Identification and Classification of Defects in Steel Sheets using Deep Learning Models

MSc Research Project
Programme Name

Akansha Bansal
Student ID: 18182615

School of Computing
National College of Ireland

Supervisor: Dr. Catherine Mulwa

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Akansha Bansal		
Student ID:	X18182615		
Programme:	Data Analytics	**Year:**	2020
Module:	MSc Research Project		
Supervisor:	Dr. Catherine Mulwa		
Submission Due Date:	17/08/2020		
Project Title:	Identification and Detection of Defects in Steel Sheets Using Deep Learning Models		
Word Count:	……………………………… **Page Count**……………………………………….…….		

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: ………

Date: ………

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Identification and Classification of Defects in Steel Sheets using Deep Learning Models

Akansha Bansal
X18182615

17th August 2020

Abstract

Defect detection is one of the critical parts of manufacturing of a product in an industry. Manually inspecting the bulk production of steel sheets for defect detection is a tiresome task and could be prone to human errors. When steel sheets are manufactured it came in contact with many heavy machines for the purpose of drying, cutting, rolling, heating due to this some defects occur on the surface. So, this technical report aims at developing a deep learning model which can identify the surface defects in steel sheets and classify those defects into various classes. Automating this task can improve the manufacturing standards of organization. For this purpose, four deep learning models has been implemented on Severstal dataset. The models are Xception, U-Net, Mask RCNN and UNet++ and their performance has been compared by Dice Coefficient. Among all the four models Xception achieved highest Dice Coefficient of 0.927. In addition to this, results of the previous used techniques for defect detected were discussed. Based on the identified gaps deep learning models were developed.

1 Introduction

Steel sheets are one of the important components for building materials. Manufacturing of the steel sheet is a complex process where steel sheet came in contact with heavy machines starting form heating, casting, drying to drying and cutting. During its processing some defects may occur on the surface on steel sheets such rust, crack, scratches and hole Sharifzadeh et al. (2008). Casting a steel into a sheet is a delicate process which causes changes into their physical property. Before delivering the product to the consumer, manually detecting the defects could be a tedious task and human work could be prone to errors. So, automating this task can help industries to improve their quality standards as well as improve the standards of steel manufacturing process Sun, Cai and Sun (2016). The research is focused on the implementation of different machine learning techniques to identify the location of defects and classify those defects into four classes.

1.1 Project Background and Motivation

From 16th century there has been many techniques developed for pressing the steel into sheets. However, since than some of the new manufacturing techniques does not use its resources sufficiently. Previously steel metal was hand hammered for producing the sheet. Then this manual work has been replaced by rolling mills were large iron cylinders were used for pressing the metal. In this process of flattening, metal tin is used to coat the iron cylinder and steel sheet to prevent them rust. But when the end product was examined it has shown the traces the rust which came under one of the classes of defects (A short sheet metal history - Metal Working World Magazine, 2020). It has also been seen that the steel sheet has been inspected manually for the final check. This manual check is an unreliable task as steel sheets are produced in bulk and there are some defects which cannot be seen from naked eyes.

Maintaining the quality of the end product is one of the significant aspects in the metal industry. With the increase in demand of the steel sheet by consumers which majorly emphasizes on quality, supplier not only has the pressure to increase the production of product but also to maintain the quality level of steel sheet Wu, Xu and Xu (2007). In recent years most of the manufacturers are using hot rolling mills for pressing the metal which may cause several defects such as spots, edge cracks, scratches, flanges and these types of defects occurs when metal has been passed from one process to other such casting, rolling, pressing, heating etc. Saridis and Brandin (1979). If these defects are not detected on time or before delivering to customers it may cause huge loss in economy and effects the reputation of the organization. Unfortunately, it has been seen that today's technology it not completely advanced for detecting the defects and not feasible for real life cases Li, Tso, Guan and Huang (2006). There are disadvantages of partially automated and manual process for detecting the defects such as:

- Manually checking the defects on steel sheets could be a tiresome task and results could be unreliable. Different people have different approaches for inspecting a product, thus causing the inconsistent standard for quality check.
- Partially automated process is still prone to errors as these measures are incorporated between the manufacturing process because it can only detect one single kind of defect and could affect the order performance of remaining processing steps. Thus, there is an urgent need of a technique which can detect the multiple kinds of defects.

Thus, automating the process can help the organization to maintain the standard of their production and it will help waste reduction. Industry can also recycle the defected product which will maintain the ecological balance and sometimes minor defected products could be used for secondary products.

With the tremendous growth in the field of image recognition where many machine learning models has shown state-of the art results. Convolution Neural Network (CNN) has also shown impressive result in metal industry like classifying the types of metal shape and size, segregating the metals from the dump, identification and classification of textures in metal, identification of cracks in metals, detection of defects etc. It has been seen not much research has been done in fully automating defect detection process of steel sheets so, machine learning models can be used. As it has an ability to learn from the given knowledge and could also understand the processing of steel. So, the goal of this technical report is to implement different deep learning models to identify and classify the defects on steel sheets. The performance of the models is compared based Dice Coefficient.

1.2 Research Question

The manufacturing of steel sheet is a complex task where sheet has been touched by several kind of heavy machines which could be the major reason for the defects. The dataset contains the images of steel sheet which were taken from high infrared camera to improve the detection algorithm. The technical report aimed at identifying the defects on steel sheet and classify those defects into four classes. The dataset contains four class IDs where each id represents one kind of defect.

RQ: "To what extend can deep learning models (U-Net, Mask R-CNN, UNet++ and Xception) enhance/improve identification and classification of steel sheet defects to help the industry in automating the process of defect detection?"

Sub RQ: "Can the classification of the defect into four class in steel sheet help the industry to maintains the quality of the product as well as reduction of waste?" Automating the task can help the manufacturing industry in maintain the quality of the product and also helps in the reduction of waste as the defected steel sheet could be reused for the secondary purpose.
To solve this research question following objectives are specified and implemented in this technical report.

1.3 Research Objectives

Table 1: Research Objectives

Objectives	Description	Evaluation Metrics
Objective 1	A critical review of the present literature work done in the same field or similar field	
Objective 2	Steel sheet defect detection methodology approach design used	
Objective 2.1	Modified methodology approach used	
Objective 2.2	Architectural and process flow diagram	
Objective 3	Deep learning models used for steel sheet defects	
Objective 3.1	Data Exploration and pre-processing	
Objective 3.2	Implementation, Evaluation and Result of Xception	Dice Coefficient
Objective 3.3	Implementation, Evaluation and Result of U-Net	Dice Coefficient
Objective 3.4	Implementation, Evaluation and Result of Mask R-CNN	Dice Coefficient
Objective 3.5	Implementation, Evaluation and Result of UNet++	Dice Coefficient
Objective 3.6	Comparison of developed models	

1.4 Contribution

The major contribution of this project is to develop a deep learning model which can effectively use to detect the defects on steel sheet during its manufacturing. The goal of this research is to improve the standards of production in industry. The developed models will identify the defect as well as classify the defect into various classes which will help industries to understand more about the manufacturing process (reason for the frequently occurring defect).

The rest of the technical report is structure as follows. Section 2 is a review of the related work surrounded to techniques used for defect detection in steel sheet, Image Segmentation and Semantic Segmentation, Section 3 represents modified methodology used for defect detection in steel sheet, Section 4 represents the implementation, evaluation and results of four deep learning models and finally Section 5 represents the conclusion and future work summary.

2 Literature Review on Defect Detection on Steel Sheet

2.1 Introduction

Detecting the defects on steel sheet is a serious problem and many researchers work is an inspiration to solve this complex problem by using different approaches. This section of the technical report investigates about the different approaches and their accuracy taken to defect the defect in metal. The literature review is dived into subsection which are: 2.2) Literature Review of Techniques used for Defect Detection and Identified Gaps 2.3) Instance Segmentation and Semantic Segmentation for the Images of Defect Detection 2.4) Comparison of Reviewed Techniques in Steel Sheet Defect Detection 2.5) Conclusion.

2.2 Literature Review of Techniques used for Defect Detection and Identified Gaps

Over the past few years there has been attempts made to use different machine learning approaches to automate this procedure of defect detection. The algorithms which were used to identify the defects are called image detection algorithm. And it is further dived into two parts image classification and image localization. In image classification, objects of the image are classified into different classes and in image localization boundary of the important objects from the image are predicted. Li, Su, Geng and Yin (2018) has tried to detect defects on steel strips with the help of improved You Only Look Once (YOLO) network for all convolution. The dataset contains the six types of defects of cold-rolled steel strip. The six types of defects were scar, scratch, inclusion, burr, seam and iron scale. The paper describes an end to end solution for detecting the defects where the network consists of 27 layers. The network achieves 99 percent detection rate. However, the research is implemented on small size dataset and not implemented on variety of steel strip images. Also, the location and scale of the defects has shown little accuracy. Sharifzadeh et al. (2008) has investigated on detection and classification of steel surface defects. The image processing algorithm is applied on four kinds

of defects which are hole, scratch, coil break and rust. This method was implemented on 250 steel defect images and each defect is treated differently. The result shows of accuracy of 88 percent, 78 percent, 90 percent and 90 percent for hole, scratch, coil break and rust defects respectively. The implementation is not feasible to real life scenario as there could as algorithm can defect only defect in a single time. So, placing this method for defect detection in the organization which has bulk production steel sheet can be very time consuming.
Zhou et al. (2017) has improved the CNNs by initializing its kernel with the learned filter kernels which makes a feature vector and single multi-class SoftMax regression learned simultaneously. This proposed method is simple, effective and robust for classifying the hot rolled steel sheet defect. The experiment was performed on small dataset which achieves moderate level of accuracy but the average classification accuracy can be up to 99 percent. The proposed method can be applicable on different kinds of steel sheet such cold-rolled, galvanised sheet, silicon sheet etc. Kun Qian. (2019) has implemented series of machine learning algorithms of semantic segmentation and neural networks with encoder and decoder architectures using U-Net and feature pyramid network (FPN). The experiment has performed well and achieved F1 score over 0.915 and 0.905 at a speed of 1.5 images per second. The proposed ensemble algorithm model improves the efficiency, quality and saves labour cost. However, the proposed model requires complex processing which increases computational cost. In the future faster neural network models can be adopted which reduces the computational cost. Luo and He (2016) has developed the cost effective AOI system for hot rolled steel. The research follows the following steps: firstly, a detailed topology is constructed with the lighting setup then image enhancement method is designed to for ruling out uneven lighting and over-or-under exposure. Thirdly defect detection model is developed and, in the end, developed algorithm is implemented. The proposed is effective and also quick with an accuracy of 92.11 percent and 5.54 percent of false-negative rate. However, this method does not classify the defect and also the quality grade estimation.

He, Song, Meng and Yan (2020) has developed deep learning system for defect detection steel plate. The proposed method uses CNN which generates feature map on each step and then introduce multilevel-feature fusion network which can includes number of defect locations. Based on that region proposed network is adopted to generate region of interest (ROI). ROI there is a detector which consists of a classifier and a bounding bond regressor and it outputs final result. The experiment achieves 99.67 percent of accuracy and detection speed is 20fps. However, the data augmentation computation is very expensive and does not perform precise defect boundary. Liu, Xu and Xu, (2019) has developed the defect detection technique for periodic defects. The method uses the combination of CNN and LSTM for defects like roll marks for strong time-sequenced characteristics of such defects. The detection rate is 81.9 percent which 10 percent higher than CNN. The method has been compared between three methods which are VGG16, VGG16+LSTM and VGG16+LSTM+Attention. Though the attention mechanism increases the accuracy to 86.2 percent but it also increases the complexity. The work claims to use more powerful feature extractor for training the samples. Song, Yan and Hu (2013) has studied the steel defect recognition method using scattering convolution network method (SCN). SCN was very effective in building the large-scale variant and to extract the features of defect recognition. The SCN method achieved good recognition accuracy even with the least number of training. For this study, the North-eastern University (NEU) surface detect database is constructed. The study shows that the SCN method presents highly accurate performance even under the presence of feature variation.

2.3 Instance Segmentation and Semantic Segmentation for the Images of Defect Detection

Instance Segmentation is used to identifies the boundary of the object from the image in a detailed pixelwise level. Bukharev et al. (2018) has develops the method for instance segmentation for mineral grains in thin section image of sandstone. The challenges faced by researcher is that the grain particles were too small to give clear boundary. To solve the problem two fully-convolution neural network algorithm were used. The first model is used to specify the location of the objects and second model is used mask the identified location. The two models were cascaded which allows to construct the algorithm for small training samples. Highly accurate results were achieved and high quality of prediction increase the size of samples. Paste and Chickerur (2019) has implemented Mask R-CNN on entertainment data. The experiment was performed for instance segmentation on training data. The research paper claims huge amount of data between 1000 to 2000 images in training set improves the capability of instance segmentation to identifying the boundary of objects from any kind of image.

Semantic Segmentation is used to label each pixel of the object to the corresponding class. It is also known as pixel level image classification. Liu, Deng and Yang (2019) has divided the process of semantic segmentation into ways that are: deep neural network and traditional method. Traditional method used to extract feature from the image dataset. The features use SURF, pixel colour and histogram. On the other side, Deep neural network has the ability to shift weights based on translation invariance characteristics. The process has achieved good level of results in segmentation, classification and detection. Roberts et al. (2019) has demonstrated the feasibility of automatic identification system for detecting the three kinds of crystallographic defects. The three kinds of defects are: precipitates, dislocations, voids in steel metal. The author uses hybrid model of CNN and DefectSegNet. The data has been trained by inputting defect image using pixelwise semantic segmentation. This method not only not improves the model performance but also increases the accuracy.

2.4 Comparison of Reviewed Techniques in Steel Sheet Defect Detection

Comparison of techniques, model and classifiers are obtained in table 2 Li, Su, Geng and Yin (2018) has attained the accuracy of 99 percent in defect detection techniques by using the improved YOLO model. However, this research is only implemented on small size dataset and thus not feasible to adopt in real-time. Sharifzadeh et al. (2008) has achieved the mean accuracy of 88 percent but thus method can only detect one defect at a time. Kun Qian. (2019) has implemented U-Net and FPN with the F1-score of 0.91 and 0.90. This model requires complex processing and requires lot of computational cost.

Table 2: Comparison of Literature in Steel Defect Detection

Techniques	Compared Results	Authors
YOLO	99%; Technique only implemented on small size dataset	Li, Su, Geng and Yin, (2018)
Segmentation, Gaussian Function and Renyi Entropy	88%; Can only one defect at a time	Sharifzadeh et al. (2008)
U-Net and FPN	F1-score 0.91 and 0.90; requires complex processing which increases computation cost	Kun Qian. (2019)

2.5 Conclusion

From the reviewed literature work it can be observed that the reviewed results have identified gaps and gives the clear evidence for the need to develop a model which is feasible to implement in the industry and improves their standard of manufacturing. This leads to the answer of research question section 1.2 and research objectives 1. The next section represents the modified methodology used for steel sheet defect detection which throw the light on how to develop the deep learning model in order to provide the feasible results.

3 Steel Sheet Methodology Approach and Design Specification

3.1 Introduction

This chapter represents the data mining methodology used. In this research report Knowledge Discovery in Databases fits very well for the implementation of models. The motivation behind this technical report is to bring insights to business so that it can improve the standards of manufacturing. A three-tier architecture approach represents the steps taken to and the technology for its implementation.

3.2 Steel Sheet Methodology Approach

Project planning is an important stage for implementing a project. In this research project Modified KDD methodology (refer figure 1) is used for steel sheet defect detection which consists of five stages. Following describes each stage of this methodology with the aspect of localization and classification of defect detection.

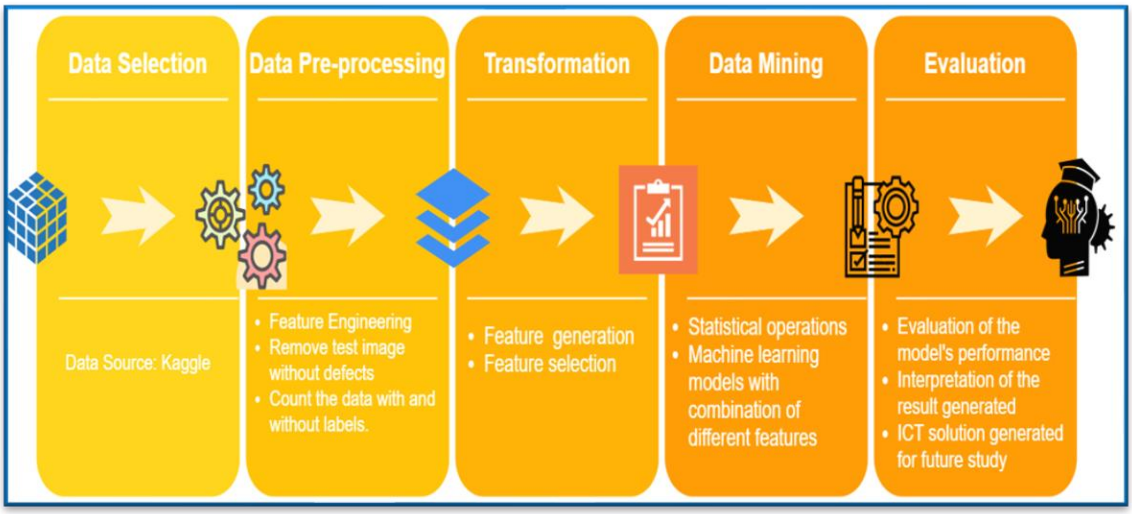

Figure 1: Methodology of defect detection in steel sheets

Data Selection: The dataset has been downloaded from Kaggle. It consists for high frequency camera images of steel sheet. The dataset set consists of 12, 568 images and the size of the image is 1600x256. All the images are of high quality. Figure 2 is the sample of the image of the defected steel sheet with their respective labels.

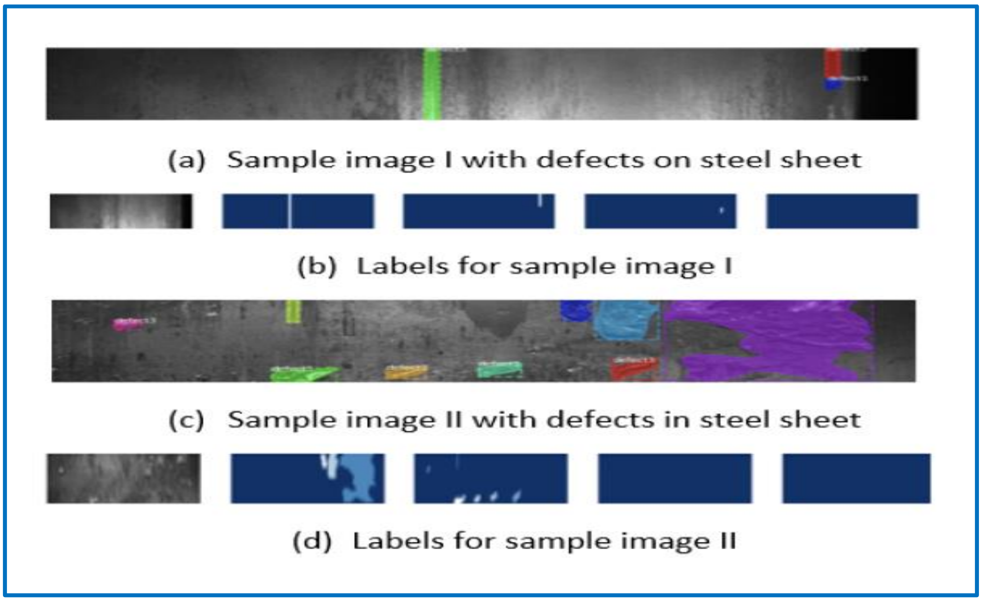

Figure 2: Sample images of steel sheets with with defects and respective labels

Data Pre-processing: In this phase dataset cleaning has been done in Google Collaboratory using Python. The image id and class id columns were separated into two entities. The images were examined based on how frequently each class is labelled, determines images with and without labels and size of defects per class.

Transformation: The features has been extracted from the grey scale image and are transformed by feature vectors. Also, the concept of running length encoder is used which determines the size and location of the defect.

Data Mining: In this phase deep learning models are applied on the dataset for the identification and classification of steel sheet defect detection. The models are U-Net, Mask R-CNN, Xception and UNet++.

Evaluation: Models are interpreted and evaluated based on Dice Coefficient. Also, best fit model is selected for the real-time application.

$$\text{Dice Coefficient} = \frac{2.|X \cap Y|}{|X|+|Y|} \tag{1}$$

3.3 Design Specification

This technical report of identification and classification of defects in steel sheets using deep learning models is designed using three tier architecture (figure 3). The design shows different technologies used in the implementation of the project and consists of three layer which are: Data Persistent Layer, Business Logic Layer and Client Layer.

First tier describes the processing of the dataset which includes the source from where the data is downloaded, exploratory data analysis done on the data so that image can be transformed for the identification of the defect. Graphics Processing Unit (GPU) used to coding and different libraries used such as 'Keras', 'plotly', 'datagenerators' etc. The programming language used is python. The second tier describes about the different deep learning models used for the classification of the defects. And, third tier is used for visualizing the results of the different models and accuracy of each models were measured.

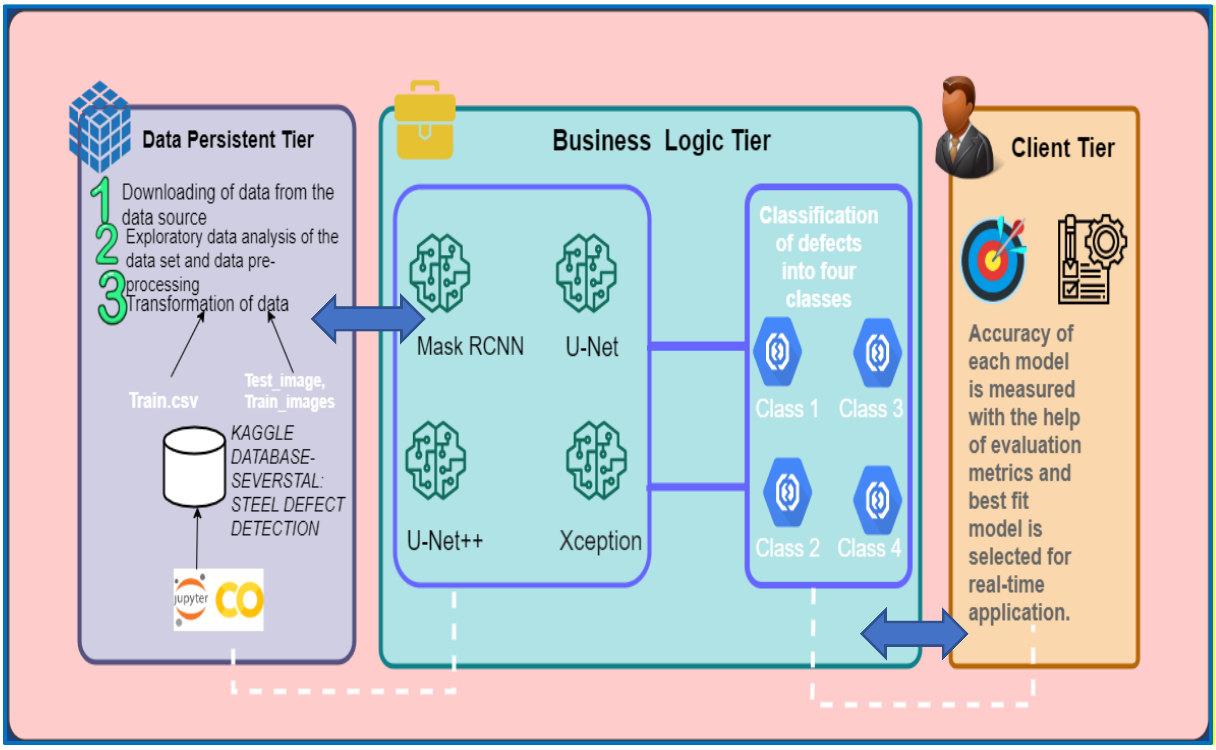

Figure 3: Defect detection architecture design

3.4 Conclusion

The steel sheet methodology has been chosen as the implementation plan for this research work. The methodology has been modified according to the need of the project. This modified methodology is used for process flow diagram and based on that three-tier architecture design is created which depicts its implementation. Hence, this section marks to completion of research objective section 2. The implementation, evaluation and results for the classification of the defect detection is continued in the next section.

4 Implementation, Evaluation and Results of Deep Learning Models for Steel Sheet Defects

4.1 Introduction

This section of the technical report is used to describe the implementation, evaluation and result of the deep learning models used in identification and classification of defects in steel sheets. Along with the implementation of models, exploratory data analysis has been done on the image dataset which explains about the nature of the data as well as gives the interesting insights about the data which is described in detail in this section. For the evaluation of the Dice Coefficient is used. The chapter later represents the comparison with of the results of different deep learning models and best fit model is selected.

4.2 Data Exploration and Pre-processing

Data pre-processing and exploratory data analysis is a fundamental objective to understand about the nature of the data. Through this important characteristics, variables and features can be visualized. This part of the technical report fulfils the research objective section 3.1. Steel is one of the important components for building an infrastructure. Through this project, the production of the steel could become more efficient by automating the process of identifying the defects. For conducting the data pre-processing and exploratory data analysis, data has been downloaded from Kaggle. The dataset consists of 12,568 images of defected and not defected steel sheet. All the images were captured from high infrared camera. There are 6,666 images are without defects and rest are with defects. All the images are labelled with classes. Each class represent four kind of defects. Through the analysis it is known that 897 imaged has type class 1 defect, 247 type class 2 defect, 5150 images have type class 3 defect and 801 images has type class 4 defect. Dataset also contains the images which has more than one of kind defect. For that 6,239 images have only one type of defect, 425 images have two types of defects and two images has three types of defects. The implementation has been done in Google Collaboratory using python library. Initially four classes of defects have been seen on steel sheet with the mask over the defect.

4.2.1 Visualizing the Mask and the Images

The steel sheet images shown in figure 4 gives an idea how each type of defects looks like as well the understanding about the mask. Through the analysis, count of the images has been found out for the missing labels and also the frequency of occurring of each defect is checked.

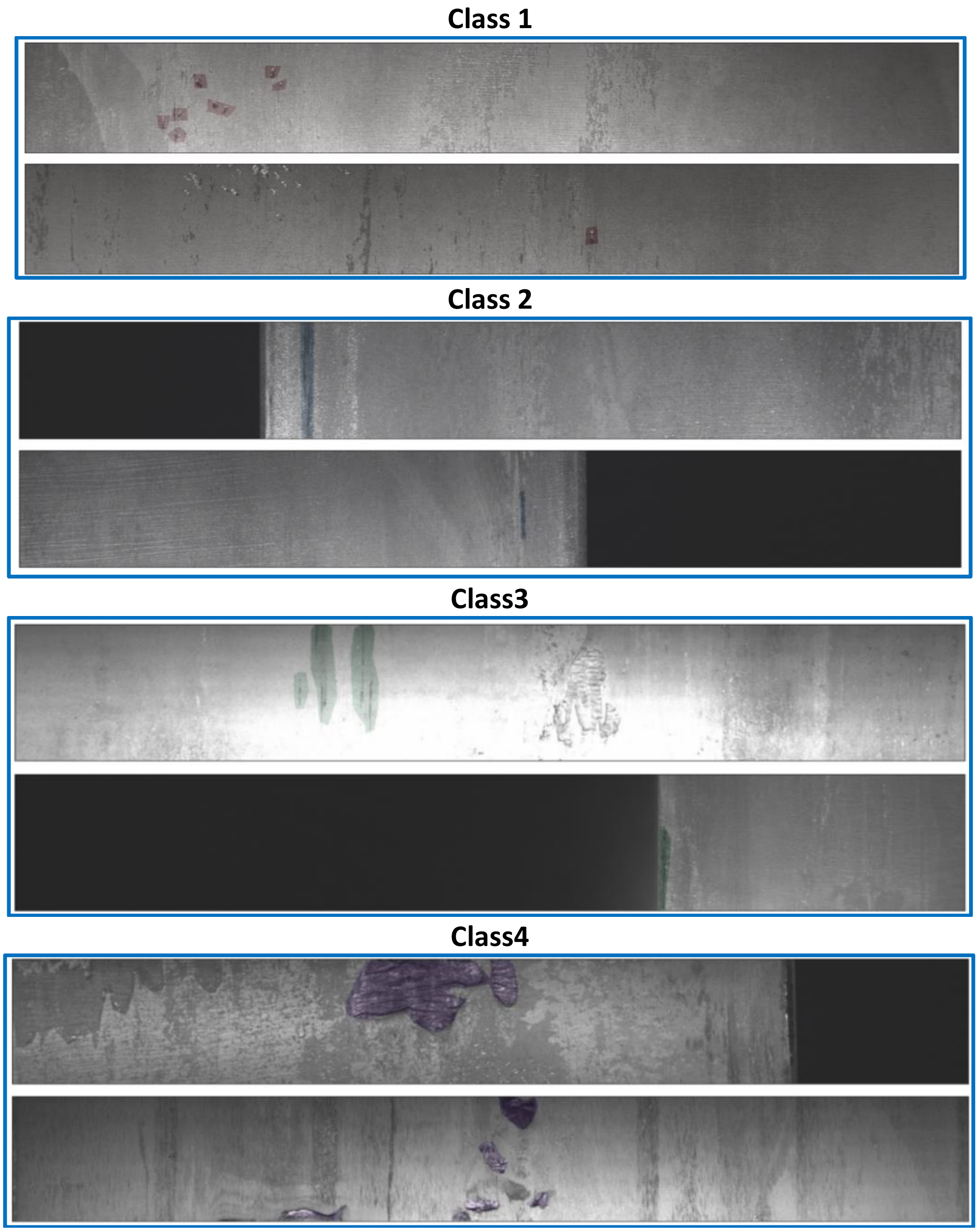

Figure 4: Class wise defects on steel sheet images

4.2.2 Defects per Image and Missing Labels

Through the analysis figure 5 it has been observed that there is no image without the label and most of the images has only one kind of defect and very few images have two and three classes of defect in a single image. It can be concluded that in the dataset labels are not missing for any of the images.

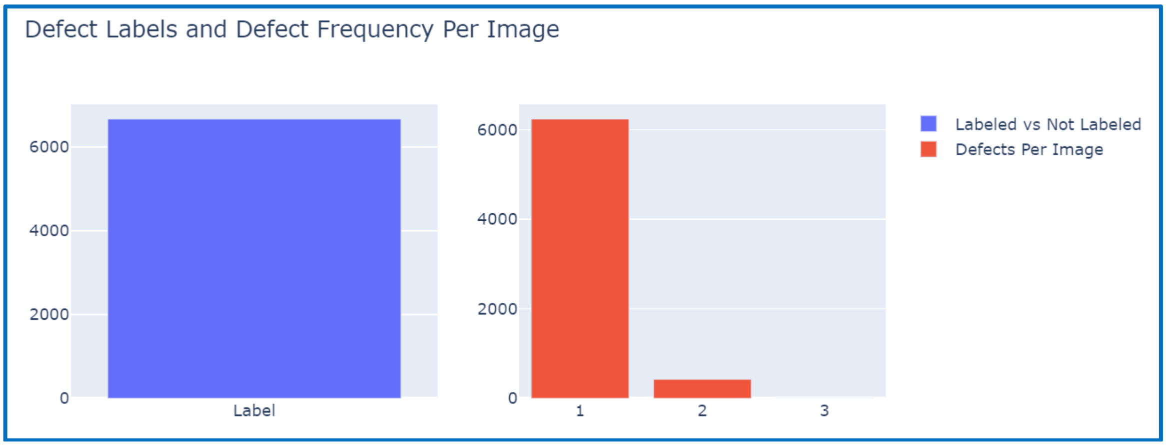

Figure 5: Defect labels and defect frequency per image

4.2.3 Mask Size and Class Wise Defect

In this analysis binary masks for the representing the defect were used where 0 represents no defect and 1 represents defect. So, by counting the number of pixels of a mask can gives an understanding of the size of the size of the defect bases on its class. It is observed from the figure 6 that approximately 73 percent of defects are from class 3 following 13 percent to class 1, 11 percent to class 4 and 4 percent to class 2. Whereas, from the perspective of total area occupied by each defect it is seen through the analysis that class 4 has the area of 17 percent which denotes that usually class 4 defects are larger in size on steel sheet.

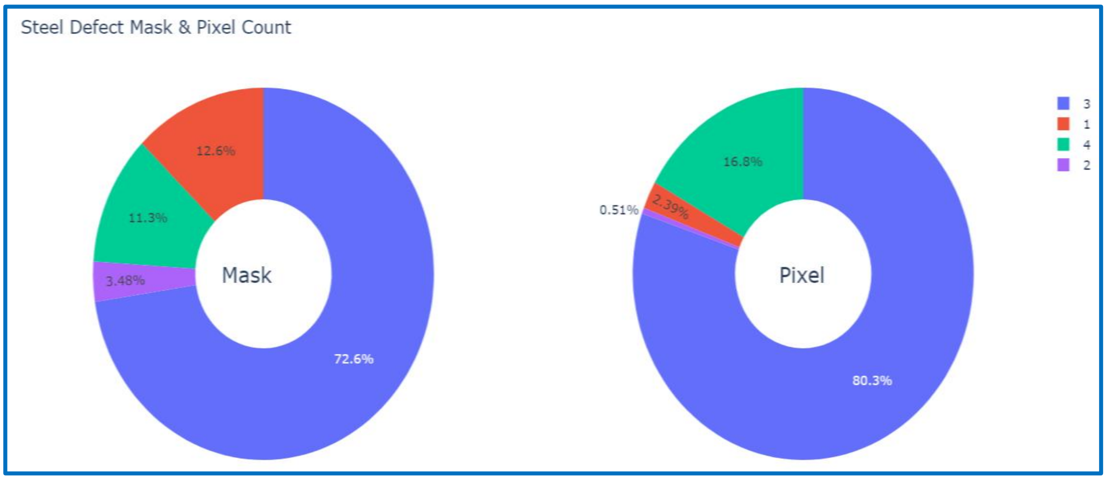

Figure 6: Steel defect mask and pixel count

4.2.4 Segments per Defect Type

In this phase, number of defected regions on steel sheet were counted for a mask. Through this it can visualize multiple regions of the defect in an image. The analysis throws the light on the behaviour of segments for different classes of defects in an image. Figure 7 shows the number of segments of defects in steel image whereas figure 8 and 9 shows number of segments with respect to mask size based on class id and mask pixel size.

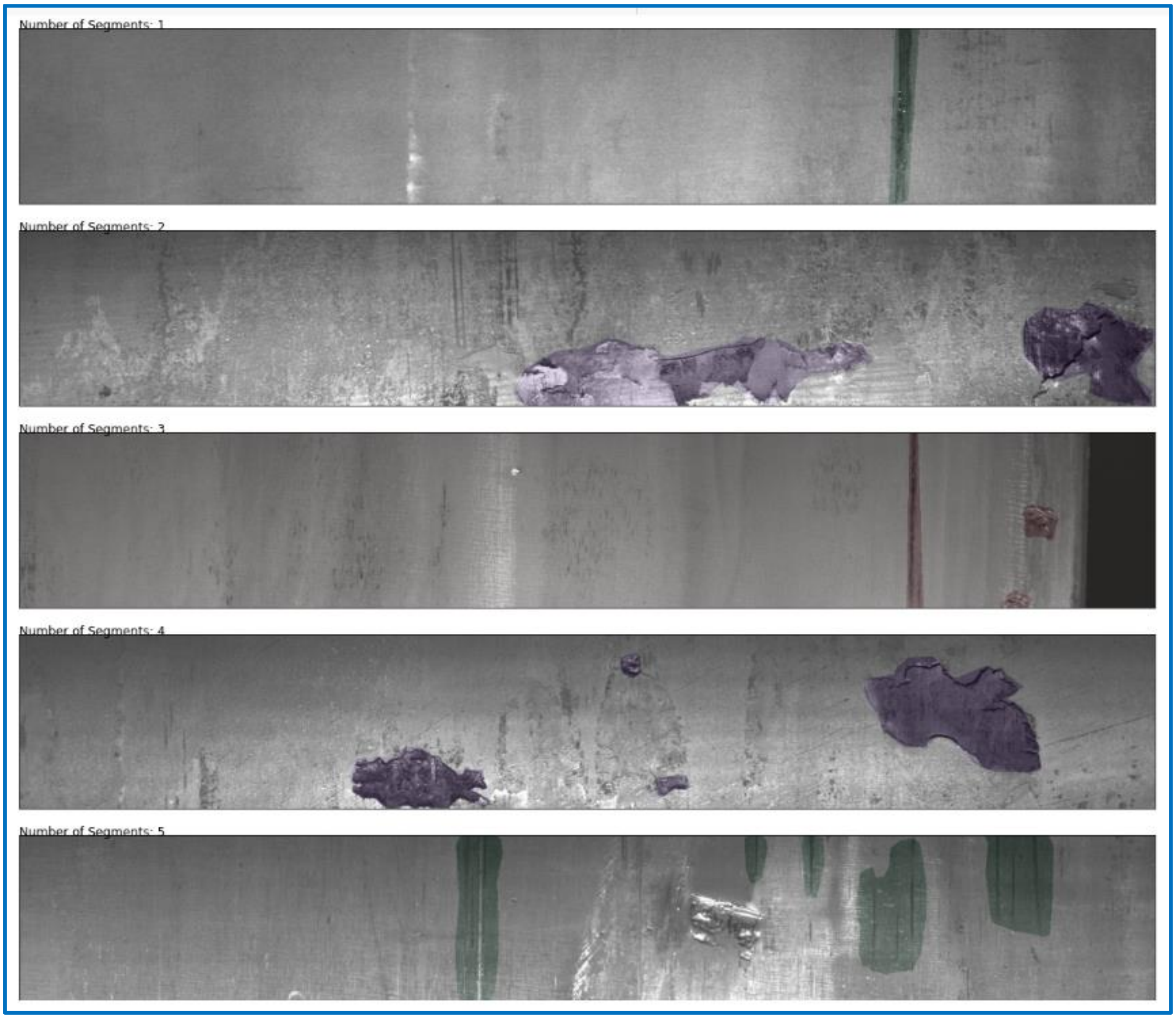

Figure 7: Number of defect segments in steel sheet image

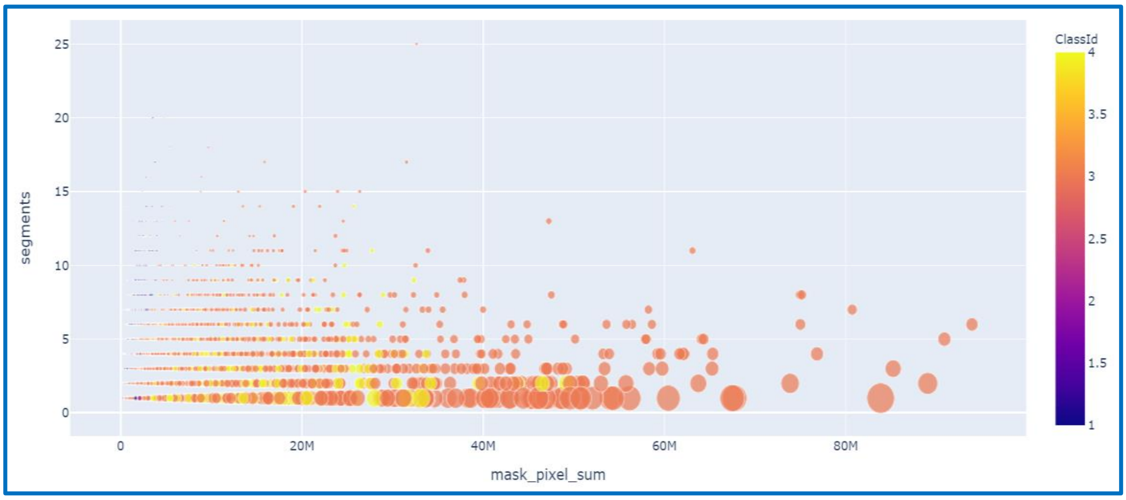

Figure 8: Number of segments vs mask size based on class id

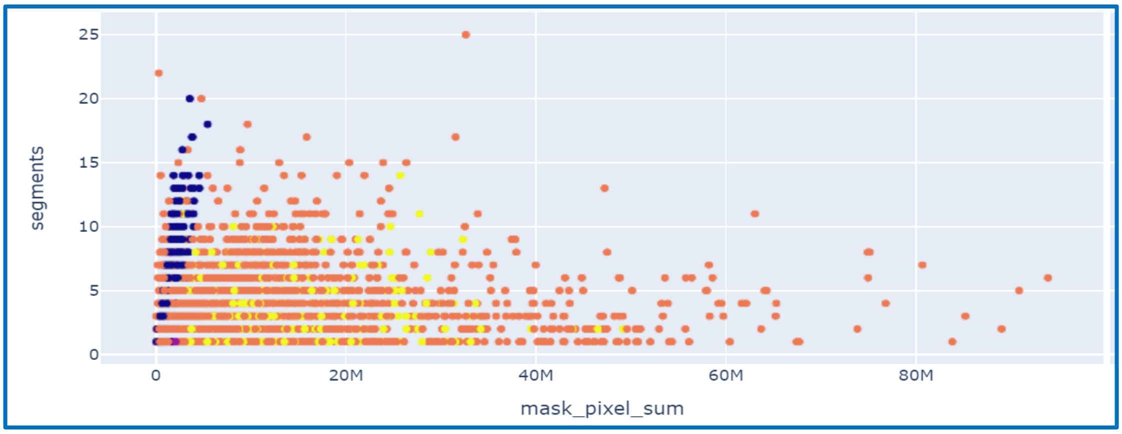

Figure 9: Number of segments vs mask pixel size

4.2.5 Steel Defect Segment Count

It is observed from the figure 10 that class 1 defects are usually smaller in size and has many segments. It has the highest percentage of more than 5 segments and overall has small defect area on steel sheet. Class 2 does not have many segments and small in size. It has one or two segments in an image. Class 3 and class 4 has the frequency of more than three segments.

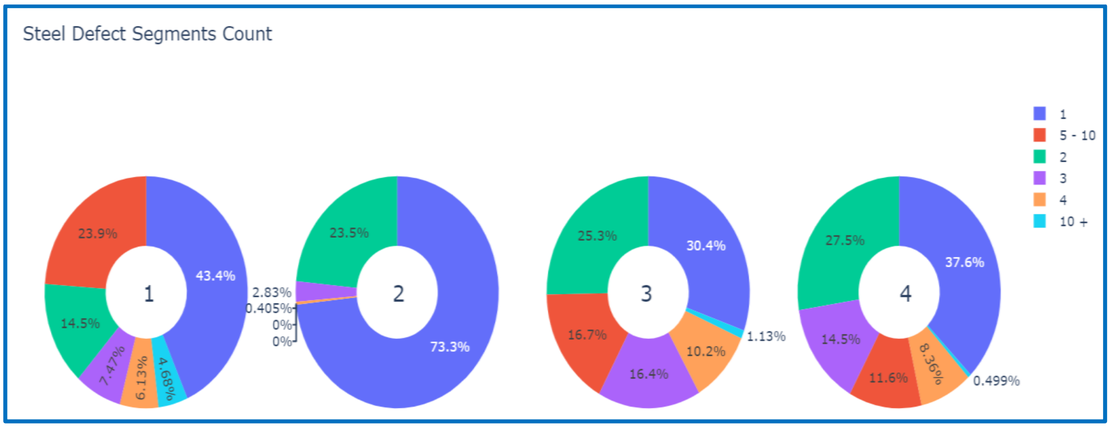

Figure 10: Steel defect segments count

4.2.6 Frequent Pattern Mining

In the dataset there are many images with more than one class of defect which brings an analysis of how frequently every class of defect occur. For its purpose frequent pattern mining is implemented which outputs the pattern of types of defects in pair. It is observed from the figure 11 that the occurrence of defect class 3, 1 and 4 is the most frequent pattern. Combination of class defect 3 and 4 is more common than alone class 2. And the frequency of combination defect class 3 and 1 is less frequent which is less than 10 percent.

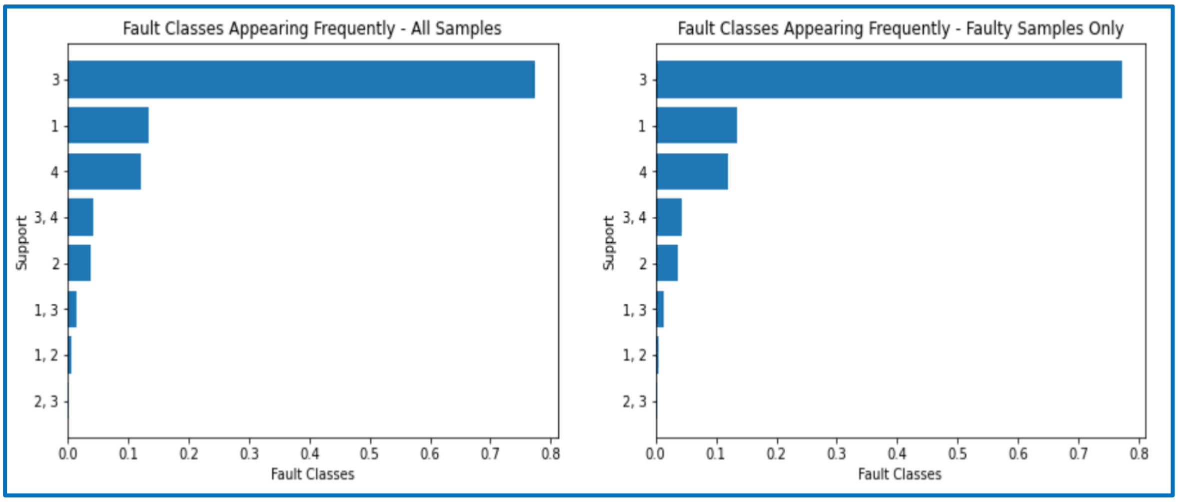

Figure 11: Frequent pattern mining bar chart based on class id

4.3 Xception Model

Exception model was developed by Google, which means extreme version of Inception model. It is convolution neural network model which has 71 layers and involves Depthwise Separable Convolutions. The Depthwise Separable Convolutions were modified into pointwise convolutions. The order of operation has changed which first performs 1x1 convolutions followed by channel-wise spatial convolutions. And this model does not have intermediate ReLU non-linearity due to which it has high accuracy. Its architecture has three flows: entry flow, middle flow and exit flow. First the data enters through entry flow and goes through middle flow for the repeated eight times and finally enters exit flow.

4.3.1 Implementation

Through the EDA it has been seen that data is imbalanced, due to this right training data needs to be fed in the model. Binary classifier is trained for all the images in the data set to classify whether the image is defected or not, whereas Multi label classifier is used only on images with defects. Multi label classifier has five output images where four output neurons represent four classes of defect and fifth neuron represents "no defect". The four segmentation models were trained for each type of class. Hence, six model architecture is created to implement the Xception model. The model has trained over 30 epochs. For pre-training the Xception model for classification efficientnetb1of U-Net architecture is used for segmentation. The model uses the library such as 'ImageDataGenerator', 'Keras', 'TensorBoard' and 'segmentation_models' in the python using Goggle Collaboratory. 'TensorBoard' is used to check the performance of the model in each epoch.

4.3.2 Evaluation and Results

Dice Coefficient is used as an evaluation metrics. The data set has been divided into three parts train, validation and test with the ratio of 72:18:10. Data generators library is used for pixel value scaling for model training. The model achieved Dice Coefficient of 0.944, 0.982, 0.804 and 0.978 for defect class I, II, III and IV respectively as shown in figure 12. The mean average of dice coefficient is 0.927 as shown in figure 13. From the figure 12 it is observed that the model outputs high precision value for multi label classification and high recall value for binary classification. Hence, this marks the completion of research objective section 3.2 where model has successfully performed.

```
Found 1257 validated image filenames.
--------------------------------------------------
Mean Dice coefficient of each defect: [defect_1, defect_2, defect_3, defect_4]
[0.9442220594192524, 0.9826448541766108, 0.8047021536276849, 0.9787651186873508]
--------------------------------------------------
Classification Report:
              precision    recall  f1-score   support

 hasDefect_1       0.85      0.49      0.62        91
 hasDefect_2       1.00      0.20      0.33        25
 hasDefect_3       0.98      0.75      0.85       524
 hasDefect_4       0.97      0.89      0.93        81
    NoDefect       0.80      0.99      0.88       590

   micro avg       0.87      0.84      0.85      1311
   macro avg       0.92      0.66      0.72      1311
weighted avg       0.89      0.84      0.84      1311
 samples avg       0.87      0.85      0.86      1311

--------------------------------------------------
```

Figure 12: Dice Coefficient of each defect class

```
--------------------------------------------------
Mean Dice Coefficient (Overall): (Competition Metric)
0.9275835464777248
--------------------------------------------------
CPU times: user 6min 40s, sys: 52.3 s, total: 7min 32s
Wall time: 6min 9s
```

Figure 13: Mean Dice Coefficient

4.4 U-Net Model

U-Net is a Fully Convolutional Network model. This model is more accurate in terms of pixel-based image segmentation. U-Net used to convert image into vector by using feature mapping and uses same mapping to convert it into image which maintains the structure integrity of an image. Its architecture consists of three parts: the contraction, the bottleneck and expansion section. The contraction section is used to learn more about the features of the image as it consists of 3x3 convolution layers followed by 2x2 max pool layer. In expansion layer important features of the image are learned while reconstructing the image. This layer consists of 3x3 CNN layer followed by 2x2 un-sampling layer. It uses loss weighing scheme where higher weights are at the border of the segmented objects which causes to classify each pixel as one class.

4.4.1 Implementation

The U-Net model has been modified for the implementation of defect detection Ronneberger, Fischer and Brox (2015). The architecture has 3x3 convolution boxes which has padding and rectifies linear unit (ReLU) averaging along with 2x2 max pooling layer. Adam is used as an optimiser with the learning rate of .0001 and binary cross entropy is used for loss function. For training the model, batch of 32 images were randomly fed. U-Net is implemented using ‘cv’, ‘keras’ and ‘sklearn’ library in the python using Google Collaboratory. The data is split into test and train and validation set were test size is 15 percent. The model uses running length encoding mask to better understand the pixels of the defected area. To load the large number of files in batches data generator class has been is instantiated using keras.utils.Sequence where ‘len’ returns the number of batches per epochs and ‘on_epoch_end()’ function shuffles the indices to create a robust model.

4.4.2 Evaluation and Results

The evaluation of the is done using Dice Coefficient. The model was trained over 7 epochs where a final dice coefficient is 0.6868. The loss value of the validation set is similar to that training set which means that model is well adjusted with other sets. However, the dataset is unbalanced where it predicts defects of class IV which has the majority in dataset. Hence, this section marks the completion of research objective section 3.3 where model has successfully performed and result id shown through figure14.

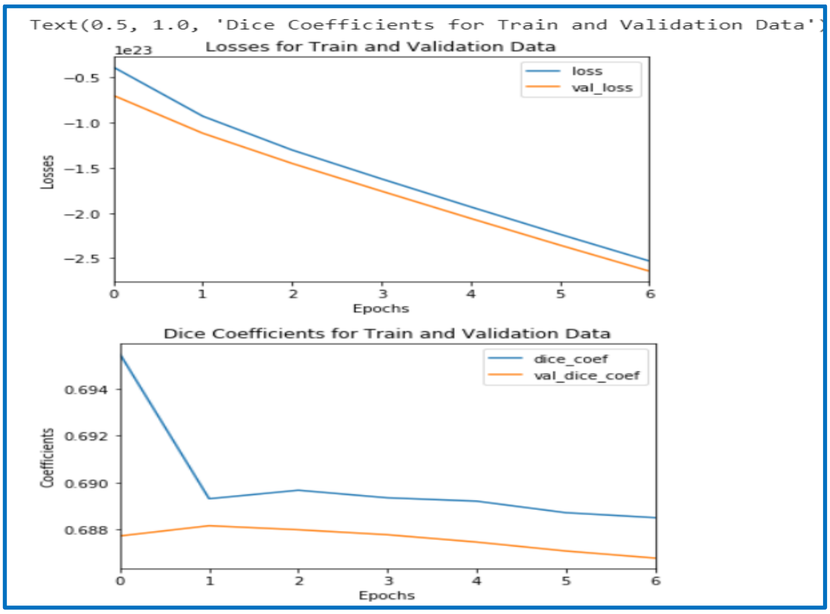

Figure 14: Loss of training and validation set and Dice coefficient

4.5 Implementation, Evaluation and Result of Mask RCNN

Mask RCNN is a deep neural network and works on instance segmentation problem. Its result separated the objects form rest of the image. It follows two stages of computation where it generates the Region of Interest (RoI) in an image then it predicts the class of an object and created bounding boxes and pixel level masks.

4.5.1 Implementation

The model created the chunks of masks which is of similar size of input image and each box is coordinated with bounding box and defect class. The model uses library such as ‘scipy’ and ‘mrcnn.config’. The loss function of the model is formed from the combination of Region Proposal Network (RPN) localize objects, Mask RCNN segment objects and Mask RCNN localize objects.

4.5.2 Evaluation and Result

The training loss drops by 1.1 and remain steady throughout the process as shown in figure 15**.** ‘rpn_loss_box’ function as highest value throughput the training process and accounts for major loss. The dice coefficient is 0.054. It shows that Mask RCNN does not provide better when compared with U-Net. It is also observed that instance segmentation is not an ideal choice for defect detection because unclear boundaries of the different shapes of defect and there is maximum loss due to region proposal network. Hence, this section marks the completion of research objective section 3.4**.**

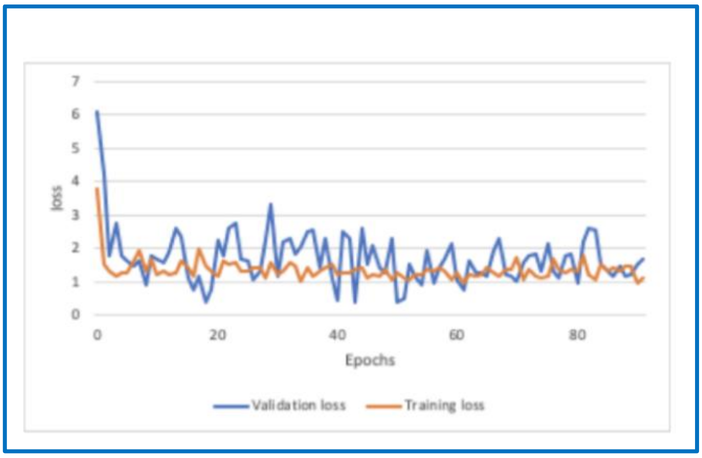

Figure 15: Loss of training and validation data set

4.6 Implementation, Evaluation and Result of UNet++

UNet++ uses DenseNet to improve the U-Net. It has the convolution layer between encoder and decoder and acts as a bridge. It has skip connection for improving the gradient flow and has deep supervision for improving the performance by just skipping one loss layer. Due to the model pruning in UNet++ its inference time is reduced by 32 percent.

4.6.1 Implementation

In the pre-processing the of the training data set only the images which has at least one type defect is considered in the training data. DataGenerator class has been created so the encoder and decoder mask can be resized. The architecture of model is modified by convolution layer by replacing the residual blocks. The model uses the various library such as ‘keras’, ‘sklearn’, ‘tqdm’ and ‘efficientnet.keras’ in the python using Google Collaboratory. ‘np_resize()’ function is used to reshape the mask of the defects, ‘H’ and ‘U’ functions are used for resolution reduction.

4.6.2 Evaluation and Result

The evaluation of the model is done using Dice Coefficient. The model score 0.5754 as shown in figure 16. The model is trained for 30 epochs and Adam is used as an optimizer with the learning rate of 3e-4. The loss value of the validation keeps on dropping throughput the training process starting from 0.625 to 0.375. Hence, this section marks the completion of research objective section 3.5 where model has performed successfully.

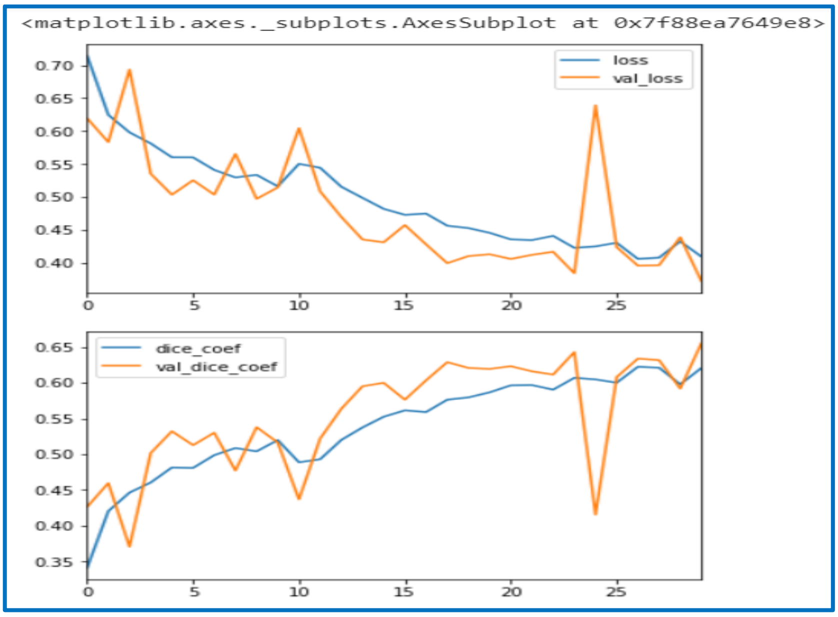

Figure 16: Loss of training and validation set and Dice coefficient

4.7 Conclusion

All the objectives of chapter 1, section 1.3 of research objective have been implemented and the results presented have solved the research question and sub research question. This section marks to the conclusion of implementation, evaluation and results of all the four deep learning models.

5 Comparison of Developed Models and Discussion

All the developed model Xception, U-Net, MaskRCNN and UNet++ are compared below in figure 17. It can be observed that Xception model has given tremendous result followed by UNet++. The Dice Coefficient of Xception is 0.927 and UNet++ is 0.575. Both the models have successfully able to identify between the classes of defects as well as able to localize the position of defects in steel sheets. On the other hand, in the comparison bar chart it can be observed that U-Net and Mask RCNN has attain the score of 0.068 and 0.054 respectively. So, it can be concluded that these models are not performed well for defect detection. This leads to the completion of research objective section 3.6.

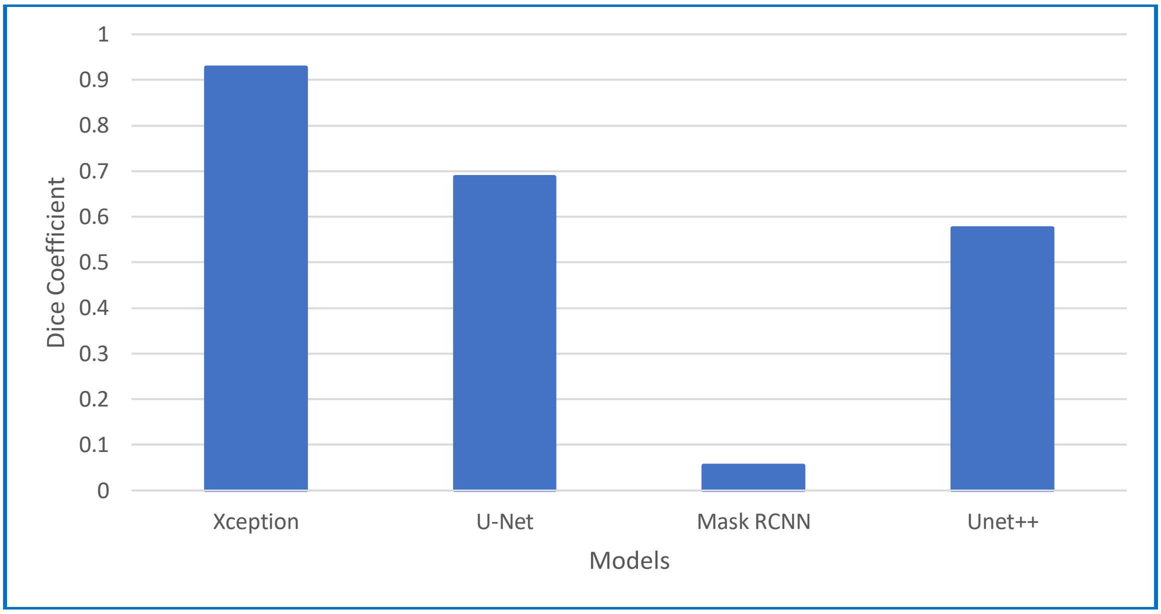

Figure 17: Comparison based on the performance of Deep Learning Model

6 Conclusion and Future Work

This research focused on solving the problem of detecting the defects in steel sheets manually. Manually detecting the defects is a tiresome task and prone to human errors. So, automating this process can help the industry to improve their standards and can maintains the quality of the end product. The research has been carried out using deep learning models in python. All the objectives of chapter 1, section 1.3 is implemented successfully and the results presented have solved the research and the sub-question.

This research project has implemented four machine learning models they are Xception, U-Net, Mask RCNN and UNet++. Out of all the models Xception has given the tremendous result of Dice Coefficient of 0.927. The model has shown good performance in test, train and validation dataset which means it is not overfitting the model. The model uses Xception model architecture for its classification and U-Net architecture with efficientnetb1 for the segmentation. This model has attained high precision multilabel classification and high recall for binary classification. This research can contribute to the steel industry to embed this model in their manufacturing process for maintain the standards of their product.

Future Work: This research has carried out just for steel sheet surface defects. It can also be used for defect detection of heavy metals and on different kind of labels. Also, the results of the proposed Xception model can be improved by increasing the layers of the architecture and work can also be done to understand the extremity of the defects so that industry can recycle the usage of defected sheets and maintains the ecological balance.

7 Acknowledgement

I would like my supervisor Dr. Catherine Mulwa for her continuous support and guidance throughout the semester. I would like to thank my mother, father and brother for their trust in me.

References

Automated Detection of Steel Defects via Machine Learning based on Real-Time Semantic Segmentation. In Proceedings of the 3rd International Conference on Video and Image Processing (ICVIP 2019). Association for Computing Machinery, New York, NY, USA, 42–46. DOI: https://doi.org/10.1145/3376067.3376113

Bukharev, A., Budennyy, S., Lokhanova, O., Belozerov, B. and Zhukovskaya, E. (2018). The task of instance segmentation of mineral grains in digital images of rock samples (thin sections), 2018 International Conference on Arti_cial Intelligence Applications and Innovations (IC-AIAI), pp. 18-23.

He, Y., Song, K., Meng, Q. and Yan, Y., 2020. An End-to-End Steel Surface Defect Detection Approach via Fusing Multiple Hierarchical Features. IEEE Transactions on Instrumentation and Measurement, 69(4), pp.1493-1504.

Li, J., Su, Z., Geng, J. and Yin, Y., 2018. Real-time Detection of Steel Strip Surface Defects Based on Improved YOLO Detection Network. IFAC-PapersOnLine, 51(21), pp.76-81.

Liu, X., Deng, Z. and Yang, Y. (2019). Recent progress in semantic image segmentation, 52: 1573-7462.

Li, X., Tso, S., Guan, X. and Huang, Q., 2006. Improving Automatic Detection of Defects in Castings by Applying Wavelet Technique. IEEE Transactions on Industrial Electronics, 53(6), pp.1927-1934.

Liu, Y., Xu, K. and Xu, J., 2019. Periodic Surface Defect Detection in Steel Plates Based on Deep Learning. Applied Sciences, 9(15), p.3127.

Luo, Q. and He, Y., 2016. A cost-effective and automatic surface defect inspection system for hot-rolled flat steel. Robotics and Computer-Integrated Manufacturing, 38, pp.16-30.

Metal Working World Magazine. 2020. A Short Sheet Metal History - Metal Working World Magazine. [online] Available at: <https://www.metalworkingworldmagazine.com/a-short-sheet-metal-history/>.

Paste, A. S. and Chickerur, S. (2019). Analysis of instance segmentation using mask-rcnn, 2019 2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT), Vol. 1, pp. 191-196.

Roberts, G., Haile, S., Sainju, R., Edwards, D., Hutchinson, B. and Zhu, Y. (2019).

Deep learning for semantic segmentation of defects in advanced stem images of steels, Scientic Reports 9.

Ronneberger, O., Fischer, P. and Brox, T., 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. Lecture Notes in Computer Science, pp.234-241.

Saridis, G. and Brandin, D., 1979. An automatic surface inspection system for flat rolled steel. Automatica, 15(5), pp.505-520.

Sharifzadeh, M., Amirfattahi, R., Sadri, S., Alirezaee, S. and Ahmadi, M., 2008. Detection of steel defect using the image processing algorithms. *The International Conference on Electrical Engineering*, 6(6), pp.1-7.

Song, Ke-Chen & Shaopeng, Hu & Yan, Y., (2014). Automatic recognition of surface defects on hot-rolled steel strip using scattering convolution network. Journal of Computational Information Systems. 10. 3049-3055. 10.12733/jcis10026.

Sun, Q., Cai, J. and Sun, Z., 2016. Detection of Surface Defects on Steel Strips Based on Singular Value Decomposition of Digital Image. *Mathematical Problems in Engineering*, 2016, pp.1-12.

Wu, G., Xu, K. and Xu, J., 2007. Application of a new feature extraction and optimization method to surface defect recognition of cold rolled strips. Journal of University of Science and Technology Beijing, Mineral, Metallurgy, Material, 14(5), pp.437-442.

Zhou, S., Chen, Y., Zhang, D., Xie, J. and Zhou, Y., 2017. Classification of surface defects on steel sheet using convolutional neural networks. Materiali in tehnologije, 51(1), pp.123-131.