Satellite Project

Name	BN	Section
Ahmed Mohmed Ahmed Lotfy	8	1

Problem statement

Satellite imagery provides valuable information for monitoring various phenomena on Earth's surface, including environmental changes, urban development, and natural disasters. One critical task in satellite image analysis is change detection, which involves identifying and analyzing differences between images captured at different times. Accurate change detection is essential for various applications, including urban planning, disaster management, and environmental monitoring

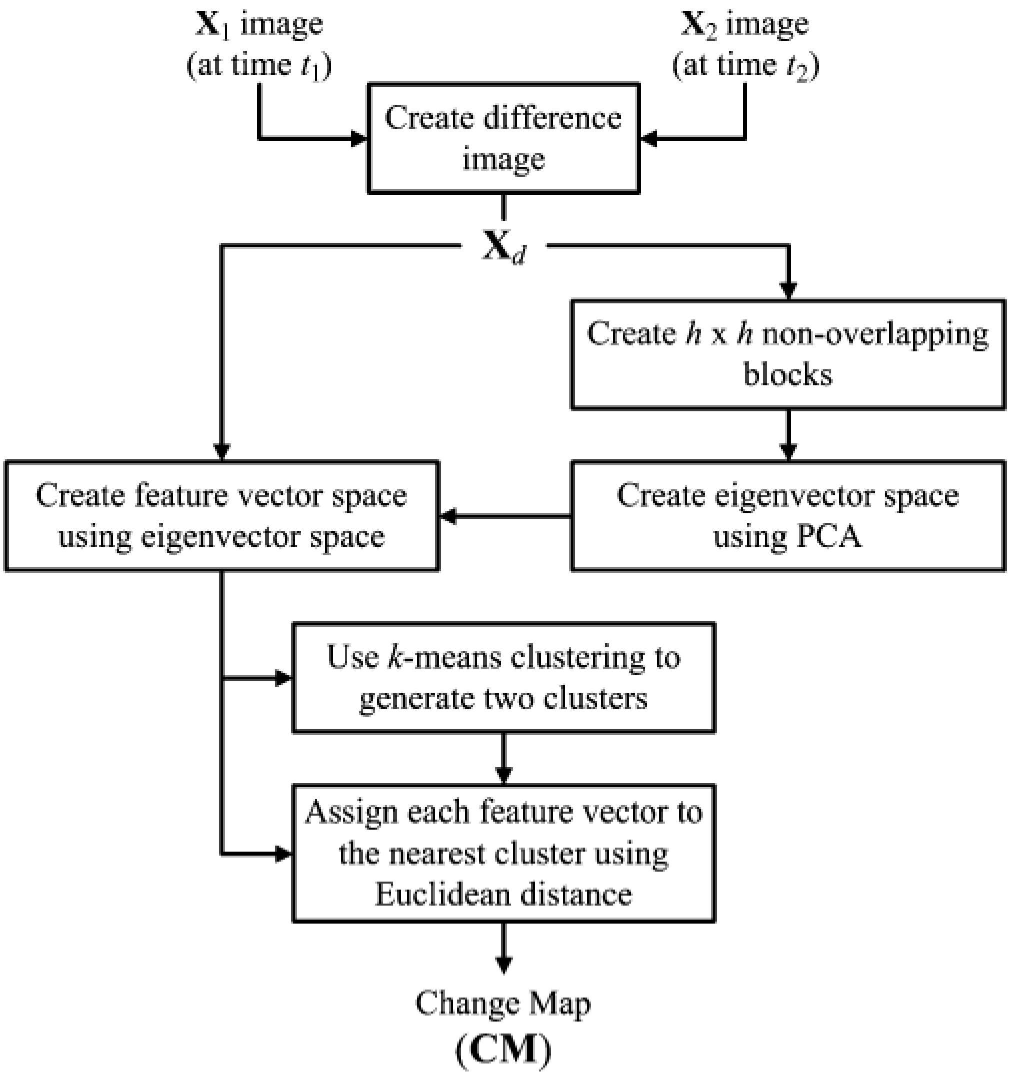
.Dataset

Link: [Building Change Detection Dataset]

Traditional Techniques

Trial (1): PCA + KMeans:

Project Pipeline



Results:

- mIoU over 1000 images →

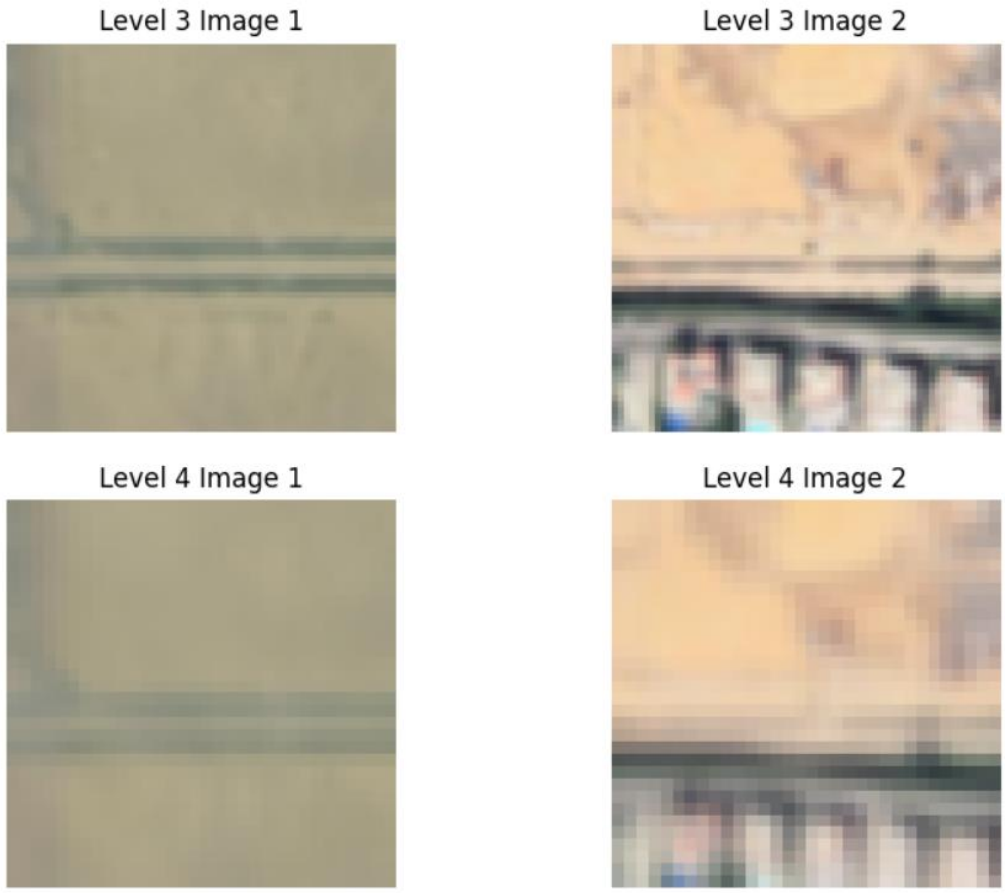
```
Average Jaccard Index: 0.812
Average Accuracy: 0.9924608459472656
```

Trial (2): CVA (not applicable with this dataset)

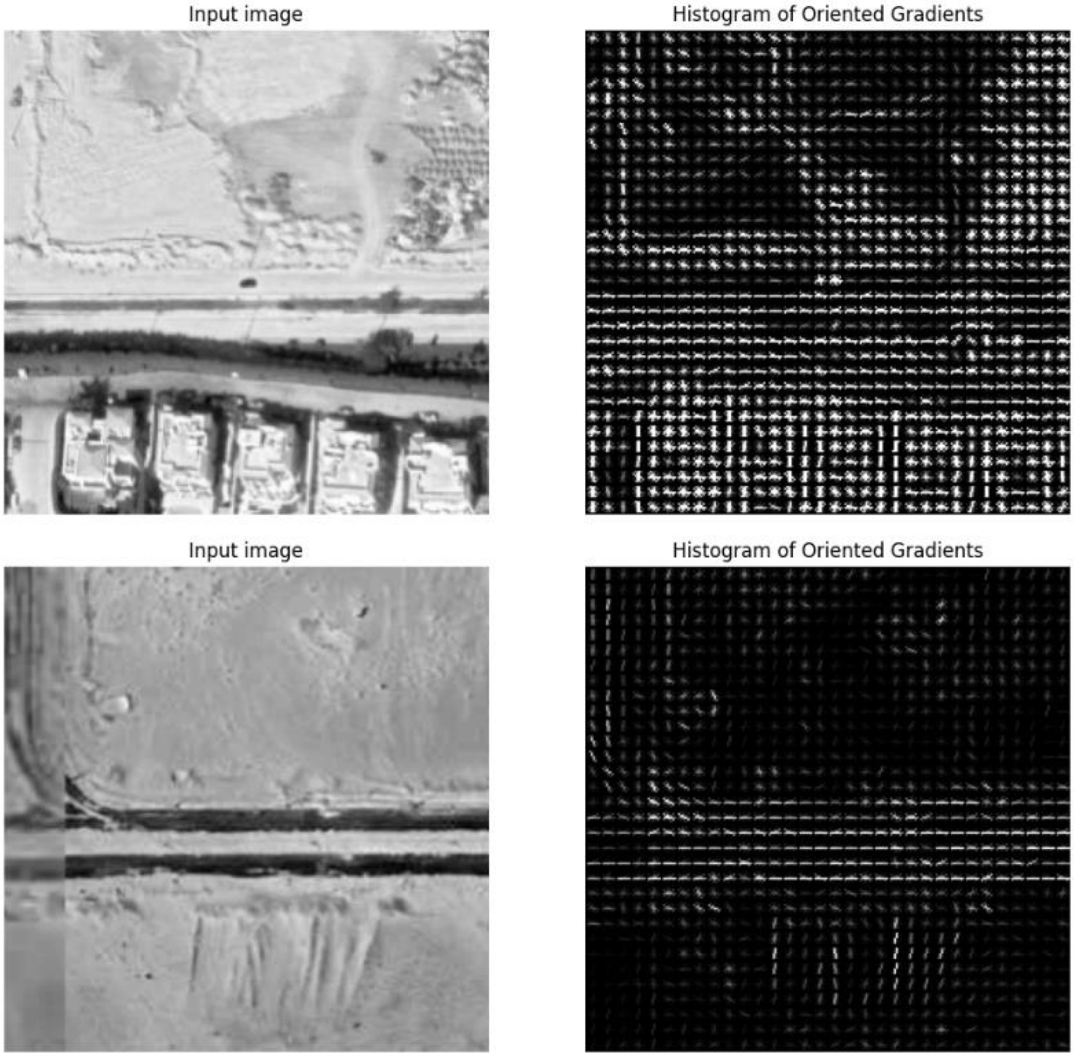
Trial (3): Extract HoG feature and thresholding.

Project Pipeline

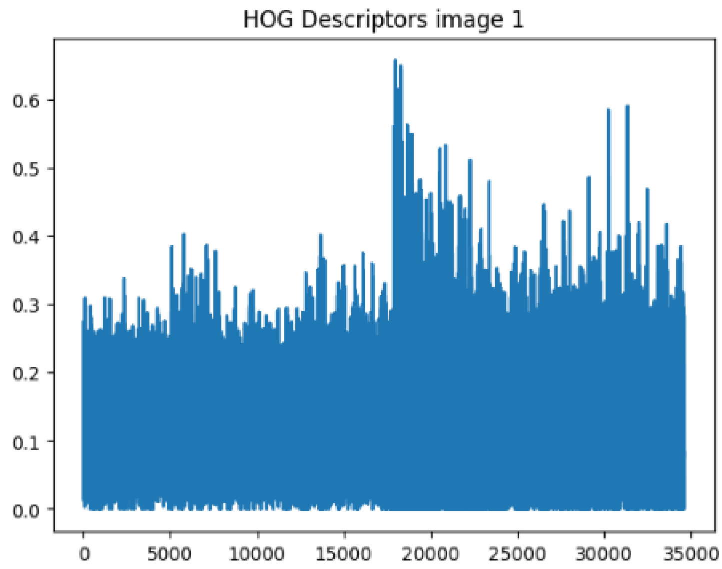
1. Applying pyramid of gaussians for noise reduction

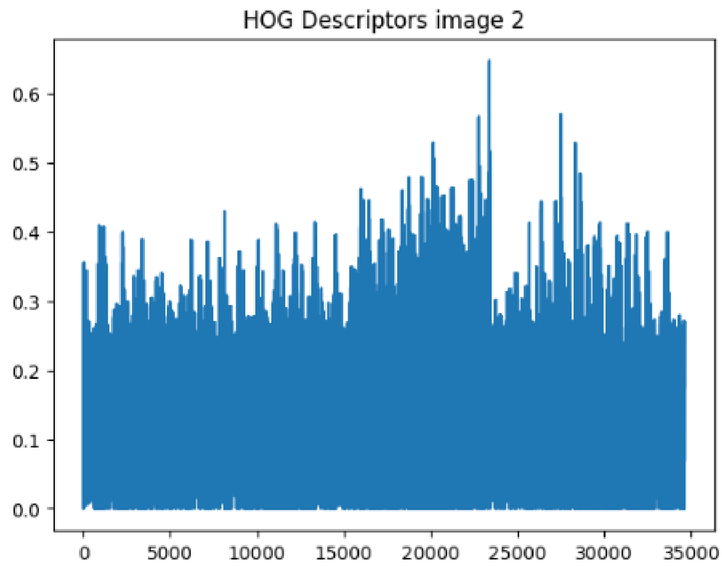


2. Get the HoG feature of each image t1,t2

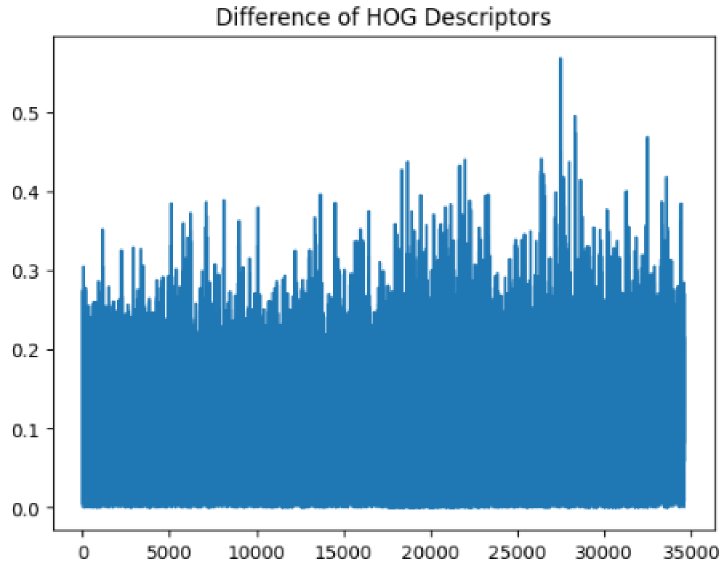


3. Get the HoG descriptors

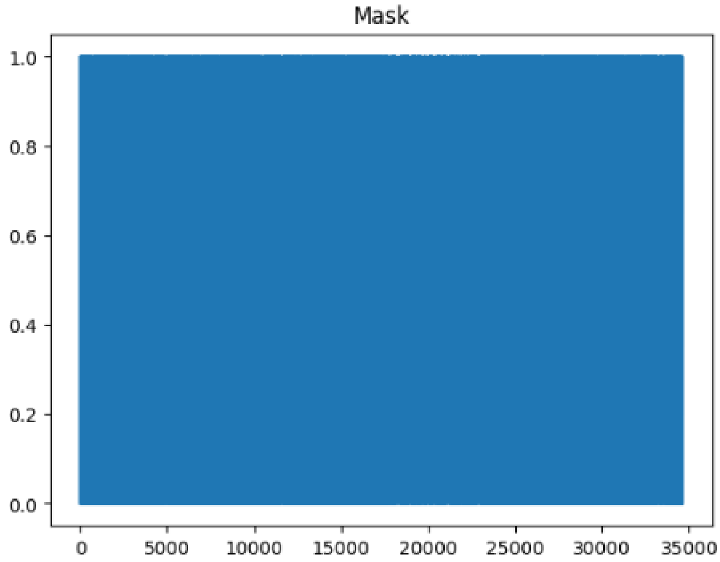


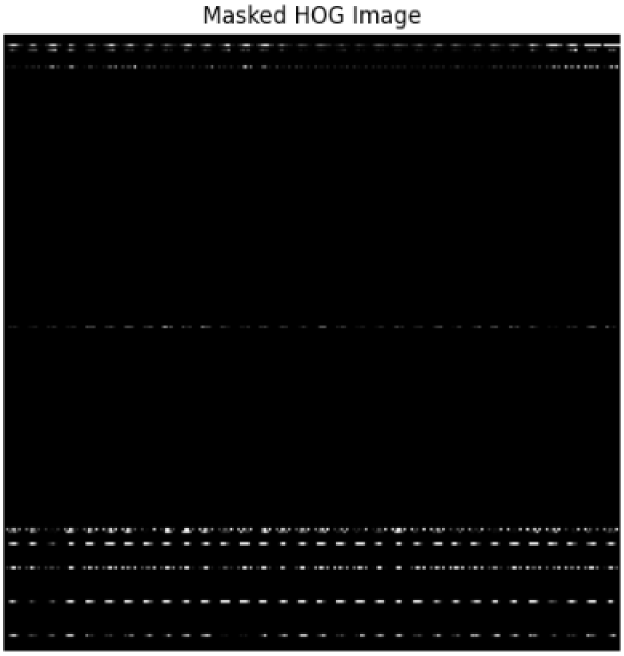


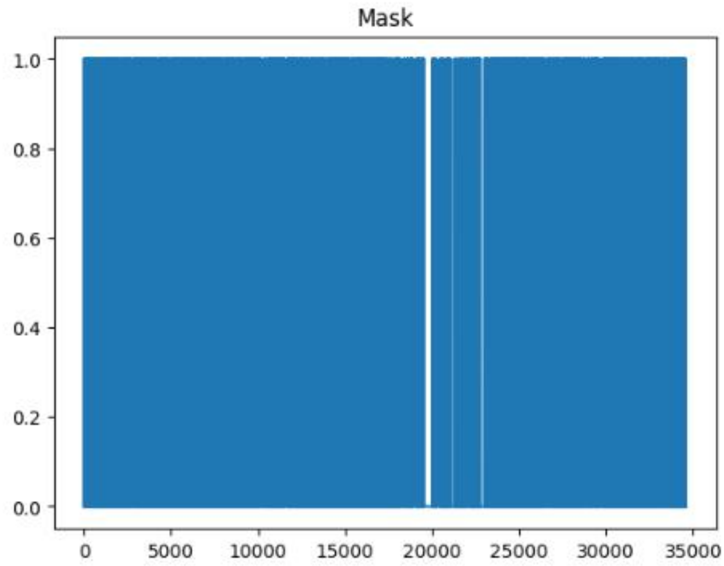
4. Get the difference of the HoG descriptors of two images

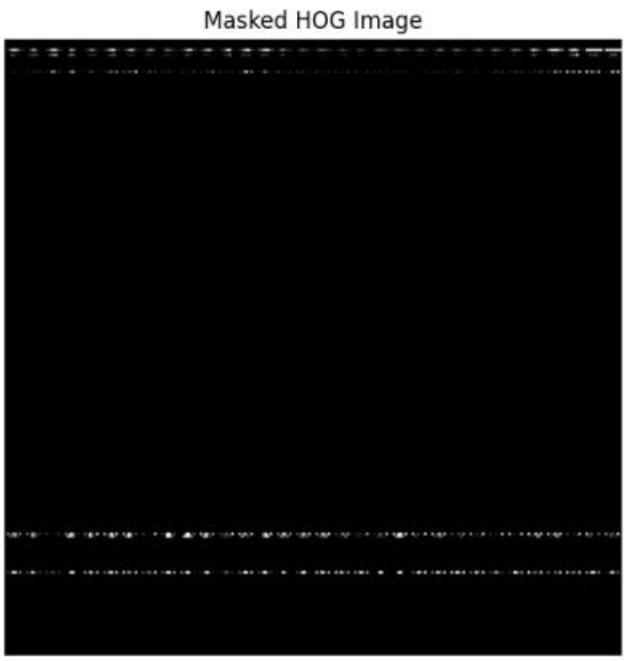


5. Apply hyperparameter tuning over the threshold to extract the regions of interest









6.

Trial (4): Structural Similarity Index

Step:

1. Convert the images into gray scale
2. Image differencing between t2 and t1.
3. Remove negative values as we focused on t2 changes.
4. Threshold the difference by the average as pixel value > avg =255 and <avg=0.
5. Calculate the SSIM score to see the similarity between the two images.

6. Tune the threshold to decide whether the two images are similar or not.
7. If ssim> threshold: make the prediction black as there is a considered similarity.
8. Compute the jaccard index for each prediction.
9. The **mIoU** =0.81

```
====================================FINISHED===============================
mIoU:  0.8127306053324157
Takes:  256.178288936615
```

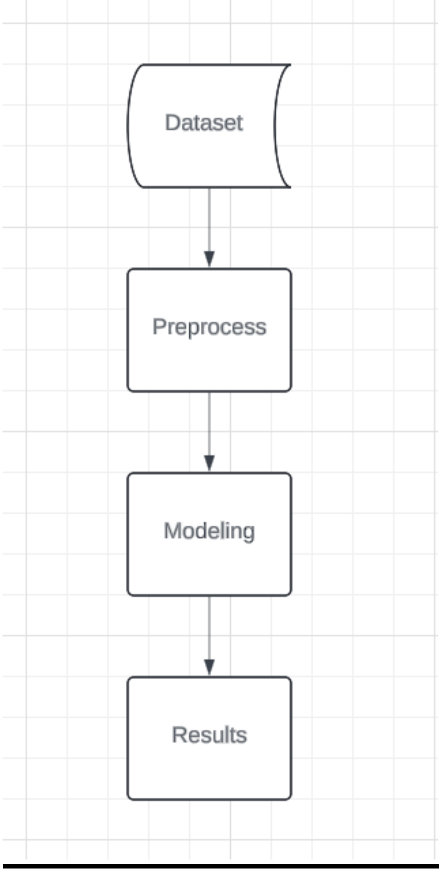
Deep Learning Technique

CGNet

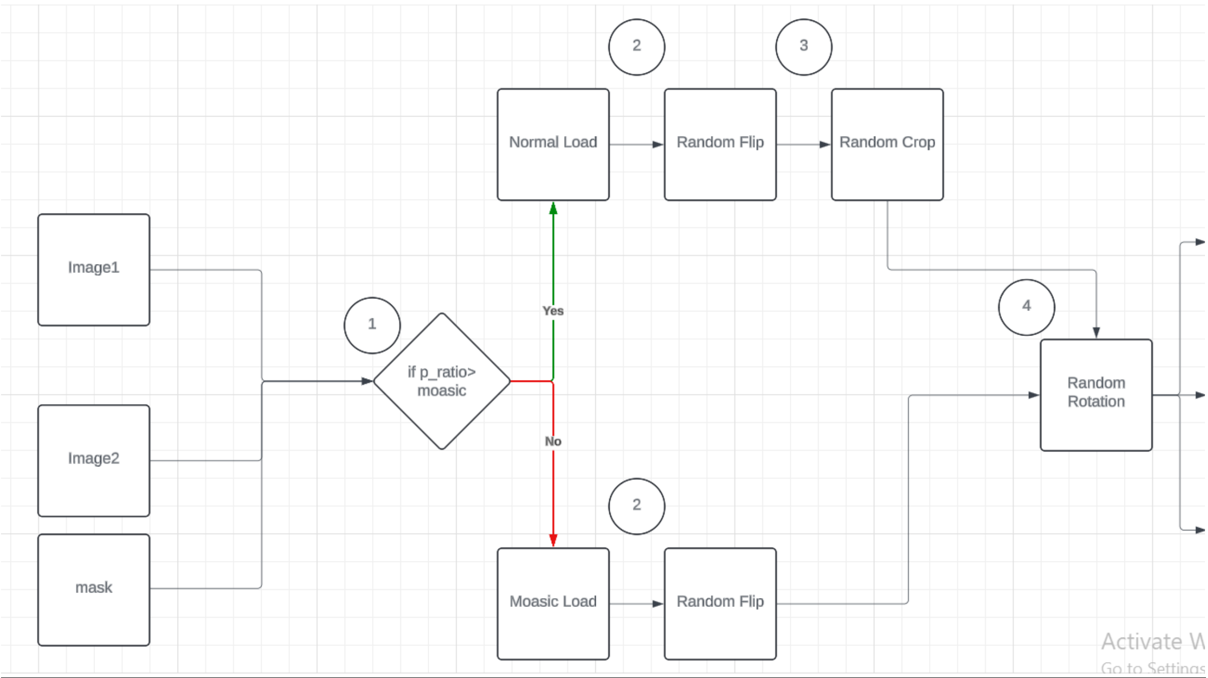
Why using CGNet (the rationale):

1. **Backbone Architecture (VGG16)**:
    - **VGG16** is chosen as the backbone architecture due to its effectiveness in capturing hierarchical features in images.
2. **Encoder-Decoder Architecture**:
    - The encoder-decoder architecture is well-suited for image segmentation tasks like change detection.
    - The encoder part of VGG16 extracts high-level features from input images, while the decoder part upsamples these features to generate segmentation maps.
3. **Change Guide Module (CGM)**:
    - CGM is introduced to adaptively adjust the feature representations based on the guidance provided by a change map.
    - It consists of query, key, and value convolutions to compute attention-based feature representations.
    - The use of CGM allows the model to focus more on regions where significant changes occur, enhancing the change detection performance.
4. **Loss Function**:
    - Suitable loss functions such as **cross-entropy loss** or **dice loss** are commonly used for training segmentation models like **CGNet**.

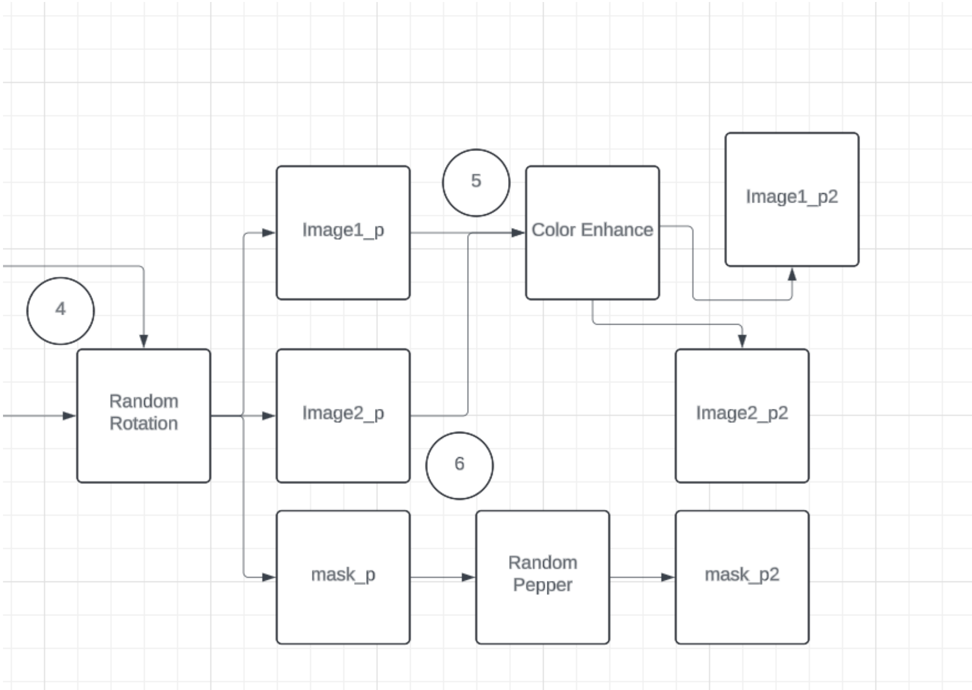
Project Pipeline



Preprocess Stage



1. Preprocess Stage consist of many parts let's explain them:
    a. Part **(1)**: based on random number **p_ratio** the images will be mosaic if **mosaic** number is greater than **p_ratio:**
        i. Each image and mask will have a random 3 images then crop each image to have a size of quadrant then after cropping they will be concatenated.
        ii. If **mosaic** number is less than **p_ratio**: the images and mask will be loaded normally.
    b. Part **(2)**: as a part of **Data Augmentation** firstly we do random flip based on a random number.
    c. Part **(3)**: as a part of **Data Augmentation** secondly we do random crop if the decision is **YES.**
    d. Part **(4)**: as a part of **Data Augmentation** thirdly we do random rotation.



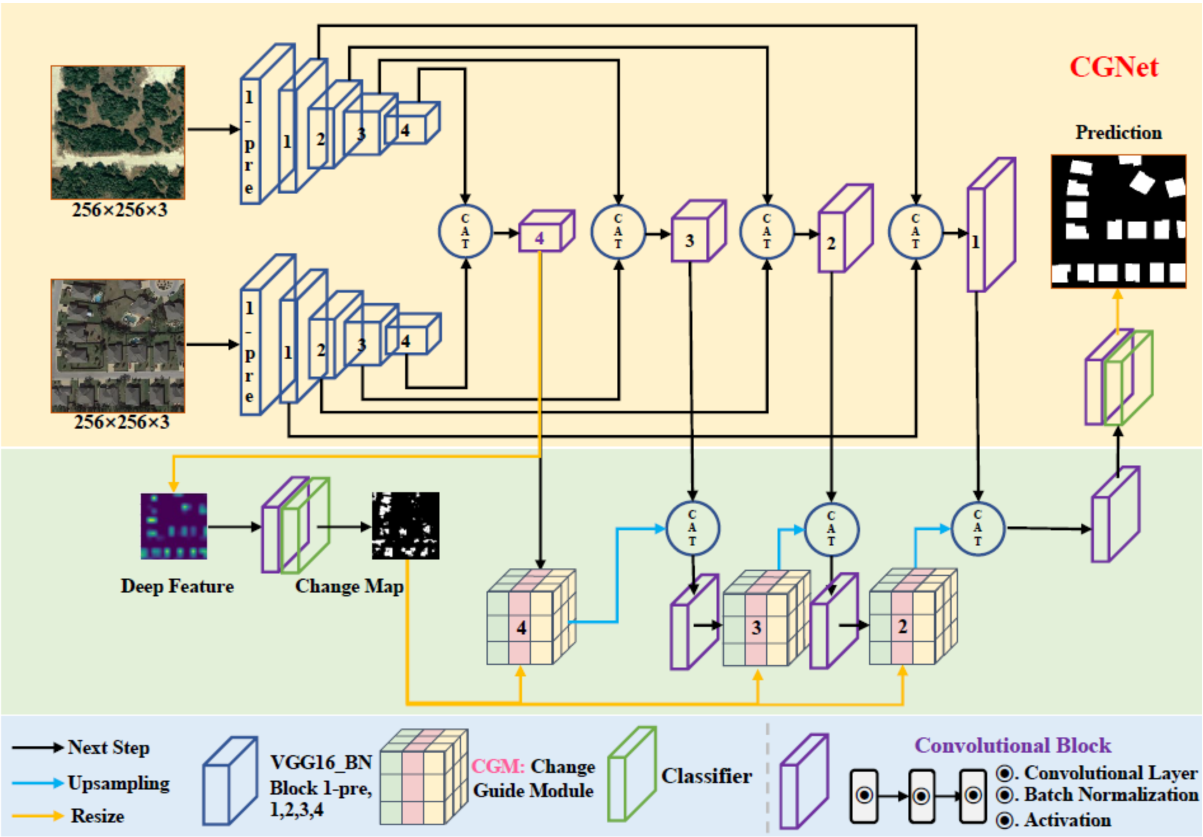
    e. Part **(5)**: as a part of **Data Augmentation** the processed images are then color enhanced by randomly adjust the **brightness, contrast, color and sharpness**.
    f. Part **(6)**: as a part of **Data Augmentation** the processed mask is then added to some random pepper.
    g. The purposes of these steps of **Data Augmentation** are to be **robust** to variations in dataset, reducing **overfitting**.

Dataset Spliting

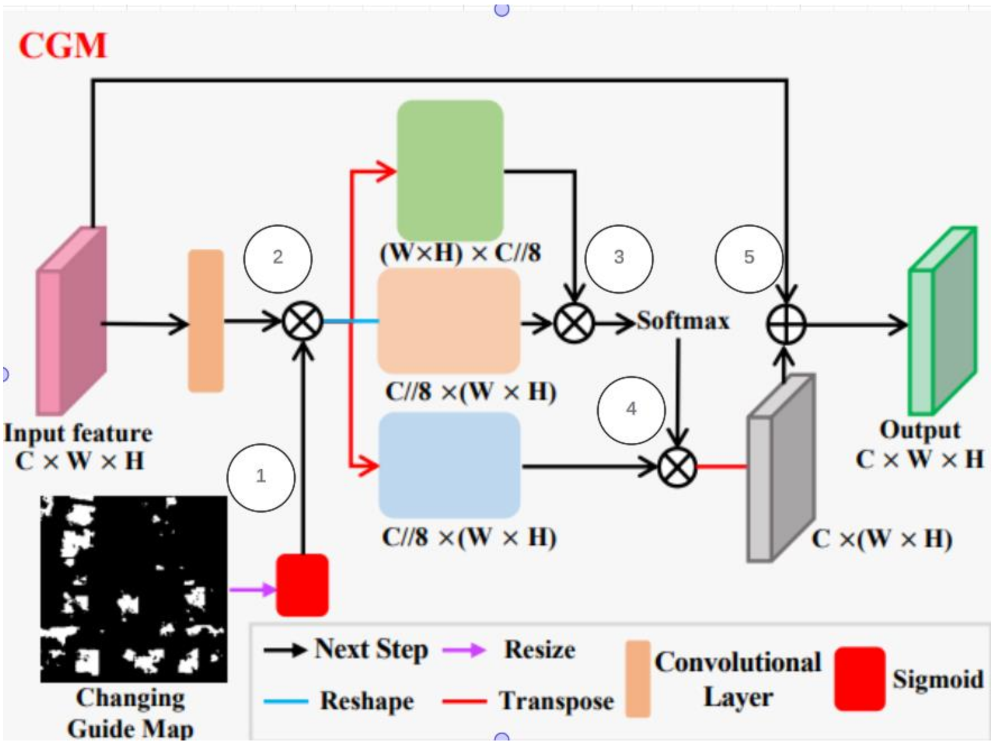
-About **70**% of the dataset is for **training** and **20**% for **validation**.

Modeling Stage

- **CGNet** (Change Guiding Network) is designed to tackle the insufficient expression problem of change features in the conventional **U-Net** structure adopted in previous methods, which causes **inaccurate** edge detection and internal holes.



-
- Let's go deep on the architecture, as we see in the diagram above blocks are defined:
  - We first will introduce the rubrik cube-shaped module which is the core **CGM** change guide module.

- 

- Let’s discuss what is happening here, First we have the input tensor ‘x’ and a changing guiding map (will be discussed in the overall architecture of CGNet).
- We want to get Q,K,V to get the attention map
- $$(Q, K, V) = (QW_Q, KW_K, VW_V)$$
- Part **(1)**: the changing guiding map is resized to match the shape of ‘x’ using billinear method then squashed by **sigmod** activation function to have values from 0 and 1 and get the weight graph **W**.
- Part **(2)**: the input tensor ‘x’ is convolved to extract features then element-wise multiplication with the change guiding map to get **Q,K,V.**
- Part **(3)**: compute the energy matrix using batch matrix multiplication between projection(**Q**) and projection(**K**) and get attention by squashing the energy matrix by **softmax** where the attention scores is normalized so their sum is 1.
- Part **(4)**: aggregating the values according to the attention scores by matrix multiplicate the projection(**V**) and the **attention**.

- Part **(5)**: the output of part **(4)** is scaled with a learnable parameter **gamma** then combined with input tensor ‘x’ to give the final output.

- **CGNet** integrates components a pre-trained model **VGG-16** (which is the encoder) feature fusion, reduction, and decoding for the two images and for each layer the output of it which are the images are concatenated and convolved and be the input of the change guide module with the changing map.
- The **Guiding map** we mentioned before is the output of the **VGG-16** model of two images after concatenation and convolution.
- The **change guide module** is the decoder part.

Performance and Results

- I trained **CGNet** over the dataset and I got **mIoU=0.6255**:

```
Best Model Iou :0.6325, mIoU:0.6255; F1 :0.7749; Best epoch : 49
Best Model Iou :0.6255; F1 :0.7749
FINISIHED
Training Time : 11006.351668834686
```

- Where it takes almost 3.1 hrs
- Dataset test mIoU is 0.54 takes 13 seconds

Model	Time	mIoU Training	mIoU Testing
CGNet	3.1 hrs	0.6255	0.54 (13 sec)
Classical: SSIM	256 sec	0.81	
Classical: PCA+KMeans		0.812	

Thank You