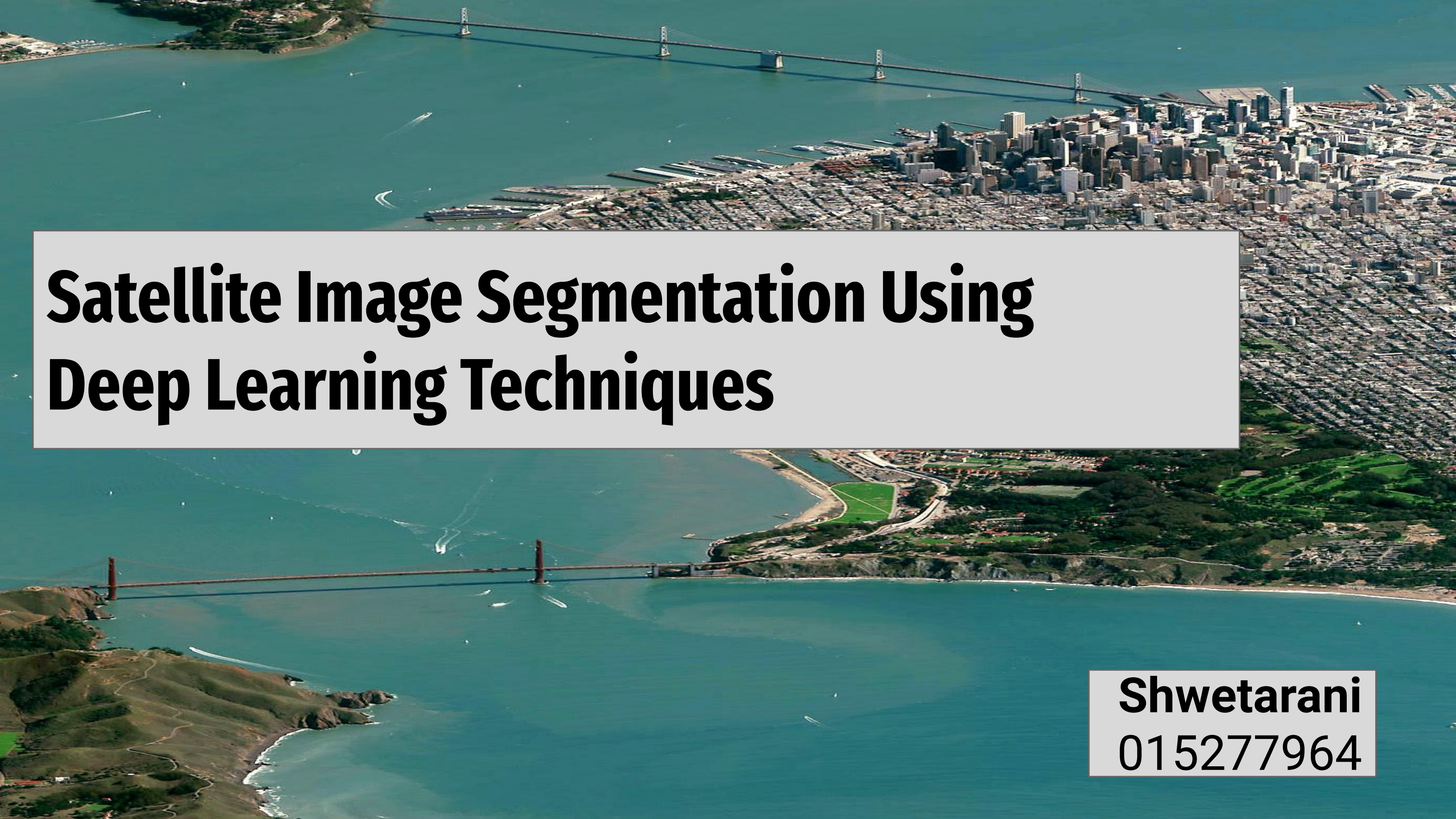


Satellite Image Segmentation Using Deep Learning Techniques

Shwetarani
015277964

Overview

- ➢ Project Background and Objective
- ➢ Project Requirements
- ➢ Technology and Literature Survey
- ➢ Project Resource Requirement and Plan
- ➢ Project data Preparation
- ➢ Machine Learning Models

Project Background

What's the first thing we do when we see an image ?

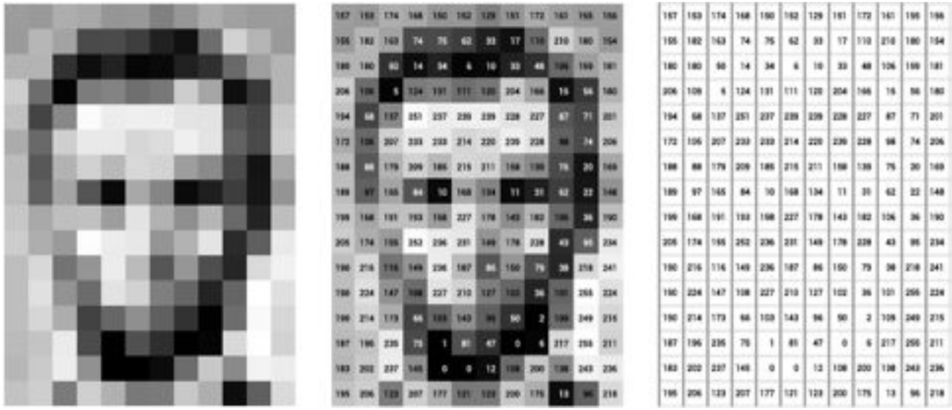
- ➢ Recognize the objects in the image
  - ○ Street/Signs/buildings
  - ○ Vehicles
  - ○ People
- ➢ Understand the context
  - ○ People crossing the road

Is it possible to do the same for the computers ???

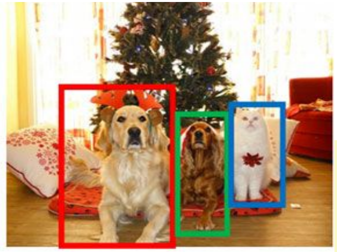
Yes, using ***Computer Vision***

How does computer vision work ??

- ➢ Image is an array of numbers
- ➢ There is an inherent pattern stored in the image depending on the objects present.
- ➢ Patterns are inturn derived from the features present in the image
- ➢ It uses convolution neural networks to recognize the features in the image.

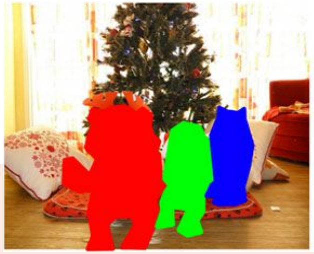


Applications of Computer Vision



Object Detection

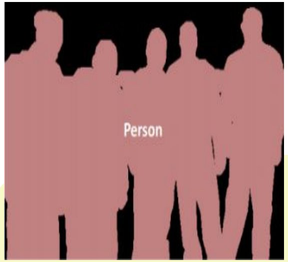
- Detect Objects of different Classes
- Bounding Box to each Class in the image
- Tells nothing about shape of the object



Segmentation

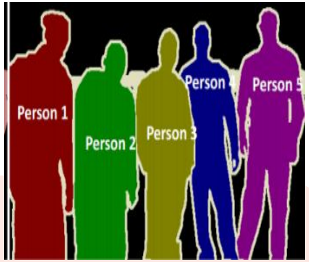
- Grouping together all pixels of similar attributes
- Pixel wise mask for each object in the image
- Tells about shape of the object
- Granular understanding of the object in an image

Image Segmentaion



Semantic Segmentation

- Every pixel belongs to a particular class
- Particular class represented by the same color (person: pink and background: black)



Instance Segmentation

- Particular class to each pixel
- Different objects of the same class have different colors
- Segments objects of the same class

In this project, we follow Semantic Segmentation!!!

Executive Summary

Project Motivation and Needs

Motivation

- Automate satellite image processing
- Huge, difficult to monitor manually

Needs

- Monitor Environmental changes
- Landscape Analysis
- Research purposes like oceanography
- Open roof space for solar panel industry.

Targeted Problem

- Satellite image segmentation (semantic) using Deep learning Techniques
- Multiclass segmentation

Deliverables

- Research paper on the problem
- PPT on implementation
- Prototype and a simple deployment model (Heroku)

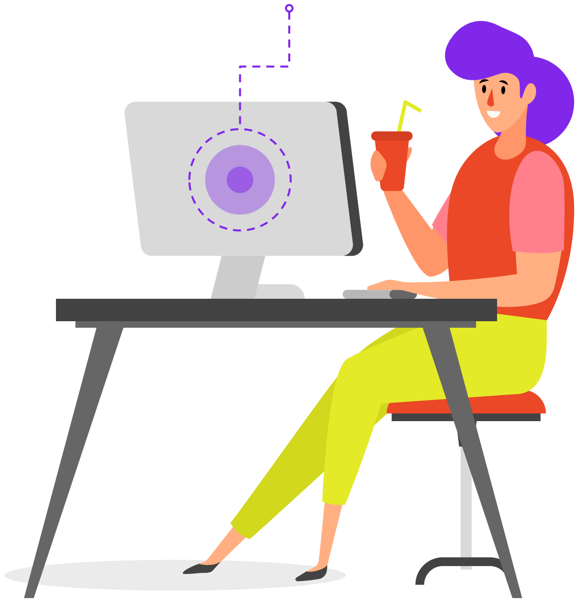
Objective of the Project

Semantic Segmentation of the Satellite Image by Using Deep learning Techniques

- Multiclass Segmentation
  - Buildings
  - Roads
  - Trees

Project Requirements

Analyze Requirements



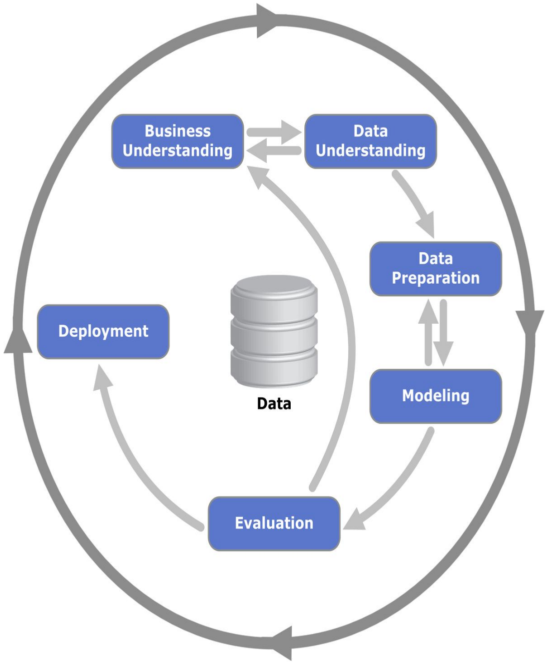
Dataset Requirements

- ➢ Collect Satellite images
  - ○ Kaggle data
  - ○ Existing Databases
- ➢ Collect Raw Satellite Data
  - ○ Use MapBox API
  - ○ Nasa earth data

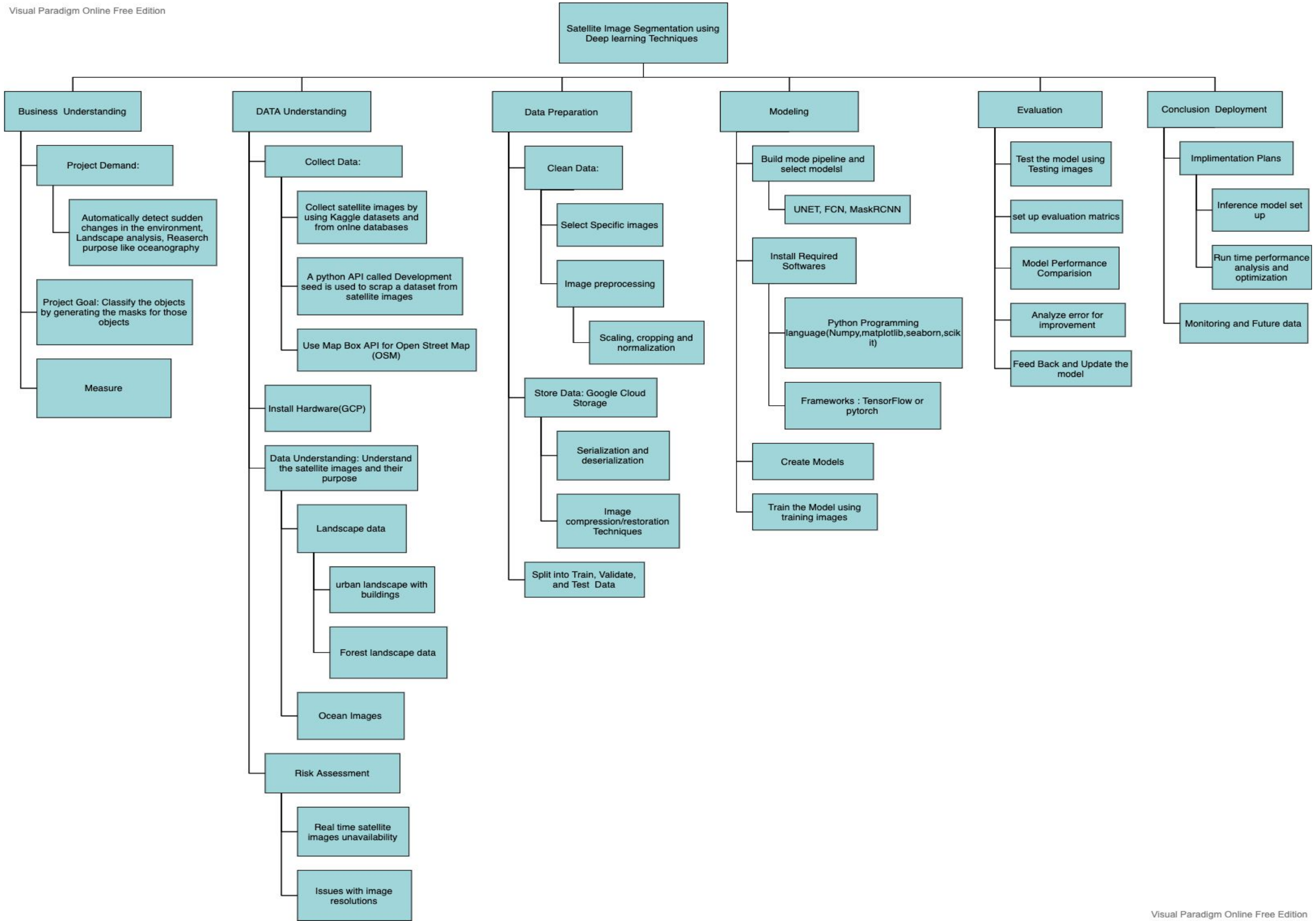
Functional & AI requirements

- ➢ AI models
  - ○ FCN
  - ○ U-NET
  - ○ LinkNet
- ➢ Google Cloud Platform
  - ○ Deep Learning VM
  - ○ DataFlow (Jupyter Lab)

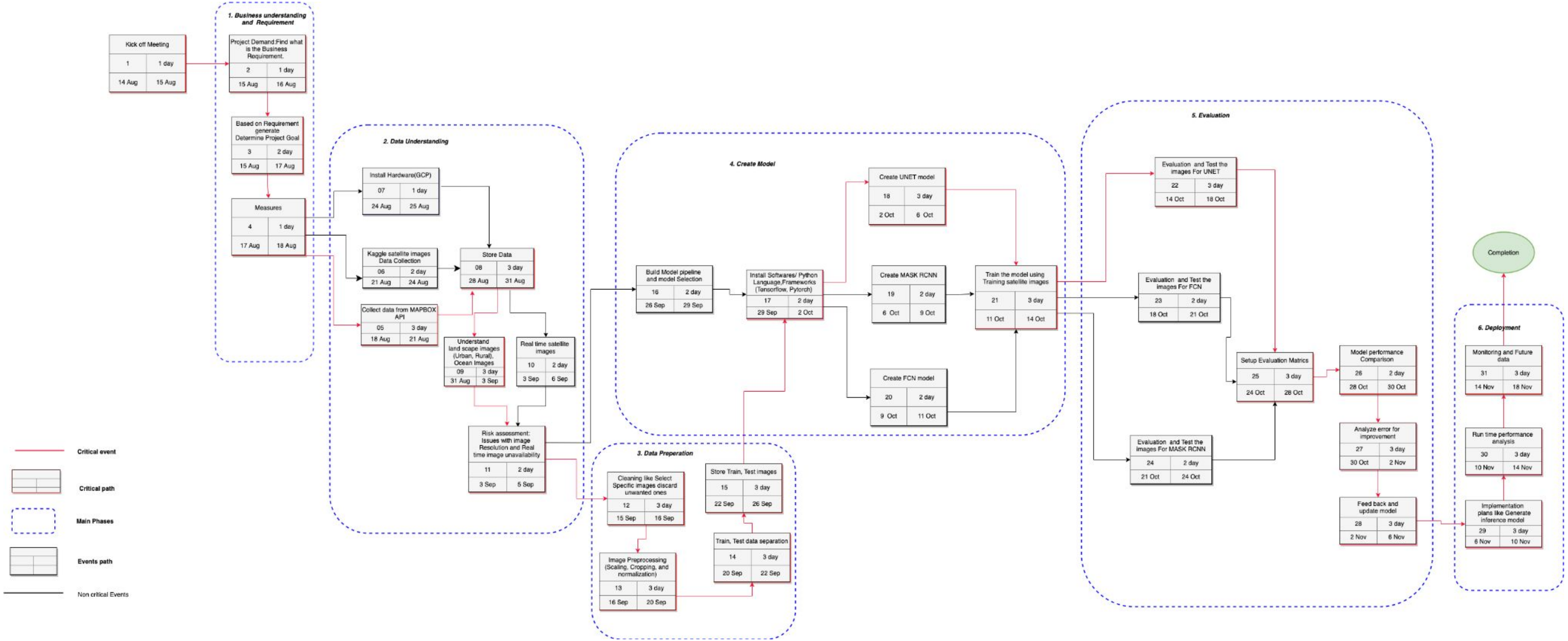
Project Organization : Work Breakdown structure



CRISP-DM Model



Project Management : PERT Chart



Technology and Literature Survey

Summary

➢ Recently most of the research papers have solved satellite image segmentation problem by using Deep Learning Technique

➢ FCN, U-NET, LinkNet, SegNet, and MaskRCNN Deep Learning Models are used in the many recent research papers

➢ While digging more, got to know that some papers also solve the segmentation problem by using manual Feature extraction technique

➢ Apart from Feature extraction, unsupervised and semi-supervised methods like clustering (K-means) and Random Walker methods are used to segment images

➢ For land cover classification of multispectral satellite images includes supervised classifiers such as support vector machine (SVM) , conditional random fields (CRF), and Random Forest (RF).

Technology and Literature Survey (contd.)

Comparison Between the Technologies

- Traditional ML methods depends on manual feature extraction
  - Difficult to automate
  - Hard to generalize
- Neural Networks are more automatic and can be generalized by having good diversity in dataset
- Auto encoder / decoder style Conv networks are better suited because they only deal with only images
  - FCN: Uses input arbitrary size and produce corresponding sized output
  - U-NET: Which is a FCN with extra skip connection concept. Uses less train data
  - LinkNet: Extension of U-NET, uses less parameters while training
  - MaskRCNN: This architecture is used to segment both semantic and instance segmentation. But need more training images

Project Resource Requirements and Plan

Resource requirements

Hardware Requirements

Google Cloud Platform:
GPUs (NVIDIA A100)
CPU AMD EPYC 7003 (Server with CPU)
Storage 100GB

Software Requirement

OS Linux
Python 3.9.5 with ML lib (Numpy, Pandas, Matplotlib, Scikit etc)
Tensor Flow, Keras

Tools and License

Google Cloud storage/instance services and APIs
Tableau, Voila, Anaconda License

Resources, Specifications, Costs, and Justification

S.no	Resources	Specifications	Costs	Justification
1	Google cloud computing resources	Google cloud GPU instance, Google Cloud TPU instance	Free tier	Models are very compute-intensive, so we are using either GPU or TPU clusters instead of CPUs. GPU: NVIDIA® Tesla® A100: 40 GB of HBM2e, 6,912 CUDA cores, 432 Tensor Cores, PCIe 4.0. TPU: TPU v3 16nm clock speed 940 MHz 32 GB HBM.
2	Google cloud storage resources	10 GB per month to store all images and software models.	Two dollars per month over a period of six months ($12)	Including the software model and the datasets as per the Google cloud cost calculator, It costs around two dollars per month, and this subscription is continued over a period of six months which is assumed to be the project duration.
3	Deploying google cloud models	Heroku	Free tier	We can deploy web apps by using Heroku on google cloud storage.

Project Data Preparation

Data Collections & manage collected data 01

- ➢ Collect Existing and raw satellite images
- ➢ Store by assigning image ID, label ID, Class of the image
- ➢ Create dictionary based on geo location

Data Preprocessing 02

- ➢ Exploration
- ➢ Cleaning like Resizing
- ➢ Discard unwanted data (Statistics)
- ➢ Convert to RGB image

Data Transformation 04

- ➢ Augmentation
  - ○ Flipping
  - ○ Rotation
  - ○ Adj.Brightness
  - ○ Mirror

Data-Preparation for labelling 05

- ➢ Collecting Labelled images already existed
- ➢ Manually label the images

Data-preparation for test, train, and validate sets 06

- ➢ Split train, test, and validate sets 75:15:10 ratio
- ➢ Cross validation method
- ➢ Region based splitting

Data Statistics 07

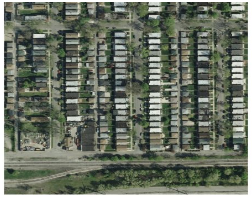
- ➢ Training data statistics of all classes by mean, median and Std

Data Collection summary and parameters

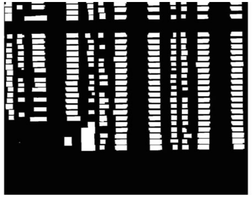
Name of the data	Data source	Data type	Size	Time stamp
Semantic Segmentation Of Aerial Image dataset	Kaggle	unit8	Average image size is about 868.56 KB and the total dataset is about 287 MB	2020-05-29 to 2020-05-29
Satellite Images Of Water Bodies	Kaggle collected by using Sentinel-2 satellite	unit8	The average image size is about 194.63 KB and the total dataset is about 34 MB.	2020-05-17 to 2020-05-17
Satellite Building Semantic Segmentation Data	Kaggle	float32	The average image size is about 128 KB and the total dataset is about 921 MB	2020-11-06 to 2020-11-24
Satellite Images Of Buildings	Inria Database	float32	180 color image tiles of size 5000×5000	2017
Satellite Images Of Buildings and Roads Networks	SpaceNet Database collected by using Worldview-2 and Worldview-3 satellites.	float32	67,000 square km of very high-resolution imagery.	2018
Vignette Landsat dataset	This dataset is collected from Landsat 8.	int16	Contain images of size 2158 * 1941	2017-02-21 to 2021-02-21
Raw Satellite Images (all kinds)	NASA earth data (GIBS)	float32	900 images, 1 MB each	Up To Date
Artificially Generated Satellite Images	MapBox API	float32	25000 images, 120 KB each	2021-08-21 to 2021-09-02

Satellite Image Data Samples

Collected Satellite image samples from various sources

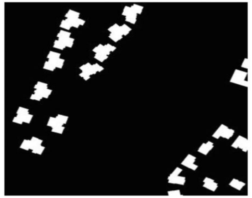


Chicago



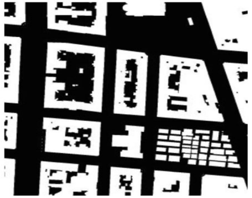
Chicago – reference

Kitsap County, WA

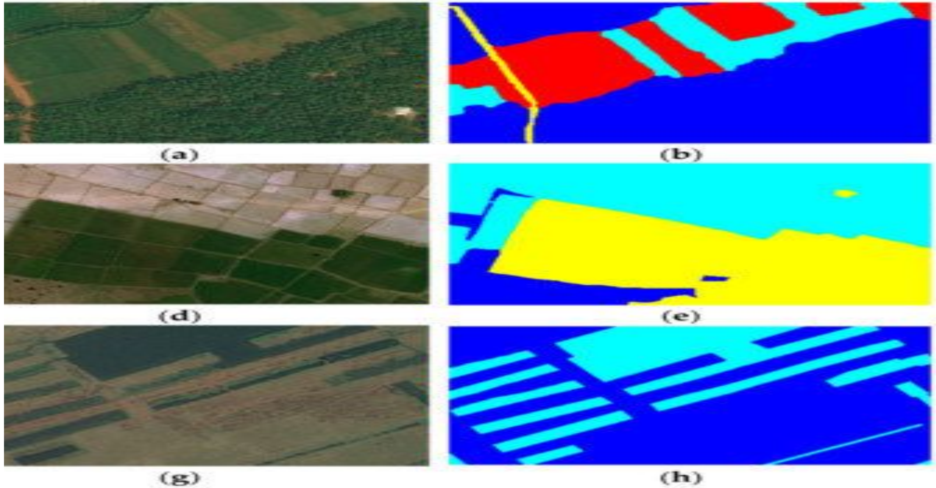


Kitsap County, WA – Reference

Vienna

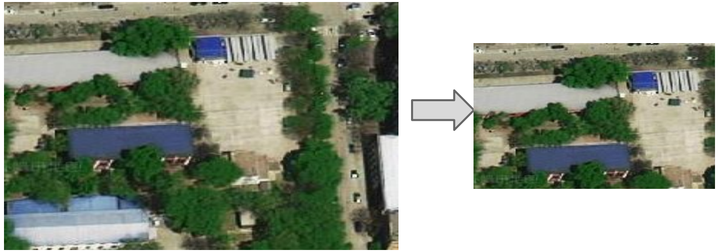


Vienna – Reference

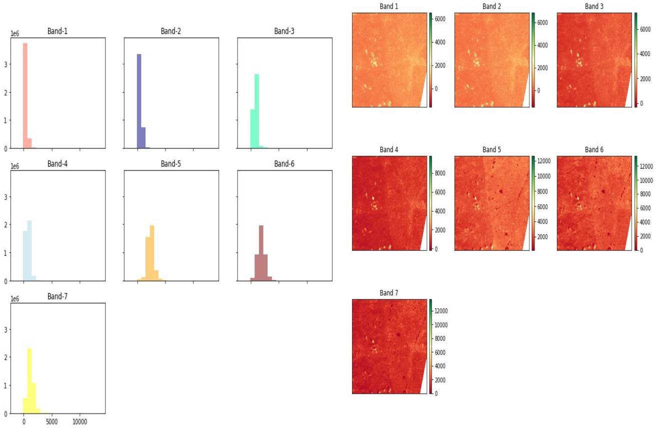


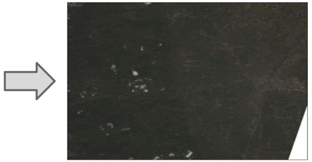
Data Preprocessing (Data Exploration and Cleaning)

Image resizing 1000 x 1000 to 256 x 256



Convert to RGB channel





RGB image

- ➢ **Convert Data type**
  - Images pixels are in different datatype convert all to float32
- ➢ **Discard unwanted images**
  - By calculating stats like mean, median and standard deviation of all images
  - Filter the outliers
- ➢ **Convert all the image bands to RGB**
  - By plotting graphs got to know some of the images are having multiple bands

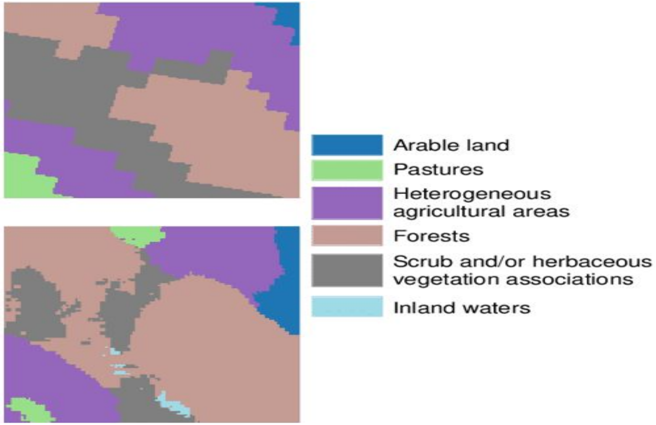
Data Transformation

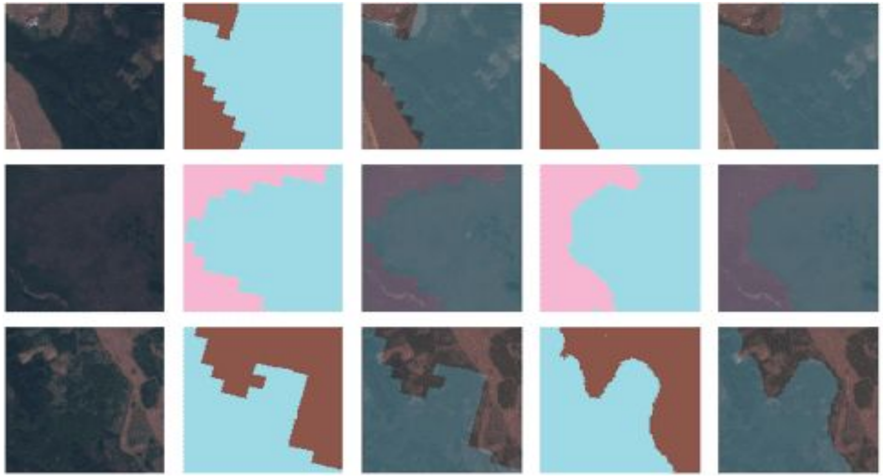
- **Data Augmentation is used**
  - Increase the training dataset
  - Train the model to handle invariance and make it more generic
    - Rotational
    - Contrast
    - Flipping
    - Mirror
    - Brightness

Data Preparation for labelling

Binary Classification

Multiclass Classification

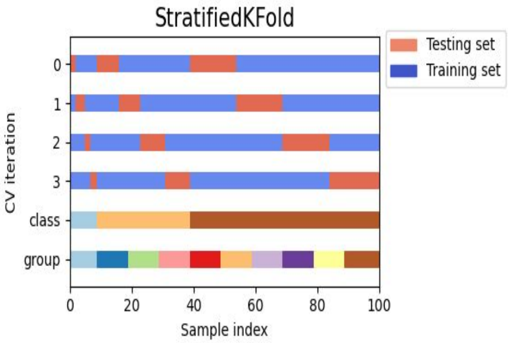




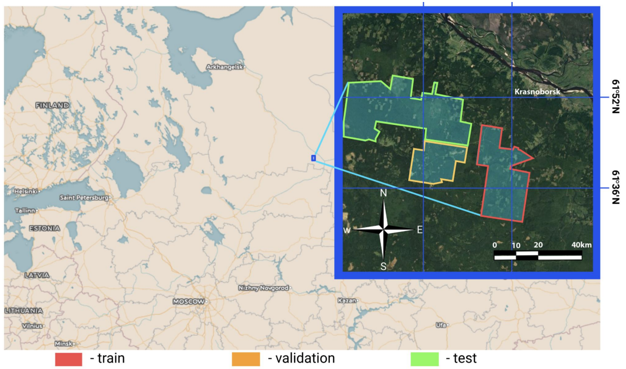
- ➢ Image annotation happens based on the classes
- ➢ Each class is assigned to an number and coloring them
- ➢ Tools : MatLab Labeller and GroundWork

Data Preparation for Train, Test, and Validate sets

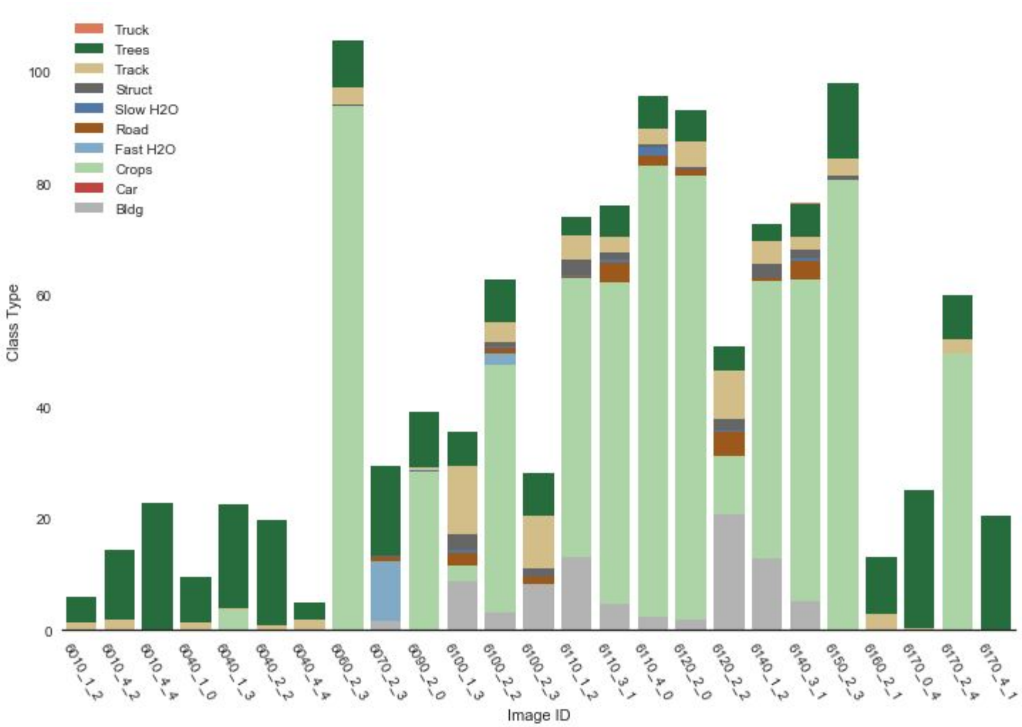
Cross validation Technique to split sets
(reduces bias and solve model overfitting)



Region Based Splitting

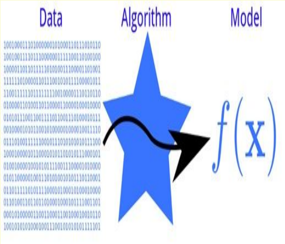


Training data Statistics of all Classes



Deep Learning Models

Convolution Models for Segmentation



Models	
Fully Convolution Network (FCN)	FCN is an extension of CNN without Dense layer Which mainly encoder Conv blocks and one decoder upsampling block
U-NET	U-NET is an extension of FCN, have encoder Conv blocks and decoder Conv blocks with skip connections
LinkNet	Similar to U-NET with encoder and decoder Conv blocks with skip connections. The encoder and decoder blocks internally also have skip connections, uses lesser parameters

Convolution Neural Network (CNN) Base Model

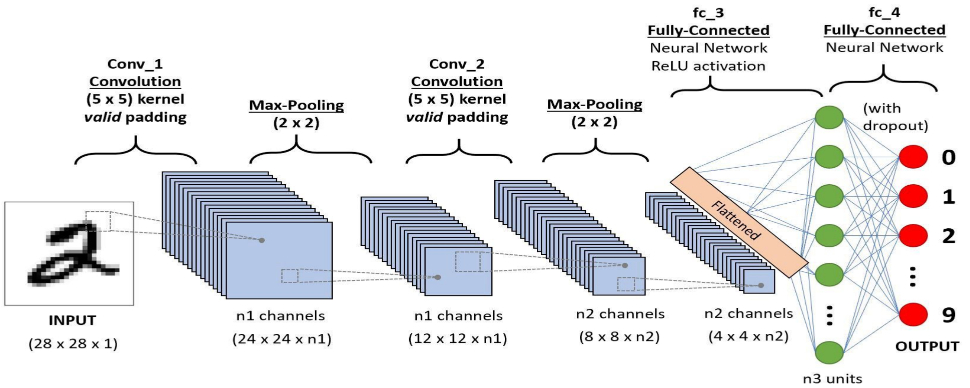
Fundamental Parts of CNN

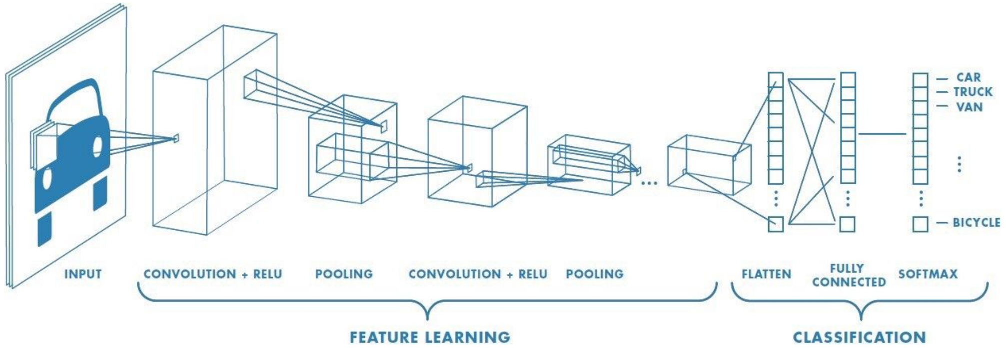
Feature Extraction

- Convolution, learn features in input size by using filters (weights)
- Non-Linearity (ReLu) Activation
- Reduce dimensionality with max pooling

Class Probabilities

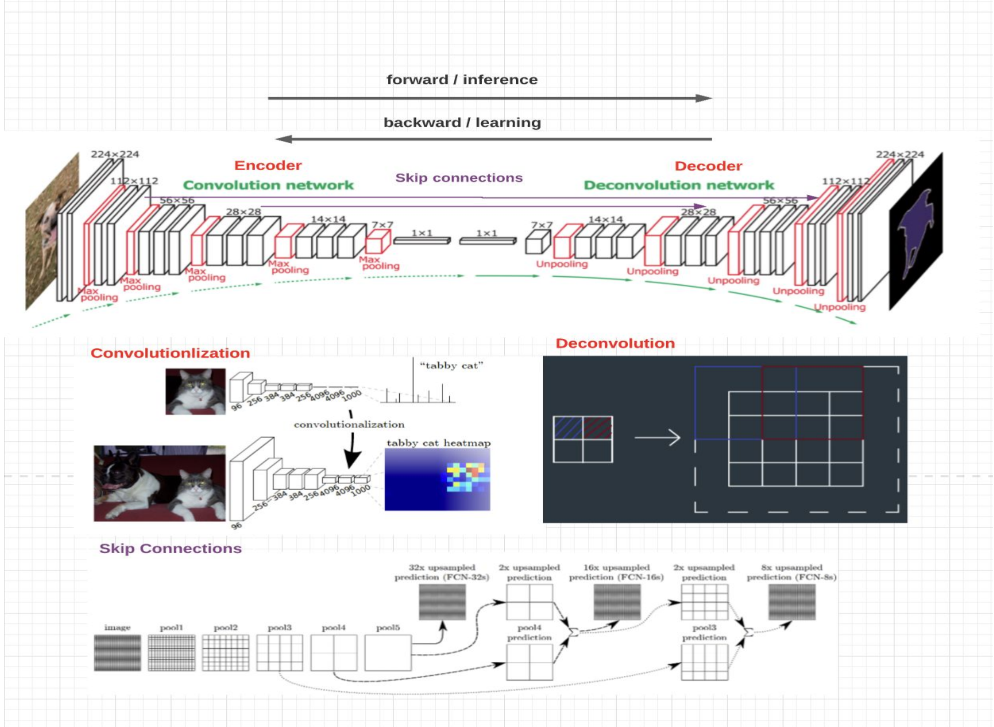
- Flatten, Conv and Pool layers(output high level features of input)
- Fully connected layers uses these features to classify image by using projection (SoftMax Classifier)





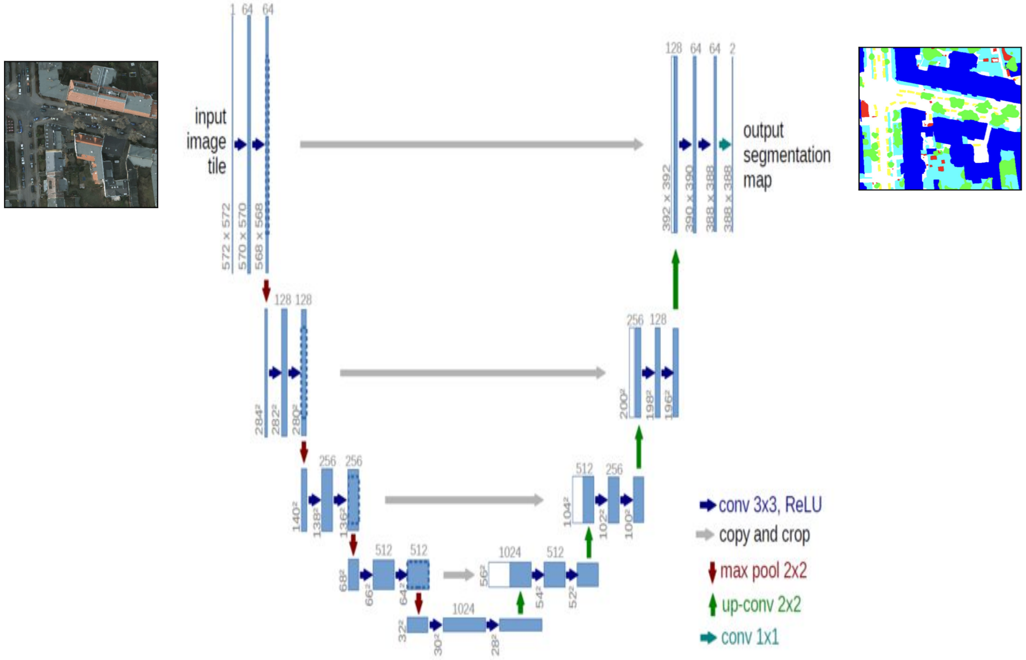
Fully Convolution Network Architecture (FCN)

- ➢ As said FCN is also a CNN without dense layer
- ➢ Main goal is to get output as image in this architecture
- ➢ To extract features, uses the same encoder part of CNN
- ➢ Instead of Flattening, we use 3D vector
- ➢ Decoder part, it reconstruct the original image by Deconvolution network (upsampling)
- ➢ Some variations of FCN have two Skip connections which solve the problem of loosing information while upsampling the image.



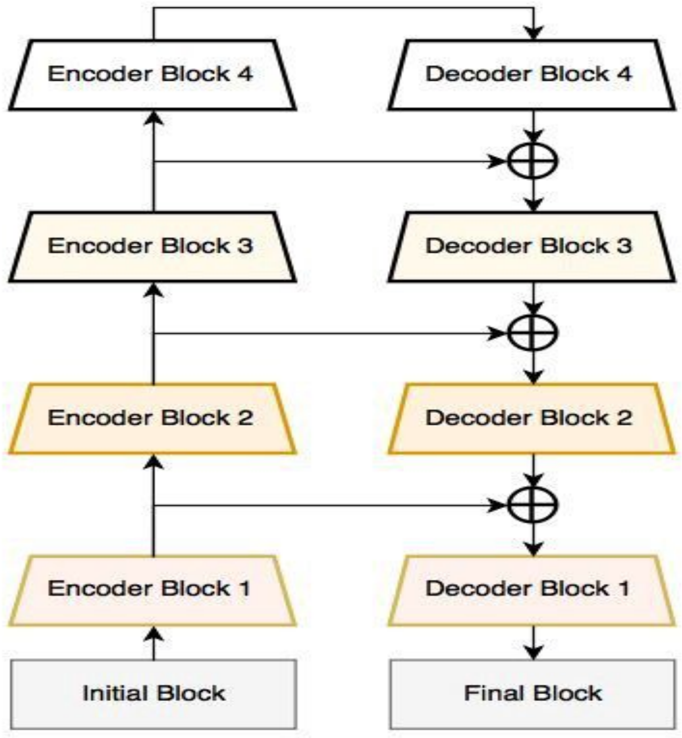
U-NET Architecture

- U-NET is a FCN, it has a symmetric structure
- Orignally introduced for Biomedical image segmentation
- Features as extracted by using multiple convolution +ReLU + maxPool
- Each decoder part is connected from the encoder network by skip connections
- U-NET has same number of upsamples or deconvolutions as encoders
- Advantage of U-NET, we can train a model by using less images



LinkNet Architecture

- ➢ LinkNET is also a U-NET, Converts U-NET normal stacking method to adding for feature extraction
- ➢ Train this model using less parameters so that the segmentation work on realtime
- ➢ Follow Encoder Decoder network architecture
- ➢ 4 Blocks of encoders and 4 Blocks of Decoders.
- ➢ Each encoder contains (Convolutions + ReLU + Pooling)
- ➢ Each Decoder same as Encoder instead of merging and max pooling it has upsampling operations
- ➢ Initial blocks 2x2 filter of batch normalization + ReLU + Convolutions layers + max pooling
- ➢ Final blocks 2 x 2 filter of batch normalization + ReLU+ upsampling + Convolutions



Model Evaluation Results

FCN:

Calculated Accuracy score, IoU, Recall, F1 score, and Precision.

Class	Accuracy		IoU		Recall		Precision		F-1	
	FCN-8	eCog	FCN-8	eCog	FCN-8	eCog	FCN-8	eCog	FCN-8	eCog
Forest	0.82	0.80	0.73	0.77	0.30	0.63	0.81	0.72	0.40	0.65
Builtup	0.83	0.73	0.71	0.58	0.52	0.30	0.51	0.19	0.45	0.21
Farmland	0.73	0.63	0.60	0.47	0.33	0.23	0.59	0.32	0.3	0.32
Water	0.93	0.73	0.86	0.59	0.76	0.69	0.78	0.40	0.76	0.48
Average	0.85	0.74	0.76	0.60	0.43	0.46	0.75	0.41	0.48	0.42

U-NET and LinkNET : Calculated metrics based on Accuracy, Quality of Segmentation (DSC)

Accuracy

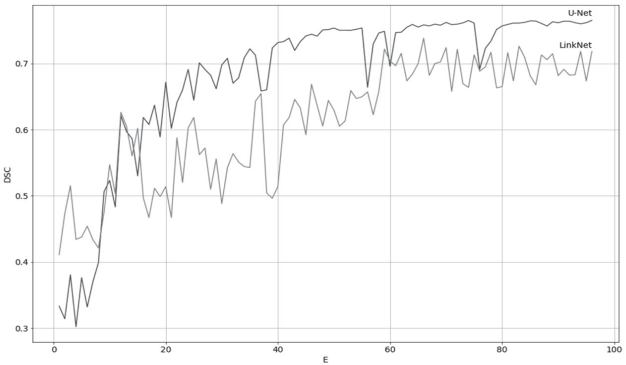
Model	Accuracy (A)
U-NET	96.31 %
LinkNet	95.85 %

DSC (Soresnsen Dice Coefficient)

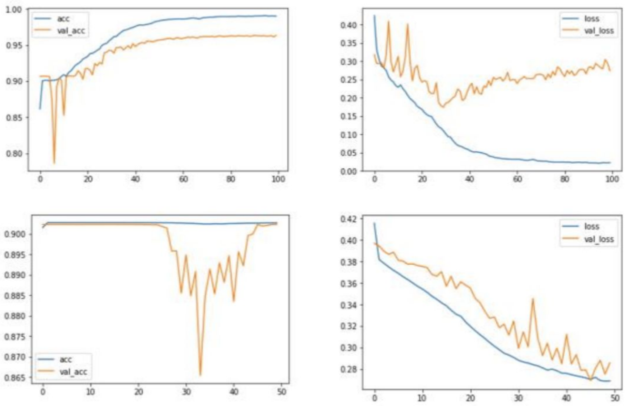
Model	DSC
U-NET	77 %
LinkNet	72 %

Model Evaluation Graphs

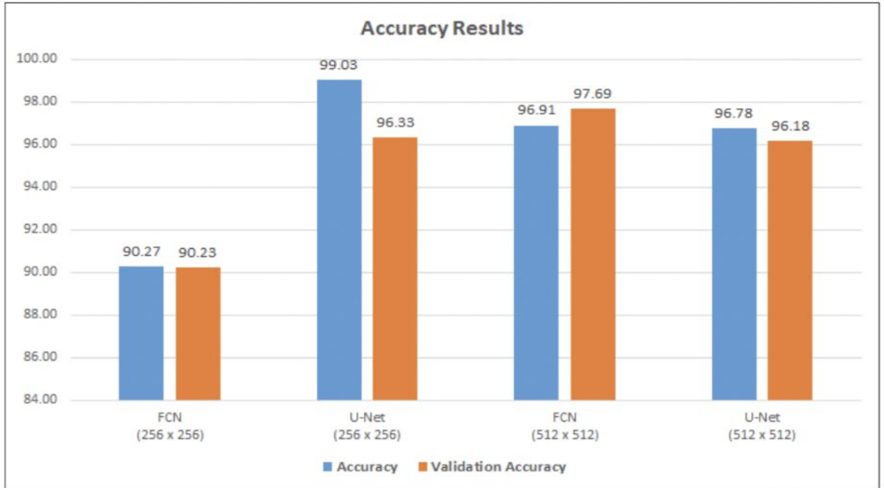
Graph 1. U-Net vs LinkNet (DSC)



Graph 2. Training and validation accuracy U-Net vs FCN



Graph 3. U-NET vs FCN training and Evaluation Result based on Accuracy



Model Comparison and Justification

Justification to use the Models

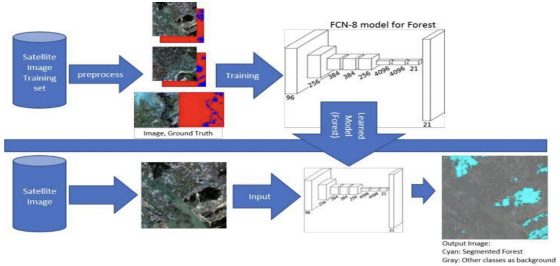
- Arbitrary image size:
  - The models are very flexible in terms of images sizes. The input or training image sizes can be or arbitrary resolution
- Number of training images:
  - The Unet model requires significantly less training samples to train for required accuracy.
  - The variants of FCN with skip connections and LinkNet also use lesser number of training samples but slightly more than Unet.
- These models a basically encoder-decoder style networks dealing with image IO

Comparison table

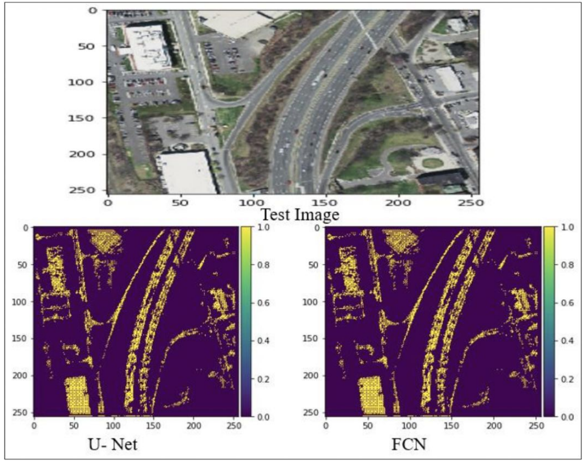
Parameters	FCN	U-NET	LinkNet
Decoder / upsampling	Only one or fewer than encoders	Same number as encoders	The same number of encoders
Skip Connections	Not present, some variants have two	Every decoder connected to the corresponding encoder	Every decoder connected to the corresponding encoder
Learnable filters in upsampling	Absent uses bilinear interpolation (static)	Present and dynamic	Present and dynamic

Satellite Image Segmtation Results

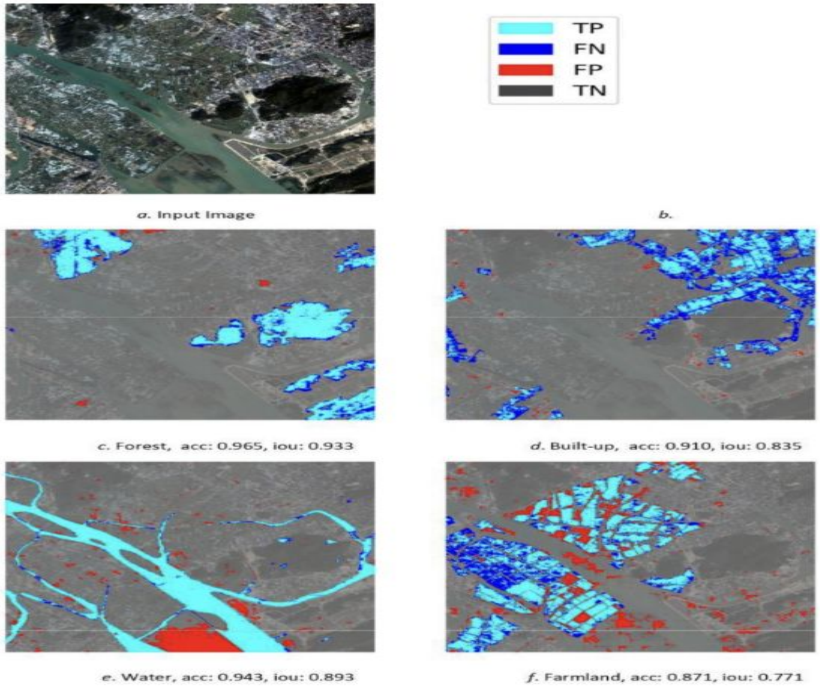
Training and Evaluation process FCN



U-NET and FCN Test and segmented images



F1 Score: Cyan color represents correctly Classified Forest Area



References

Artan, Y. (2011). Interactive image segmentation using machine learning techniques. 2011 IEEE Canadian Conference on Computer and Robot Vision, 264-129. https://doi//10.1109/CRV.2011.42

Ivanovsky, L., Khryashchev, V., Pavlov, V., & Ostrovskaya, A. (2019). Building detection on aerial images using u-net neural networks, *2019 24th Conference of Open Innovations Association (FRUCT)*, 116-122. https://doi//10.23919/FRUCT.2019.8711930.

Ronneberger, O., Fischer P., Brox T. (2015). U-Net: Convolutional networks for biomedical image segmentation. Medical Image Computing and Computer-Assisted Intervention.2015 Springer, Cham, 9351. https://doi.org/10.1007/978-3-319-24574-4_28

Thank you!!

Questions??