



FSM Mid-Internship Review

INTP23-ML-9:

Piston Defect Detection using Computer Vision

Ananya Mohapatra

National Institute of Science and Technology, Berhampur

Under Mentorship of

Keivalya Pandya

IITD-AIA FOUNDATION FOR SMART MANUFACTURING



Objectives

Objective 1: Performing Explanatory Data Analysis

Achieved: Conducted a thorough exploratory analysis of the collected piston image dataset. Examined the distribution of defect types, analyzed image quality, and identified any data inconsistencies.

Planned Deliverables: Perform statistical analysis on the dataset, such as mean, standard deviation, and image resolution. Visualize the data distribution and correlations between features.

Target completion: 100%

Objective 2: Data Collection and Preparation

Achieved: Collected a dataset of AC piston images with a diverse range of defect types. Preprocessed the images to ensure consistency and Complete data augmentation techniques. Completed data augmentation techniques to increase the dataset size and diversity.

Planned Deliverables: Complete data augmentation techniques to increase the dataset size and diversity.

Target completion: 100%



Objectives

Objective 3: Model Development and Training

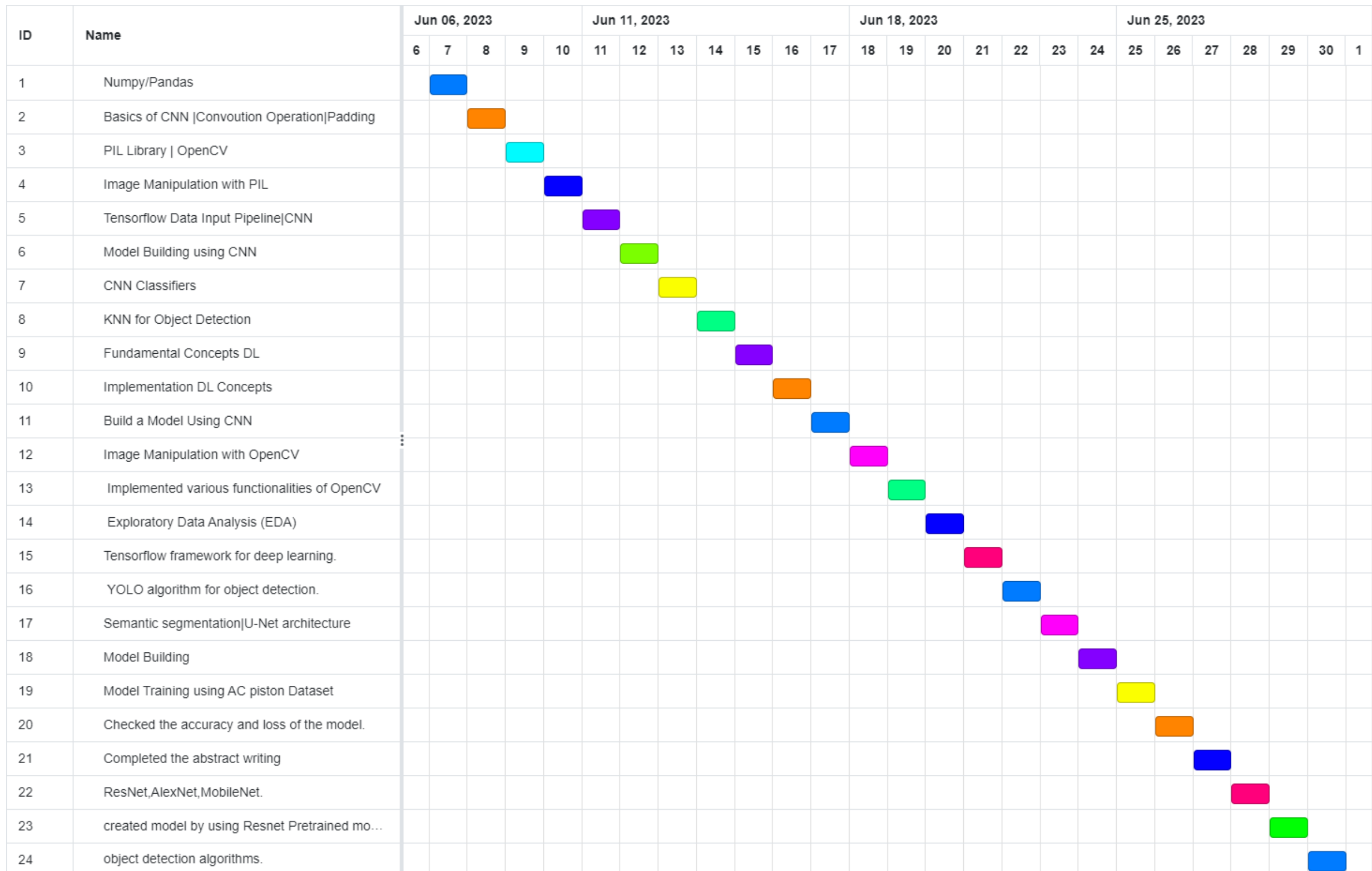
Achieved: Developed a Convolutional Neural Network (CNN) architecture for piston defect detection. Trained the model using the piston dataset to learn the features and patterns of different defects.

Planned Deliverables: Fine-tune the CNN model using transfer learning to enhance its performance. Experiment with various hyperparameters and architectures to optimize accuracy.

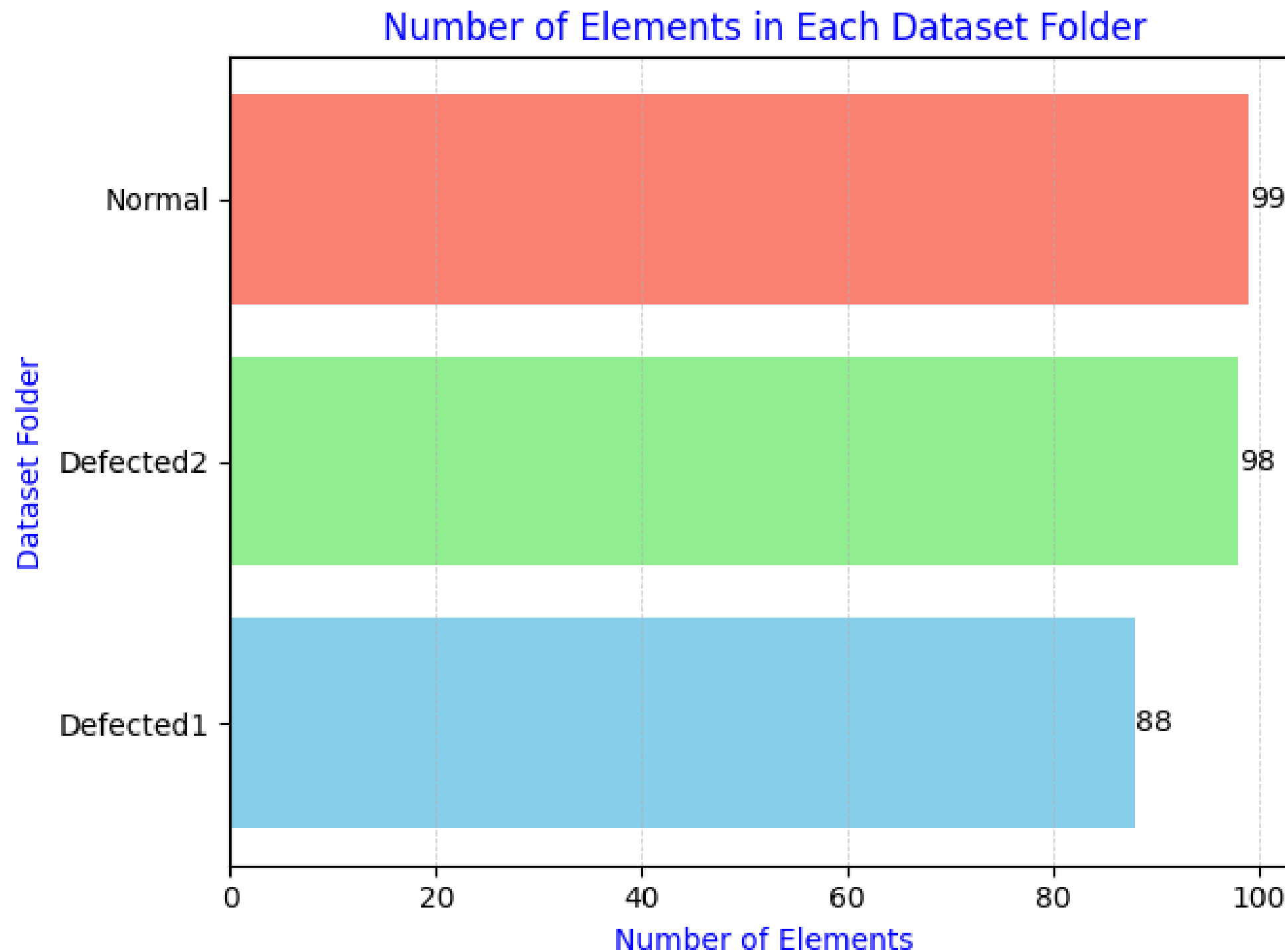
Target completion: 80%



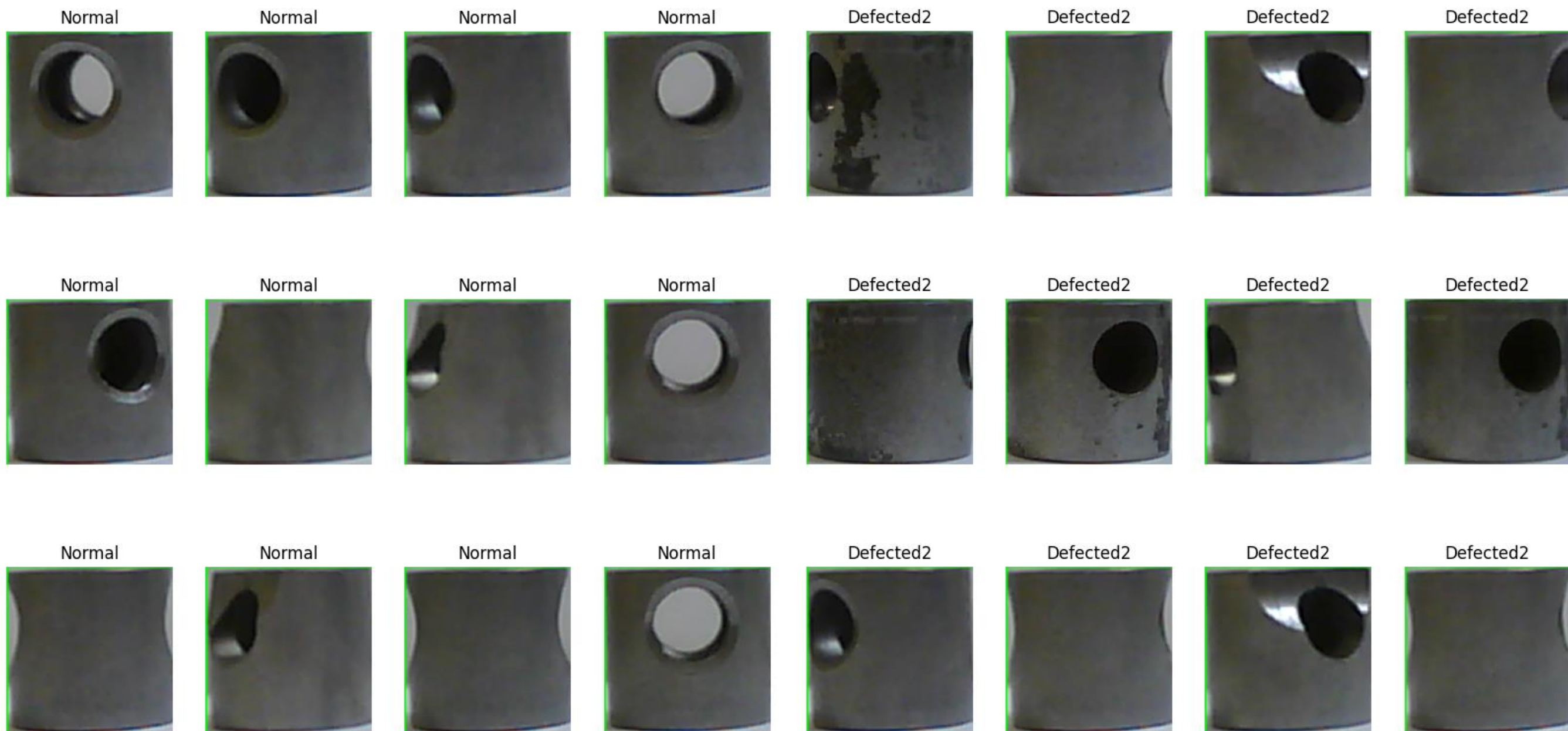
Timeline - Gantt chart



Screenshots of development



Screenshots of development





Screenshots of development



Visual Studio Code interface showing a Jupyter Notebook titled "Detect.ipynb" with the following code cells:

```
train_ds=train_ds.cache().shuffle(1000).prefetch(buffer_size=tf.data.AUTOTUNE)
val_ds=val_ds.cache().shuffle(1000).prefetch(buffer_size=tf.data.AUTOTUNE)
test_ds=test_ds.cache().shuffle(1000).prefetch(buffer_size=tf.data.AUTOTUNE)
```

[21] ✓ 0.0s Python

```
resize_and_rescale= tf.keras.Sequential([
    layers.experimental.preprocessing.Resizing(IMAGE_SIZE,IMAGE_SIZE),
    layers.experimental.preprocessing.Rescaling(1.0/255)
])
```

[22] ✓ 0.0s Python

```
data_augmentation= tf.keras.Sequential([
    layers.experimental.preprocessing.RandomFlip("horizontal_and_vertical"),
    layers.experimental.preprocessing.RandomRotation(0.2)
])
```

The interface also shows the file explorer on the left with "Piston_Defect_detection.ipynb" and "Detect.ipynb" open. The bottom status bar displays "29°C Haze", "Search", and the date "01-07-2023".



Screenshots of development



```
input_shape=(BATCH_SIZE,IMAGE_SIZE,IMAGE_SIZE,CHANNELS)
n_classes = 3
model=models.Sequential([
    resize_and_rescale,
    data_augmentation,
    layers.Conv2D(32,(3,3),activation='relu',input_shape=input_shape),
    layers.MaxPooling2D((2,2)),
    layers.Conv2D(64,kernel_size=(3,3),activation='relu'),
    layers.MaxPooling2D((2,2)),
    layers.Conv2D(64,kernel_size=(3,3),activation='relu'),
    layers.MaxPooling2D((2,2)),
    layers.Conv2D(64,(3,3),activation='relu'),
    layers.MaxPooling2D((2,2)),
    layers.Conv2D(64,(3,3),activation='relu'),
    layers.MaxPooling2D((2,2)),
    layers.Flatten(),
    layers.Dense(64,activation='relu'),
    layers.Dense(n_classes,activation='softmax'),
])
```




Screenshots of development



Visual Studio Code interface showing a Jupyter Notebook titled "Detect.ipynb" with the following code and output:

```
model.compile(  
    optimizer='adam',  
    loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=False),  
    metrics=['accuracy']  
)
```

[27] ✓ 0.0s Python

```
history=model.fit(  
    train_ds,  
    epochs=EPOCHS,  
    batch_size=BATCH_SIZE,  
    verbose=1,  
    validation_data=val_ds  
)
```

[28] ✓ 8m 26.9s Python

Epoch 1/30
WARNING:tensorflow:Using a while_loop for converting RngReadAndSkip cause there is no registered converter for this op.



Thank You

amb120@nist.edu