Smart Construction Site Monitoring using Deep Learning and UAV Technology

Our solution captures live images from site cameras and UAVs, detects cracks and defects in real-time, and displays the results on a user-friendly dashboard for efficient decision making

01. Introduction

Need for crack detection in construction sites and challenges involved

02. Data Collection

Process of collecting site photographs and importance of good dataset

03. Model Training

The deep learning model and the techniques used to train it, Including the libraries and framework used

04. Deployment

How the model is deployed on a smart camera device and considerations to be made when choosing a deployment location

05. Data Transmission

The process of sending the images to the deployed model, and the role of gateway in compression and encryption

06. Dashboard

The design and functionality of the dashboard, and the methods used to display result of the model's predictions

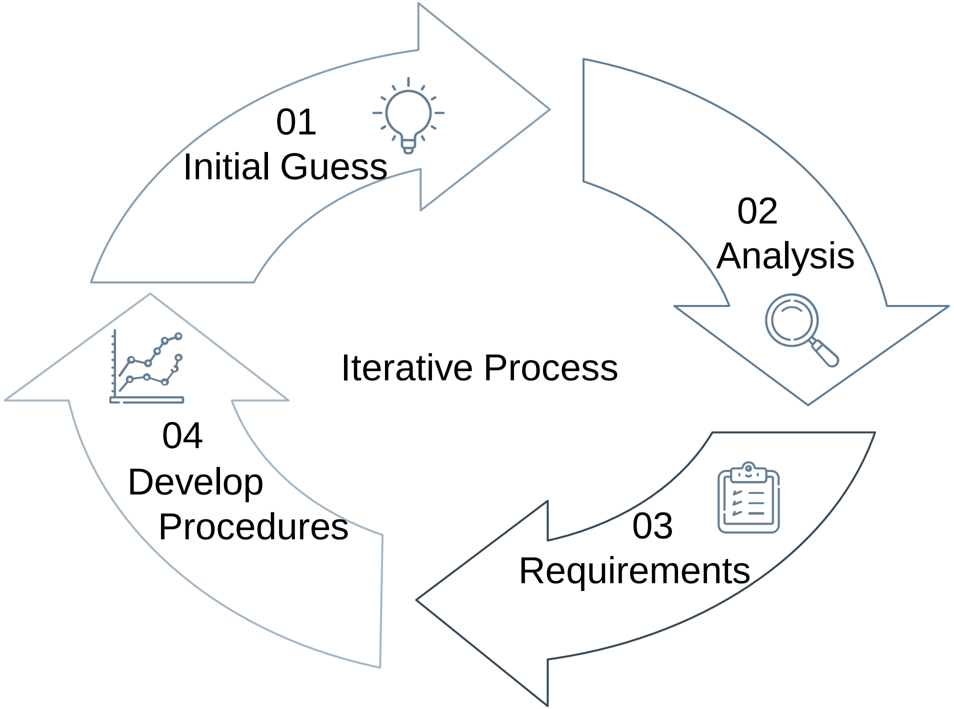
07. Improvement

The ways to improve the efficiency of the framework, including image compression and data encryption

08. Conclusion

The key points and the benefits of using deep learning for crack detection in construction sites

How to start



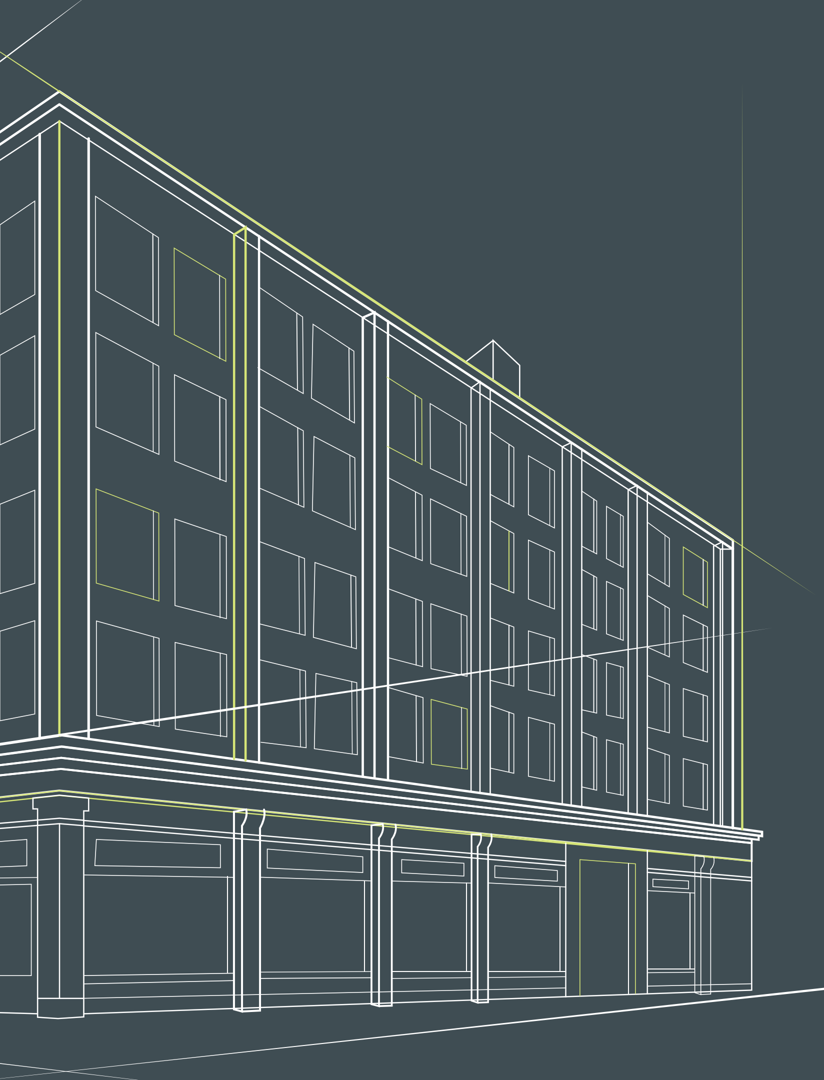
01
Initial Guess

02
Analysis

03
Requirements

04
Develop
Procedures

Iterative Process



01

Introduction

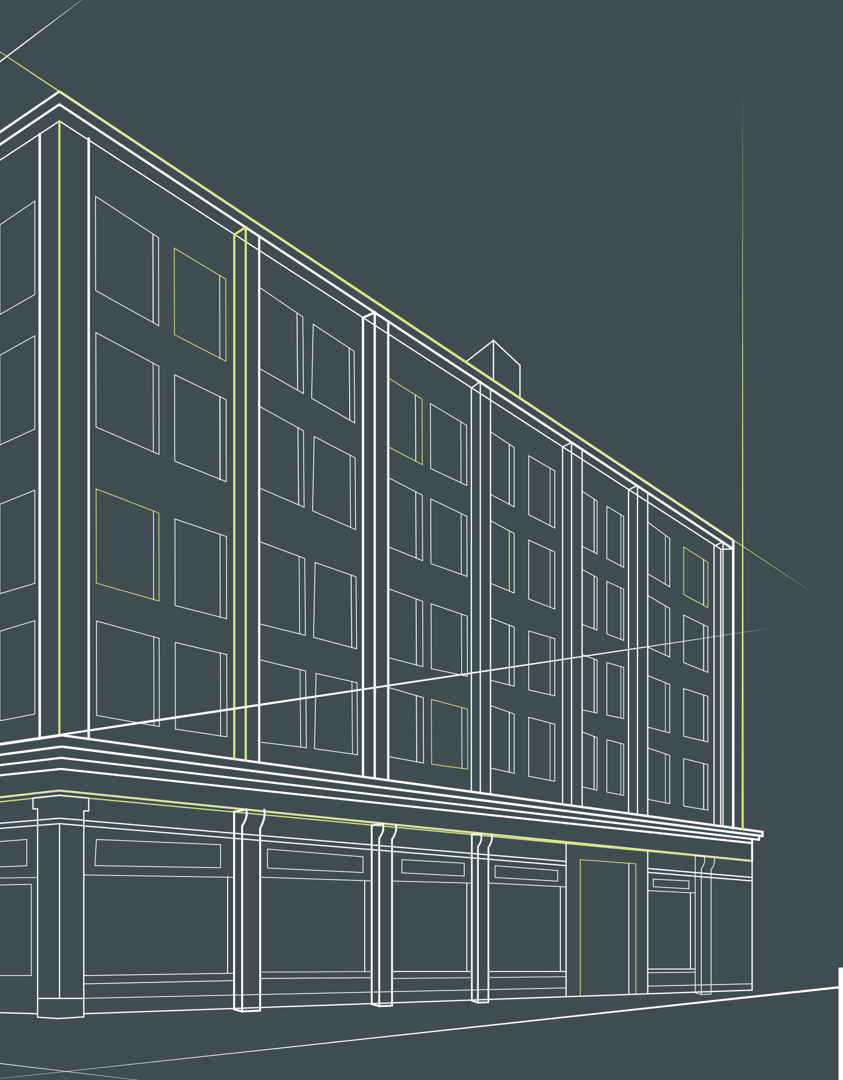
Introduction

Need for Crack Detection

- Early detection of cracks in building or infrastructure can help prevent further damage and potential collapse.
- Regular crack inspections can help identify structural ideas and guide maintenance and repair efforts.
- Traditional methods of crack detection, such as manual inspections, can be time-consuming, costly and may not provide a complete picture.
- Automated crack detection using technology such as drones and machine learning can improve the efficiency, accuracy and safety of inspections.

Introduction

- There is a need for **robust** and sophisticated **machine learning** models to be able to effectively analyze images and detect and classify cracks with high accuracy.



02

Data Collection

Data Collection

Process of collecting good dataset

- **Setting up cameras at the construction site:** Site cameras, such as UAVs or drones, are used to collect high-resolution images of construction sites, typically on a regular basis.

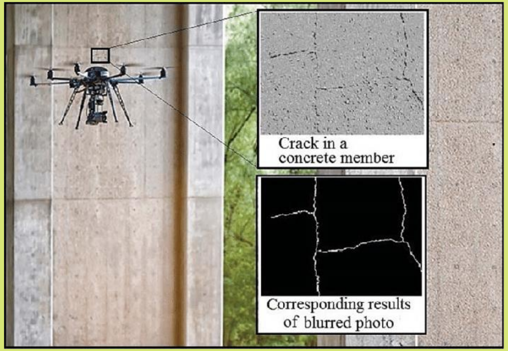
- **Capturing images:** The cameras will be set up to capture images of the construction site at regular intervals, or when triggered by certain events.

- **Transferring images to a central location:** The images will be transferred to a central location, such as computer or server , for storage and processing.

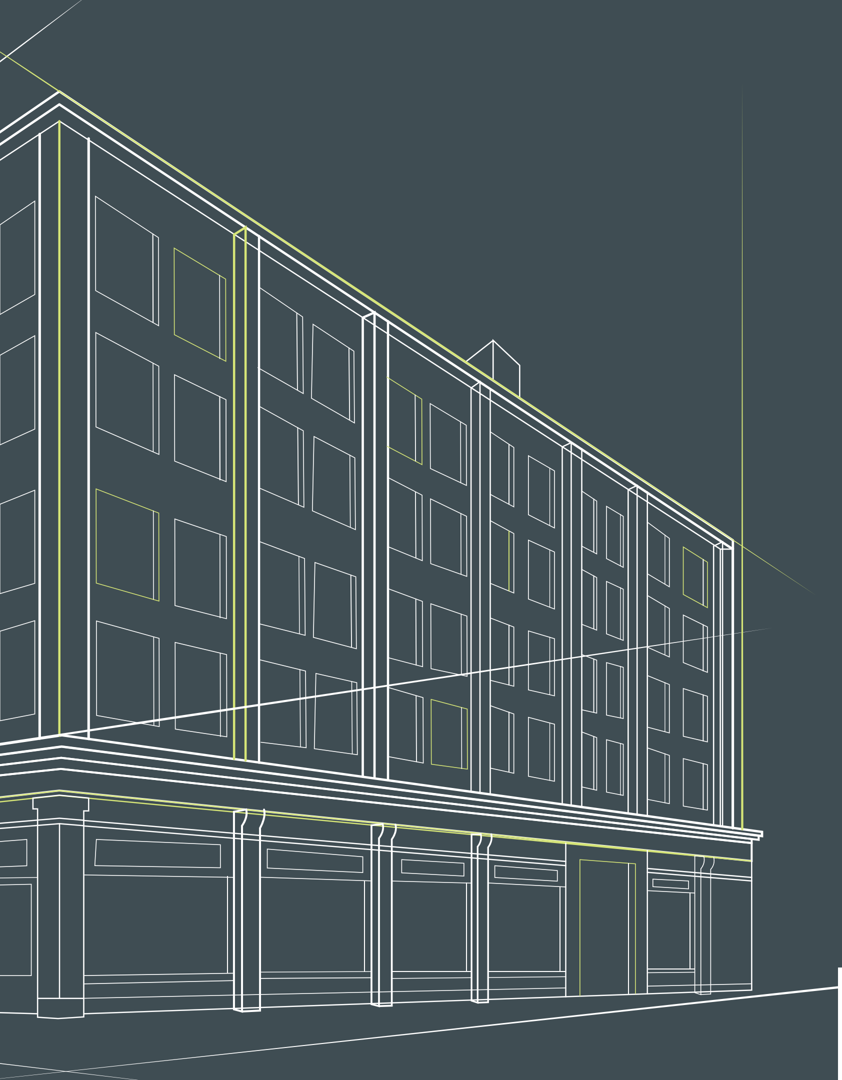
- **Preprocessing of images:** Before the images can be used for analysis , they will need to be preprocessed to remove any unwanted noise or artifacts, and to ensure that they are of a consistent size and quality.

Data Collection

- **Annotating images:** The images will be annotated with information about the location, time, and any relevant details about the site.
- **Creating a Data set:** The images and annotations will be used to create a dataset that can be used for training and testing machine learning models.


Crack in a concrete member
Corresponding results of blurred photo

It is important to have a good dataset because, a well-annotated dataset is important for accuracy training of machine learning models that can be used for crack detection. A dataset with a diverse range of images captured under different lighting conditions and with varying severity is essential to train models that can generalize well to new images



03

Model Training

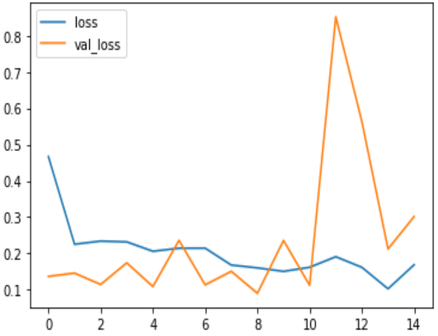
Model Training

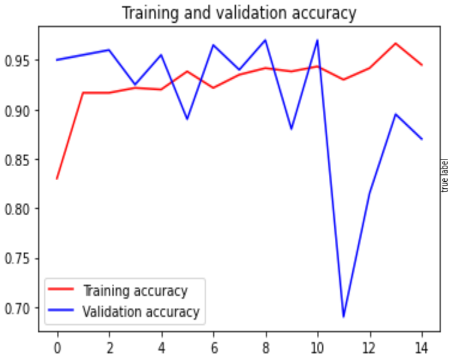
The techniques used to train deep learning models

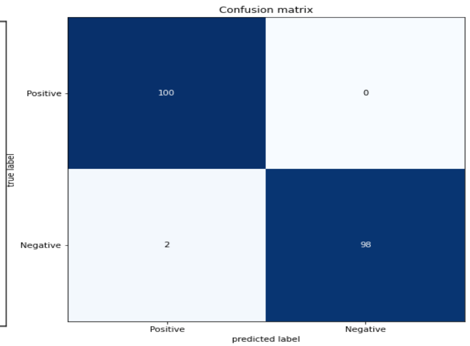
- Importing necessary libraries and frameworks (Tensorflow, Keras)
- Extracting data from training directories
- Inspecting sample images
- Preprocessing images (resizing, normalizing,etc)
- Creating the pre-trained model using transfer learning
- Downloading inception v3 weights and loading the model
- Applying data augmentation techniques using tensorflow
- Pipelining the pre-trained model with additional fully connected layers

Model Training

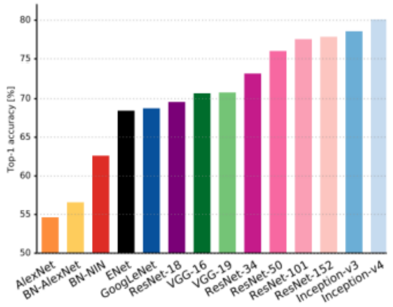
- Using softmax activation function in final dense layer
- Compiling the model using Adam optimizer
- Creating checkpoints to save the model with lowest validation loss
- Training the model on the dataset and evaluating the model’s performance

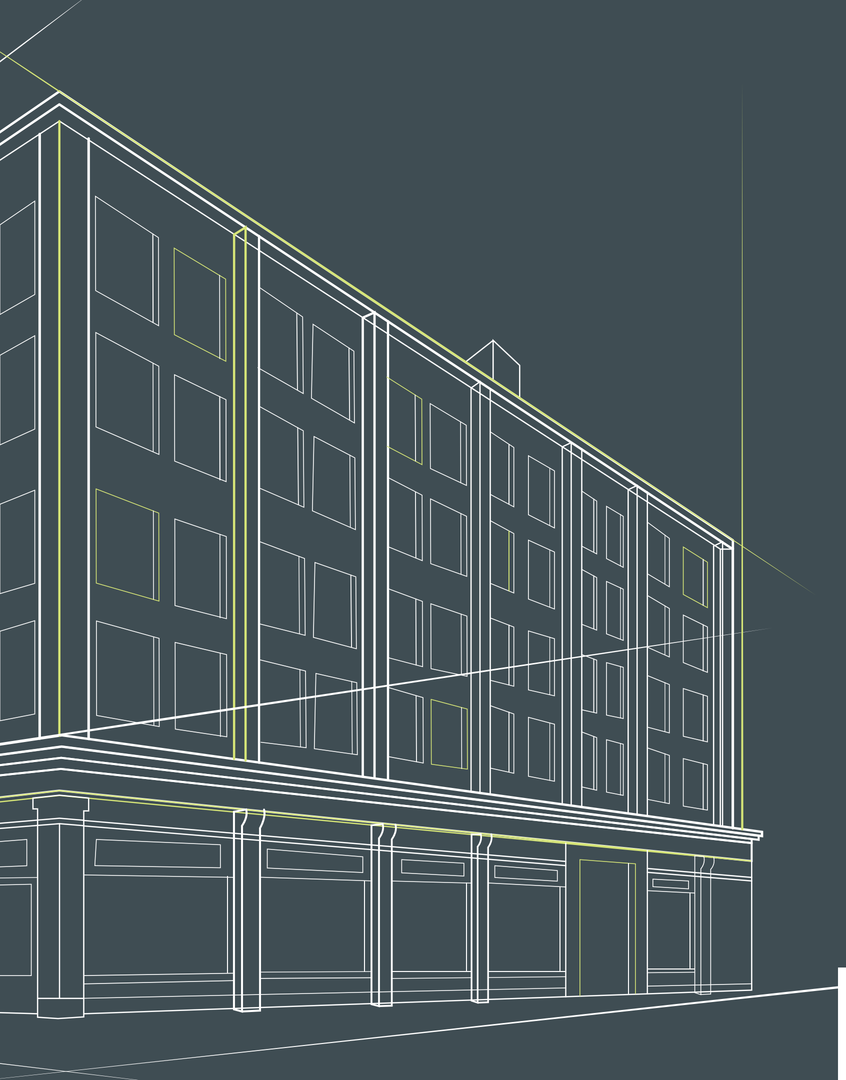






	Positive	Negative
Positive	100	0
Negative	2	98



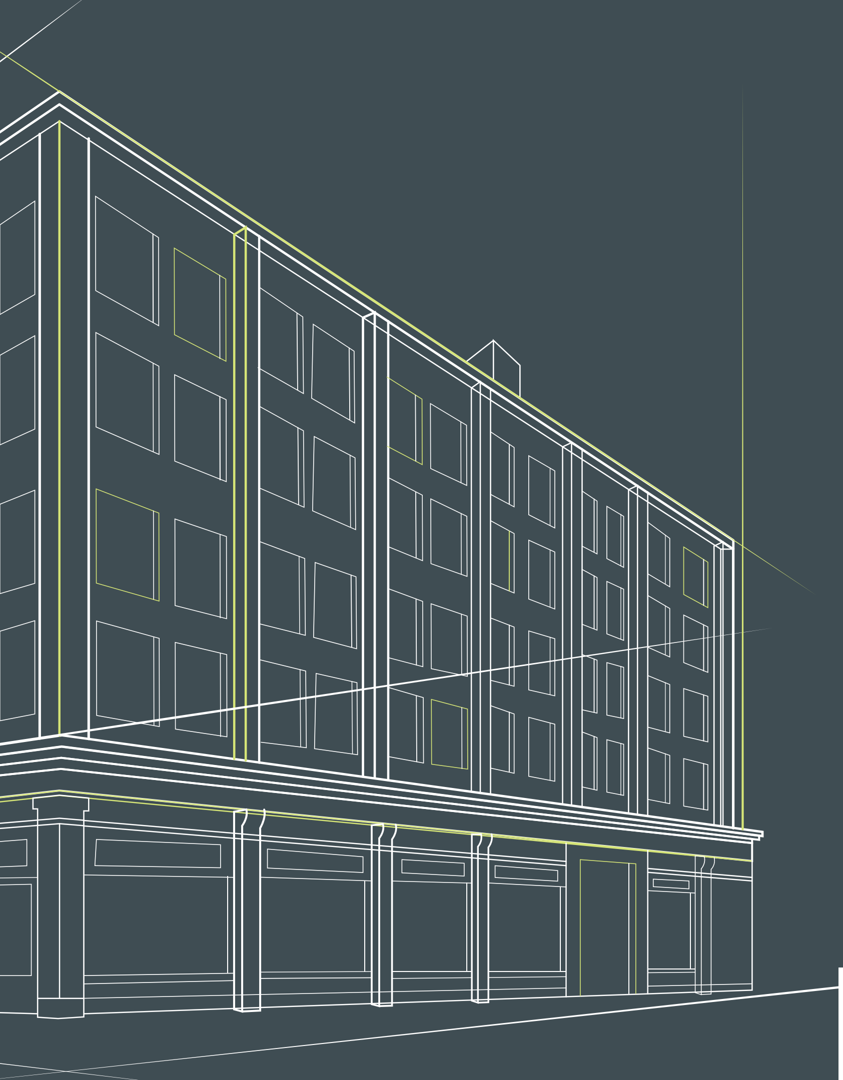


04

Deployment

Deployment

- **Scalability:** AWS allows for easy scaling of resources, such as memory and computing power to handle changes in traffic and usage
- **Reliability:** AWS offers a wide range of services and features to ensure high availability and fault tolerance, such as auto-scaling and load balancing
- **Ease of management:** AWS provides a web-based management console and various tools to easily manage and monitor the deployed resources and services
- **Security:** AWS offers a variety of security features and compliance certifications to help the secure the deployed resources and data
- **Global coverage:** AWS has a global infrastructure with availability zones and regions, which enables to deploy and run your application in multiple locations**.**



05

Data Transmission

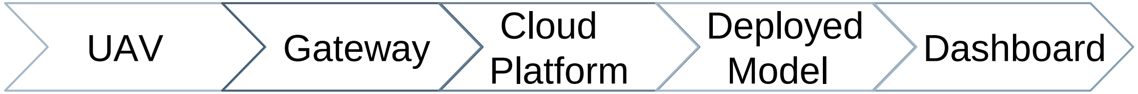
Data Transmission

- The smart camera on the UAV capture images of the construction site
- The images are sent to a gateway device, which is responsible for compressing the images to reduce transmission time and cost
- The gateway device also encrypts the images to ensure secure transmission
- The compressed and encrypted images are then sent to the deployed model, which is hosted on a cloud platform such as aws
- The deployed model processes the images and sends back the classification results to the gateway
- The gateway then decrypt the classification results and sends it back to the UAV or any other device which is connected to it.

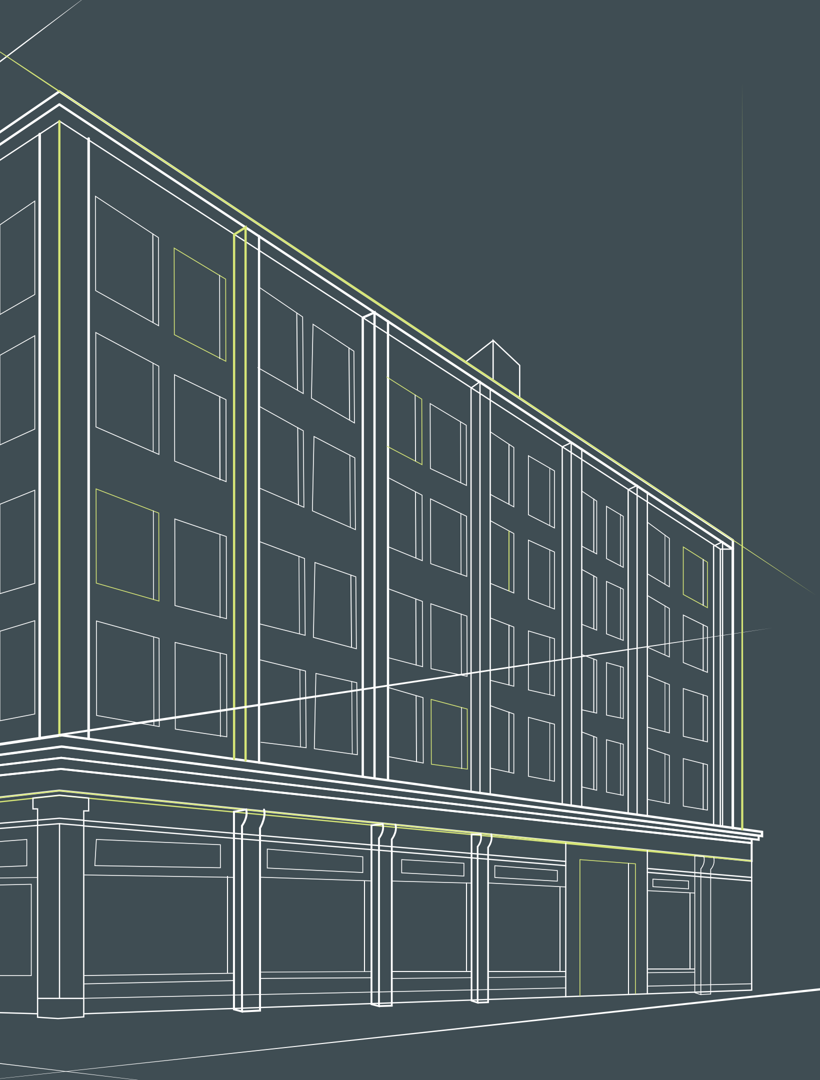
Data Transmission

How to create gateway in such project

- Choose a suitable gateway device such as Raspberry Pi or Intel NUC
- Install necessary software on the gateway device such as an operating system, SSH, and any necessary libaries for compression and encryption
- Configure the device to connect to the deployed model and the UAV device’s smart camera by using the protocol such as **MQTT** or **HTTP** to send the images to the deployed model
- We can write a script to handle the compression and encryption of the images before sending them to the deployed model
- Integrate the gateway into the overall system architecture, ensuring proper communication and data flow between the UAV, gateway , and deployed model.



UAV → Gateway → Cloud Platform → Deployed Model → Dashboard



06

Dashboard

Dashboard

- ➢ Gradio is a web-based tool that allows for the deployment of machine learning models as interactive web apps
- ➢ It is easy to use and allows for the creation of a user-friendly interface with the model
- ➢ The deployed model can be accessed through the dashboard by uploading an image for classification or by using a live webcam feed
- ➢ The dashboard displays the image along with the classification results, such as the predicted class and probability scores
- ➢ The dashboard also has the ability to save classified images for future reference and to view previous classifications
- ➢ The dashboard can be accessed remotely through cloud servers such as **AWS**, allowing users to access the model from anywhere with an internet connection.

Dasboard

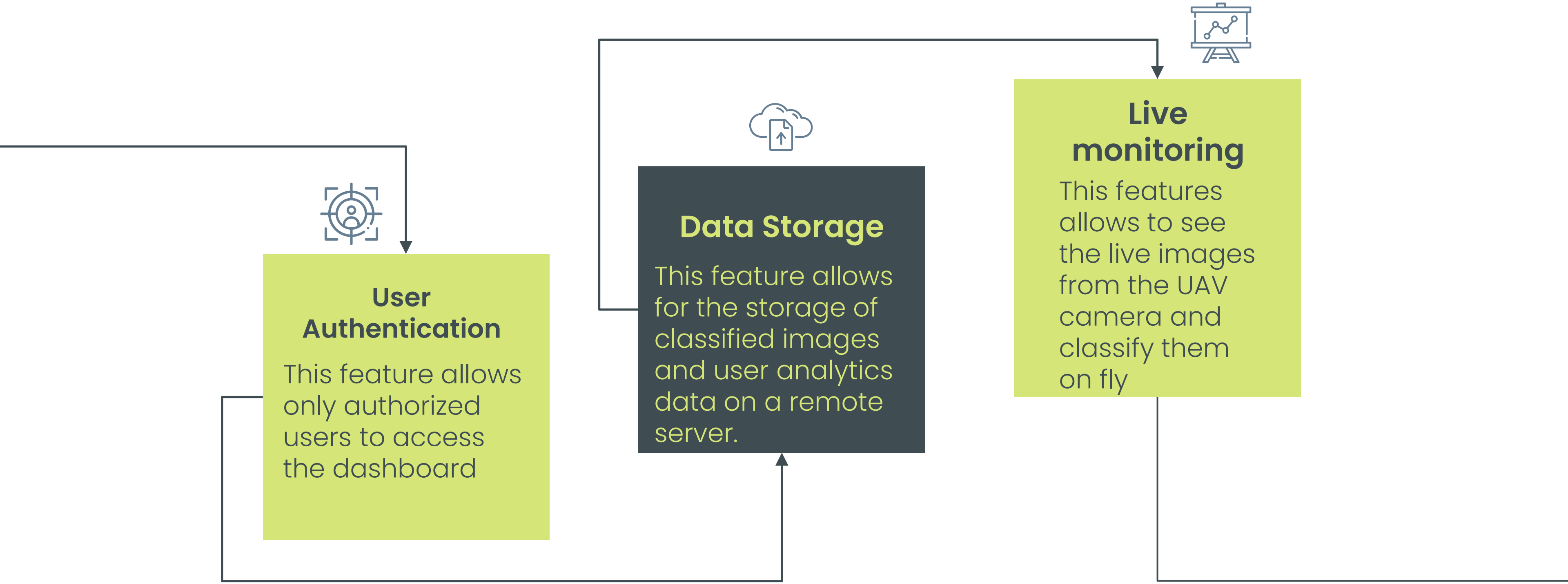
Why only Gradio ?

- Gradio is an open-source framework that allows developers to easy build and deploy machine learning models as web applications. It provides a simple and intuitive interface for users to interact with the model, making it a great choice for creating a dashboard for a project

- In the case of a crack detection project, Gradio can be used to create a dashboard that allows users to upload images of construction sites, view the model's prediction for the presence of cracks, and download the images with the crack regions highlighted. The dashboard could also include options for users to adjust the model's threshold for crack detection, view statistics on the number of images processed, and access to the historical data.

Overall, Gradio allows developers to easily create a user-friendly and interactive dashboard that effectively communicates the results of the model to users and makes the process of using the model more efficient.

Dashboard Stages



User Authentication

This feature allows only authorized users to access the dashboard

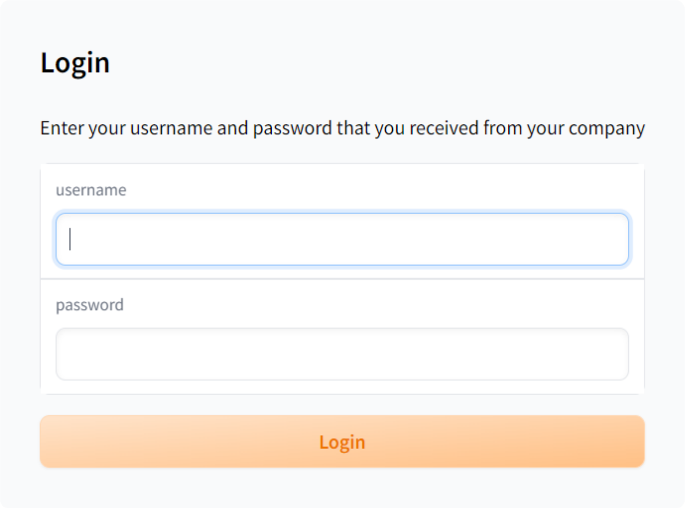
Data Storage

This feature allows for the storage of classified images and user analytics data on a remote server.

Live monitoring

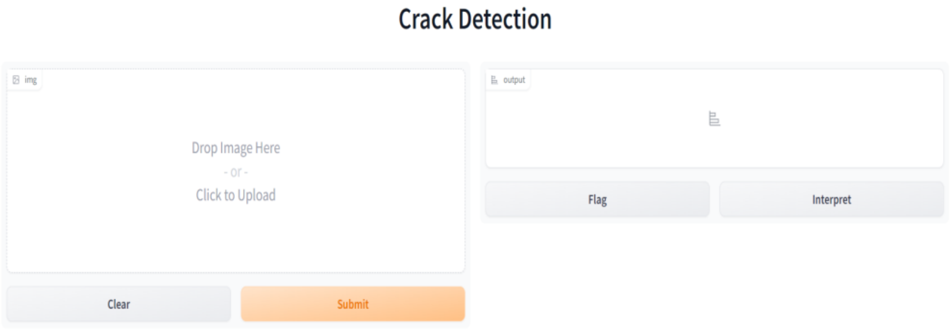
This features allows to see the live images from the UAV camera and classify them on fly

Dashboard



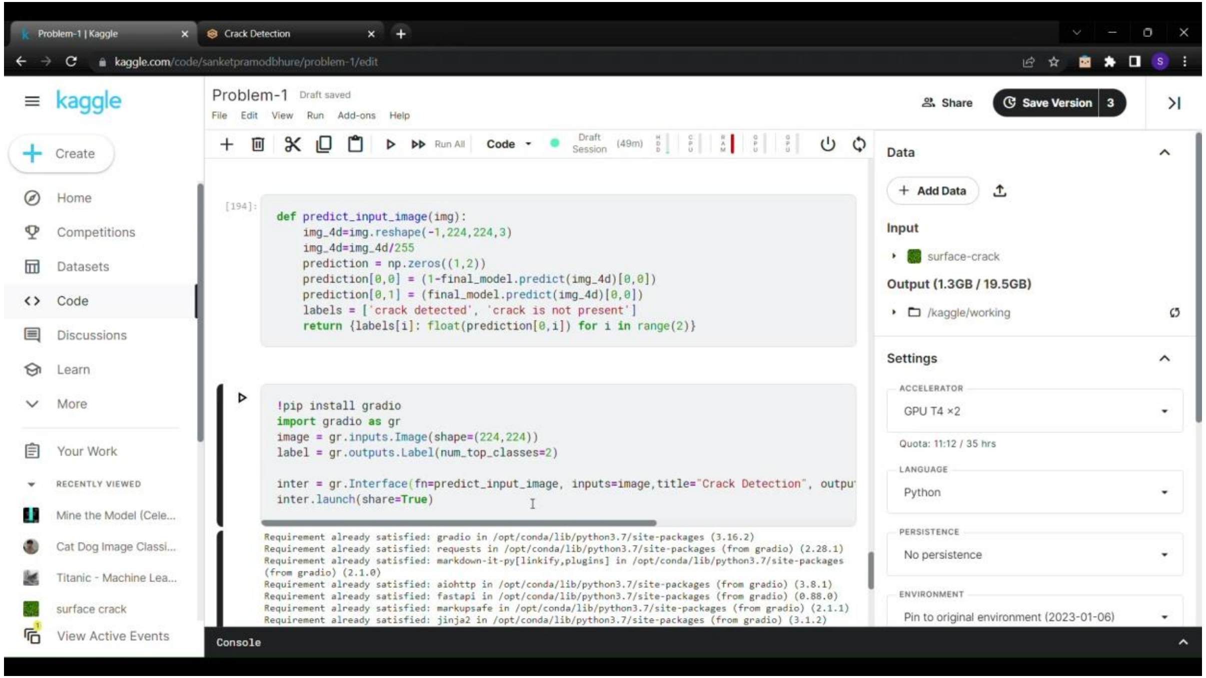
Login

Enter your username and password that you received from your company



Crack Detection

Dashboard



```
[194]:
def predict_input_image(img):
    img_4d=img.reshape(-1,224,224,3)
    img_4d=img_4d/255
    prediction = np.zeros((1,2))
    prediction[0,0] = (1-final_model.predict(img_4d)[0,0])
    prediction[0,1] = (final_model.predict(img_4d)[0,0])
    labels = ['crack detected', 'crack is not present']
    return {labels[i]: float(prediction[0,i]) for i in range(2)}
```

```
!pip install gradio
import gradio as gr
image = gr.inputs.Image(shape=(224,224))
label = gr.outputs.Label(num_top_classes=2)

inter = gr.Interface(fn=predict_input_image, inputs=image,title="Crack Detection", outpu
inter.launch(share=True)
```

```
Requirement already satisfied: gradio in /opt/conda/lib/python3.7/site-packages (3.16.2)
Requirement already satisfied: requests in /opt/conda/lib/python3.7/site-packages (from gradio) (2.28.1)
Requirement already satisfied: markdown-it-py[linkify,plugins] in /opt/conda/lib/python3.7/site-packages
(from gradio) (2.1.0)
Requirement already satisfied: aiohttp in /opt/conda/lib/python3.7/site-packages (from gradio) (3.8.1)
Requirement already satisfied: fastapi in /opt/conda/lib/python3.7/site-packages (from gradio) (0.88.0)
Requirement already satisfied: markupsafe in /opt/conda/lib/python3.7/site-packages (from gradio) (2.1.1)
Requirement already satisfied: jinja2 in /opt/conda/lib/python3.7/site-packages (from gradio) (3.1.2)
```

ACCELERATOR: GPU T4 ×2

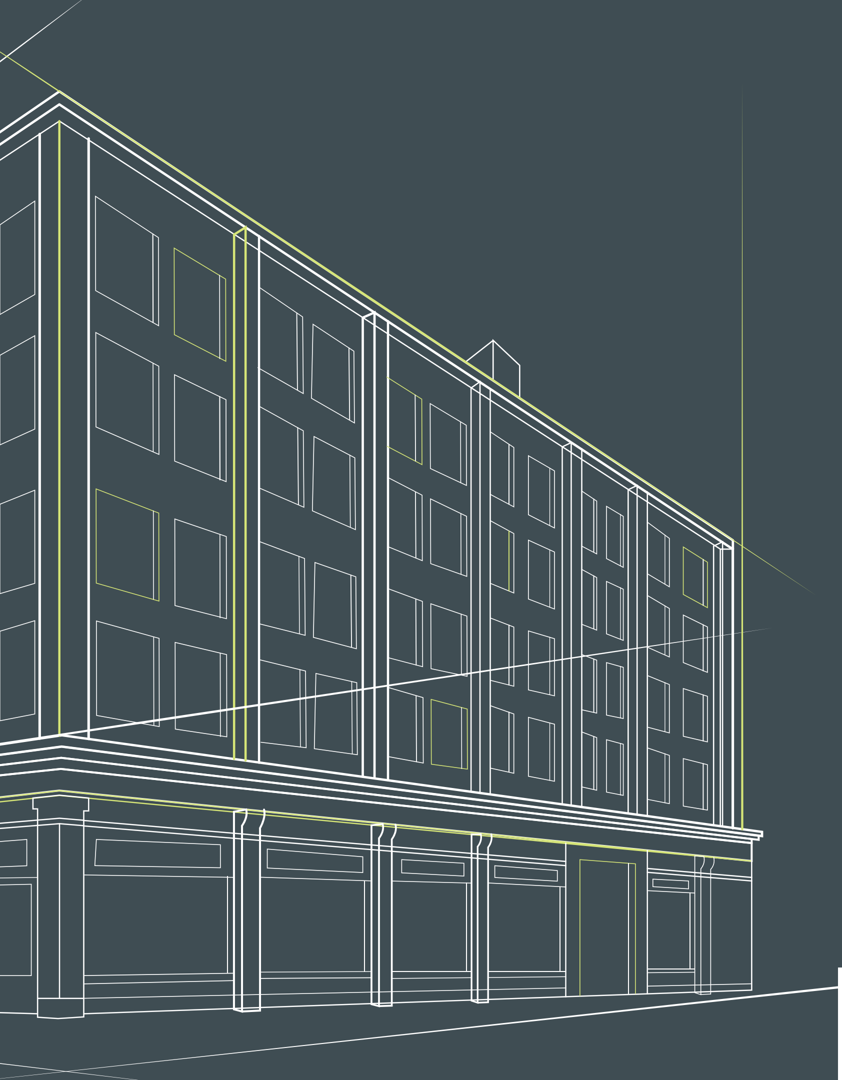
Quota: 11:12 / 35 hrs

LANGUAGE: Python

PERSISTENCE: No persistence

ENVIRONMENT: Pin to original environment (2023-01-06)

Crack detection dashboard



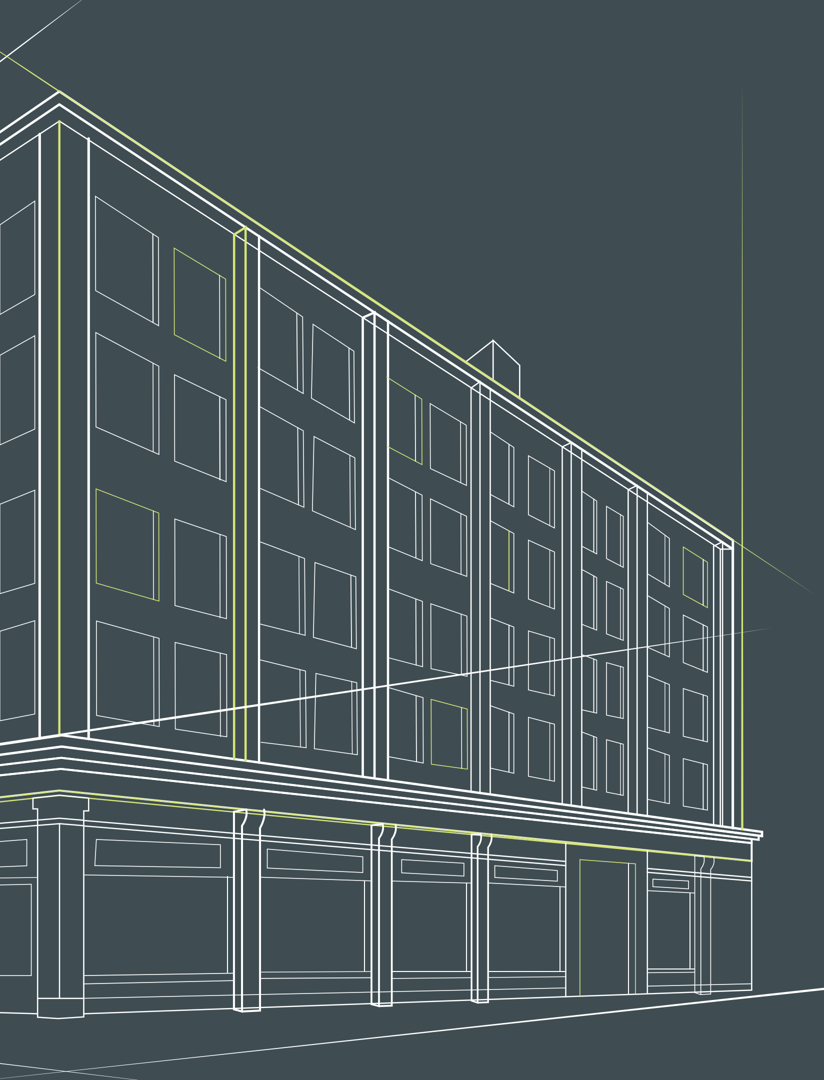
07

Improvement

Improvement

Some future plans that can be considered to make the project idea more perfect:

- Incorporating more advanced deep learning models such as R-CNN for better accuracy and faster image processing.
- Implementing real-time object detection and tracking to monitor the progress of the construction site and identify potential issues more quickly
- Integrating machine learning learning algorithms to predict the likelihodd of future defects based on historical data and patterns.
- Developing a mobile app to access the dashboard and receive alerts on the go, allowing for more efficient and convenient site monitoring.
- Creating a cloud-based system to store and analyze large amounts of data and make it accessible to multiple users.
- Testing the solution on different types of construction sites and gathering feedback from users to improve the system further.



08

Conclusion

Conclusion

- ➢ The accuracy of the model is crucial for the success of the system, and it may require fine-tuning the model, and testing the model on different scenarios, to ensure that it can detect cracks effectively.

- ➢ Using pre-trained models, such as models trained on large image datasets like inception v3 , as a starting point for the CNN model, can help improve the model's performance and reduce the amount of data and computational resources required for training

However, challenges such as variability in lighting, weather conditions, and the need for high-resolution images can make automated crack detection difficult.

Thanks!

Do you have any questions?
ae20b108@smail.iitm.ac.in
+91 8010602448

Team – SANKY18