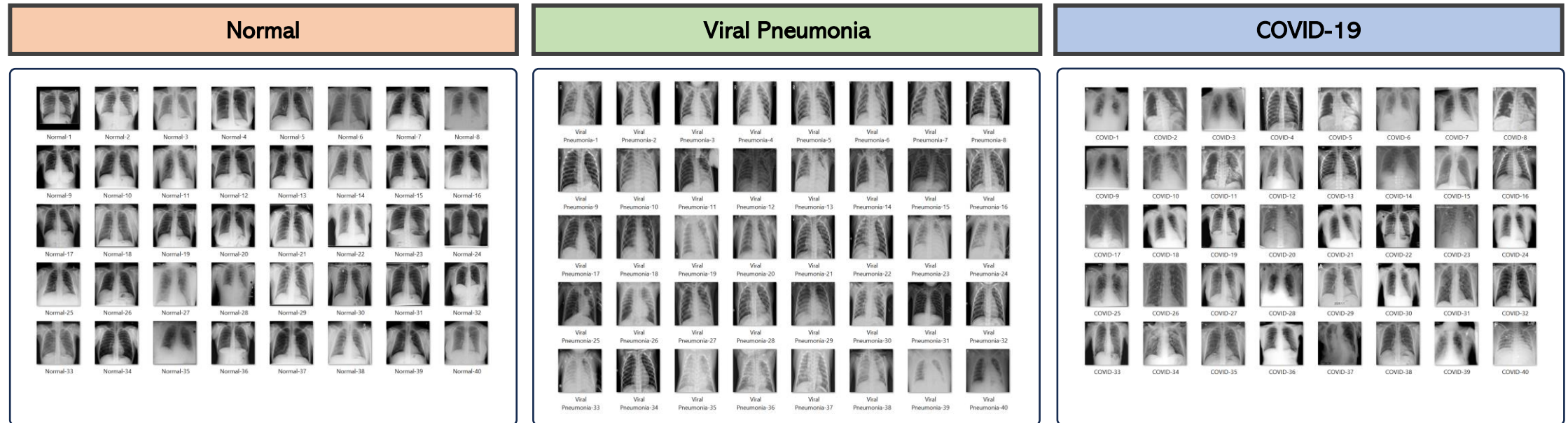
Section 2

Results Summary



RESULTS – DATA GATHERING

- A subset of an open [COVID-19 Radiography Dataset](#) from the [Kaggle](#) data repository (with all credits attributed to [Preet Viradiya](#), [Juliana Negrini De Araujo](#), [Tawsifur Rahman](#), [Muhammad Chowdhury](#), and [Amith Khandakar](#)) was used for the analysis as consolidated from different primary sources^A.

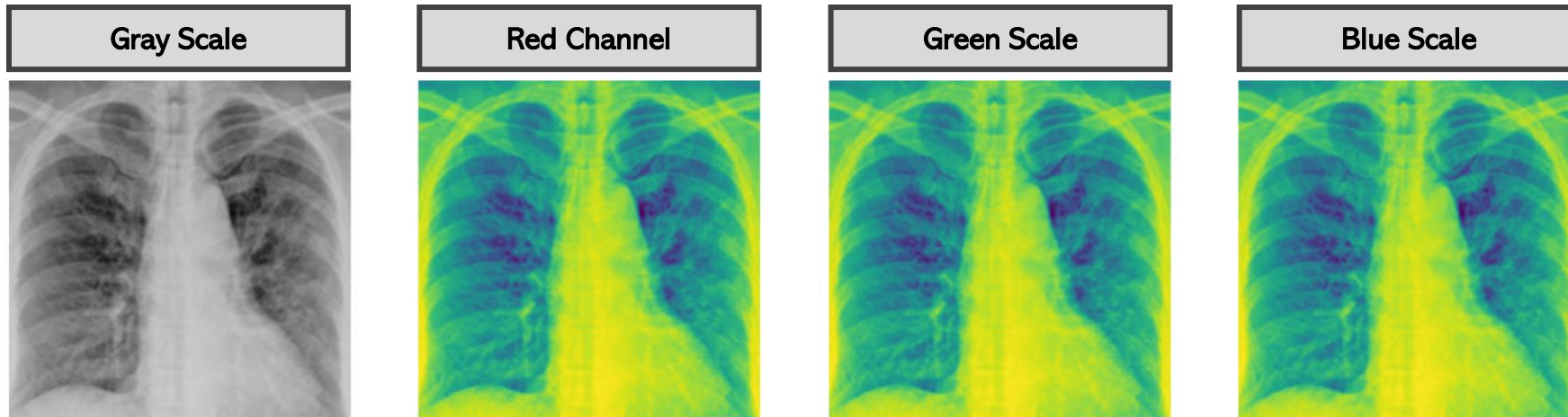


^ASources

- Covid19 X-Ray Images from [BIMCV Medical Imaging Databank of the Valencia Region](#)
- Covid19 X-Ray Images from [GitHub: ML Group](#)
- Covid19 X-Ray Images from [Italian Society of Medical and Interventional Radiology](#)
- Covid19 X-Ray Images from [European Society of Radiology](#)
- Covid19 X-Ray Images from [GitHub: Joseph Paul Cohen](#)
- Covid19 X-Ray Images from [Publication: COVID-CXNet: Detecting COVID-19 in Frontal Chest X-ray Images using Deep Learning](#)
- Pneumonia and Normal X-Ray Images from [Kaggle: RSNA Pneumonia Detection Challenge](#)
- Pneumonia and Normal X-Ray Images from [Kaggle: Chest X-Ray Images \(Pneumonia\)](#)

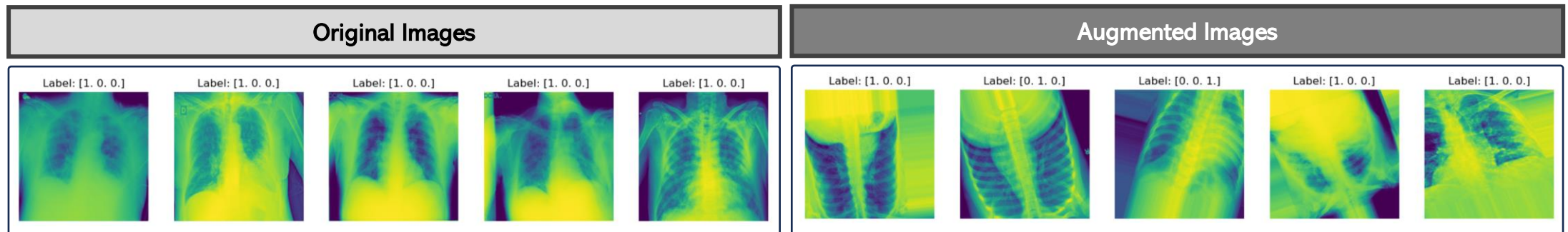
RESULTS – DATA DESCRIPTION

- Overall dataset contains 3600 images broken down equally into 3 classes:
 - 1200 images labeled as **Normal**
 - 1200 images labeled as **Viral Pneumonia**
 - 1200 images labeled as **COVID-19**
- There were no duplicated or null images noted.
- Each image contains 3 channels – red, green, and blue with pixel values ranging from 0 to 255.
- Each image is in gray scale indicating that the pixel values for each channel are exactly the same.



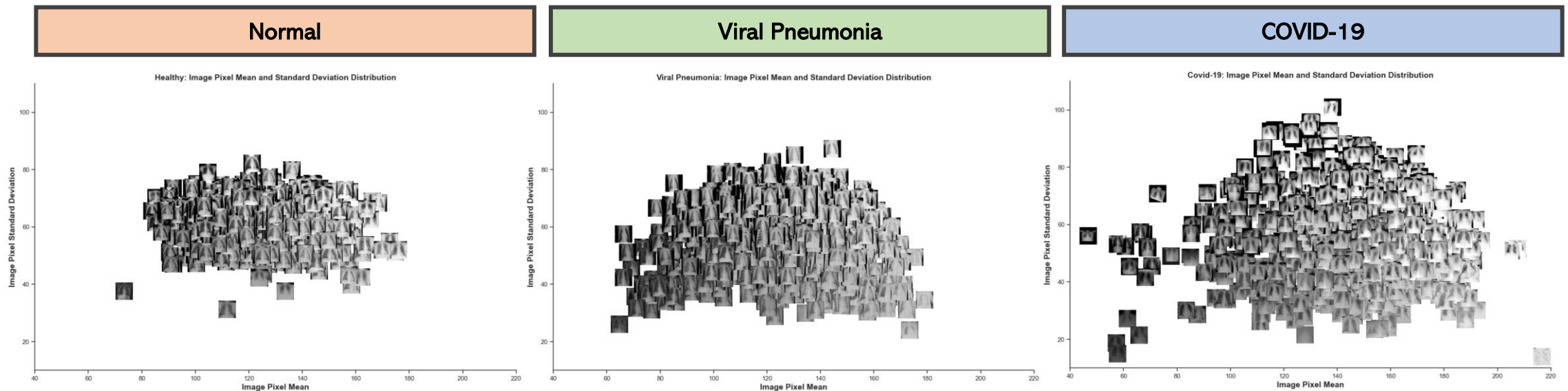
RESULTS – DATA PREPROCESSING

- Different image augmentation techniques were applied using various transformations to the training images to artificially increase the diversity of the dataset and improve the generalization and robustness of the model:
 - **Rescaling** - normalization of the pixel values within the 0 to 1 range
 - **Rotation** - random image rotation by 20 degrees
 - **Width Shift** - random horizontal shifting of the image by 20%
 - **Height Shift** - random vertical shifting of the image by 20%
 - **Horizontal Flip** - random horizontal flipping of the image
 - **Vertical Flip** - random vertical flipping of the image
 - **Shear Transformation** - image slanting by 20 degrees along the horizontal axis
 - **Zooming** - random image zoom-in or zoom-out by a factor of 20%
- Training data included 2880 augmented images representing 80% of the dataset.
- Validation data included 720 original images representing 20% of the dataset.



RESULTS – DATA EXPLORATION

- Distinct patterns were observed between the image categories.
- Images labeled as **Normal** were noted with lower mean pixel values (generally darker images) and a compact range of image pixel standard deviation values (stable and sufficient contrast).
- Images labeled as **Viral Pneumonia** were noted with higher mean pixel values (generally lighter images) and a wider range of image pixel standard deviation values (higher variation in contrast).
- Images labeled as **COVID-19** were noted with higher pixel values (collectively lighter images) and a wider range of image pixel standard deviation (extreme variation in contrast).



RESULTS – MODEL DEVELOPMENT

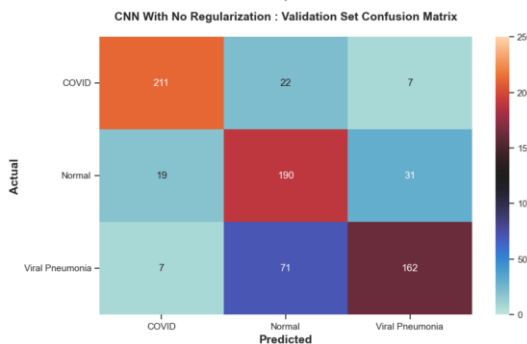
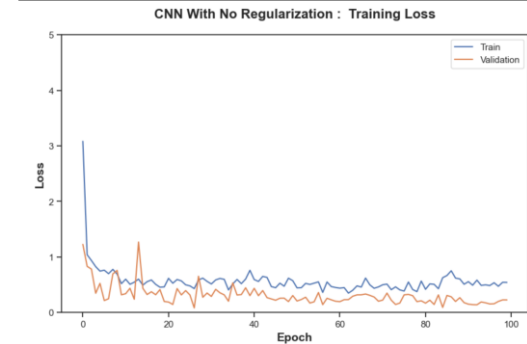
- 4 convolutional neural network (CNN) models with varying levels of regularization were formulated.

Model 1 CNN With No Regularization	Model 2 CNN With Dropout	Model 3 CNN With Batch Normalization	Model 4 CNN With Dropout + Batch Normalization
Conv2D Filters=32 Kernel Size=3x3 Activation=RELU Padding=Same	Conv2D Filters=32 Kernel Size=3x3 Activation=RELU Padding=Same	Conv2D Filters=32 Kernel Size=3x3 Activation=RELU Padding=Same	Conv2D Filters=32 Kernel Size=3x3 Activation=RELU Padding=Same
Max_Pooling2D Pool Size=2x2	Max_Pooling2D Pool Size=2x2	Max_Pooling2D Pool Size=2x2	Max_Pooling2D Pool Size=2x2
Conv2D Filters=32 Kernel Size=3x3 Activation=RELU Padding=Same	Conv2D Filters=32 Kernel Size=3x3 Activation=RELU Padding=Same	Conv2D Filters=32 Kernel Size=3x3 Activation=RELU Padding=Same	Conv2D Filters=32 Kernel Size=3x3 Activation=RELU Padding=Same
Max_Pooling2D Pool Size=2x2	Dropout Rate=0.25	Batch Normalization	Batch Normalization
Flatten	Max_Pooling2D Pool Size=2x2	Activation Activation=RELU	Activation Activation=RELU
Dense Units=128 Activation=RELU	Flatten	Max_Pooling2D Pool Size=2x2	Dropout Rate=0.25
Dense Units=3 Activation=SOFTMAX	Dense Units=128 Activation=RELU	Flatten	Max_Pooling2D Pool Size=2x2
	Dense Units=3 Activation=SOFTMAX	Dense Units=128 Activation=RELU	Flatten
		Dense Units=3 Activation=SOFTMAX	Dense Units=128 Activation=RELU
			Dense Units=3 Activation=SOFTMAX

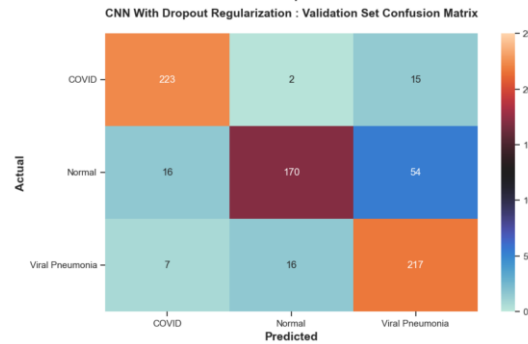
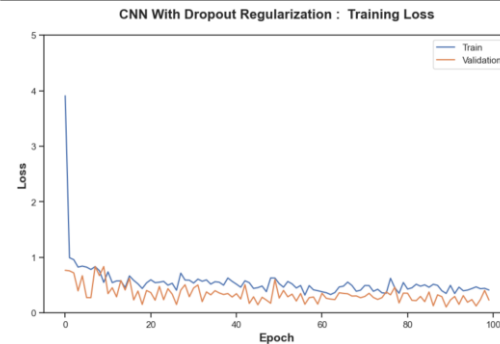
RESULTS – MODEL VALIDATION

- Model 1 demonstrated 79.06% Precision, 78.19% Recall, and 78.25% F-Score on the validation set.
- Model 2 demonstrated 85.65% Precision, 84.72% Recall, and 84.57% F-Score on the validation set.
- Model 3 demonstrated 91.11% Precision, 90.97% Recall, and 90.97% F-Score on the validation set.
- Model 4 demonstrated 88.42% Precision, 87.36% Recall, and 87.44% F-Score on the validation set.

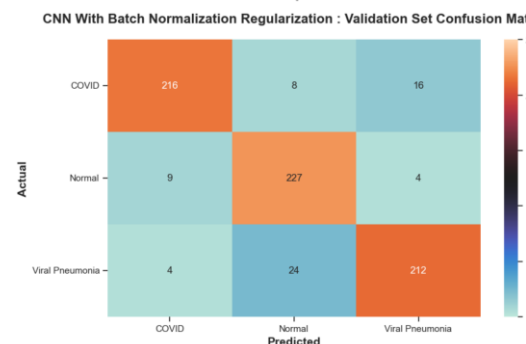
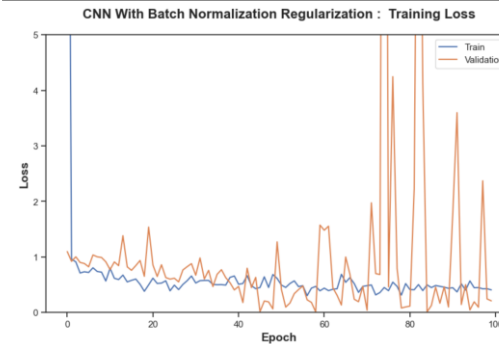
Model 1: CNN_NR
CNN With No Regularization



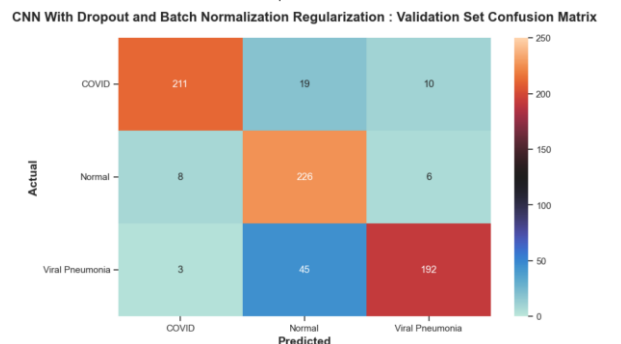
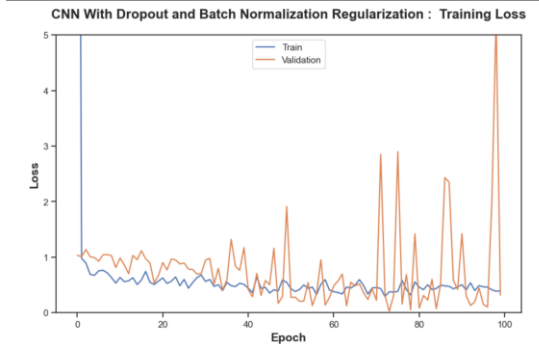
Model 2: CNN_DR
CNN With Dropout



Model 3
CNN With Batch Normalization



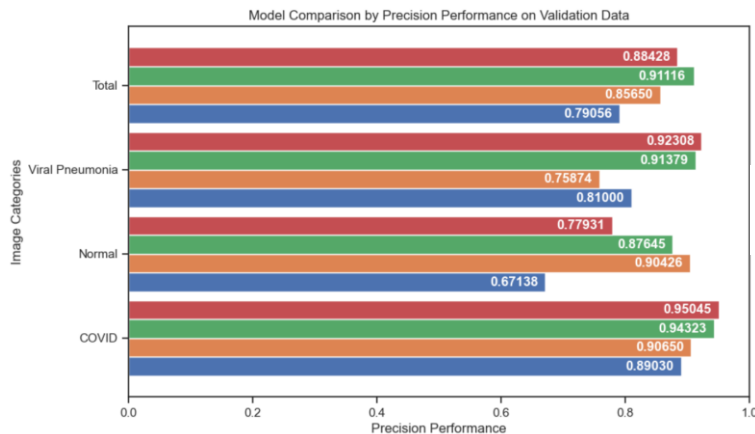
Model 4
CNN With Dropout + Batch Normalization



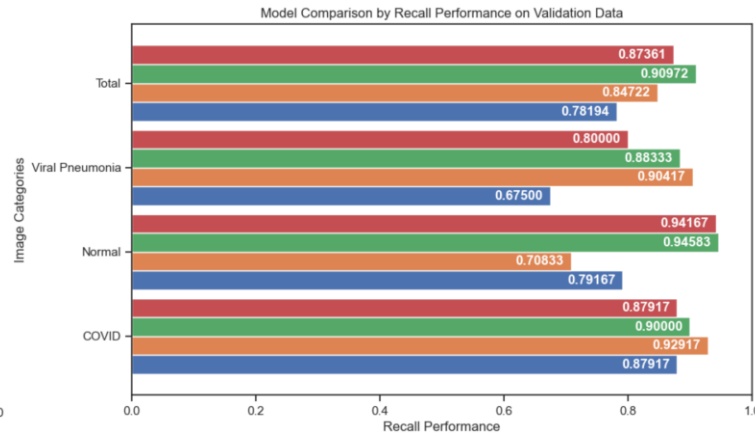
RESULTS – MODEL SELECTION

- Model 3 – the CNN model with Batch Normalization had the best validation performance and was selected as the final model among all candidate models.
 - Overall - 91.11% Precision, 90.97% Recall, 90.97% F-Score
 - COVID-19 - 94.32% Precision, 90.00% Recall, 92.11% F-Score
 - Viral Pneumonia - 91.38% Precision, 86.33% Recall, 89.83% F-Score
 - Normal - 87.65% Precision, 94.58% Recall, 90.98% F-Score

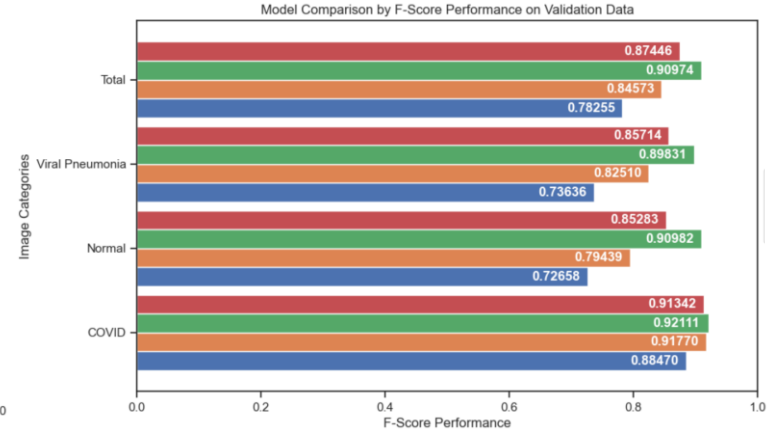
Precision by Image Category



Recall by Image Category



F-Score by Image Category



Model 1: CNN_NR
CNN With No Regularization

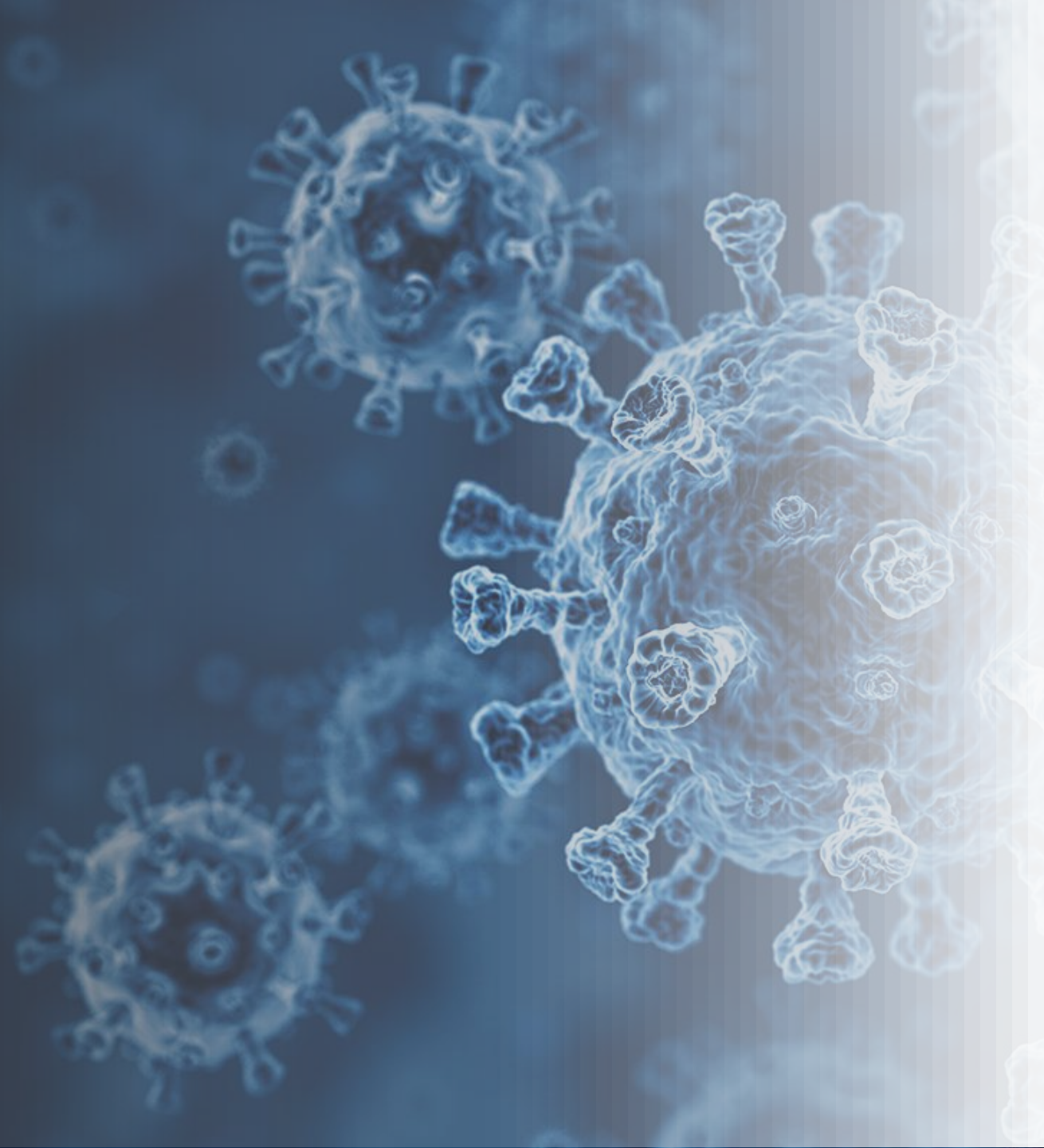
Model 2: CNN_DR
CNN With Dropout

Model 3
CNN With Batch Normalization

Model 4
CNN With Dropout + Batch Normalization

Section 4

Summary



CONCLUSION

- **Key Findings**

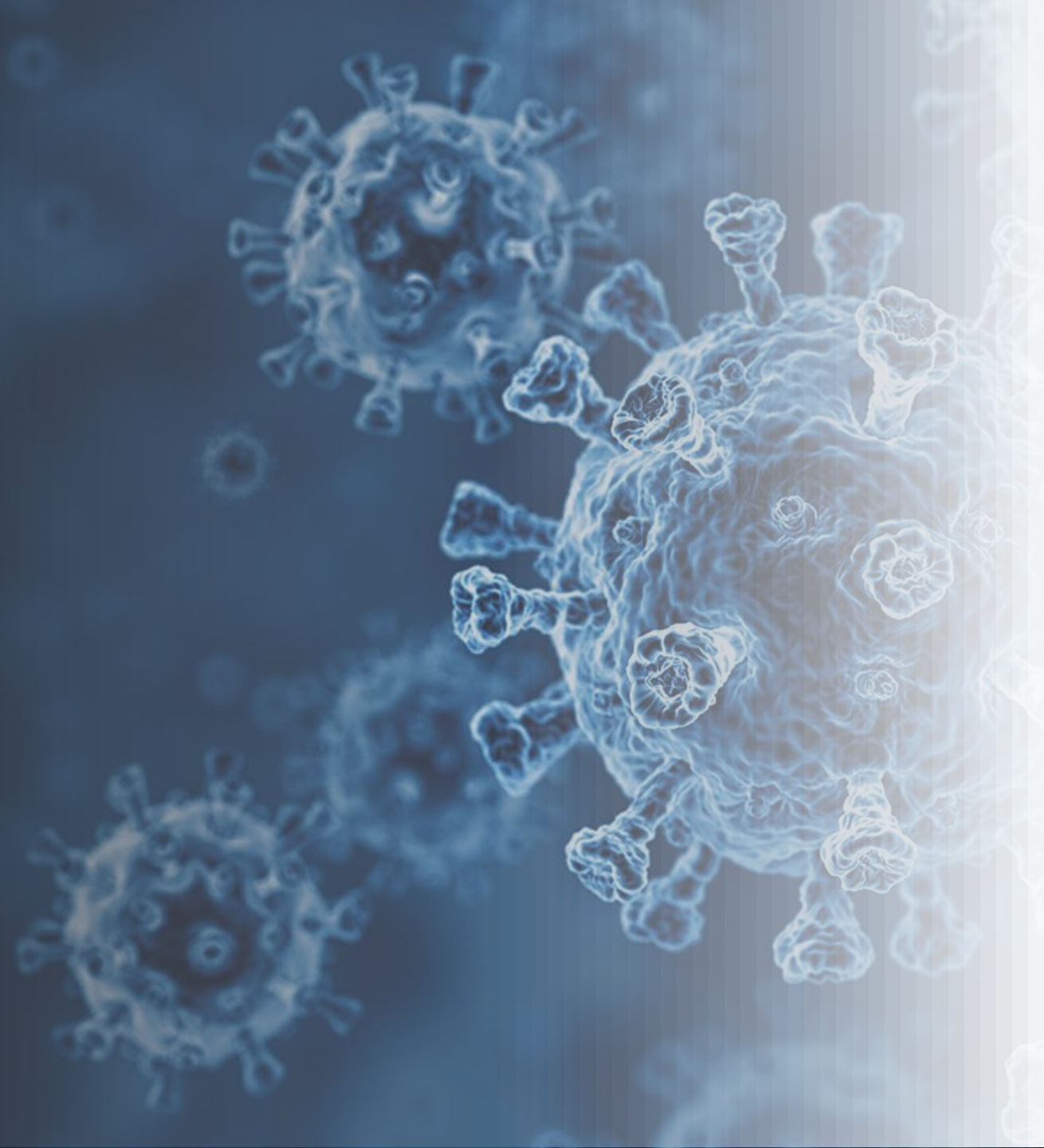
- The final classification model using the **CNN model** which applied **batch normalization** as a regularization method provided robust and reliable image category predictions based on the estimated validation set metrics as follows.
 - Precision: 91.11%
 - Recall: 90.97%
 - F-Score: 90.97%

- **Next Steps**

- While the classification results have been sufficiently high, the current study can be further extended to achieve optimal model performance through the following:
 - Conduct model hyperparameter tuning given sufficient analysis time and higher computing power
 - Formulate deeper neural network architectures to better capture spatial hierarchies and features in the input images
 - Apply various techniques to interpret the CNN models by understanding and visualizing the features and decisions made at each layer
 - Consider an imbalanced dataset and apply remedial measures to address unbalanced classification to accurately reflect a real-world scenario

Section 5

Appendix



APPENDIX

- Source Data

- Covid19 X-Ray Images: [BIMCV Medical Imaging Databank of the Valencia Region](#)
- Covid19 X-Ray Images: [GitHub: ML Group](#)
- Covid19 X-Ray Images: [Italian Society of Medical and Interventional Radiology](#)
- Covid19 X-Ray Images: [European Society of Radiology](#)
- Covid19 X-Ray Images: [GitHub: Joseph Paul Cohen](#)
- Covid19 X-Ray Images: [Publication: COVID-CXNet: Detecting COVID-19 in Frontal Chest X-ray Images using Deep Learning](#)
- Pneumonia and Normal X-Ray Images: [Kaggle: RSNA Pneumonia Detection Challenge](#)
- Pneumonia and Normal X-Ray Images: [Kaggle: Chest X-Ray Images \(Pneumonia\)](#)

APPENDIX

- Python Notebooks | Codes

- GitHub URL: [Data Background](#)
- GitHub URL: [Data Description](#)
- GitHub URL: [Data Quality Assessment](#)
- GitHub URL: [Data Preprocessing](#)
 - GitHub URL: [Image Description](#)
 - GitHub URL: [Image Augmentation](#)
- GitHub URL: [Data Exploration](#)
 - GitHub URL: [Exploratory Data Analysis](#)
- GitHub URL: [Model Development](#)
 - GitHub URL: [Premodelling Data Description](#)
 - GitHub URL: [CNN With No Regularization](#)
 - GitHub URL: [CNN With Dropout Regularization](#)
 - GitHub URL: [CNN With Batch Normalization Regularization](#)
 - GitHub URL: [CNN With Dropout and Batch Normalization](#)
- GitHub URL: [Consolidated Findings](#)

APPENDIX

- **References**

- [Book] [Deep Learning with Python](#) by Francois Chollet
- [Book] [Deep Learning: A Visual Approach](#) by Andrew Glassner
- [Book] [Learning Deep Learning](#) by Magnus Ekman
- [Book] [Practical Deep Learning](#) by Ronald Kneusel
- [Book] [Deep Learning with Tensorflow and Keras](#) by Amita Kapoor, Antonio Gulli and Sujit Pal
- [Book] [Deep Learning](#) by John Kelleher
- [Book] [Generative Deep Learning](#) by David Foster
- [Book] [Deep Learning Illustrated](#) by John Krohn, Grant Beyleveld and Aglae Bassens
- [Book] [Neural Networks and Deep Learning](#) by Charu Aggarwal
- [Book] [Grokking Deep Learning](#) by Andrew Trask
- [Book] [Deep Learning with Pytorch](#) by Eli Stevens, Luca Antiga and Thomas Viehmann
- [Book] [Deep Learning](#) by Ian Goodfellow, Yoshua Bengio and Aaron Courville
- [Book] [Deep Learning from Scratch](#) by Seth Weidman
- [Book] [Fundamentals of Deep Learning](#) by Nithin Buduma, Nikhil Buduma and Joe Papa
- [Book] [Hands-On Machine Learning with Scikit-Learn, Keras and Tensorflow](#) by Aurelien Geron
- [Book] [Deep Learning for Computer Vision](#) by Jason Brownlee

APPENDIX

• References

- [Python Library API] [numpy](#) by NumPy Team
- [Python Library API] [pandas](#) by Pandas Team
- [Python Library API] [seaborn](#) by Seaborn Team
- [Python Library API] [matplotlib.pyplot](#) by Matplotlib Team
- [Python Library API] [matplotlib.image](#) by Matplotlib Team
- [Python Library API] [matplotlib.offsetbox](#) by Matplotlib Team
- [Python Library API] [tensorflow](#) by TensorFlow Team
- [Python Library API] [keras](#) by Keras Team
- [Python Library API] [pil](#) by Pillow Team
- [Python Library API] [glob](#) by glob Team
- [Python Library API] [cv2](#) by OpenCV Team
- [Python Library API] [os](#) by os Team
- [Python Library API] [random](#) by random Team
- [Python Library API] [keras.models](#) by TensorFlow Team
- [Python Library API] [keras.layers](#) by TensorFlow Team
- [Python Library API] [keras.wrappers](#) by TensorFlow Team
- [Python Library API] [keras.utils](#) by TensorFlow Team
- [Python Library API] [keras.optimizers](#) by TensorFlow Team
- [Python Library API] [keras.preprocessing.image](#) by TensorFlow Team
- [Python Library API] [keras.callbacks](#) by TensorFlow Team
- [Python Library API] [keras.metrics](#) by TensorFlow Team
- [Python Library API] [sklearn.metrics](#) by Scikit-Learn Team

APPENDIX

• References

- [Article] [Convolutional Neural Networks, Explained](#) by Mayank Mishra (Towards Data Science)
- [Article] [A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way](#) by Sumit Saha (Towards Data Science)
- [Article] [Understanding Convolutional Neural Networks: A Beginner's Journey into the Architecture](#) by Afaq Umer (Medium)
- [Article] [Introduction to Convolutional Neural Networks \(CNN\)](#) by Manav Mandal (Analytics Vidhya)
- [Article] [What Are Convolutional Neural Networks?](#) by IBM Team (IBM)
- [Article] [What is CNN? A 5 Year Old guide to Convolutional Neural Network](#) by William Ong (Medium)
- [Article] [Convolutional Neural Network](#) by Thomas Wood (DeepAI.Org)
- [Article] [How Do Convolutional Layers Work in Deep Learning Neural Networks?](#) by Jason Brownlee (Machine Learning Mastery)
- [Article] [Convolutional Neural Networks Explained: Using PyTorch to Understand CNNs](#) by Vihar Kurama (BuiltIn)
- [Article] [Convolutional Neural Networks Cheatsheet](#) by Afshine Amidi and Shervine Amidi (Stanford University)
- [Article] [An Intuitive Explanation of Convolutional Neural Networks](#) by Ujjwal Karn (The Data Science Blog)
- [Article] [Convolutional Neural Network](#) by NVIDIA Team (NVIDIA)
- [Article] [Convolutional Neural Networks \(CNN\) Overview](#) by Nikolaj Buhl (Encord)
- [Article] [Understanding Convolutional Neural Network \(CNN\): A Complete Guide](#) by LearnOpenCV Team (LearnOpenCV)
- [Article] [Convolutional Neural Networks \(CNNs\) and Layer Types](#) by Adrian Rosebrock (PyImageSearch)
- [Article] [How Convolutional Neural Networks See The World](#) by Francois Chollet (The Keras Blog)
- [Article] [What Is a Convolutional Neural Network?](#) by MathWorks Team (MathWorks)

APPENDIX

• References

- [Kaggle Project] [Covid 19 Radiography Data - EDA and CNN Model](#) by Juliana Negrini De Araujo (Kaggle)
- [Kaggle Project] [Pneumonia Detection using CNN \(92.6% Accuracy\)](#) by Madhav Mathur (Kaggle)
- [Kaggle Project] [COVID Detection from CXR Using Explainable CNN](#) by Manu Siddhartha (Kaggle)
- [Kaggle Project] [Class Activation Mapping for COVID-19 CNN](#) by Amy Zhang (Kaggle)
- [Kaggle Project] [CNN XRay COVID Classification](#) by Gabriel Mino (Kaggle)
- [Kaggle Project] [Detecting-Covid-19-Images | CNN](#) by Felipe Oliveira (Kaggle)
- [Kaggle Project] [Detection of Covid Positive Cases using DL](#) by Sana Shaikh (Kaggle)
- [Kaggle Project] [Deep Learning and Transfer Learning on COVID-19](#) by Digvijay Yadav (Kaggle)
- [Kaggle Project] [X-ray Detecting Using CNN](#) by Shivan Kumar (Kaggle)
- [Kaggle Project] [Classification of COVID-19 using CNN](#) by Islam Selim (Kaggle)
- [Kaggle Project] [COVID-19 - Revisiting Pneumonia Detection](#) by Paulo Breviglieri (Kaggle)
- [Kaggle Project] [Multi-Class X-ray Covid19 Classification-94% Accuracy](#) by Derrel Souza (Kaggle)
- [Publication] [Gradient-Based Learning Applied to Document Recognition](#) by Yann LeCun, Leon Bottou, Yoshua Bengio and Patrick Haffner (Proceedings of the IEEE)

Thank You!

