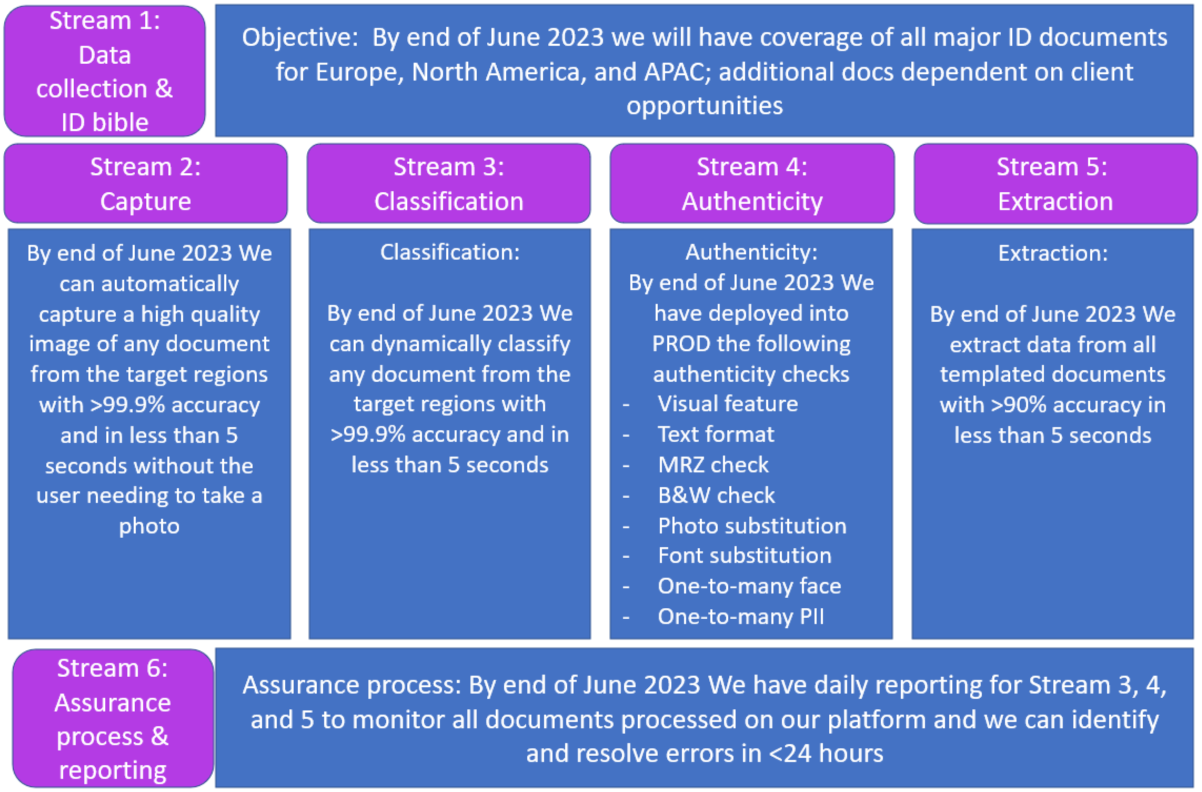
Reflective Journal Entries

Week 1:

The goals of the first week are to get started with Onboarding and get to know about the task and deadlines. The Activities of the first week involve attending a Kick-off workshop hosted by the company CEO. Mike Simpson the CEO & Co-founder Introduced the Machine Learning teams and explained the workflow inside the organization. Got assigned a list of tasks to perform during the internship period, and was able to ask and clarify the point of contact and more about the assigned task in person.



Stream 1: Data collection & ID bible

Objective: By end of June 2023 we will have coverage of all major ID documents for Europe, North America, and APAC; additional docs dependent on client opportunities

Stream 2: Capture

By end of June 2023 We can automatically capture a high quality image of any document from the target regions with >99.9% accuracy and in less than 5 seconds without the user needing to take a photo

Stream 3: Classification

Classification:

By end of June 2023 We can dynamically classify any document from the target regions with >99.9% accuracy and in less than 5 seconds

Stream 4: Authenticity

Authenticity:
By end of June 2023 We have deployed into PROD the following authenticity checks

- Visual feature
- Text format
- MRZ check
- B&W check
- Photo substitution
- Font substitution
- One-to-many face
- One-to-many PII

Stream 5: Extraction

Extraction:

By end of June 2023 We extract data from all templated documents with >90% accuracy in less than 5 seconds

Stream 6: Assurance process & reporting

Assurance process: By end of June 2023 We have daily reporting for Stream 3, 4, and 5 to monitor all documents processed on our platform and we can identify and resolve errors in <24 hours

Figure 3 Roadmap for the company to achieve by June

The Organisation introduced 6 Streams where all the selected interns will be working this includes processes like Data Collection, Capture, Classification of Data, authentication, Extraction, and Quality Assurance and reporting. Design, building, improving Text format check, and Designing, building, and improving Unique Security Feature check were the two important tasks assigned to me during the meeting. These were part of the ID document Authenticity check.

The Task Design, building, and improving Text format check includes the following functions to perform

- Gather text format metadata for all documents, Define the logic of matching data against the defined metadata.

Week 2:

Goals of week 2 involves getting access to most of the logins, which the company uses, like access to emails, teams, etc., and planning the 1:1 meeting. Activities involve getting In touch with the security teams for access. Outcomes: Received the credentials for the work platforms and got in touch with the supervisor and gave the internship availability dates Every Tuesday, Wednesday, and Thursday from 9:00 am to 6:00 pm, and scheduled weekly meetings for reporting with the supervisor.

Every Thursday 30 minutes meeting is arranged between the Technical product owner(Sai Kiran Ravula) and the CEO (Mike Simpson) to discuss the progress made in the particular week and to discuss the roadmaps for the upcoming week including giving suggestions and comments on the performance done so far.

Week 2 the work mostly deals with researching the Non-ID document Extraction. Week 2 mostly worked on finding different text patterns and finding how to extract the data from the text or metadata it has. Has to go through numerous real-time projects and research papers and implementations to find out the different ways to extract data from the non-ID particularly since nothing has a unique structure. Used Google Scholar for the research papers.

Researching part was rewarding, never found it easy but the process was insightful about the latest or possible implementation of the task. This also helped me to think and process differently than the traditional approach. As part of the onboarding did Sign off on Policy and Standards or Truuth Organisation, did setup Jira Confluence for the documentation part, followed the company guidelines to encrypt the system, and also went through security awareness training modules As part of Locii Information Security Management System Security Training & ISO27K Awareness Training for Interns 2020 - 2021 - locii ISMS.

The Security Awareness module mostly deals with how to secure the data inside the organization and how to deal with vulnerability and threats and a detailed introduction about the cyber attacks through spear phishing emails, baiting, Scareware, and Pretexting. A few of the take away from the four module Training are listed below.

- Cyber Security in the Workplace is Everyone's Responsibility Staffs should ensure that
- Do not access Locii's data from public Wi-Fi internet
- Proper security measures are in place when handling confidential and High confidential data
- When working in a shared workspace, privacy filters and screen locks are on for the endpoint devices used.
- Are in a protected closed environment when discussing confidential ideas in a shared working space
- The URL links are legitimate and not malicious before clicking it
- Antivirus and patching are always on in their Locii devices and BYOD
-

At some point I felt communicating with the team was difficult, especially when no response to the mail sent regarding the task to proceed next. The access set up with the security consultant was smooth. Getting to know more about what you do and trying to implement is a good way to handle the tough situation, I spent extra hours after work researching the topic and going through the multiple research paper. Maybe frequent follow-ups and guidelines to proceed further, I find it difficult to understand how to proceed further.

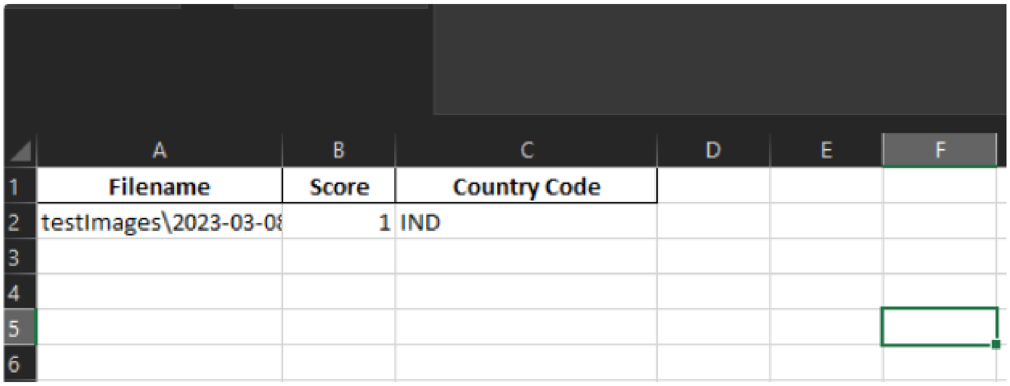
Week 3:

The goals of week 3 involve checking the authenticity of the documents. Here is the brief about it (An authenticity check is needed to determine whether the document submitted by the user is an original document or a B&W photocopy, particularly in situations where authenticity is critical such as legal or financial transactions or verifying government-issued identification documents.)

As part of this process, it's important to have a clear idea of the data before processing with and checking Authenticity using Python script. On Weekly calls, CEO mentioned he wanted the Geographical distribution of the data the organization has in a visualization format. The process involved fetching the document and passing the documents through Python open cv API created in Azure. Then Analysis the geographical diversity of the documents.

Below is the step-by-step method followed to achieve the same.

- The dataset from the two data sources is carefully screened and merged into one source with 469 total passports.
- Wrote a Python script to fetch the API and gather the score & country code of the passport. The reference sample is attached below.
- This process ran through all the individual passports in the dataset and got this output.csv file.



Filename	Score	Country Code
testImages\2023-03-0	1	IND

Figure 4 Output format received in the CSV

- Ran the Visualization code for the bar plot, and geo plot to understand the distribution of the passport dataset.

Data Source

For the Analysis of the Geography Distribution of the passport dataset, the organization Initially collected the dataset from the ID Bible, where manually downloaded the file from the A-Z Countries, and obtained 269 Passports. Another dataset has been provided with a collection of 200 different passports. Later combined these two sets of data sources into one to proceed further.

Total Dataset of passport: 469

Table 1 Table showing the Dataset which Visualization performed

Dataset	count
ID Bible	269
Dataset #1	200
Total	469

Analysis

The entire process is finished on 469 dataset images and received an output of 431. The remaining 38 passport file couldn't fetch the result either of the APIs not able to fetch the data. Most of the passports on the dataset are from Australia(133), New Zealand (12), the UK (8), Philippines(7). Most of the diverse passports are covered as part of the dataset but are limited to 1 or <6.

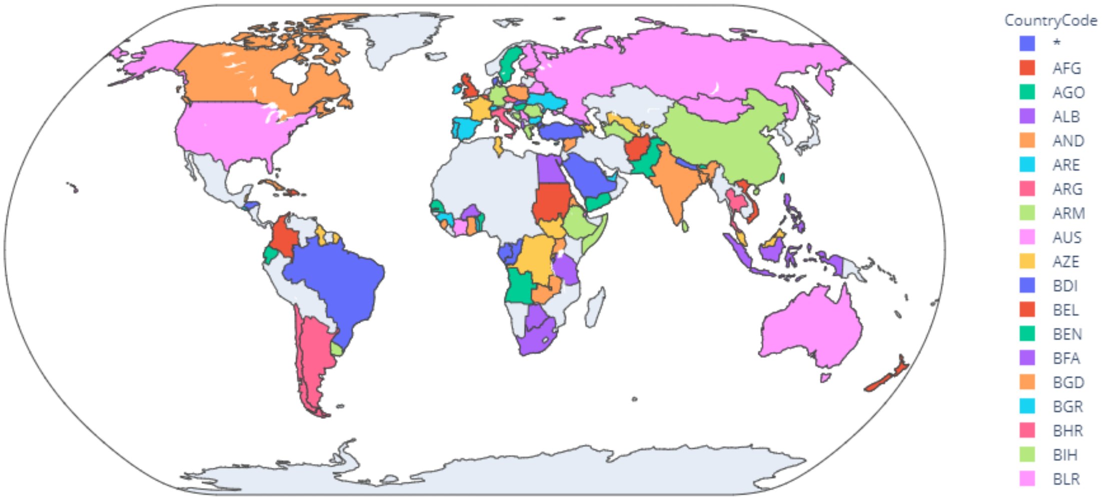


Figure 5: Geo plot representing the wide coverage of passport

Getting to know more about the data and being clear about the task I'm supposed to do helped to deal better with the challenge. Reaching out to the supervisor after not being able to get the solution and looking for help better. Improved skills in Python and visualization, received proper guidance on extracting the text from the document from the supervisor assigned. Followed a proper communication channel for the same.

Week 4:

The goals of the week are to Implement a Visual Feature check on the document database and generate a matching score and generate a hypothesis on the MRZ Positive and Negative Testing + worked on the Classification Errors APCER and BPCER. Activities involve writing Python scripts to check the visual feature of sample documents and user-submitted documents and generate a matching score.

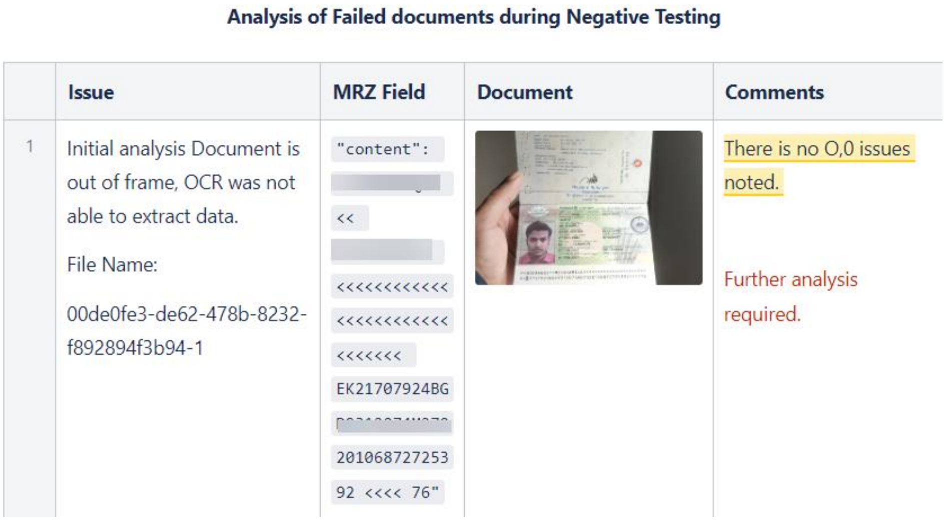
BPCER (Bona fide presentation classification error rate), is the percentage ratio at which bona fide examples are misidentified as presentation attack examples. A higher value indicates higher user friction. APCER Attack presentation classification error rate, the percentage ratio at which presentation attack examples are misidentified as a bona fide example. A higher value indicates higher security vulnerability.)

Run the Testing on both positive and Negative by generating Negative samples, the negative sample is created by forging the authentic file like changing the value in the MRZ field. Negative Testing has been performed on the 200 authentic datasets from the org. A sample of 70 document images has been taken and changed the data based on the Date of Expiry, Passport Number, and Date of issue.

Table 2 Table showing how the document has been falsified for Neg testing

MRZ Dataset Changes	count
Change in Passport No	16
Change in DOB	24
Change in DOE	30
Total	70

Outcomes: Negative Testing has been performed on the 200 authentic datasets from the org. A sample of 70 document images has been taken and changed the data based on the Date of Expiry, Passport Number, and Date of issue. The output of the above run has been attached below, where 66/70 of the false document API run was able to recognize its fake document.



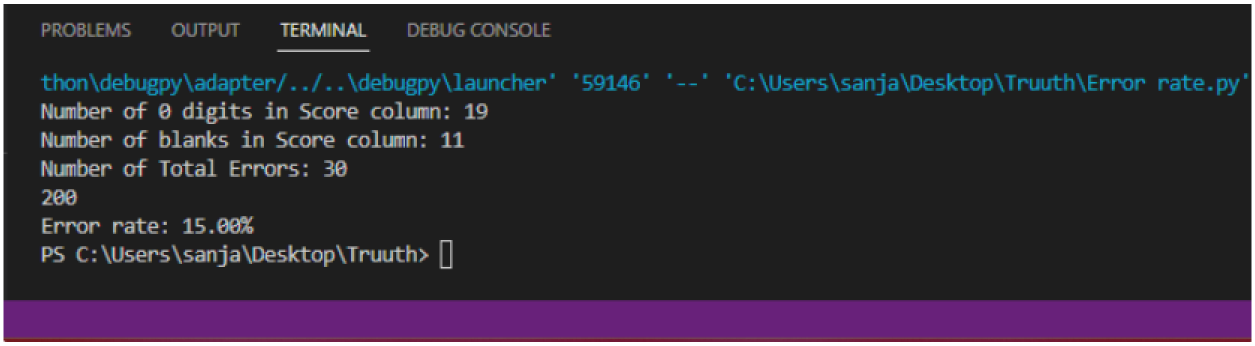
Analysis of Failed documents during Negative Testing

	Issue	MRZ Field	Document	Comments
1	Initial analysis Document is out of frame, OCR was not able to extract data. File Name: 00de0fe3-de62-478b-8232-f892894f3b94-1	"content": [redacted] << [redacted] <<<<<<<<<<<< <<<<<<<<<<<< <<<<<<< EK21707924BG [redacted] 201068727253 92 <<<< 76"		There is no O,0 issues noted. Further analysis required.

Figure 6 Further Analysis of failed documents to understand why they failed.

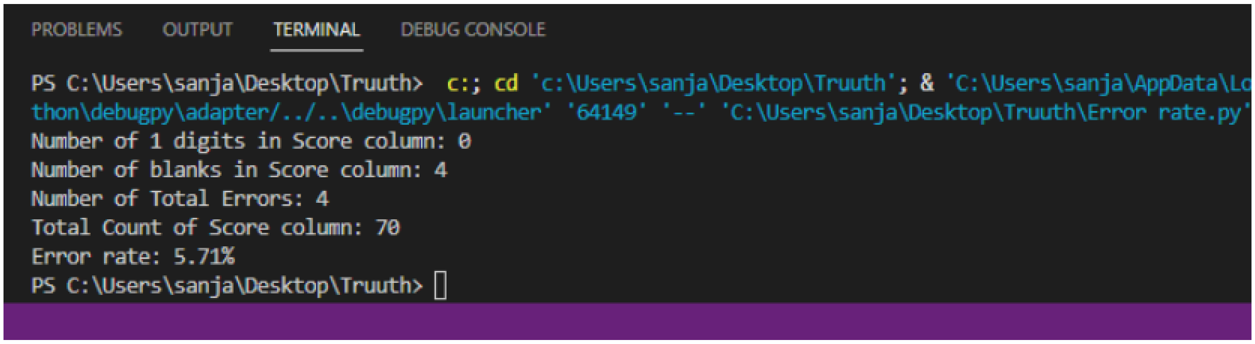
For the positive Testing considered the 200 Authentic passports dataset and ran the script on the same. The result obtained is 30/200 files turned out to be not authentic. 19/200 files turned to be MRZ score of 0 instead of 1 and the remaining 11 files didn't produce any score. Few of the files produced blank results this has been analyzed in the below approach.

Then the final stage is to classify errors and get the error rate for APCER and BPCER. The error rate for BPCER: is 15.00%, and The error rate for APCER: is 5.71%



```
thon\debugpy\adapter/../..\debugpy\launcher' '59146' '--' 'C:\Users\sanja\Desktop\Truuth\Error rate.py'
Number of 0 digits in Score column: 19
Number of blanks in Score column: 11
Number of Total Errors: 30
200
Error rate: 15.00%
PS C:\Users\sanja\Desktop\Truuth> 
```

Figure 7: Figure showing BPCER Error rates



```
PS C:\Users\sanja\Desktop\Truuth>  c:; cd 'c:\Users\sanja\Desktop\Truuth'; & 'C:\Users\sanja\AppData\Lo
thon\debugpy\adapter/../..\debugpy\launcher' '64149' '--' 'C:\Users\sanja\Desktop\Truuth\Error rate.py'
Number of 1 digits in Score column: 0
Number of blanks in Score column: 4
Number of Total Errors: 4
Total Count of Score column: 70
Error rate: 5.71%
PS C:\Users\sanja\Desktop\Truuth> 
```

Figure 8: Figure Showing APCER Error rate

From the activity from this week got to know why sample set analysis is better than processing the entire dataset in one go. Learned to use a Python script to call the Azure API. Learned about the data complexity and chances of getting biased based on the data. A summary of this week's activity is listed below.

- Wrote a Python script to fetch the API and gather the Visual Features of the document and generate a matching score.
- Wrote Python script to redirect the OCR extracted text in the form of JSON code.
- Few documents didn’t have any output, getting to a hypothesis and detailed analysis of that particular process is a rewarding experience. This way gather knowledge to analyze the data differently.
- The coding part was a bit challenging as never worked with real-time fetching of API and doing the visual analysis of the document.

A few things noticed is Organizationrequirement keeps changing sometimes, having a platform like an Excel report sheet to track the roadmap and timeline would be easy to understand the progress and track the productivity as well.

Week 5:

Every identification document contains distinct and unchanging features or objects that are consistent in size and location within the document. These objects are utilized in determining the document's classification. This week the goal is to build a visual feature check. This can achieve by specifying the precise location of each object, and by examining the location of each object in a sample image, we determine whether the document is correctly positioned or not and find the matching percentage of the template.

Activities include checking the Validity of the documents with a respective template which also provides the matching percentage. The objectives have been categorized as the priority below. Categorize the sample into respective document types folder. A sample of max 20 or less as per the requirement. Run the Python program to find the respective match per the folders. Analyze the match percentage and further investigate failed documents which will help with the following.

- To see the positive and negative score distributions for each classification code.
- To compare the results of the 3 different curation approaches.
- To test the models on unseen positive and negative samples and calculate the FAR and FRR.
- To store the logistic regression parameters.

Knowledge:

The visual feature check is a sub-check within our Authenticity API version 2 that tests whether the client-submitted ID document image has similar visual features to the authentic template we have saved for that classification code. Visual characteristics like banners, logos, emblems, and any special symbols present on the document are considered visual features.

- Learned about the data complexity and chances of getting biased based on the data.
- Wrote a Python script to fetch the API and gather the Visual Features of the document and generate a matching score.
- Wrote Python script to get a matching percentage and further used logistic regression model for the outputs.
- The Data Curation part was a Tricky scenario:
- This is the scenario where we don't have sufficient positive samples to perform the data gathering.
- For example, we don't have 200 documents of the NSW_FULL_DL to run the positive analysis. In this case, we would like to do the following data curation approach:

The coding part was a bit challenging as never worked with real-time fetching of API and doing the visual analysis of the document.

Note: This week has taken leave from work on Thursday due to the back pain issue I faced, and my productivity was less compared to the other weeks so far due to the sick. The organization's work culture is amazing without a prior email communication regarding the sick leave they approved the leave and made sure I'm taking rest by giving off to the task assigned. Most of the document tasks have been covered this week comparing to the coding.

Week 6:

The Task of Week 6 involves doing the pending task from the last week of visual feature check and the below ones.

- Collect quality samples for a wide range of global identity documents and create a template for each document class. Our aim for the POC is to extend the visual feature check to all identity documents that have been templated (AU + NZ DLs). A stretch goal is to extend the POC to global passports.
- Categorize the samples into respective document types folders. A sample of max 50 or less as per the requirement.
- Use SIFT to calculate the number of matching features for each sample image vs the chosen template for the document class

Activities include running the generic testing. For a positive set, actual documents run against the actual templates, and for negative (generic negative documents) documents which are not Australian documents like debit cards, debit cards. Here selecting Max of 200 files for each category. Image Enhance will be run before proceeding with every test for the same and running the logistic regression.

Sample Dataset - Ideal

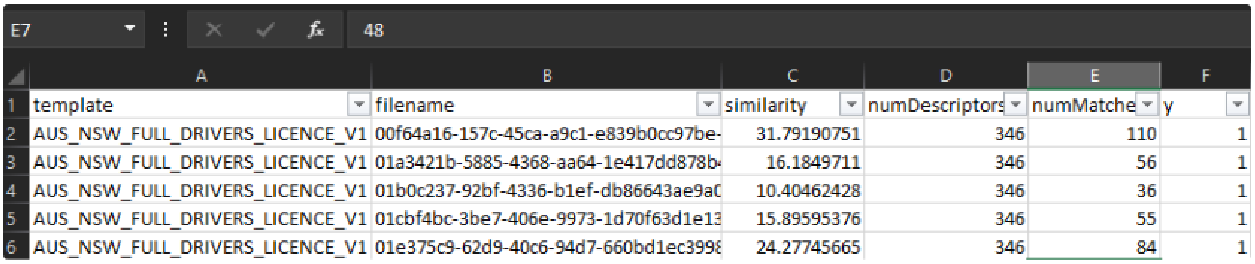
Table 3 Shows the Ideal dataset to process

Dataset Quality	**Count**	**Dependant Variable**	**Independent Variable.**
AUS_NSW_FULL_DRIVERS_LICENCE	200	1	Matching feature template with AUS_NSW_FULL_DRIVERS_LICENCE_V
Other Samples from the Data source used are (AUS_ACT_FULL_DRIVERS_LICENCE_V1)	200	0	Unmatching Features

Note: The dataset has mixed samples of other types of documents, so we need a manual approach to select 200 license front page AUS_NSW_FULL_DRIVERS_LICENCE_V1. Data curation in the specific session is shown below.

Positive Testing

For the positive testing got the similarity score ranged from 0 to 31.79 and analyzed the distribution of the similarity using a Histogram. 27 records are falling under a similarity score of 5.



E7 48

	A	B	C	D	E	F
1	template	filename	similarity	numDescriptors	numMatche	y
2	AUS_NSW_FULL_DRIVERS_LICENCE_V1	00f64a16-157c-45ca-a9c1-e839b0cc97be-	31.79190751	346	110	1
3	AUS_NSW_FULL_DRIVERS_LICENCE_V1	01a3421b-5885-4368-aa64-1e417dd878b	16.1849711	346	56	1
4	AUS_NSW_FULL_DRIVERS_LICENCE_V1	01b0c237-92bf-4336-b1ef-db86643ae9a(	10.40462428	346	36	1
5	AUS_NSW_FULL_DRIVERS_LICENCE_V1	01cbf4bc-3be7-406e-9973-1d70f63d1e13	15.89595376	346	55	1
6	AUS_NSW_FULL_DRIVERS_LICENCE_V1	01e375c9-62d9-40c6-94d7-660bd1ec399	24.27745665	346	84	1

Figure 9 shows the similarity score obtained for positive files.

Negative Testing

For the positive testing got the similarity score ranged from 0 to 4.62 and analyzed the distribution of the similarity using a Histogram.

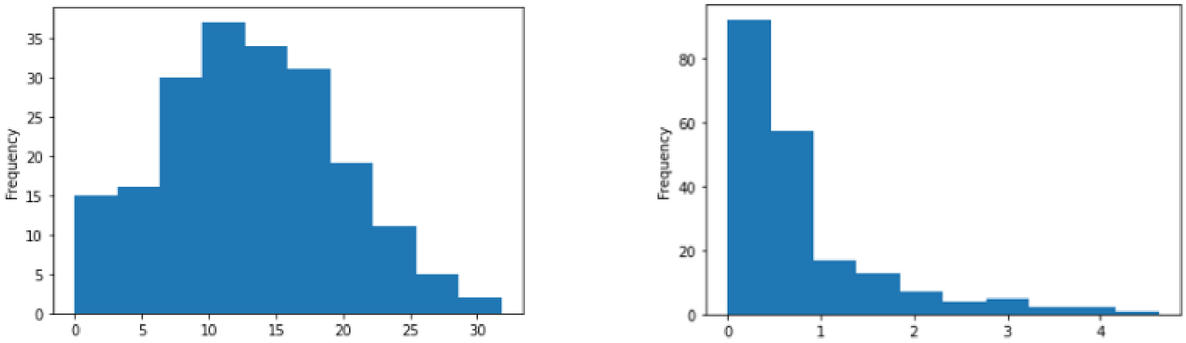


Figure 10 On the left is the Postive Distribution & on the right Negative Distribution.

From the Hypothesis Tests it has been found out the **similarity score could improve by enhancing the image**, using the image enhancer script for the same, and running the same process of negative and positive testing to understand further differences in the similarity score obtained. Similarly below is the hypothesis test to run to improve the output.

- The image enhancement model has improved the image quality it enhanced the blurred images which eventually improve the similarity scores and thereby better results.

Experience: Learned about the data complexity and chances of getting biased based on the data. The documentation part was rewarding during the time, Documentation makes it easier to maintain the code over time. As code evolves and changes, Well-documented code is more likely to be reusable. The coding part was a bit challenging as never worked with real-time processing and image enhancement of 1000 documents in a go. The organization team was supportive of providing a token to the Azure API every time it expires which is an integral part of the image enhancement.

Week 7:

Goals involve converting the number of matching features into a match percentage that takes account of variations in the max visual features for each document class and using logistic regression further. Analyze the match percentage and further investigate failed documents. Continue last week's work on Generic Models as there was some skewness observed.

Activities: Selecting Max of 200 positive and negative files for each category. Image Enhance will be run before proceeding with every test for the same and running the logistic regression. Then write code to find visual feature matches for random files based on their similarity score.

Table 4 Dataset for Hypothesis Test 2

Dataset Quality	**Count**	**Dependant Variable**	**Independent Variable.**
AUS_NSW_FULL_DRIVERS_LICENCE_V1 (Positive & Negative combined)	400	0,1	Similarity Scores

From the Analysis got logistic regression score of 0.98 typically refers to the accuracy or the performance of the logistic regression model on the evaluation dataset or the test dataset. The score of 0.98 indicates that the model can correctly classify 98% of the observations in the test dataset.

Overall, a logistic regression score of 0.98 suggests that the model is performing well and is a strong predictor of the binary outcome variable being modeled.

For the below observation took 10 positives from AUS_NSW_FULL_DRIVERS_LICENCE_V1 and Negative sample files randomly from the dataset AUS_NT_DRIVERS_LICENCE_V1, ACCESS_CARD_KIWIetc.

Table 5 Calculating Visual Feature Match based on the similarity score obtained.

File Name	Similarity Score	Visual Feature Match
04698be2-6b54-4ae9-82f0-02785c5a3230-1.jpg	12.13872832	0.9998575724388266
4617c1bb-55ee-47ff-a522-8610de9c566c-1.jpg	4.624277457	0.6760779822878397
4829fe2e-529c-4a35-9412-2cd14b4fdc27-1.jpg	2.023121387	0.11152914079973549

Note: A Detailed Explanation of how the visual feature match is calculated and a logistic regression problem statement is explained detailed in next section.

Hypothesis Test III

- Run the Generic Testing. For a positive set, actual documents run against the actual templates, and for negative (generic negative documents) documents are not Australian documents.

- Categorize the samples into respective document types folders. A sample of max 50 or less as per the requirement later run the negative testing with non-ID documents and perform the logistic regression on the same.

Table 6 Positive Dataset for Hypothesis Test iii

Data Quality	**Data quantity**	**Dependent variable**	**Independent variable**
NON-ID documents such as Visiting cards, Debit, and Credit cards.	1223	0	Similarity scores

Table 7 Table 6 Positive Dataset for Hypothesis Test iii

Data Quality	**Data quantity**	**Dependent variable**	**Independent variable**
Authentic Documents + NON-ID documents such as Visiting cards, Debit, and Credit cards.	2532	1	Similarity scores

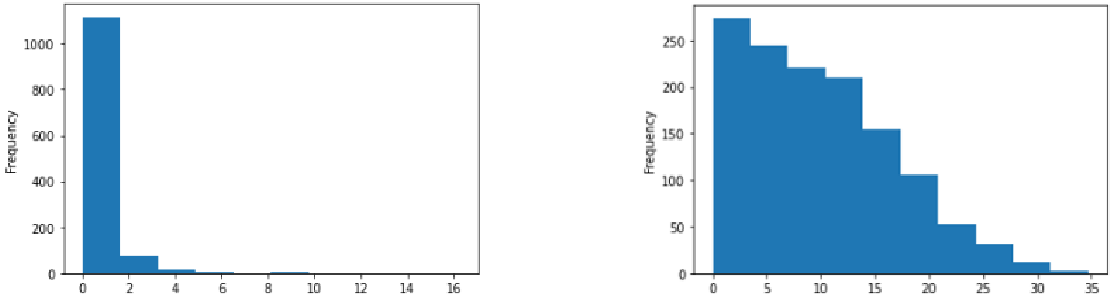


Figure 11 On Left Negative Testing and on the right Positive Testing

Analysis: It has been observed that there is right skewness in the positive data, it has to be further analyzed what causing the skewness. For this selected the positive files with a similarity score of less than 5 and negative files with a similarity score greater than 5 from the dataset and did distribution plot for the same.

The Future roadmap includes rectifying the skewness by improving the brightness of images through a deep-learning model and performing the above steps including Logistic regression. This process was the most difficult so far in the internship period, I have come up with a couple of solutions like Normalisation on images, and log transformation on images to fix the skewness, which eventually has not given any exponential difference in the skewness, I have also received support to try out different solutions to the problem from the team.

Week 8:

Goals involve converting the number of matching features into a match percentage that takes account of variations **Goals:**

- Improve the skewness of the positive dataset image processed.
- Hypothesis 1: Increase the image's overall brightness by a specific percentage and see an improvement in the document scores.

Activities:

Activities include trying out multiple deep-learning image enhancement papers, selecting the best suits from them according to our requirements, and executing the same on the positive dataset.

The detailed objectives are:

- Find out the best paper and try to replicate the codebase.
- Tweak and apply the codebase to the organization's existing Positive dataset.
- Improve the accuracy score on the positive dataset and check the changes in skewness.

For this approach selected, 20 images from the daily processed document files with an image confidence score of less than <0.7. Wrote a Python script to improve the existing image brightness by 30%. The function returns the original image if the image is already bright enough (determined by checking its mean pixel value).

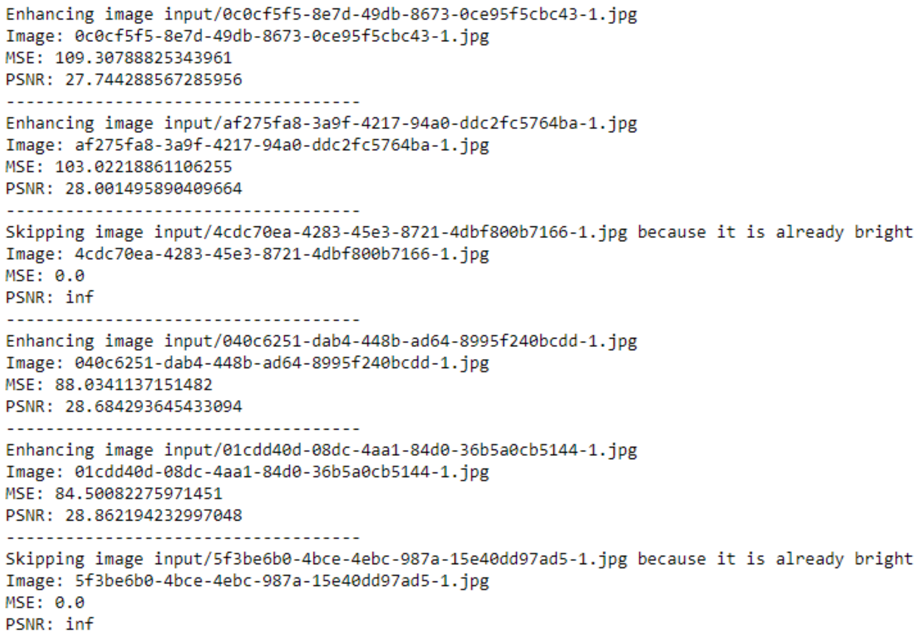
Image Performance Metrics

Use the metrics as signal-to-noise ratio (PSNR), and mean squared error (MSE). These metrics can be used to compare two images and evaluate their visual quality or fidelity similarity.

Using MSE Metrics on the above

MSE calculation assumes that the original image is the ground truth and the enhanced image is the approximation. They used PSNR Metrics as well. It compares the original image to the modified image and calculates how much noise has been introduced in the modified image. Higher PSNR values indicate that the modified image is closer to the original image in terms of visual quality.

Outcomes: Out of 20 documents, 11 document scores have been improved, two scores noted the same as before, and slight changes in the score of the other 7 papers were noted.



```
Enhancing image input/0c0cf5f5-8e7d-49db-8673-0ce95f5cbc43-1.jpg
Image: 0c0cf5f5-8e7d-49db-8673-0ce95f5cbc43-1.jpg
MSE: 109.30788825343961
PSNR: 27.744288567285956
------------------------------------
Enhancing image input/af275fa8-3a9f-4217-94a0-ddc2fc5764ba-1.jpg
Image: af275fa8-3a9f-4217-94a0-ddc2fc5764ba-1.jpg
MSE: 103.02218861106255
PSNR: 28.001495890409664
------------------------------------
Skipping image input/4cdc70ea-4283-45e3-8721-4dbf800b7166-1.jpg because it is already bright
Image: 4cdc70ea-4283-45e3-8721-4dbf800b7166-1.jpg
MSE: 0.0
PSNR: inf
------------------------------------
Enhancing image input/040c6251-dab4-448b-ad64-8995f240bcdd-1.jpg
Image: 040c6251-dab4-448b-ad64-8995f240bcdd-1.jpg
MSE: 88.0341137151482
PSNR: 28.684293645433094
------------------------------------
Enhancing image input/01cdd40d-08dc-4aa1-84d0-36b5a0cb5144-1.jpg
Image: 01cdd40d-08dc-4aa1-84d0-36b5a0cb5144-1.jpg
MSE: 84.50082275971451
PSNR: 28.862194232997048
------------------------------------
Skipping image input/5f3be6b0-4bce-4ebc-987a-15e40dd97ad5-1.jpg because it is already bright
Image: 5f3be6b0-4bce-4ebc-987a-15e40dd97ad5-1.jpg
MSE: 0.0
PSNR: inf
```

Figure 12 Output of the file after running the image enhancement

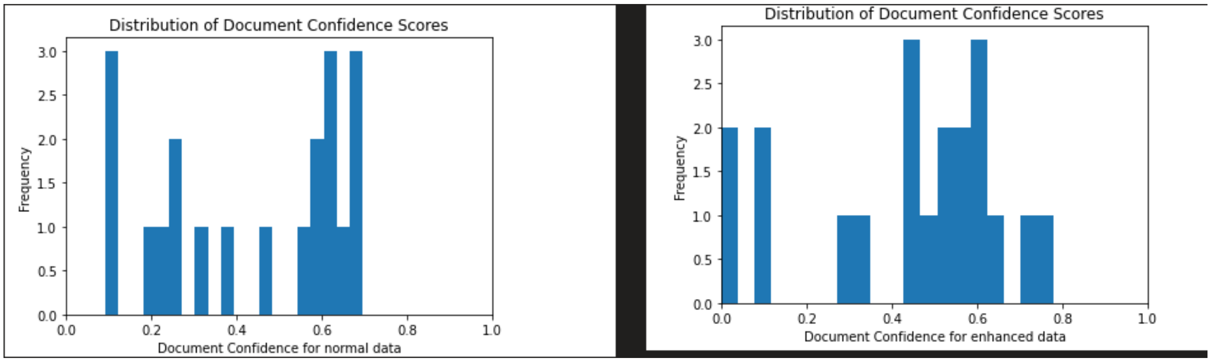


Figure 13 Change in the distribution of the data after enhancement

Overall: Mean Absolute Error: 0.13245

Code reads the data into a panda's data frame, applies the MAE formula to each row of the data frame, and calculates the average of the MAE values to get the final error rate. A lower MAE indicates better accuracy in the predictions.

Going through multiple papers and replicating the code was challenging, especially when trying to replicate the old paper, where the author has used TensorFlow 2 as those functions still need to be supported.

A few of the images provided excellent values after image enhancement and running the OCR checks; understanding what caused this issue was also tiering way.

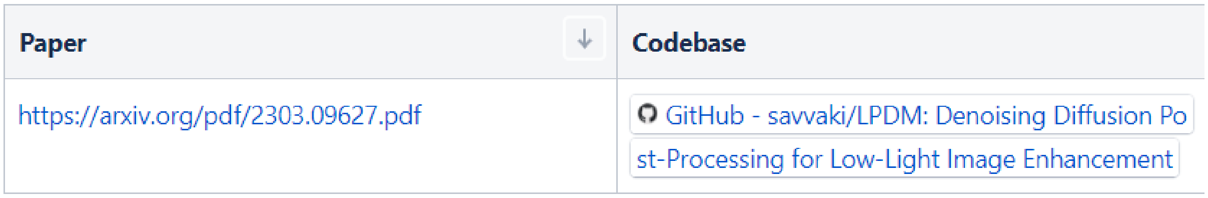
Week 9

The goals involve Implementing different deep-learning model papers to solve the brightness issues.

The detailed objectives were:

- Find out the best paper and try to replicate the codebase.
- Tweak and apply the codebase to the organization's existing Positive dataset.

Paper 1: Denoising Diffusion for Low-Light Image Enhancement



Paper	Codebase
https://arxiv.org/pdf/2303.09627.pdf	GitHub - savvaki/LPDM: Denoising Diffusion Post-Processing for Low-Light Image Enhancement

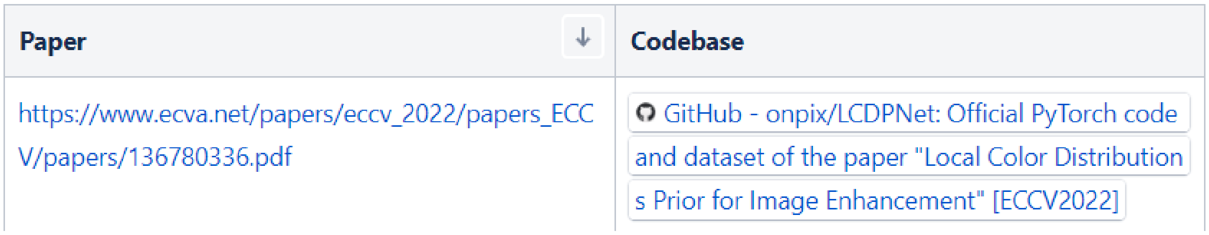
The paper proposes a new approach for the post-processing of low-light images, called the Low-light Post-processing Diffusion Model (LPDM). Low-light image enhancement techniques often introduce various image degradations, such as noise and color bias, which post-processing denoisers address. However, these denoisers often produce smoothed results lacking detail.

Issues:

The model replication was successful and was able to generate the denoised images; the issue highlighted below made not to proceed with the paper.

- We replicated the code base and applied the same to the positive dataset but eventually figured out the pre-trained model size is more than 1 GB**,** which would be a problem executing the PROD system.

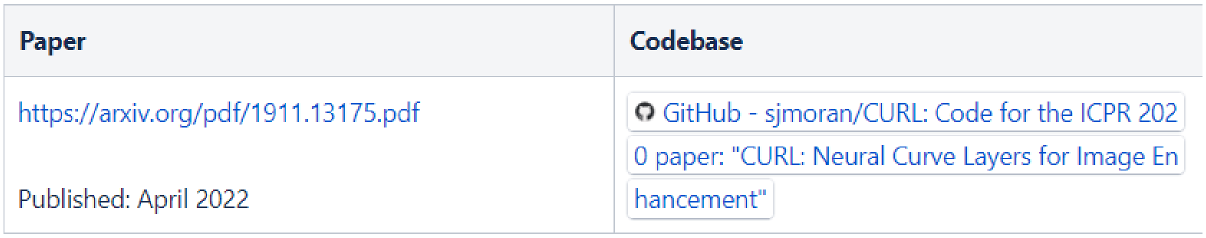
Paper 2: Local Color Distributions Prior to Image Enhancement



Paper	Codebase
https://www.ecva.net/papers/eccv_2022/papers_ECCV/papers/136780336.pdf	GitHub - onpix/LCDPNet: Official PyTorch code and dataset of the paper "Local Color Distributions Prior for Image Enhancement" [ECCV2022]

This paper proposes a novel method called LCD-Net, which consists of three main steps. In the first step, we extract the LCDs of the input image using a clustering-based algorithm. In the second step, we use a segmentation network to locate the over- and underexposed regions based on their LCDs. Finally, in the third step, we use a color enhancement network to enhance these regions.

Paper 3: Neural Curve Layers for Global Image Enhancement



Paper	Codebase
https://arxiv.org/pdf/1911.13175.pdf Published: April 2022	GitHub - sjmoran/CURL: Code for the ICPR 2020 paper: "CURL: Neural Curve Layers for Image Enhancement"

The article further demonstrates the effectiveness of CURL by combining this global image transformation block with a pixel-level (local) image multi-scale encoder-decoder backbone network. The experiments show that CURL produces state-of-the-art image quality compared to recently proposed deep learning approaches in both objective and perceptual metrics, setting new state-of-the-art performance on multiple public datasets.

Testing

- Taken Daily Analysis Report, which has documented confidence score reported, and selected 50 rows of data.
- Sorted the 50 data sources with a confidence score < 0.7 and checked the change in confidence score by performing the step below: Hypothesis Testing.
- Sorted the 50 data sources with a confidence score < 0.8 and checked the change in confidence score by performing the step below: Regression Testing.
- Pass the document through an image enhancement process using a deep learning model.
- Run the document through OCR API on the above 2 sorted documents based on the confidence score.

Hypothesis Test 1

Performing Image Enhancement on the sorted image files with a document confidence score of less than 0.7. From the below Histogram analysis of the image before and after deep learning enhancement, it is evident that there has been a change in the scores, but concerns the ideal score has been given to 8 documents.

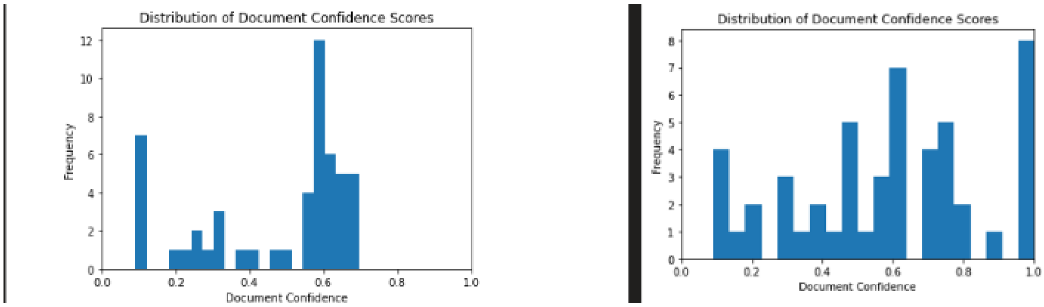


Figure 14 On the left, Before Image Enhancement. On the right, After Image Enhancement.

Regression Testing -

Performing Image Enhancement on the sorted image files with a document confidence score of greater than 0.8. From the below Histogram analysis of the image before and after deep learning enhancement. We need further analysis as it includes the idea score of 1 and observed some change in scores.

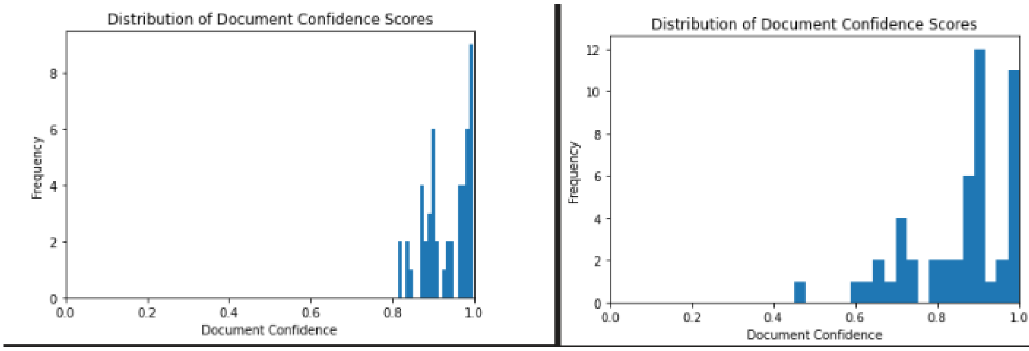


Figure 15 On the left Before Image Enhancement On the right After Image Enhancement

Week 10:

Last week tried a few papers on deep-learning image enhancement and found the curl model performs better, ideated code from the paper and implement it according to current requirements then finally execute the same on the positive dataset.

The detailed objectives are:

- Find out the best paper and try to replicate the codebase.
- Tweak and apply the codebase to the existing Positive dataset the organization has.
- Improve the accuracy score on the positive dataset and check the changes in skewness.

Activities:

For this approach selected 20 failed ocr images from one drive and wrote a code that results in an Excel sheet with respective filenames and brightness and overall brightness threshold value.

To process the brightness threshold and values, I used the Python pillow library, which reads the image and calculates the respective brightness index, whereas the openpyxl library helps to put these values in Excel format at the end.

The code calculates the brightness index for each image using the ImageStat module from the Pillow library. The brightness index is a single value that represents the image's overall brightness. The average brightness threshold value is calculated by taking the mean of the brightness values for all the images. This value can be used as a threshold to determine which images have adequate brightness levels and which images may require adjustment.

Image Brightness:

The average brightness threshold for 20 images processed is 82.216

Image Performance Calculations:

MSE calculation assumes that the original image is the ground truth and the enhanced image is the approximation. Also, the MSE metric has its limitations, and its values may not always reflect the perceptual quality of the images. Therefore, using multiple metrics and human evaluations to measure image performance accurately is recommended.

So using PSNR Metrics as well. it compares the original image to the modified image and calculates how much noise has been introduced in the modified image. Higher PSNR values indicate that the modified image is closer to the original image in terms of visual quality, while lower PSNR values indicate that the modified image has more noise and is farther from the original image value ranging from 0 - 90db. A higher MSE score indicates that there is more distortion or noise in the modified image.

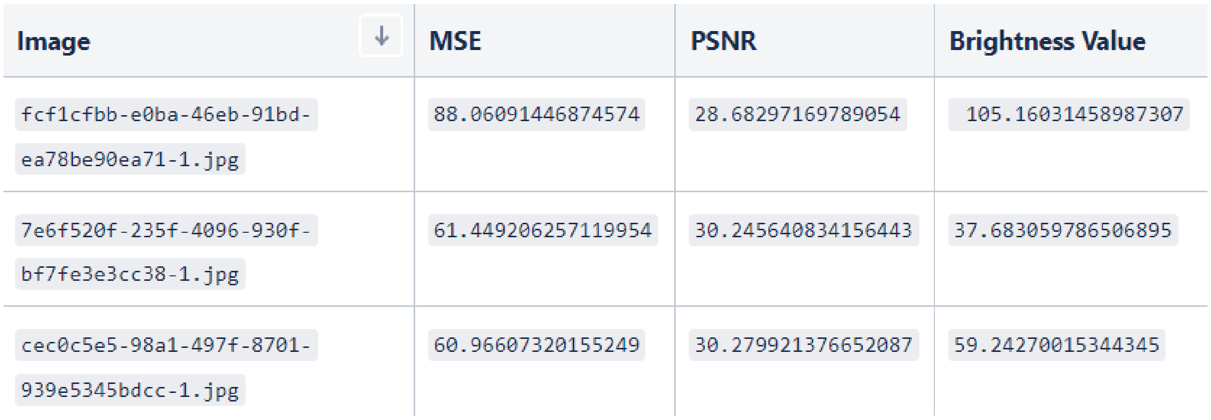


Image	MSE	PSNR	Brightness Value
fcf1cfbb-e0ba-46eb-91bd-ea78be90ea71-1.jpg	88.06091446874574	28.68297169789054	105.16031458987307
7e6f520f-235f-4096-930f-bf7fe3e3cc38-1.jpg	61.449206257119954	30.245640834156443	37.683059786506895
cec0c5e5-98a1-497f-8701-939e5345bdcc-1.jpg	60.96607320155249	30.279921376652087	59.24270015344345

Figure 16 Overview of Performance Metrics

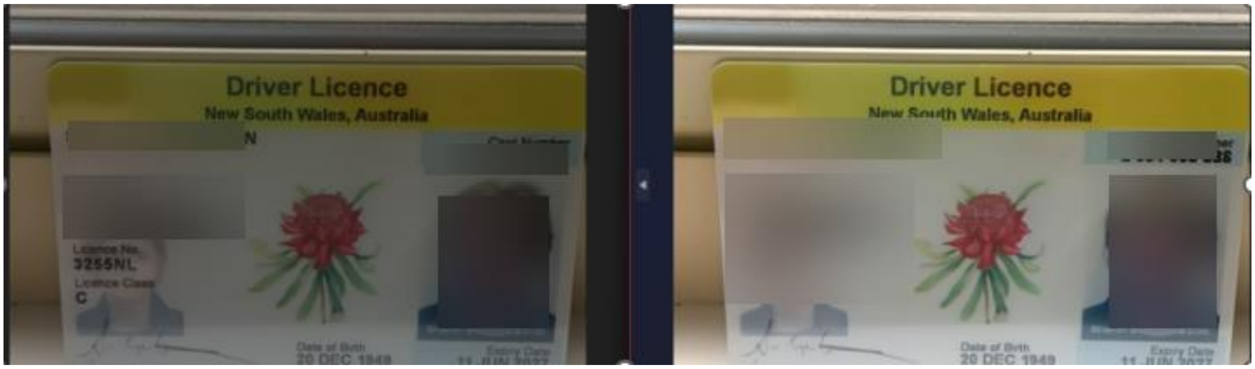


Figure 23 Sample image showing an increase in brightness

Code reads the data into a pandas data frame, applies the MAE formula to each row of the data frame, and calculates the average of the MAE values to get the final error rate. **A lower MAE indicates better accuracy in the predictions.**

Comparison of OCR performance before and after image brightness enhancement.

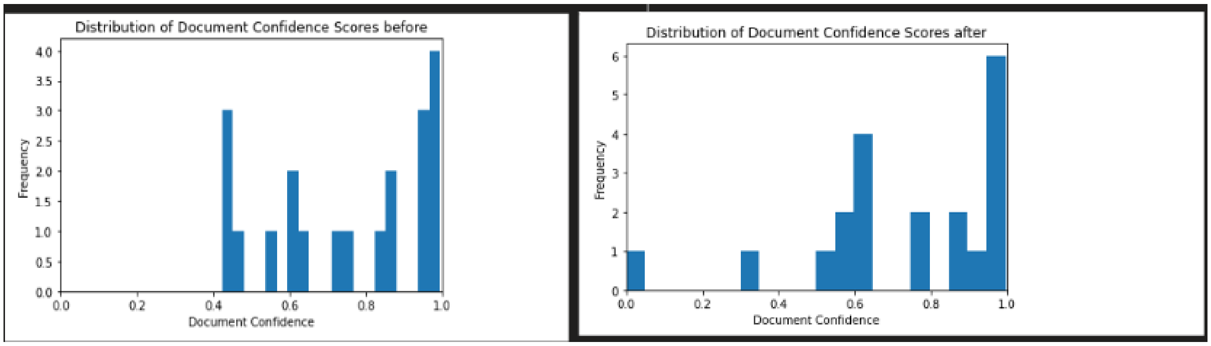


Figure 17 Distribution of the data

Week 11:

The goal is to find the optimal target brightness could be improved by selecting with more reasoning. For this reason, finding an optimal target brightness for image enhancement is important.

- Define the optimal target Brightness.
- Determine there is a difference in targets of different target brightness.

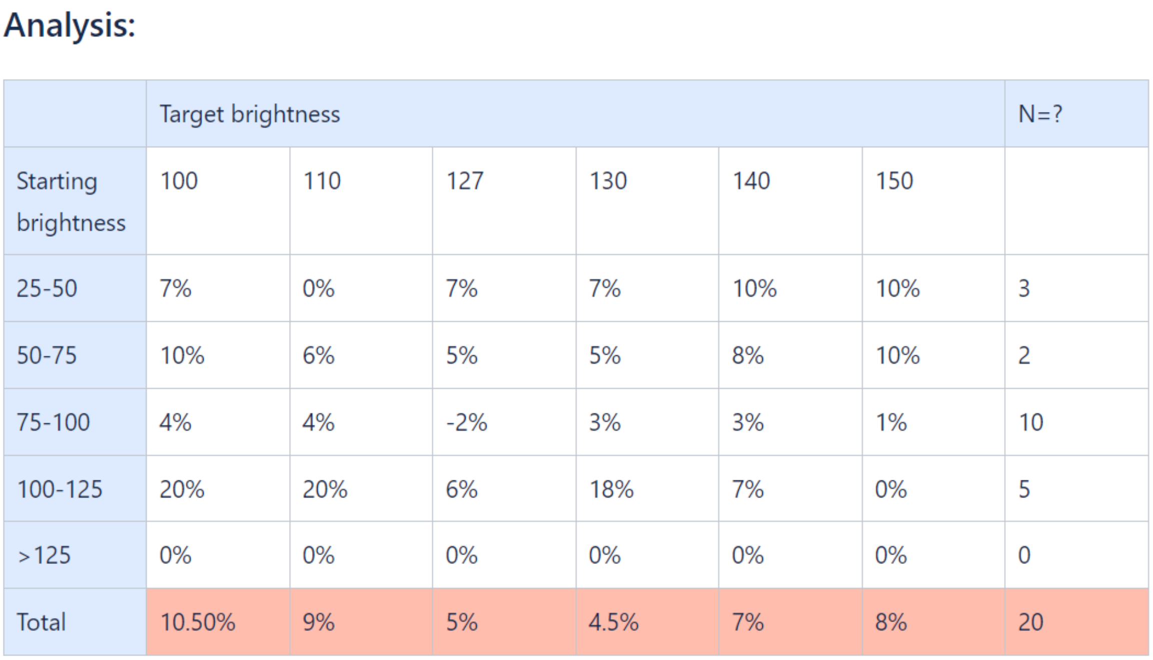
Activities: Below is the target brightness with the respective starting brightness, the respective column represents the error rate percentage comparing the image before pre-processing and post-processing. The target brightness is calculated with the formula as shown below.

E.g., if the image brightness index is 127, the function subtracts the image brightness from the brightness index divided by the brightness value.

(127 - brightness value)/(Brightness Value))

where brightness index is the average image brightness which got by calculating the mean from the PIL python package 'ImageStat' library. The above formula determines how much brightness needs to be increased to reach the desired level of 127 target value.

Once the pixel value load through the (load) function and the size function helps to calculate the width and length of the picture and using for loops it iterates over each pixel, hence new value for the RGB is calculated by multiplying the original value.



Analysis:

	Target brightness						N=?
Starting brightness	100	110	127	130	140	150	
25-50	7%	0%	7%	7%	10%	10%	3
50-75	10%	6%	5%	5%	8%	10%	2
75-100	4%	4%	-2%	3%	3%	1%	10
100-125	20%	20%	6%	18%	7%	0%	5
>125	0%	0%	0%	0%	0%	0%	0
Total	10.50%	9%	5%	4.5%	7%	8%	20

Hypothesis:

- Error rates improve across all brackets of starting brightness for target = 130, 140, 100
- Found out 127 is not the optimal target brightness.
- For some brackets of starting brightness applying the target brightness results in higher OCR errors. eg: 110. 127
- The best way to define is to implement target brightness according to the starting brightness.
- Use Target Brightness 140 if starting brightness is 25 -40 and 150 if 50-75.
- For the higher starting brightness from 75- 125 use 110 as target brightness.

In this way, a jump from 4% to 10.5% in Error rate improvement has been noticed.

Issue Analysis:

In the above analysis, it's evident that enhancing image brightness increases the overall percentage of the error rate improvement but the sample dataset is not viable to proceed. So need to further proceed with the bigger dataset. In each starting image brightness category we make sure there is 10 variable in total to proceed this way it gives some confidence in the analysis of the data.

Week 12:

The goal is to redo the last week's work as the sample dataset required for the initial analysis of finding the optimal brightness, it has been found that the optimal target brightness could be improved by selecting with more reasoning.

- Define the optimal target Brightness.
- Use a sample set of 50 with different starting brightness from 30 to 150.
- Perform Brightness enhancement on pitch-dark images to find out if the black image happened due to a lens/phone issue while the user taking the document photo.

Activities: For the pitch-black image it has been found that it is not possible to increase the brightness value of a completely black (0 value) image. Since there is no data to amplify or increase. Brightness adjustments can be made to images that have existing pixel values representing different levels of brightness.

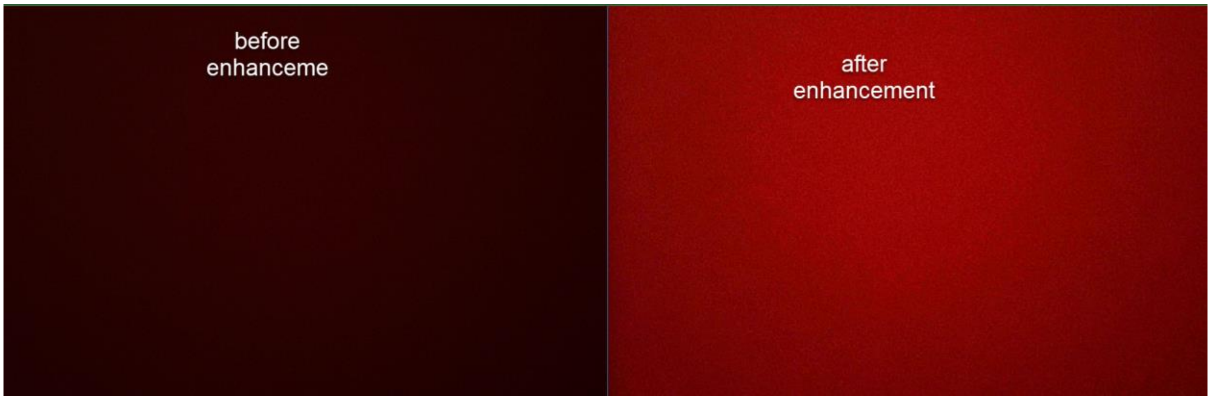


Figure 18 Analysing of the pitch black document images.

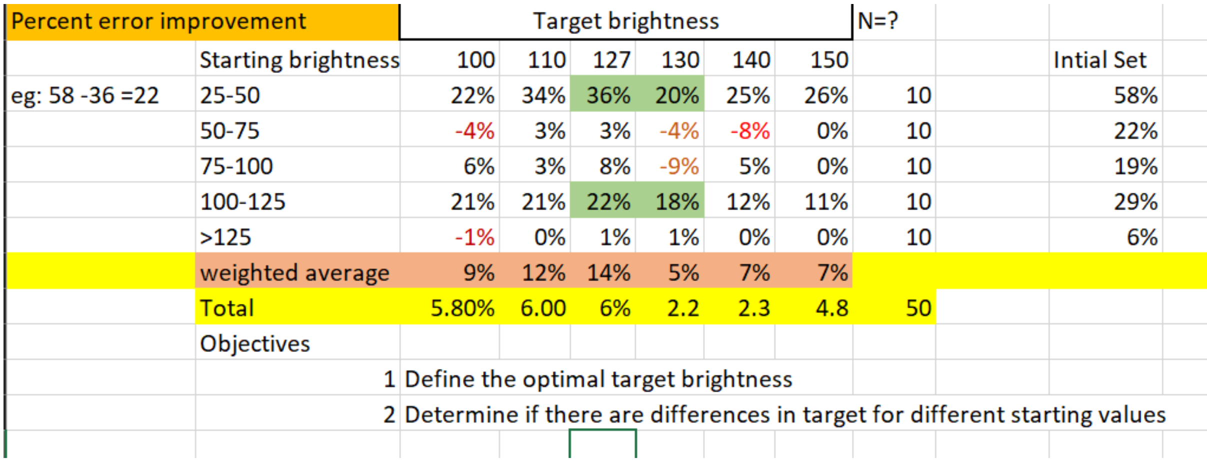
There was 2 image in each dataset with a brightness index value starting at 35 before enhancement increasing the target brightness to 127 resulted in the not expected result. Thus concluded that the document was not scanned properly by the user in the beginning and there have some other issues like lens issues and phone software-related issues in dealing with a dataset of pitch-black images received.

Another task is to increase the target brightness with the respective starting brightness, the respective column represents the error rate percentage comparing the image before pre-processing and post-processing. The target brightness is calculated with the formula as shown below.

E.g., if the image brightness index is 127, the function subtracts the image brightness from the brightness index divided by the brightness value.

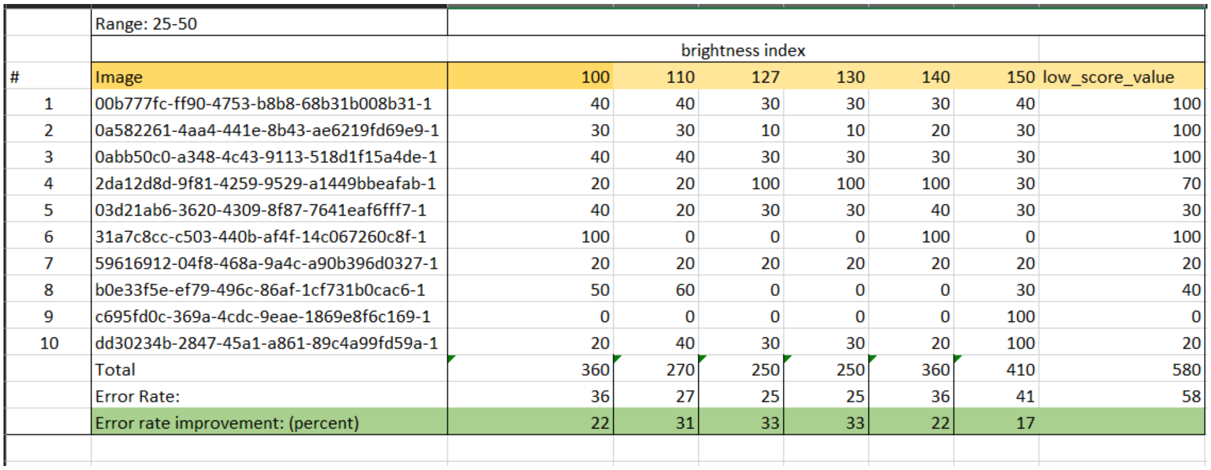
(127 - brightness value)/(Brightness Value))

where brightness index is the average image brightness which got by calculating the mean from the PIL python package ‘ImageStat’ library. The above formula determines how much brightness needs to be increased to reach the desired level of 127 target value.



Percent error improvement		Target brightness						N=?		
	Starting brightness	100	110	127	130	140	150			Intial Set
eg: 58 -36 =22	25-50	22%	34%	36%	20%	25%	26%	10		58%
	50-75	-4%	3%	3%	-4%	-8%	0%	10		22%
	75-100	6%	3%	8%	-9%	5%	0%	10		19%
	100-125	21%	21%	22%	18%	12%	11%	10		29%
	>125	-1%	0%	1%	1%	0%	0%	10		6%
	weighted average	9%	12%	14%	5%	7%	7%			
	Total	5.80%	6.00	6%	2.2	2.3	4.8	50		
	Objectives									
	1	Define the optimal target brightness								
	2	Determine if there are differences in target for different starting values								

Figure 19 Final percentage improvement.



	Range: 25-50							
		brightness index						
#	Image	100	110	127	130	140	150	low_score_value
1	00b777fc-ff90-4753-b8b8-68b31b008b31-1	40	40	30	30	30	40	100
2	0a582261-4aa4-441e-8b43-ae6219fd69e9-1	30	30	10	10	20	30	100
3	0abb50c0-a348-4c43-9113-518d1f15a4de-1	40	40	30	30	30	30	100
4	2da12d8d-9f81-4259-9529-a1449bbeafab-1	20	20	100	100	100	30	70
5	03d21ab6-3620-4309-8f87-7641eaf6fff7-1	40	20	30	30	40	30	30
6	31a7c8cc-c503-440b-af4f-14c067260c8f-1	100	0	0	0	100	0	100
7	59616912-04f8-468a-9a4c-a90b396d0327-1	20	20	20	20	20	20	20
8	b0e33f5e-ef79-496c-86af-1cf731b0cac6-1	50	60	0	0	0	30	40
9	c695fd0c-369a-4cdc-9eae-1869e8f6c169-1	0	0	0	0	0	100	0
10	dd30234b-2847-45a1-a861-89c4a99fd59a-1	20	40	30	30	20	100	20
	Total	360	270	250	250	360	410	580
	Error Rate:	36	27	25	25	36	41	58
	Error rate improvement: (percent)	22	31	33	33	22	17	

Figure 20 Calculating the error rates.

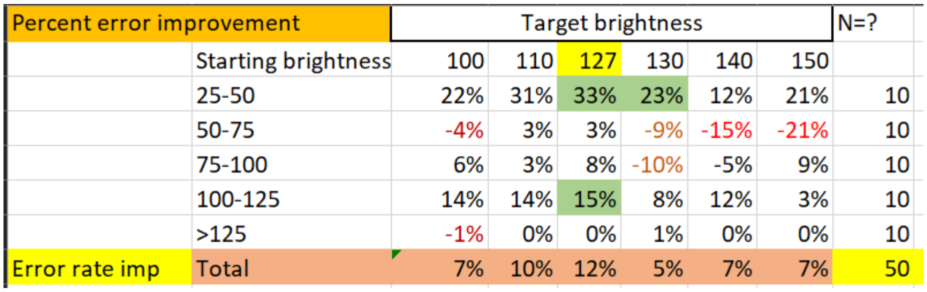
Hypothesis:

- Error rates improve majorly on optimal brightness 127
- The weighted total average of 127 sees major changes.

In this way, a jump from 6 to 14% Error rate improvement has been noticed.

Week 13:

The goal of this week is to rectify the manual error in the error rate analysis and generate the hypothesis from the inference from the table. Each of the below-starting brightness error rates from 25-50,50-75,75-100,100-125 and above is calculated the below way.

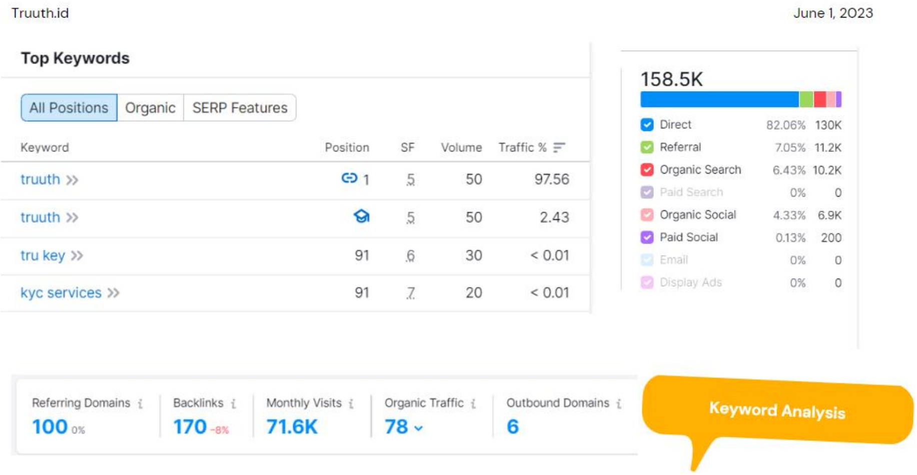


Percent error improvement		Target brightness						N=?
	Starting brightness	100	110	127	130	140	150	
	25-50	22%	31%	33%	23%	12%	21%	10
	50-75	-4%	3%	3%	-9%	-15%	-21%	10
	75-100	6%	3%	8%	-10%	-5%	9%	10
	100-125	14%	14%	15%	8%	12%	3%	10
	>125	-1%	0%	0%	1%	0%	0%	10
Error rate imp	Total	7%	10%	12%	5%	7%	7%	50

Figure 21 Overall percentage improvement.

Analysis: From the analysis it has been found that enhancing an image beyond 100 doesn't add more value and sometimes the error rate improvement declines to an extent. So further apply the enhancement to the image with brightness less than <50 as an index.

Further from the task I have also helped the organization a glimpse into the analysis of the Market performance by fetching the data curated from the Search console and providing insights on the market trends and keywords to focus to build brand awareness and further lead generation.



Truuth.id

June 1, 2023

Top Keywords

Keyword	Position	SF	Volume	Traffic %
truuth	1	5	50	97.56
truuth		5	50	2.43
tru key	91	6	30	< 0.01
kyc services	91	7	20	< 0.01

158.5K

☑ Direct	82.06%	130K
☑ Referral	7.05%	11.2K
☑ Organic Search	6.43%	10.2K
☑ Paid Search	0%	0
☑ Organic Social	4.33%	6.9K
☑ Paid Social	0.13%	200
☑ Email	0%	0
☑ Display Ads	0%	0

Referring Domains	Backlinks	Monthly Visits	Organic Traffic	Outbound Domains
100 0%	170 -8%	71.6K	78	6

Keyword Analysis

Figure 22 Keyword Analysis of the company.

As this was the last week of the internship there was a review/ feedback session organized by the company supervisor where he has given insights about the performance of the organization so far in detail and suggestions to go ahead and mentioned to summarise the overall work done on the confluence documentation page of the organization.