DETECTION AND VISUALIZATION OF CORRODED SURFACES USING MACHINE LEARNING

Major Project –II (CV499) Report
submitted in partial fulfilment of the requirement for the degree of

BACHELOR OF TECHNOLOGY
in
Civil Engineering

by

Meghana Nayak D (191CV134)

Dhanya R (191CV215)

Shrivathsa B J (191CV246)

DEPARTMENT OF CIVIL ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA

MANGALORE -575 025

APRIL 2023

DETECTION AND VISUALIZATION OF CORRODED SURFACES USING MACHINE LEARNING

Major Project –II (CV499) Report
submitted in partial fulfilment of the requirement for the degree of

BACHELOR OF TECHNOLOGY
in
Civil Engineering

by

Meghana Nayak D (191CV134)

Dhanya R (191CV215)

Shrivathsa B J (191CV246)

under the guidance of

Dr. Pavan G S

DEPARTMENT OF CIVIL ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA

MANGALORE -575 025

APRIL 2023

DECLARATION

We hereby declare that the report of the Major project-II (CV499) entitled **DETECTION AND VISUALIZATION OF CORRODED SURFACE USING MACHINE LEARNING,** which is being submitted to the **National Institute of Technology Karnataka, Surathkal** in partial fulfilment of requirements for the award of the degree of **Bachelor of Technology in Civil Engineering** is a bonafide record of the work carried out by us. The material contained in this report has not been submitted to any university or Institution for the award of any degree

1. Meghana Nayak D (191CV134)
2. Dhanya R (191CV215)
3. Shrivathsa B J (191CV246)

Place: NITK, Surathkal

Date: April 2023

CERTIFICATE

This is to certify that this report entitled **DETECTION AND VISUALIZATION OF CORRODED SURFACE USING MACHINE LEARNING**, submitted by,

1. Meghana Nayak D (191CV134)
2. Dhanya R (191CV215)
3. Shrivathsa B J (191CV246)

as the record of the work carried out by them in the Department of Civil Engineering, is accepted as the B.Tech Major Project –II Report (CV499) submission in partial fulfilment of the requirement for the award degree of **Bachelor of Technology in Civil Engineering.**

Dr. Pavan G S
Guide
Department of Civil Engineering

Chairperson – DUGC
Department of Civil Engineering

ACKNOWLEDGMENTS

We express our heartfelt gratitude to Dr. Prasad Krishna, Director of the National Institute of Technology Karnataka, Surathkal for giving us the opportunity to undertake this major project and for their noble patronage. We express our deepest gratitude to Dr. B R Jayalekshmi, Professor and Head of the Department of Civil Engineering, for extending all necessary support to complete the work reported in this report.

We extend our heartfelt thanks to Dr. Mattur C Narasimhan, our faculty advisor, and all the Teaching and Non-Teaching Staff of the Civil Engineering Department for all the help extended in completing the experimental part of this project.

We express our deepest sense of gratitude to our guide Dr. Pavan G S, Department of Civil Engineering- NITK. Without him, this project would not have been possible. His valuable suggestions and expert guidance and constant encouragement throughout the project played a vital role in bringing the project to good form. We would like to express our gratitude to our friends who helped us directly or indirectly in the successful completion of this report.

MEGHANA D NAYAK

DHANYA R

SHRIVATHSA B J

ABSTRACT

The use of artificial intelligence in asset management is greatly assisting the industry and structural health monitoring systems. Using Machine Learning techniques for asset inspections can increase safety, reduce access costs, provide objective classification, and improve digital asset management systems. The detection and visualization of corrosion from digital images present significant advantages like automation, access to remote locations, mitigation of risk of inspectors, cost savings, and detecting speed. In this work, we used deep learning convolutional neural networks to build simple corrosion detection models and used an extreme gradient boosting algorithm to visualize the corroded surfaces. A large dataset of 1900 images with corrosion and without corrosion was collected using web scraping techniques and labeled accordingly. To approach high-level accuracy, training a deep learning model requires massive and high-resolution image datasets and intensive image labeling. The results and findings will improve the development of deep learning models for detecting and visualizing specific features on corroded surfaces.

Keywords: Structural Health Monitoring, Corrosion, Rust, Image Scraping, Machine Learning, Convolutional Neural Network, Extreme Gradient Boosting

TABLE OF CONTENTS

LIST OF TABLES

Page no.

LIST OF FIGURES

Page no.

CHAPTER 1

INTRODUCTION

1.1 General

The word corrosion is derived by the Latin word corrode which means "to wear away gradually". Science defines corrosion as the electrochemical reaction of a metal or an alloy with its environment. It constitutes one of the two main causes of metal deterioration alongside mechanical loss by erosion, abrasion, or wear. All metal structures are prone to corrosion. Sometimes it can be prevented but economically it makes more sense to control it. Therefore, engineers choose materials based on the assumption that the structure will be maintained during its life cycle.

Corrosion can lead to the loss of the purity of the metal. When a metal structure undergoes corrosion, it loses its strength, and the tendency to experience structural collapse increases. For example, ships, tankers, pipelines, wind turbines, and concrete rebars are often subject to the dangerous effects of corrosion. A study by NACE estimates the global annual cost of corrosion at US$2.5 trillion, which is about 3.4% of the worldwide GDP (2013). These numbers solely represent the direct costs such as forced shut-downs or accidents; neither individual safety nor environmental consequences are included. Therefore, effective corrosion control methods become highly critical in preventing the damaging effects of corrosion. Various methods are widely used in the industry to control and prevent corrosion. These methods include cathodic and anodic protection, corrosion inhibitors, material selection, application of internal and external protective coatings, corrosion monitoring, and inspections. Early detection of structural degradation prior to failure does not only have financial benefits. Still, it can also prevent catastrophic collapses of structures and avoid harmful situations for humans and the environment.

The recent improvements in Artificial Intelligence (AI) for object recognition are largely attributed to the emergence of deep learning artificial neural networks. As one of the major fields of A.I., deep learning mimics the working of the human brain in processing data for use in detecting objects, recognizing speech, and creating patterns for use in decision-making. Deep learning has developed as the natural progression from 'shallow networks' to multi-layered networks of neurons that are able to transform representations (of data, including images) from simple to complex, with increasing layer depth. For corrosion protection, the first step towards the maintenance of structures is the visual inspection. Nowadays, this is mainly done by humans to collect qualitative data. Despite that these inspectors are certificated and experienced, the performance of this time-consuming method is subjective and largely dependent on the experience and qualifications of the individual.

On top of that, some locations of structures are difficult or completely inaccessible because of safety reasons, such as deep-sea pipelines, oil tanks, wind turbines, and some hindering constructions.

1.2 Corrosion Cause, Loss, and Remedy

Metals begin to corrode when it reacts with other substances such as oxygen, hydrogen, an electric current, or even bacteria or dirt. Corrosion can also take place when metals such as steel are occupied under so much stress compelling materials to crack.

Corrosion can cause various types of losses including:

1. Structural Integrity: Corrosion can weaken or damage structural components, compromising the safety and performance of buildings, bridges, airplanes, vehicles, and many other objects.
2. Financial Losses: Corrosion can lead to unexpected and expensive repair or replacement costs. It can also reduce the lifespan of equipment or infrastructures, hurting the bottom line of companies and governments.
3. Environmental Impact: Corrosion can release harmful substances, such as lead, mercury, and cadmium, into the environment. These substances can contaminate soil, water, and air, and cause health problems for humans and wildlife
4. Operational Problems: Corrosion can interfere with the normal functioning of machines and devices, leading to decreased productivity, efficiency, and reliability. It can also cause interruptions or failures in critical processes, such as electricity generation or transportation.
5. Aesthetics: Corrosion can make structures look old, dirty, or unappealing, which can lower their value, attractiveness, and usability. It can also reduce the quality of life for people who have to live or work in corroded environments.

There are several remedies for corrosion that can be used to prevent or control the effects of corrosion:

1. Protective Coating: Applying protective coatings, such as paint, zinc, or powder, can prevent the surface of metals from coming into contact with corrosive agents.
2. Cathodic Protection: Adding an anode, such as a sacrificial metal or impressed current, can protect a metal object by attracting the corrosive agents away from the object.
3. Corrosion Inhibitors: Adding corrosion inhibitors to metals or fluids can help to slow down the corrosion process by reducing the rate of oxidation.
4. Design Modification: Altering the design of an object or structure can help to reduce exposure to corrosive agents. This can include using non-corrosive materials, changing material thickness or orientation, and improving drainage or ventilation.
5. Fluid Control: Controlling the acidity, alkalinity, moisture, and temperature of fluids can help to prevent or reduce the rate of corrosion. This can be done by pH buffering, dehumidifying, heating or cooling, or adding corrosion inhibitors.
6. Regular Maintenance: Regular maintenance, inspection, and cleaning of metal objects and structures can help to identify and repair any damage or weaknesses that may cause corrosion.

1.3 Convolutional Neural Network

Neural networks are a subset of machine learning, and they are at the heart of deep learning algorithms. They are comprised of node layers, containing an input layer, one or more hidden layers, and an output layer. Each node connects to another and has an associated weight and threshold. If the output of any individual node is above the specified threshold value, that node is activated, sending data to the next layer of the network. Otherwise, no data is passed along to the next layer of the network.

There are various types of neural nets, which are used for different use cases and data types. For example, recurrent neural networks are commonly used for natural language processing and speech recognition whereas convolutional neural networks (ConvNets or CNNs) are more often utilized for classification and computer vision tasks. Prior to CNNs, manual, time-consuming feature extraction methods were used to identify objects in images. However, convolutional neural networks now provide a more scalable approach to image classification and object recognition tasks, leveraging principles from linear algebra, specifically matrix multiplication, to identify patterns within an image. That said, they can be computationally demanding, requiring graphical processing units (GPUs) to train models. Convolutional neural networks are distinguished from other neural networks by their superior performance with image, speech, or audio signal inputs. They have three main types of layers, which are: Convolutional layer, Pooling layer, Fully-connected (FC) layer. The convolutional layer is the first layer of a convolutional network. While convolutional layers can be followed by additional convolutional layers or pooling layers, the fully-connected layer is the final layer. With each layer, the CNN increases in its complexity, identifying greater portions of the image. Earlier layers focus on simple features, such as colors and edges. As the image data progresses through the layers of the CNN, it starts to recognize larger elements or shapes of the object until it finally identifies the intended object.

1.4 Data Scraping

Data scraping, also known as web scraping or data harvesting, refers to the process of extracting information or data from websites using automated tools or software. It involves the use of computer programs or scripts that can navigate through web pages, extract relevant data, and store it in a structured format such as a database or a spreadsheet. There are various tools and techniques available for data scraping, ranging from simple web scraping libraries such as Beautiful Soup and Scrapy to more advanced machine learning-based tools that can extract and analyze data from unstructured sources such as social media and news articles. It is important to ensure that any data scraping activities are conducted ethically and in compliance with legal requirements and website terms of service. In this work, data scraping is automated using Selenium. The top 100 images resulting in Google for chosen keywords are collected and stored.

1.5 Structural Health Monitoring

Structural performance of various infrastructure projects such as buildings, bridges, dams, etc, deteriorate over time, and sometimes natural disasters such as landslides, earthquakes, and storms can speed the process. Hence, structural health monitoring is extremely important in predicting the lifespan of buildings and extending that if possible. Structural health monitoring has multiple applications. Structural health monitoring assesses the state of structural health and, through appropriate data processing and inter-predation, may predict the remaining life of the structure. Many aerospace and civil infrastructure systems are at or beyond their design life; however, it is envisioned that they will remain in service for an extended period. SHM is one of the enabling technologies that will make this possible. It addresses the problem of aging structures, which is a major concern of the engineering community. SHM allows condition-based maintenance (CBM) inspection instead of schedule-driven inspections. Another potential SHM application is in new systems; that is, by embedding SHM sensors and associated sensory systems into a new structure, the design paradigm can be changed and considerable savings in weight, size, and cost can be achieved

Corrosion is one of the main root causes that can produce offshore structural failure. Corrosion monitoring is the process of obtaining very frequent corrosion measures to evaluate the progress of corrosion within a specific environment. Corrosion inspections are done to evaluate the material condition at any given time based on a pre-scheduled routine or the available risks. Usually, inspections are performed much less frequently than corrosion monitoring. Frequent corrosion measures are always advantageous in early detecting the risks, repairing conditions, and, consequently, reducing the operational and maintenance cost associated with corrosion. The rust formation due to corrosion depends on the environment and the type of material, as well. Carbon steel corrodes mostly by general corrosion (uniform corrosion), but localized types of corrosion can also take place, e.g., pitting, which usually is not considered in conventional corrosion analysis. Different reasons can be identified behind the growth of the corrosion process in an offshore structure, such as temperature, salinity, pH, and coating damage due to tidal fluctuations, dissolved oxygen, variable cyclic load due to wave and wind impact, etc. Even for a well-designed structure, with a long time of exposure to the harsh marine environment, corrosion causes the degradation of the metal leading to create corrosion fatigue cracks and, consequently, structural failure. This scenario increases the O&M cost because of the possibility of more frequent maintenance, repair activities, and any replacement if needed. Therefore, a successful corrosion monitoring approach can potentially decrease the ongoing operations and maintenance cost of wind turbines directed at reducing economic losses and environmental damages. However, corrosion monitoring is one of the major challenges that an SHM system developed for offshore wind turbines faces, mainly due to the accessibility difficulties and the area the corrosion monitoring system must cover, which is very large even when one considers that some zones are more affected than others, e.g., the splash zone. Thinking about an automated solution, ideally, the corrosion sensor should work unattended 24/7 for several months or even years. This imposes very hard specifications that the monitoring system must meet. Therefore, prior to the selection of an appropriate corrosion monitoring technique and sensors, the zone of interest (splash zone, atmospheric zone, submerged zone) of the wind tower and the respective environmental conditions must be considered separately.

Manual corrosion inspection techniques

There are different types of corrosion inspection and monitoring techniques to find out the corrosion condition of metals. Figure 1 shows a detailed classification of these techniques based on the corrosion detection method and sensing parameters.

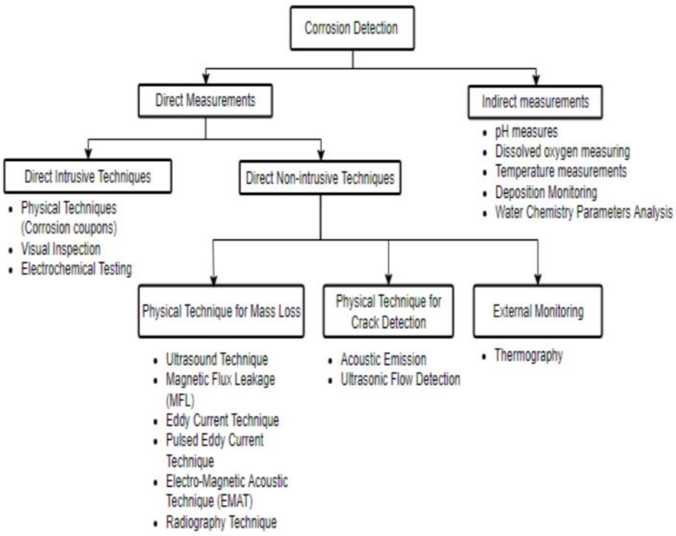


Fig 1 Corrosion detection methods.

(https://www.researchgate.net/publication/357816918_Ultrasound-Based_Smart_Corrosion_Monitoring_System_for_Offshore_Wind_Turbines)

In general, corrosion detection measures can be directly measuring metal loss due to corrosion or corrosion rate or indirectly measuring any parameter that may be a cause or consequence of the metal loss or corrosion. Some of these techniques need direct contact/exposure to the same corrosive fluids or to access the internal environment where the corrosion process takes place. They are known as intrusive techniques and can alter the corrosion process and create disturbances to the operation during the installation and reinstallation of measuring probes or inspection processes. Therefore, it is beneficial if corrosion testing can be performed from the outside of the test object. This is possible with non-intrusive techniques by detecting physical and geometry changes due to corrosion (mass loss, crack, or surface discontinuities), and there is no need for any direct contact with the same corrosive fluids or access to the internal environment. On the other hand, it is very important to perform corrosion detection activities in field applications without destroying the engineering properties of the material and without affecting its long-term future performance during the inspection process. This kind of evaluation can be achieved by using Non-Destructive Testing (NDT) methods for material testing. Among the techniques listed in Figure 1, if this condition is satisfied, they are known as an NDT method. Therefore, we can propose that the most appropriate corrosion detection technique for offshore wind turbines shall be both non-intrusive and non-destructive.

1.6 Review of Literature

The study by Kuo-Wei Liao et al. (2016) aims to develop an automatic process for the detection of the rusted surface of a steel bridge coating using digital images. Specifically, this study investigates images with non-uniform illumination and rust stains. In this, the digital image recognition algorithm consists of three different detection techniques: K-means in H, DCDR in RGB, and DCDR in H. These techniques were integrated and can be executed automatically. The LS-SVM was used to predict the radii in the DCDR approaches. To detect a rusted area, the images were first categorized into three groups based on the Hue percentages and COVs of the grey levels. To verify the proposed algorithm, 100 images were collected. The overall correct percentage was 86%, indicating that the proposed approach can provide consistent and promising detection results.

The study by Yang Tian et al. (2022) aims to develop an automatic visual-based rust removal approach has been introduced. The proposed rust detection approach can automatically and robustly detect rust under critical conditions. Based on the proposed rust detection approach, a visual servo rust removal controller is designed to combine the rust condition information with SEA fuzzy force control.

The study by M. Khayatazad et al. (2020) reports on the implementation and use of an algorithm that quantifies and combines two visual aspects – roughness and color – in order to locate the corroded area in a given image. The uniformity metric calculated from the gray-level co-occurrence matrix is considered for the roughness analysis. The histogram of corrosion-representative colors extracted from a data set in HSV color space is used for the color analysis. The algorithm has been applied to a large dataset of photographs of corroded and non-corroded components and structures. The main part of the algorithm first consists of a roughness analysis after resizing the image. The identified rough area is transferred as a candidate corroded region to the second step, i.e. the color step, for further investigation. In the color step, the color of the candidate area is compared with the predefined colors of corrosion. Finally, the outcome of this algorithm is a map showing the locations of detected corrosion

The study by Mojtaba K et al. (2020) has proven the outstanding merit of artificial neural networks compared to a previously developed image-processing algorithm for corrosion detection. As stated, the required size of the proposed ANN's training dataset is a few orders of magnitude of that of a typical CNN, demonstrating its practical relevance. The paper's findings prove that combining the deep learning feature of CNNs with the pixel-based nature of ANNs builds a robust ANN for corrosion detection. In the present work, different color spaces and textural properties are examined for their impact on the robustness of the ANN. Results reveal that the best color channels can be achieved by combining CIE Luvand YUV color spaces.

1.7 Objectives

Corrosion can cause serious damage. It can weaken the body of the metal material, causing malfunction, leakage, and other damage. On 30 October 2022, Morbi Bridge collapsed in Gujarat resulting in 135 deaths. A forensic report presented in court said that the bridge could not withstand the weight of the new heavy flooring given the cables were rusted. A well-detailed and robust Structural Health Monitoring would have saved lives and monetary resources. The development of automated corrosion detection models which can be integrated with mobile devices can save time, money, and lives in the future benefiting economy as well as society. In order to detect corrosion in remote, inaccessible structures and to analyze its scale this project aims to develop a CNN-based machine-learning model which detects corrosion on a given image of a Steel Structure. The model with better evaluation metrics is used to predict the corrosion on sample images taken on the campus. This project also aims to develop an XGBoost pixel-level color classification model to visualize and retrieve the approximate area of corroded surfaces.

CHAPTER 2

METHODOLOGY

2.1 General

Convolutional Neural Networks (CNNs) were used for corrosion detection. The data set of 1050 corrosion images and 850 non-corroded images was run through 2 different CNN models shown in Fig 2 and the accuracy of these models was compared to each other to see which one functioned the best. Post this, image pre-processing was performed on the images to visualize the corroded area in the image.

2.2 Data Collection

Image data is collected through web scraping and stored in the folder for which the path is specified. Labeled data is required for supervised learning for the identification of corrosion presence. ‘CORROSION’ and ‘NO CORROSION’ are the folders created to store the data. A 10-digit random id is assigned to the image downloaded. Selenium was used to automate web browser interactions. It is a useful technique for web scraping, as it simulates human interaction with the website and can make the process more efficient. Chrome driver is used, which guarantees that the browser closes down ordinarily, even if something within the context raises an error. ‘search_and_download’ function allows to specify ‘number_images’, which by default is set to 100 but can be changed according to the requirement. A total of 1050 CORROSION images were scraped from Google Images using keyword searches that include eight categories of corrosion problems, such as 'Steel Corrosion/Rust,' 'Ships Corrosion,' 'Ship Propellers Corrosion,' 'Cars Corrosion,' 'Oil and Gas Pipelines Corrosion, 'Concrete Rebar Corrosion, 'Water/Oil Tanks Corrosion, 'Stainless Steel Corrosion, and a total of 850 ‘NO CORROSION’ images were also scraped from Google Images using the same terms without corrosion. Sample images for detection and visualization were taken from the premises of the Girls Hostel Block Complex of the NITK campus from a distance of 1 to 1.5 m with a Camera of resolution 48 megapixels.

2.3 Data Pre-processing

The dataset contains images with varying sizes. Pixel normalization and data augmentation processes were used on the collected data. Pixel normalization is a technique used to standardize the pixel values of an image to a range of 0 to 1. Images were resized to 128x128 pixels and were converted into grey scale. Data is split into a train, validation, and test data in the ratio of 7:2:1 respectively. Data augmentation includes techniques such as adding noise, flipping orientation, changing scale, etc. thus, exposing the model to a diverse set of training data which improves the performance of the model.

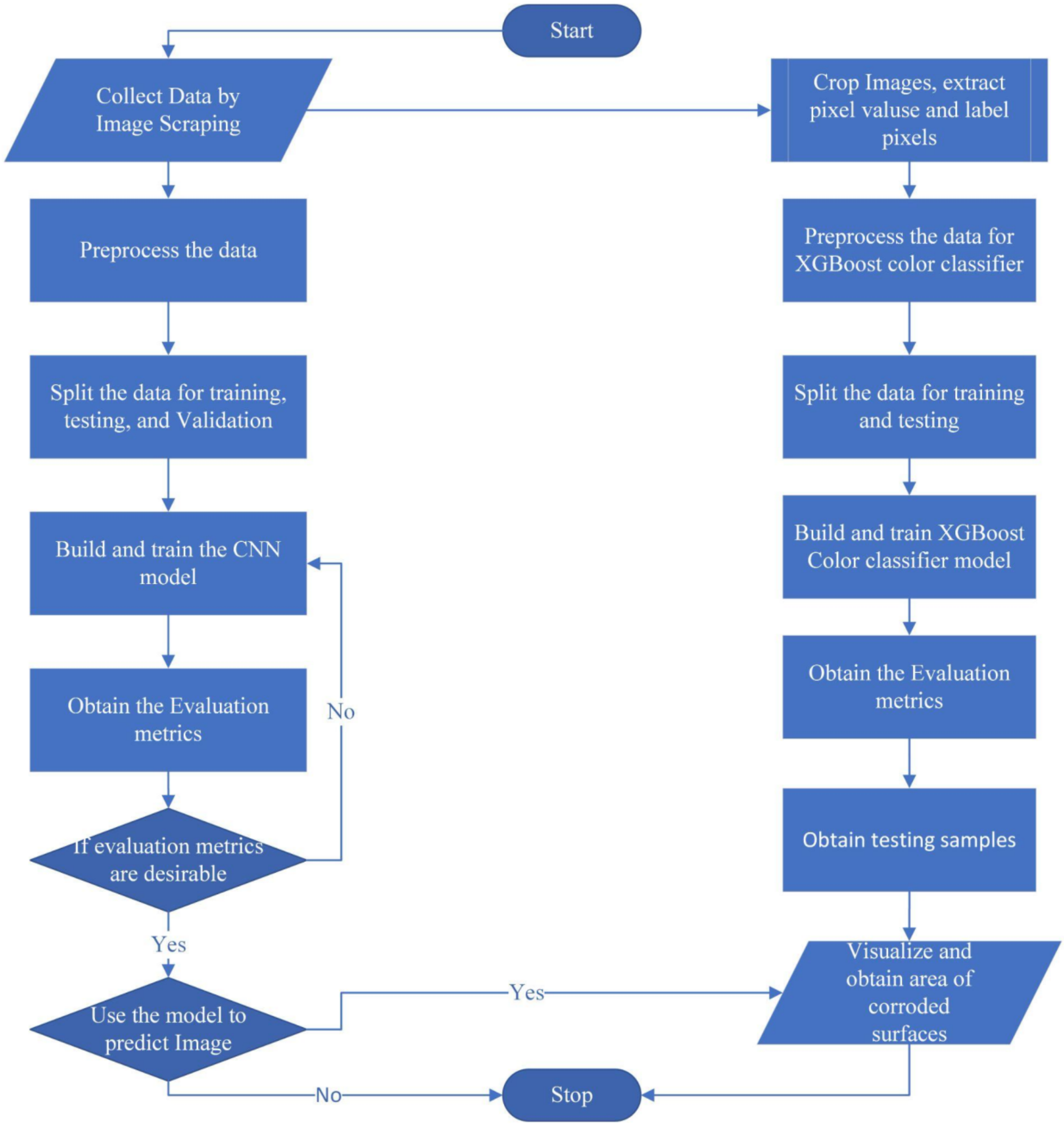


Fig. 2 Developed Algorithm for the study

2.4 Algorithm

In appearance, the surface of the corroded surface is rougher than the Not corroded ones, and its color ranges between Hue of red and brown. The dataset used for the development of the model was collected by scraping images from Google. The scraping process was automated with the help of Selenium which pretends to be a real user, opens the browser, moves the cursor around, and clicks buttons to download and save the images in a pre-selected directory. A total of 1900 images were scraped from Google containing 1050 corrosion and 850 clean(no corrosion) images of various dimensions and are labeled either "CORROSION" for corrosion or "NO CORROSION" for clean images. Since an image with small dimensions, but still showing important features, demands less time for processing, resizing large and high-quality images to smaller ones might be an option for easy training. So all images were resized to 128 x 128 pixels. These pre-processing settings are used from the inbuilt libraries. These pre-processed images are then split into the ratio of 70% training, 20% validation, and 10% testing purposes. With the datasets, ready 2 CNN models were built and trained. The model with better evaluation metrics was chosen and used for detection purposes. It should be mentioned that the images taken as samples for detection have been acquired from a distance of 1 to 1.5 m with a Camera of resolution 48 megapixels. With these resolutions and distances, our study revealed that 128 pixels in one direction are sufficient for feature extraction. The images which are classified as detected are then fed into the XGBoost color classifier to visualize corroded surfaces and to find the area of corroded surfaces. The entire algorithm has been encapsulated in the form of a flowchart in Figure 2.

2.5 Models to detect corrosion on a given image

Two models are used in this study to detect corrosion on steel surfaces. The baseline model is of four Conv 2D layers, and the second model has 4 Conv 2D layers with dropouts. The second model is the best performing among the two. XGBoost color classifier was used to find the area of corrosion on the image using color attributes.

2.5.1 Model Definition

Baseline model

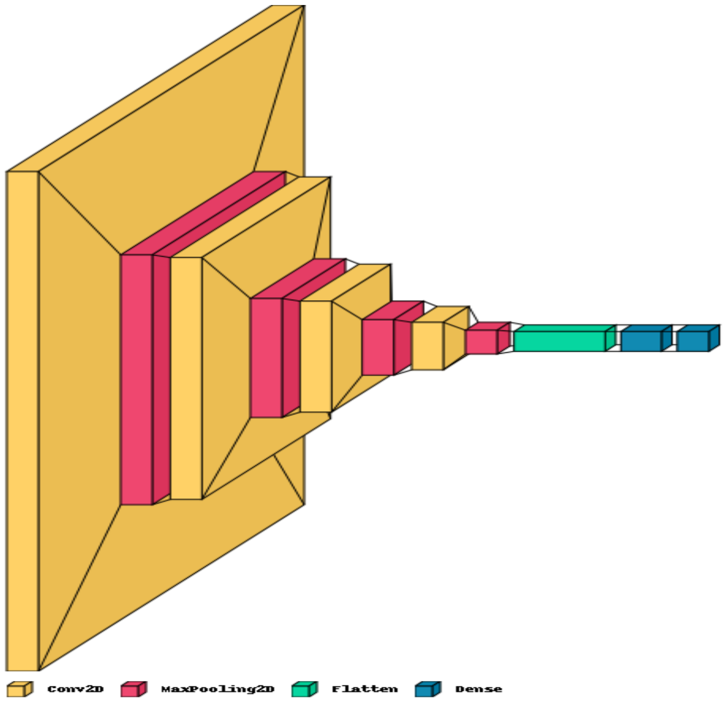


Fig 3 Baseline model

The baseline model is a simple model that provides a benchmark for evaluating the performance of more complex models. It is often used as a reference point for comparing the effectiveness of different models and techniques.

• The first layer is a convolutional layer (Figure 3, yellow layer) with a 3×3 kernel and uses 128 filters that result in 126×126×128 volumes with the same convolutions. The filters are always 3×3 with a stride of 1. The ReLu activation function is used.

• After this, the max pooling layer (Figure 3, red layer) was used with a max-pool of 2×2 size and stride 1 which reduces the height and width of a volume from 126×126×128 to 63×63×128.

• After these 3 convolutional layers are used with a 3×3 kernel each layer followed by the max pooling layer with a max-pool of 2×2 size and stride 1, volume is reduced to 6×6×16.

• Second convolutional layer has 64 filters, the third has 32 filters and the last layer has 16 filters. All layers have the ReLu activation function.

• After the final pooling layer, 6×6×16 volume is flattened into a Fully Connected (FC) layer with 576 channels. it flattens the output tensor from a 3D tensor to a 1D tensor (Figure 3 green layer).

• A fully connected layer (Dense) with 256 neurons is added (Figure 3, blue layer), which applies the Rectified Linear Unit (ReLU) activation function. Then a fully connected layer with a single neuron applies the Sigmoid activation function. The Sigmoid function maps the output of the previous layer into a probability between 0 and 1. A detailed diagram of the baseline model is given in Figure 3.

Best performing model

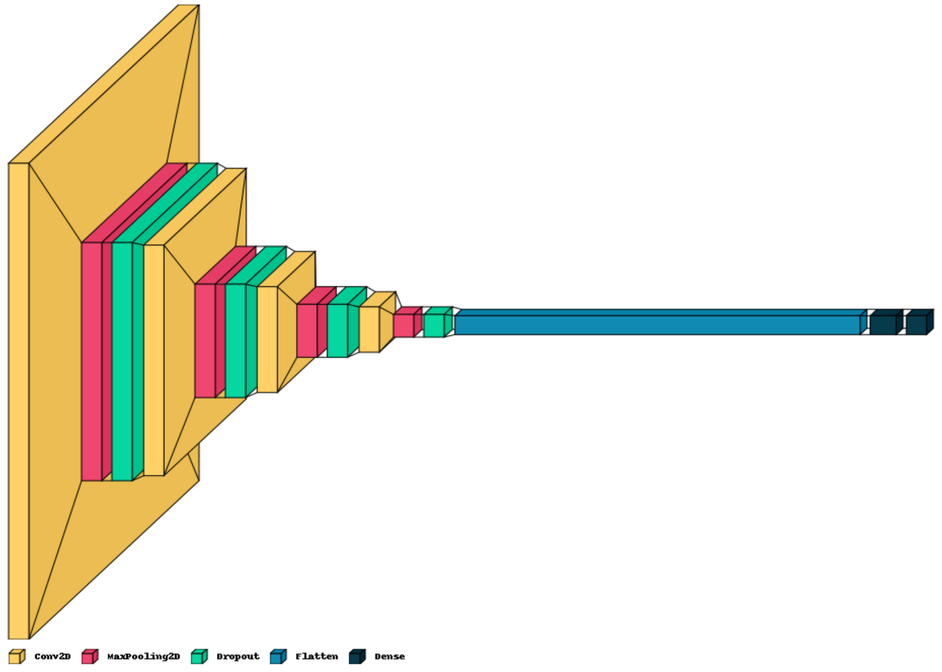


Fig 4 Best performing model with dropouts

This is the model with the best performance. It has four Conv 2D layers with max-pooling and dropouts. Dropout is a regularization technique commonly used in deep learning models to prevent overfitting. In this specific case, some of the nodes in the layer that precedes the dropout layer will be randomly dropped out with a probability of 0.2.

• The first layer is a convolutional layer with a 3×3 kernel and uses 64 filters that result in 126×126×64 volumes with the same convolutions. The filters are always 3×3 with a stride of 1. The ReLu activation function is used.

• After this, the max pooling layer was used with a max-pool of 2×2 size and stride 1 which reduces the height and width of a volume from 126×126×64 to 63×63×64.

• Then a dropout layer is added with the dropout rate of 0.2, and the resulting volume remains 63×63×64.

• After these 3 convolutional layers are used with a 3×3 kernel each layer followed by max pooling layer with a max-pool of 2×2 size and stride 1, volume is reduced to 6×6×128.

• Second convolutional layer has 64 filters, the third has 128 filters and the last layer has 128 filters. All layers have the ReLu activation function.

• After the final dropout layer, 6×6×128 volume is flattened into a Fully Connected (FC) layer with 4608 channels. it flattens the output tensor from a 3D tensor to a 1D tensor.

• a fully connected layer (Dense) with 256 neurons is added, which applies the Rectified Linear Unit (ReLU) activation function. Then a fully connected layer with a single neuron applies the Sigmoid activation function. The Sigmoid function maps the output of the previous layer into a probability between 0 and 1. A detailed diagram of the Best performing model is given in Figure 4.

2.5.2 Model Training

The model is trained using 1900 images of which 1050 are corrosion images and 850 are no corrosion images. We trained both models by specifying 100 epochs and applied an early stopping call-back (patience = 7 is assigned in this study) where the training stops if the validation loss is not improved for a certain number of epochs and the epoch with the best loss is considered. Hyperparameters like learning rate are set to 0.01, batch size is set to 32.

2.5.3 Model Input

The input data will be a 3-dimensional array i.e, it's a tensor and will have 3 color channels (RGB). The width and height of the input data will both be 128 pixels. A data generator is created using the 'ImageDataGenerator' in-built library in Python which generates image data to be automatically fed into the model while training. The data set was run through the two aforementioned models and the results of the same are presented below along with a comparative study of their accuracy. Morphological operations were performed using Python that gave the area of corrosion on the image. Excerpts from the labeled data set are shown in Figure 5.

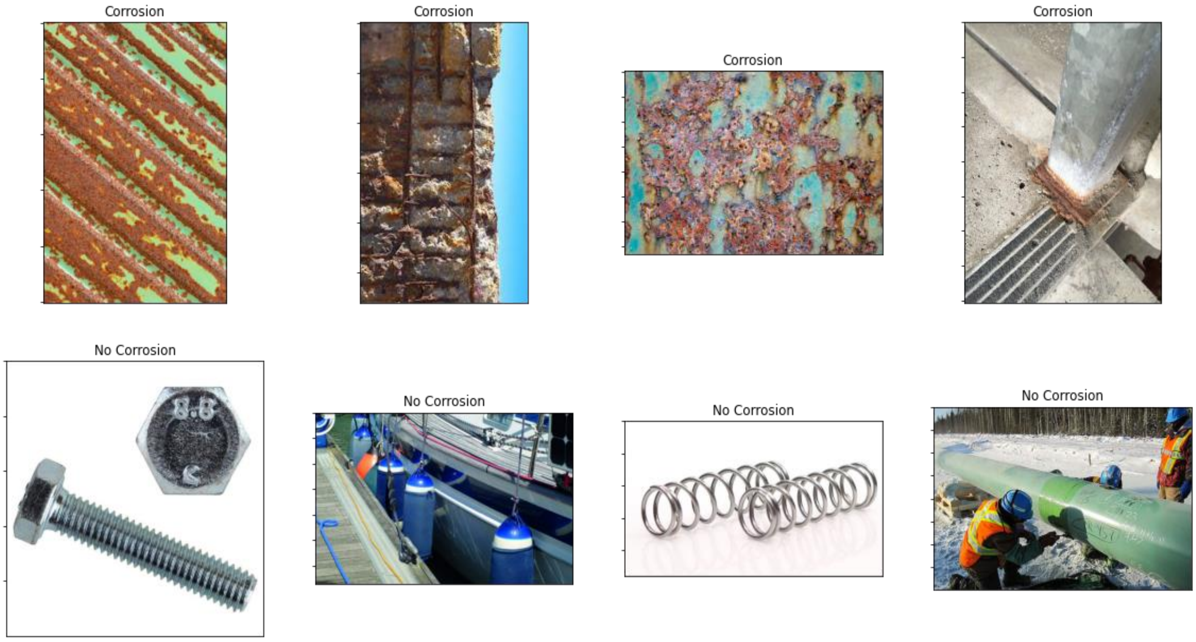


Fig 5 Labelled dataset

2.5.4 Model Output

Model output is a scalar output between 0 and 1. As the sigmoid activation function produces a scalar output between 0 and 1. It delivers output in a list of 2 values, where probabilities of a given image belonging to two classes are present.

2.6 Model to visualize and retrieve area of corrosion

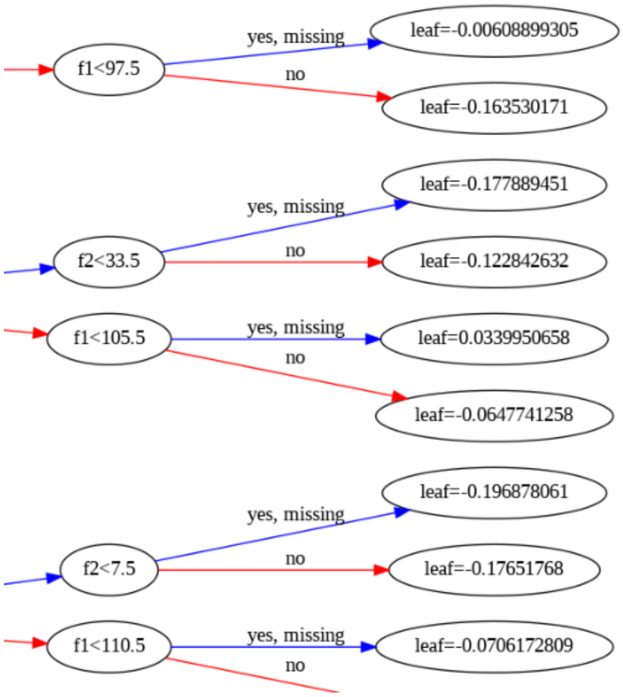
XGBoost color classifier model is used to find the area of corrosion on the image using color attributes. XGBoost (eXtreme Gradient Boosting) is a popular machine-learning algorithm for classification and regression problems. The same data used for the corrosion detection model is used for this model but with some modifications to the data. We cropped images containing corroded parts and clean parts separately and employed the image-to-pixel RGB values conversion and labeling technique. To fit a training dataset using XGBoost, an initial prediction is made. Residuals are computed based on the predicted value and the observed values. A decision tree is created with the residuals using a similarity score for residuals. The similarity score of the data in a leaf is calculated (from equation 1), as well as the gain (from equation 2), in similarity in the subsequent split. The gains are compared to determine a feature and a threshold for a node. The output value for each leaf is also calculated using the residuals. For classification, the values are typically calculated using the log of odds and probabilities. The output of the tree becomes the new residual for the dataset, which is used to construct another tree. This process is repeated until the residuals stop reducing or for a specified number of times. Each subsequent tree learns from the previous trees and is not assigned equal weight. To use this model for prediction, the output from each tree multiplied by a learning rate is added to the initial prediction to arrive at a final value or classification. XGBoost uses the following parameters and methods to optimize the algorithm and provide better results and performance:

- Regularization—A Regularization parameter (lambda) is used while calculating the similarity scores to reduce the sensitivity to individual data and avoid overfitting.
- Pruning—A Tree Complexity Parameter (gamma) is selected to compare the gains. The branch where the gain is smaller than the gamma value is removed. This prevents overfitting by trimming unnecessary branches and reducing the depth of the trees.
- Weighted quantile sketch—Instead of testing every possible value as the threshold for splitting the data, only weighted quantiles are used. The selection of quantiles is done using a sketch algorithm, which estimates a distribution on multiple systems over a network.
- Parallel Learning—This method divides the data into blocks that can be used in parallel to create the trees or for other computations.
- Sparsity-aware split finding—XGBoost handles sparsity in data by trying both directions in a split and finding a default direction by calculating the gain.
- Cache-aware Access—This method uses the cache memory of the system to calculate the similarity scores and output values. The cache memory is a faster access memory compared to the main memory and improves the overall performance of the model.
- Blocks for Out-of-core Computation—This method works with large datasets that cannot fit in the cache or the main memory and that must be kept in hard drives. The dataset is divided into blocks and compressed. Uncompressing the data in the main memory is faster than reading from the hard drive. Another technique called sharding is used when the data must be kept on multiple hard drives.

$$Similarity\ score = (\Sigma Residuals)^2 / (\Sigma P(1-P) + \lambda) \quad (1)$$

$$Gain = Left_{Similarity\ score} + Right_{Similarity\ score} - Root_{Similarity\ score} \quad (2)$$

XGBoost color classification model was built using Python in-built library ‘xgboost’ and a data generator was built which converts numpy array data of label and pixel values into Dmatrix and feeds into the model. Model hyperparameters like learning rate were set to 0.01, and the model ran 100 times. A detailed diagram of XGBoost decision trees were given in Figure 6.



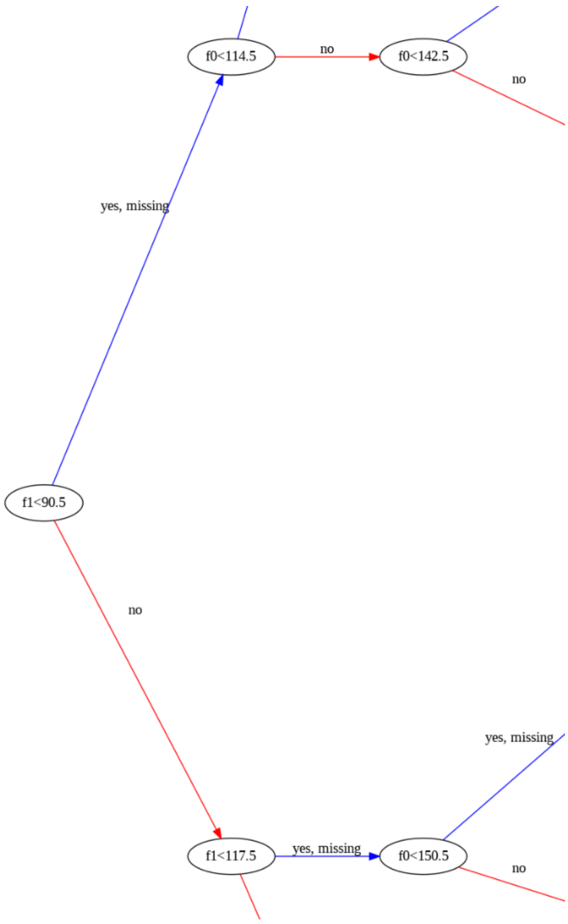


Fig 6 Decision trees of XGBoost color classifier

CHAPTER 3

RESULTS, AND DISCUSSIONS

3.1 Model Accuracy and Results

We trained each CNN model for 100 epochs and applied early stopping regularization where the training stops if the validation loss is not improved for a certain number of epochs and the epoch with the best loss is considered. Here we found that the baseline model was overfitting and the other model was able to learn and classify the images into corrosion or no corrosion. Fig shows the learning curve and the accuracy of each model. The second model with 4 Conv 2D layers with dropouts showed the best performance with a CNN AUC of 94.44%. The result has been shown graphically in Figure 8. The Baseline model shows overfitting characteristics. The graph shows that the validation loss decreases initially. However, this ceases at epoch 12 and the validation loss starts increasing quickly. The training loss keeps decreasing until it almost equals zero at epoch 20. This is expected because the model was trained to best suit the training set of data. There are several ways to lessen overfitting:

- Data augmentation: Data augmentation is an excellent practice to add more data to the existing dataset and add minor alterations and diversity to avoid the model from overfitting to training data.
- L1&L2 regulations: Apply regulation, which comes down to adding a cost to the loss function for large weights.
- Dropout layers: Use dropout layers, which will randomly remove certain features by setting them to zero.
- Batch normalization: Normalize the input layer by adjusting and scaling the activation.

XGBoost color classifier showed a CNN AUC of 86.67 % and an accuracy of 89.01% as shown in Figure 9. We cannot consider the Accuracy score (from equation 3) as an ultimate evaluation metric in our case as data is not equally divided between two classes. The following evaluation metrics (from equations 3 to 7) were evaluated for measuring the performance of a classification model.

Accuracy = (True positives+ True negatives) / Total positives and negatives classes. (3)

Precision -= True positives / (True positives +False positives) (4)

Recall = True positives / (True positives + False negatives) (5)

Specificity = True negatives / (True negatives +False positives) (6)

*F1 Score = 2 * (Precision * Recall) / (Precision + Recall)* (7)

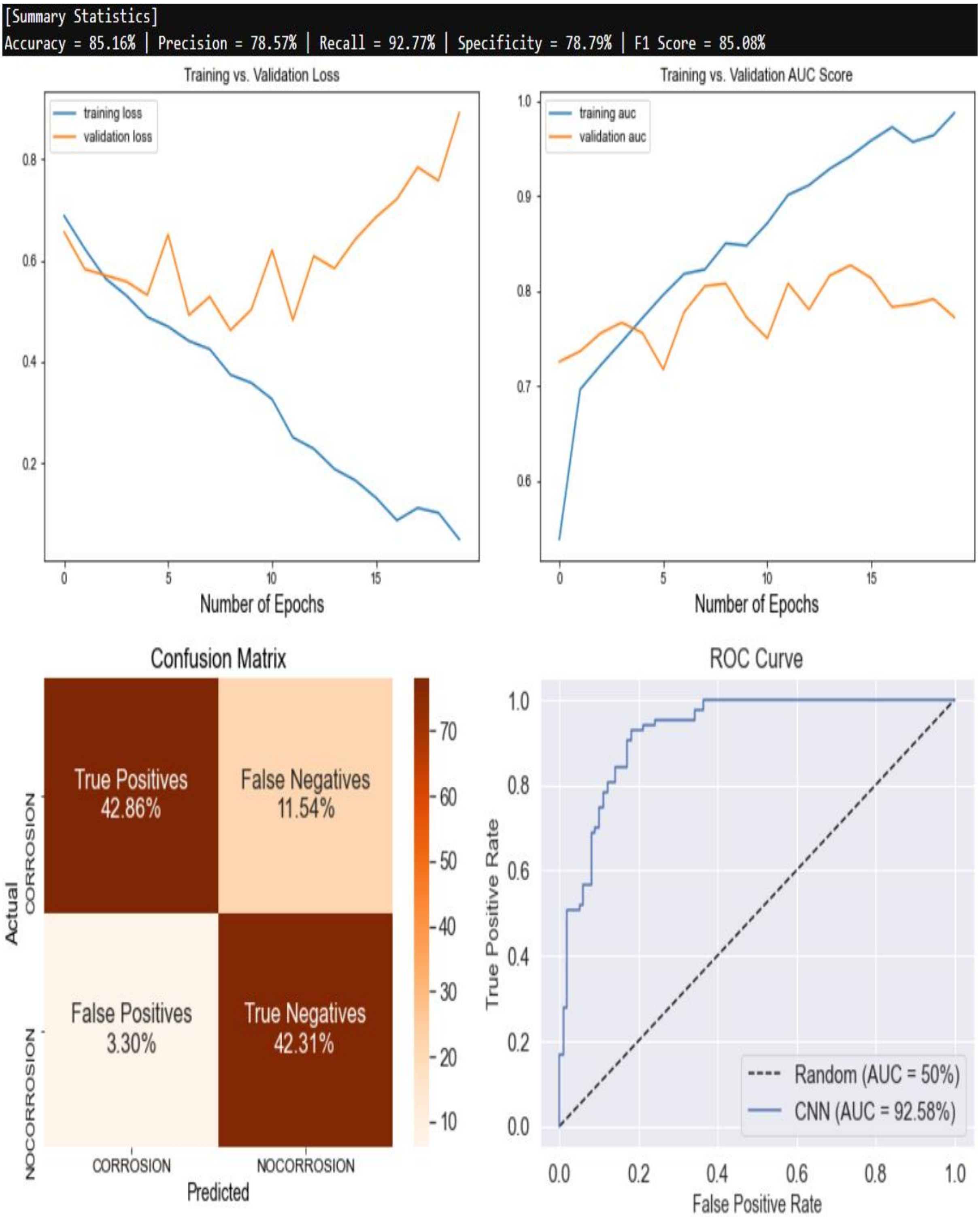


Fig. 7 Confusion matrix and other evaluation metrics of the Baseline model

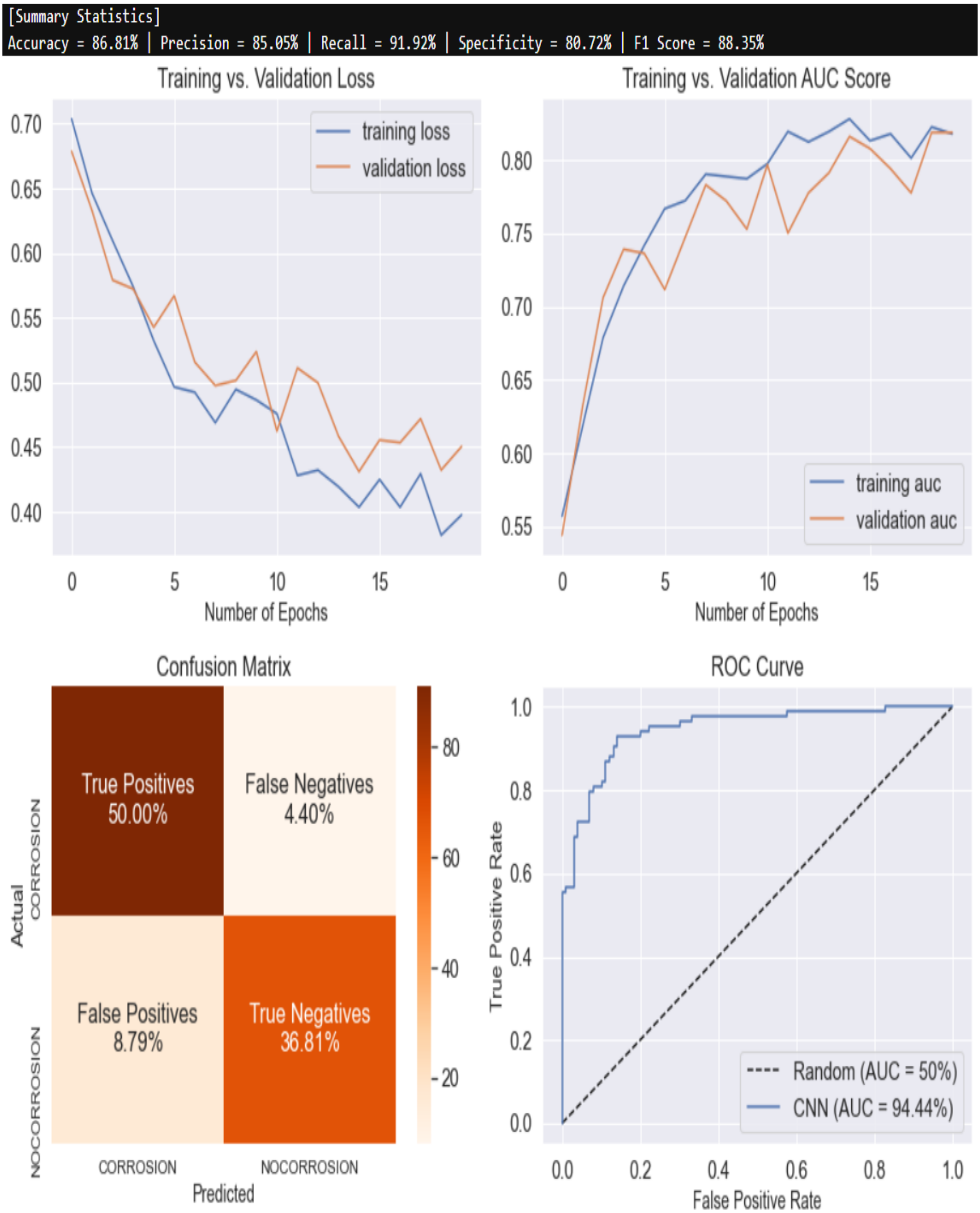


Fig. 8 Confusion matrix and other evaluation metrics of the Best performing model

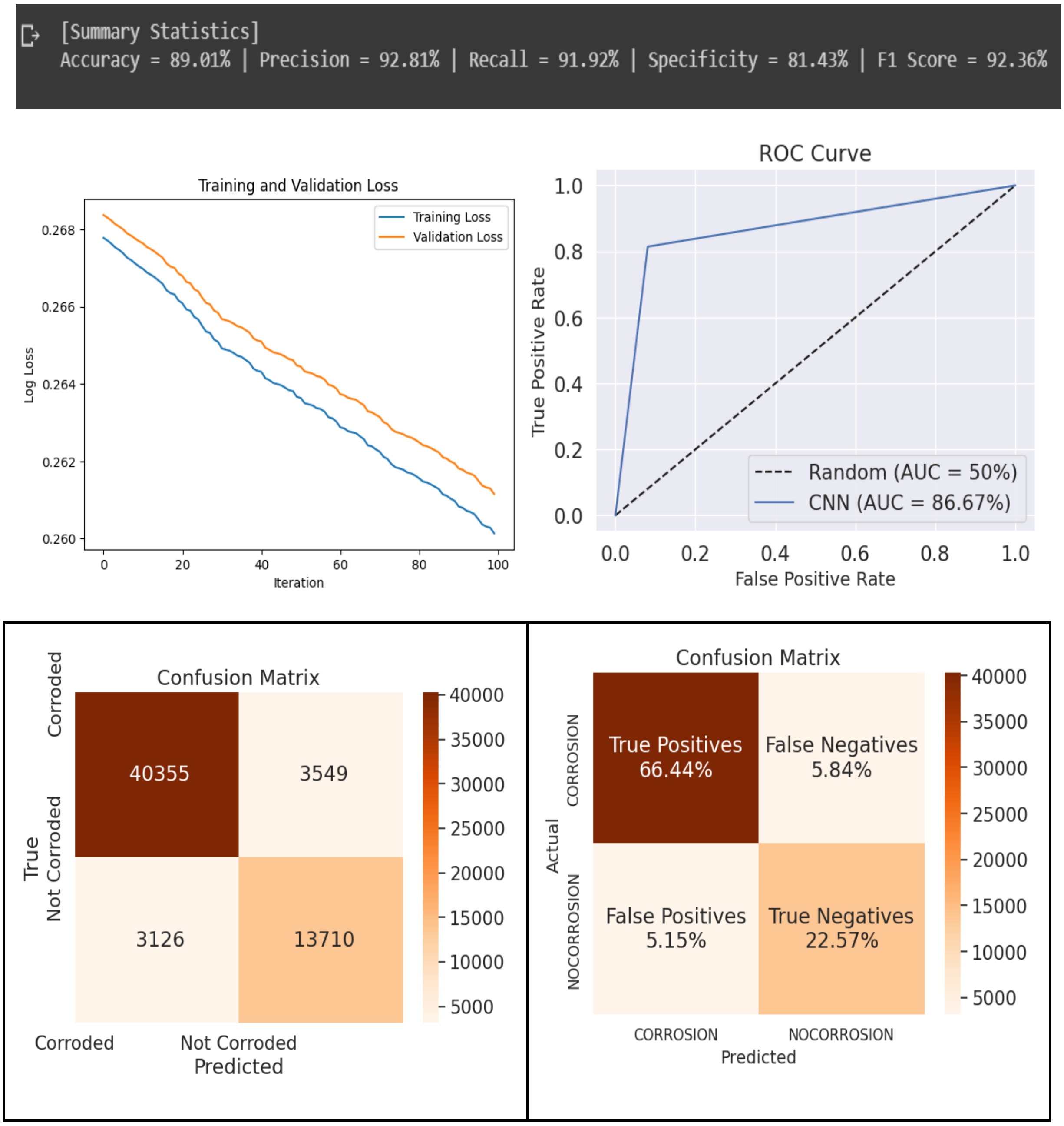


Fig. 9 Confusion matrix and other evaluation metrics of XGBoost color classification model.

3.2 Model Predictions and Discussion

The best-performing CNN model has a CNN AUC of 94.44%, which indicates that our best-performing CNN model does a great job at ranking the data by its class in the test set compared to the baseline model with a CNN AUC of 92.58% given in Figure 7 and 8.

If the predicted probability of corrosion is greater than 0.5, then a message indicating that corrosion has been detected is printed, along with the predicted probability of corrosion. If the predicted probability of corrosion is less than or equal to 0.5, a message indicating that no corrosion has been detected will be printed, along with the predicted probability of corrosion. The threshold of 0.5 used in this code may not be optimal for all situations and may need to be adjusted depending on the specific requirements of the application. The predictions of the best-performing model are given in Figure 10.

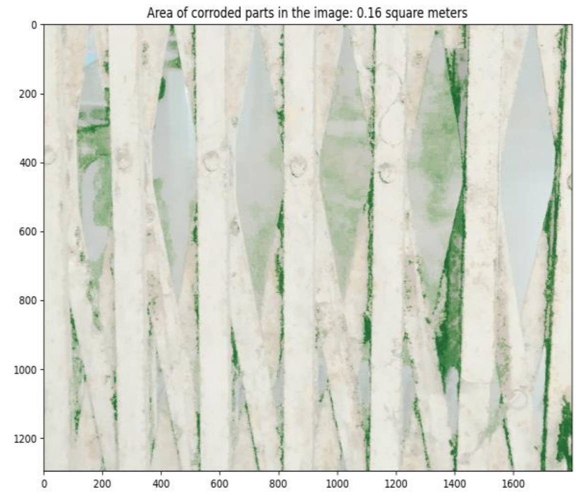
Fig 10 Predictions of the best-performing model

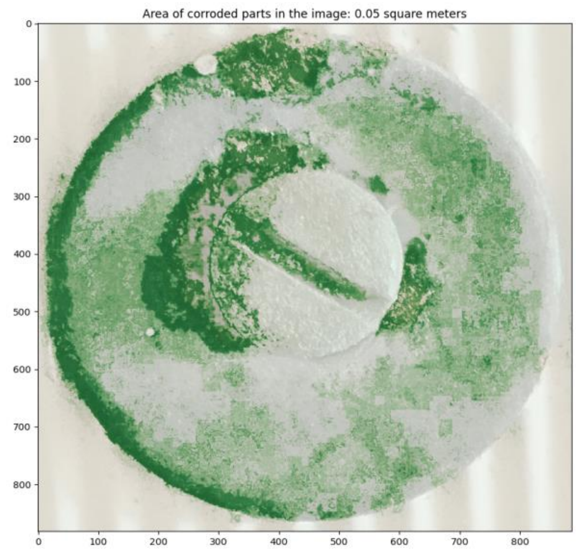
Weights, bias, and variances of the XGBoost color classifier model were saved and used later for visualizing corroded surfaces on the images taken from the Girl's hostel premises of the NITK Campus. Required pre-processing methods for the above model are applied and the 'model.predict()' method is used to classify pixels of sample images as 'Corroded / Positive' or Not Corroded / Negative'.

To find the area of the corroded surface in the given image, a number of pixels classified as positive were calculated using 'np.count_nonzero()' and were multiplied by the area of an individual pixel in a given image. For representational purpose size of an individual pixel is taken as 0.26 X 0.26 mm. Sample visualization is clearly represented in Fig

*Area of Corroded surface=Number of positively classified pixels*Area of individual pixel (8)*

To visualize the corroded surfaces python libraries like 'cv2' and 'matplotlib.pypolt' was used with 'plt.imshow(np.ma.masked_where(y_pred == 1, y_pred), cmap='Greens', alpha=0.7, vmin=0.5, vmax=1.0)' which develops a Green mask on the pixels classified as positive by the model. Sample visualization is clearly represented in Tables 1 and 2.





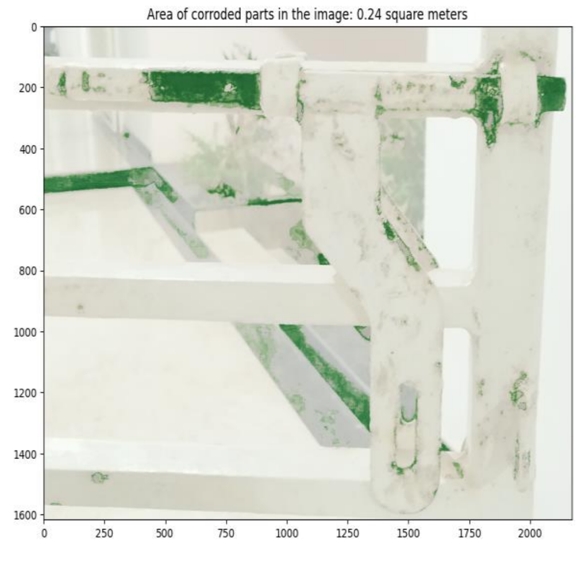
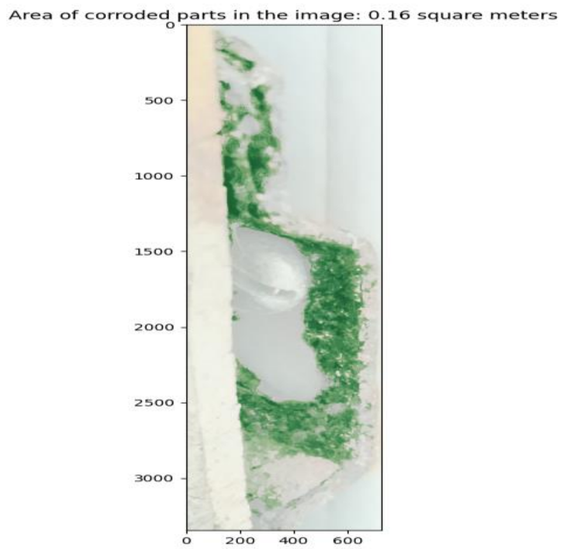
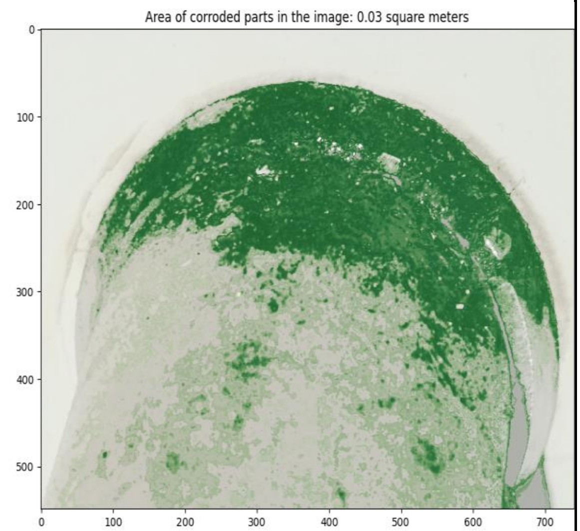


Table. 1. Visualizing corroded surface and retrieving area of corrosion using XGBoost (1)





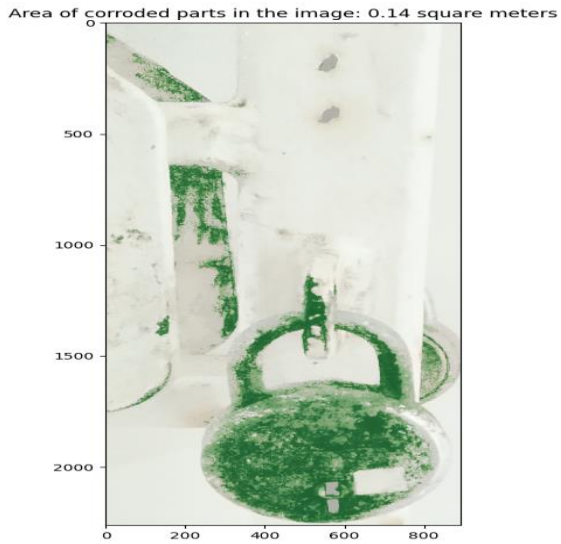


Table. 2. Visualising corroded surface and retrieving area of corrosion using XGBoost (2)

CHAPTER 4

CONCLUDING REMARKS

In this work 2 Convolutional Neural Network models were developed to detect corrosion in a given image sample and the study also found that four Conv 2D layers with dropouts are simple and effective in the classification of images. It also found that 128 pixels in one direction for image data used in training are sufficient for feature extraction by CNN models. The designed model was able to train and classify the images of steel into either “Corrosion” or “No Corrosion” categories. This model reported an accuracy of 94.44 % and an F1 score of 88.35% during testing whereas the baseline model was overfitting. Hence, 4 Conv2D layers with a drop-outs model, can be adapted to sort any new images into corrosion and no corrosion classes.

XGBoost color classifier model is used to find the area of corrosion. XGBoost (eXtreme Gradient Boosting) is a well-liked machine learning approach for classification and regression issues. A final prediction is generated by combining the predictions of various decision tree models. On a variety of machine learning problems, XGBoost has been demonstrated to produce state-of-the-art outcomes. This model demonstrated a ROC AUC of 86.67% and F1 score of 92.36% and is used for pixel-level color classification, Python in-built libraries like cv2 and ‘matplotlib.pyplot’ are used for visualization and to find the area of corroded surfaces on the image. The area of corrosion retrieved by the model differs based on pixel size which is dependent on the dpi (dots per inch) of the pixel. So, by taking the resolution of the camera used for taking image feed and the dpi of the image accurate area of the corroded surface can be retrieved. This study also found that Image resolution, Illumination, angle, and distance from an object during the capture of an image also affect the model training and its performance. To achieve a human level of accuracy we need massive and high-resolution image datasets and intensive image labeling. These results and findings will improve the development of deep learning models for detecting and visualizing specific features on corroded surfaces.

Convolutional Neural Networks and XGBoost Color Classifiers can also be used for crack detection on huge steel structures. This model can also be integrated with Mobile devices, Smartphones, and Drones. Hence, the model can be tuned/updated to determine the structural severity of corrosion. As a future work, the code used here can be edited to give accurate location/placement, area, and severity of corrosion in the structure. The area and severity of corrosion are important mainly for huge structures where manual inspection is difficult or cannot be done.

REFERENCES

[1] M. Khayatazad, L. De Pue a, W. De Waele Detection of corrosion on steel structures using automated image processing, Developments in the Built Environment (2020) Volume 3, 2020, 100022, ISSN 2666-1659,
https://doi.org/10.1016/j.dibe.2020.100022.

[2] Yamashita, R., Nishio, M., Do, R.K.G. *et al.* Convolutional neural networks: an overview and application in radiology. *Insights Imaging* 9, 611–629 (2018). https://doi.org/10.1007/s13244-018-0639-9

[3] Kuo-Wei Liao, Yi-Ting Lee, Detection of rust defects on steel bridge coatings via digital image recognition, Automation in Construction, Volume 71, Part 2, 2016, Pages 294-306, ISSN 0926-5805,
https://doi.org/10.1016/j.autcon.2016.08.008.

[4] Yang Tian, Guoteng Zhang, Kenji Morimoto, Shugen Ma, Automated Rust Removal: Rust Detection and Visual Servo Control, Automation in Construction, Volume 134, 2022, 104043, ISSN 0926-5805,
https://doi.org/10.1016/j.autcon.2021.104043.

[5] Mojtaba Khayatazad, Matthias Honhon & Wim De Waele (2022) Detection of corrosion on steel structures using an artificial neural network, Structure, and Infrastructure Engineering,
https://doi.org/10.1080/15732479.2022.2069272

[6] Koch, G., Varney, J., Thompson, N., Moghissi, O., Gould, M., & Payer, J. (2016). NACE International Impact Report: International Measures of Prevention, Application, and Economics of Corrosion Technologies Study. *NACE Int., Houston, TX, USA, Tech. Rep*.

[7] Zoran C. Petrović Serbian Armed Forces, Catastrophes caused by Corrosion, VOJNOTEHNI ČKI GLASNIK / MILITARY TECHNICAL COURIER, 2016., Vol 64, No 4 http://orcid.org/0000-0001-8053-4034

[8] Po-Han Chen, Luh-Maan Chang, Artificial intelligence application to bridge painting assessment, Automation in Construction, Volume 12, Issue 4, 2003, Pages 431-445,
ISSN 0926-5805,
https://doi.org/10.1016/S0926-5805(03)00016-5

[9] Bonnin-Pascual F, Ortiz A, Corrosion detection for automated visual inspection, Developments in Corrosion ProtectionPages 619 - 632
https://www.researchgate.net/publication/260006092_Corrosion_Detection_for_Automated_Visual_Inspection

[10] Jorge Merino, Ismael Caballero, Bibiano Rivas, Manuel Serrano, Mario Piattini, A Data Quality in Use model for Big Data, Future Generation Computer Systems, Volume 63, 2016, Pages 123-130, ISSN 0167-739X,
https://doi.org/10.1016/j.future.2015.11.024

[11] Aravind Kodandaramaiah, Mehdi E, AWS Machine Learning Blog, Rust detection using machine learning on AWS.
https://aws.amazon.com/blogs/machine-learning/rust-detection-using-machine-learning-on-aws/