PIXEL SEMANTIC SEGMENTATION

DOMAIN: MACHINE LEARNING

BY

D.V.S.SIVA DATTA

PROJECT GUIDE:

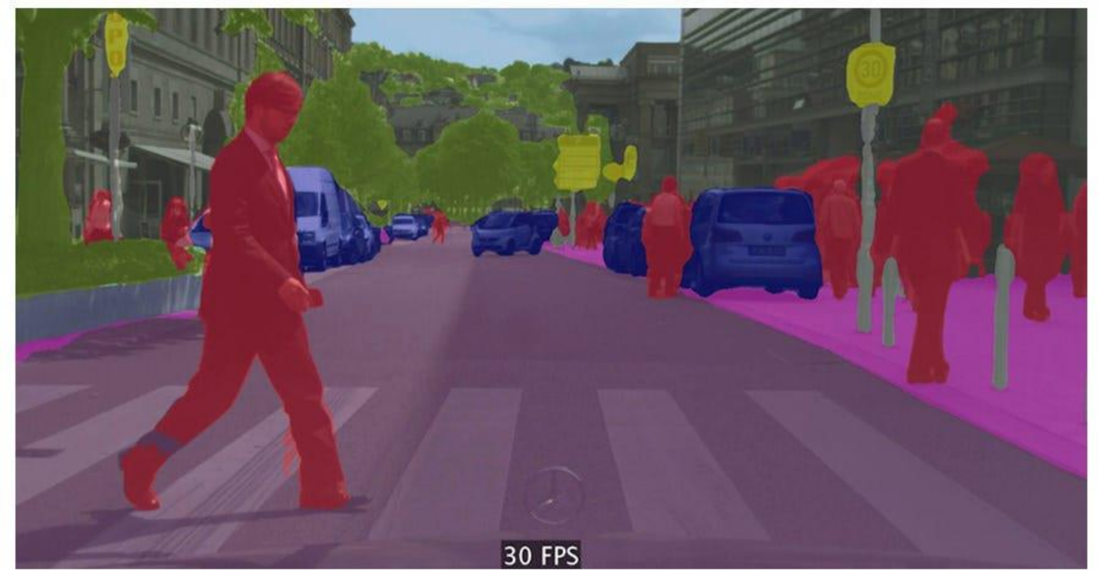
DR.NAMITA PARATI

Associate Professor,

CSE Department.

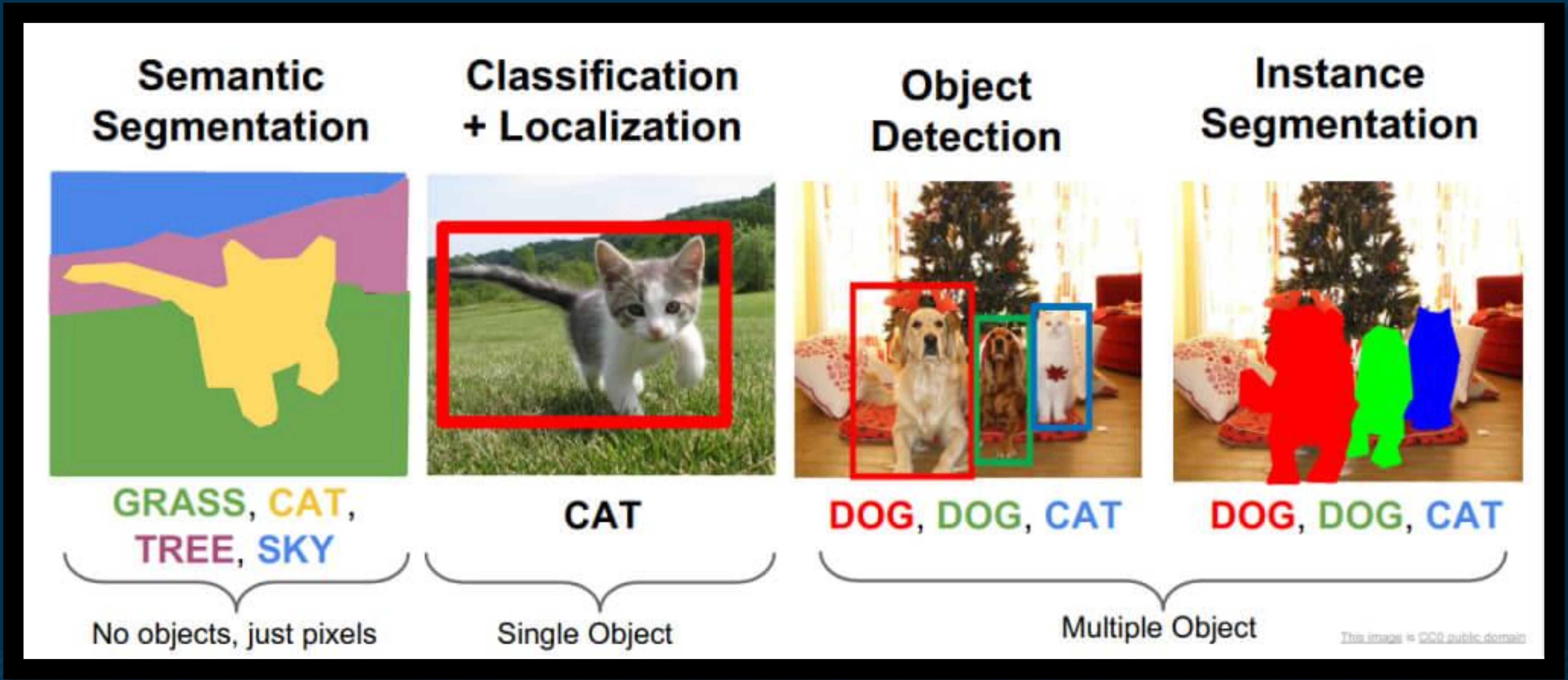
What is Semantic Segmentation?

- *For the machine to learn from its environment it needs to capture the image and classify it according to the objects present in it.*
- *This can be achieved through Pixel Semantic Segmentation.*
- *Image segmentation is a computer vision task which involves dividing various regions of image into objects that are present in it.*
- *It labels each pixel of an image with the corresponding class of what is being represented.*
- *More specifically, it does the task of clustering parts of image together which belong to the same object.*
- *In the given picture Each pixel of the image has been assigned to a class represented by a different color red for people,blue for cars,green for trees etc.*



What is Semantic Segmentation?

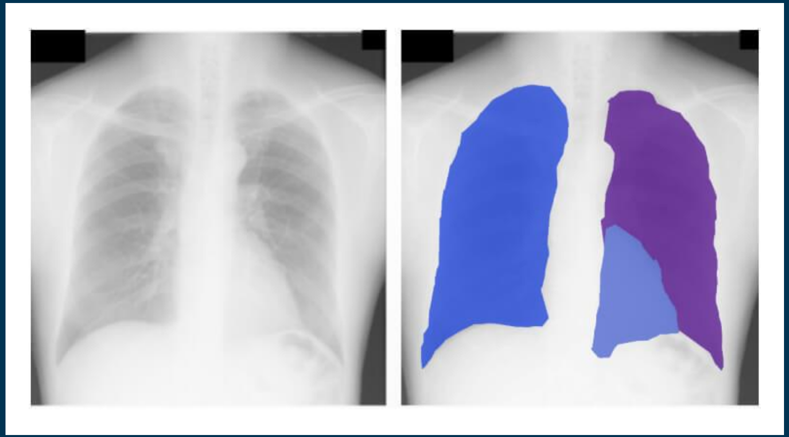
The following image presents the difference between various Computer Vision tasks:

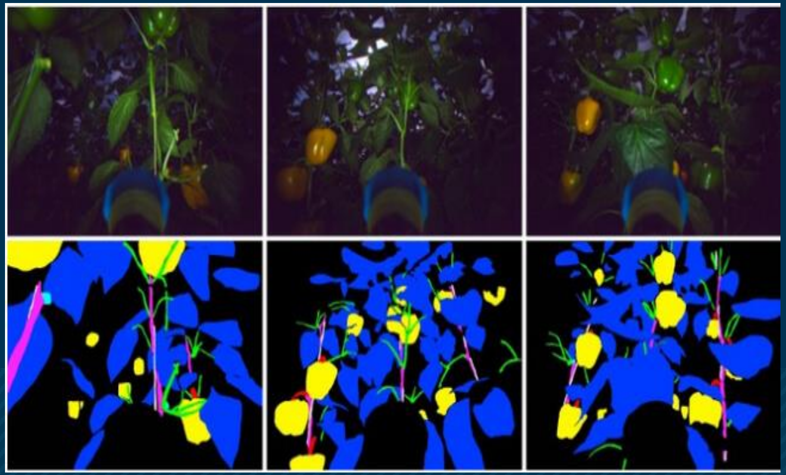


This image is CC0 public domain

APPLICATIONS

- Autonomous Self-Driving Vehicles
- Agriculture-to predict how healthy or infected the crop is.
- Virtual Try On
- Satellite/Aerial Imagery
- Medical Imaging etc
- Robotic Vision





EXISTING SYSTEM VS PROPOSED SYSTEM

- ***Existing System***

There is no real time existing system for this project . The research is going on to use this technology in self driving/autonomous cars, medical field and in satellites but they are not into application yet as there are no accurate results

- ***Proposed System***

In this proposed system we are introducing the application of U-Net Architecture which will segment the image based on its classes.Various datasets are used as input and pixel level segmentation is done.It is trained by various input datasets and ground truth datasets .Therefore every time we insert an image the machine gets trained and stores the information which is used for making the segmentations.

Software and Hardware Requirements

Software Requirements

- Google Colab/Jupyter Notebook
- Python

Hardware Requirements

- Laptop/ Personal Computer (PC)
- Random Access Memory (RAM): 8 GB or above
- Central Processing Unit (CPU): 1.7 GHz Processor and above
- Operating System (OS): Windows 10 and above

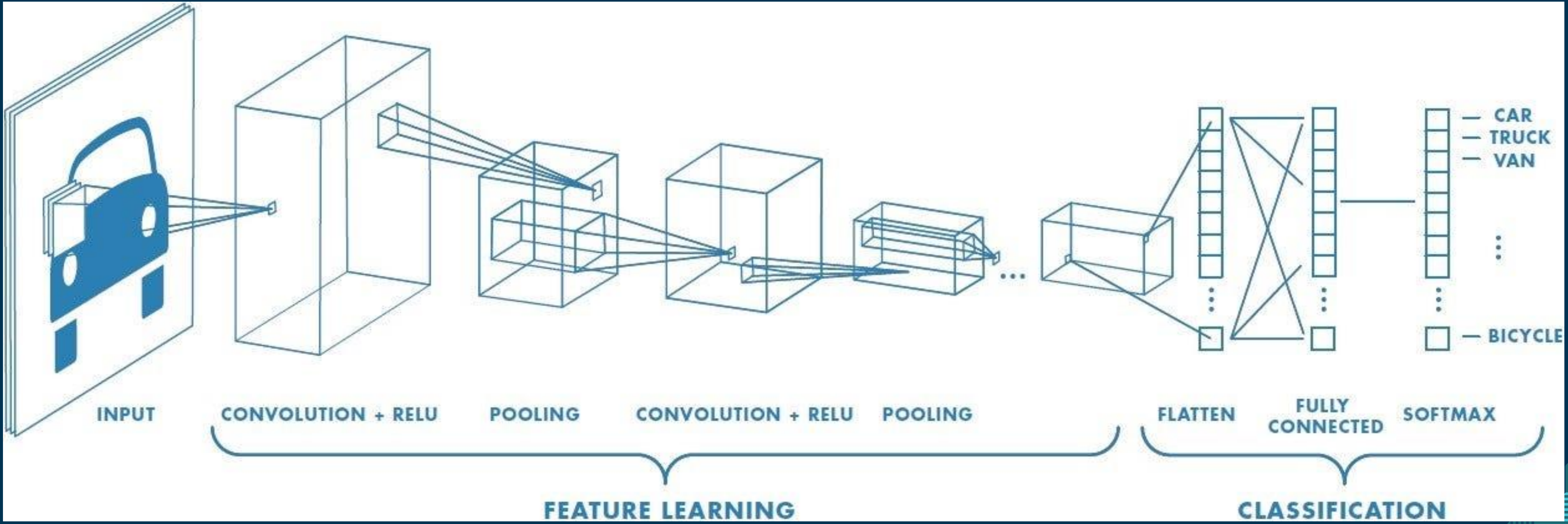
PROPOSED ARCHITECTURES AND ALGORITHMS

Convolutional Neural Networks (CNN) :

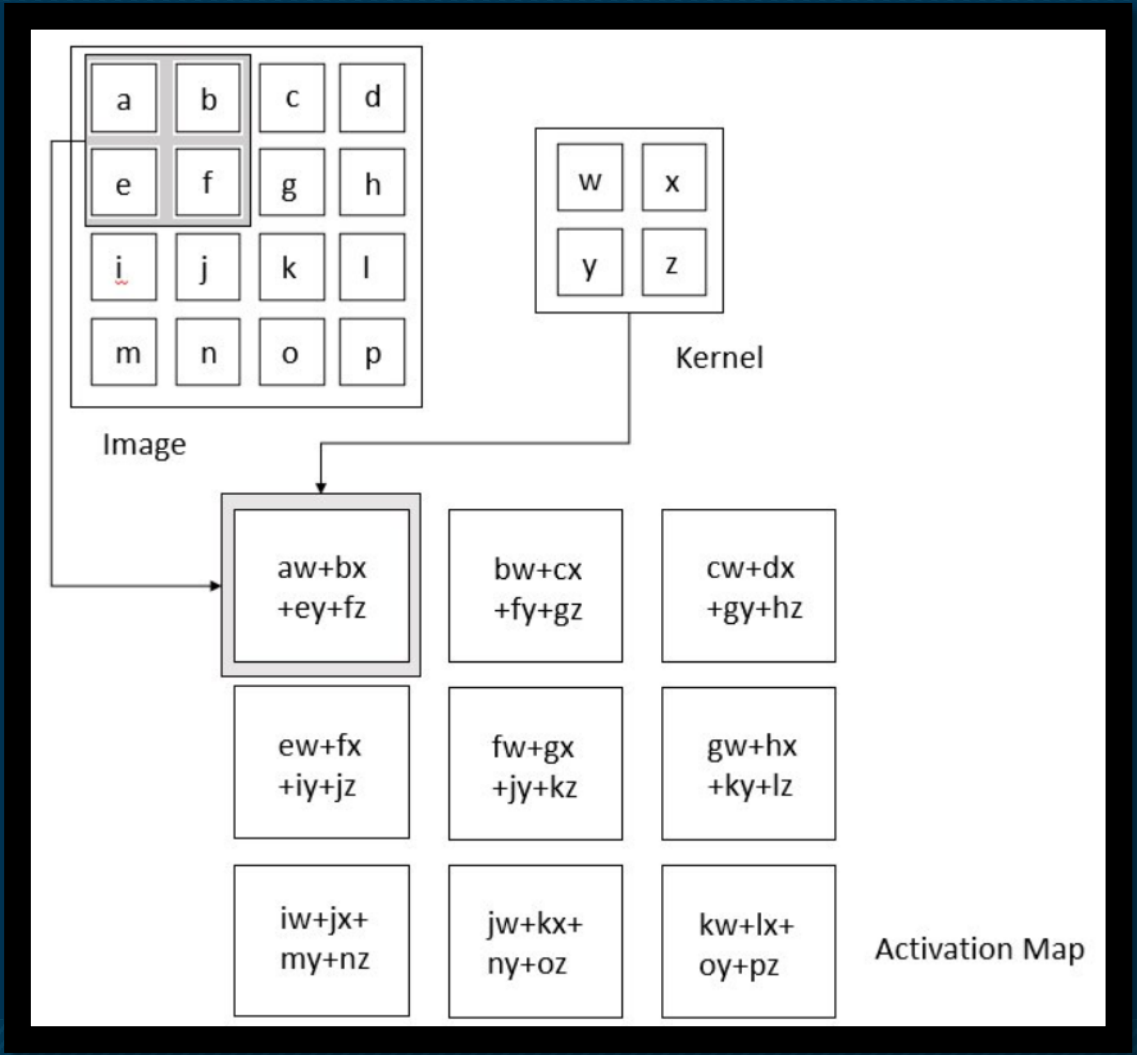
- *It works great for computer vision tasks.*
- *CNN is now the go-to model on every image related problem.*
- *The main objective of the convolution operation is to extract the high-level features such as detect objects and larger shapes in the image.*
- *CNN has an input layer, convolutional layer, pooling layer, and fully connected layer by definition.*
- *The main advantage of CNN compared to its predecessors is that it automatically detects the important features without any human supervision.*

PROPOSED ARCHITECTURES AND ALGORITHMS

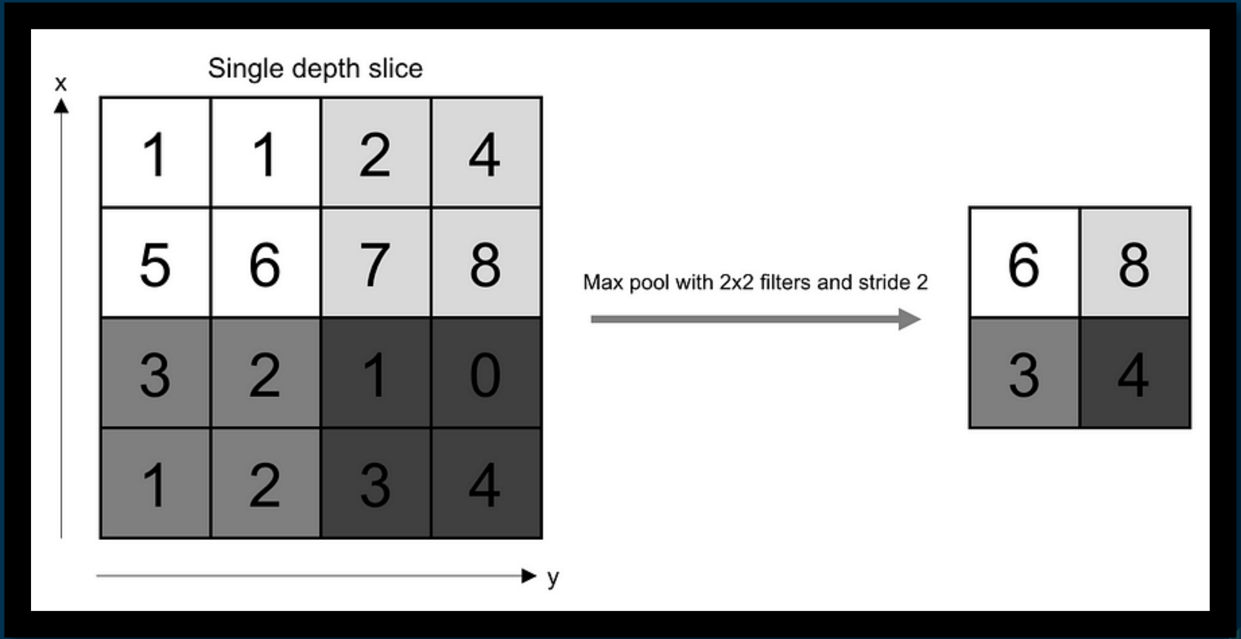
Convolutional Neural Networks (CNN) :



PROPOSED ARCHITECTURES AND ALGORITHMS



Convolution Operation



Pooling Operation

PROPOSED ARCHITECTURES AND ALGORITHMS

U-Net Architecture:

- *It has now become a very popular end to end encoder-decoder network for semantic segmentation.*
- *It has unique Up-Down architecture which has a contracting path and an expansive path.*
- *It's called U-NET because it's looks like U Shape and it was originally invented for biomedical image segmentation.*
- *The architecture contains two paths the path on the left side is called Encoder network or contraction path and on other side decoder network or expansion path.*
- *In between data from left side to right side concatenation happens and that are put together and this context gives the benefit of feature mapping & this is the reason why we get localized information and this is what it makes semantic segmentation possible using U-NET.*

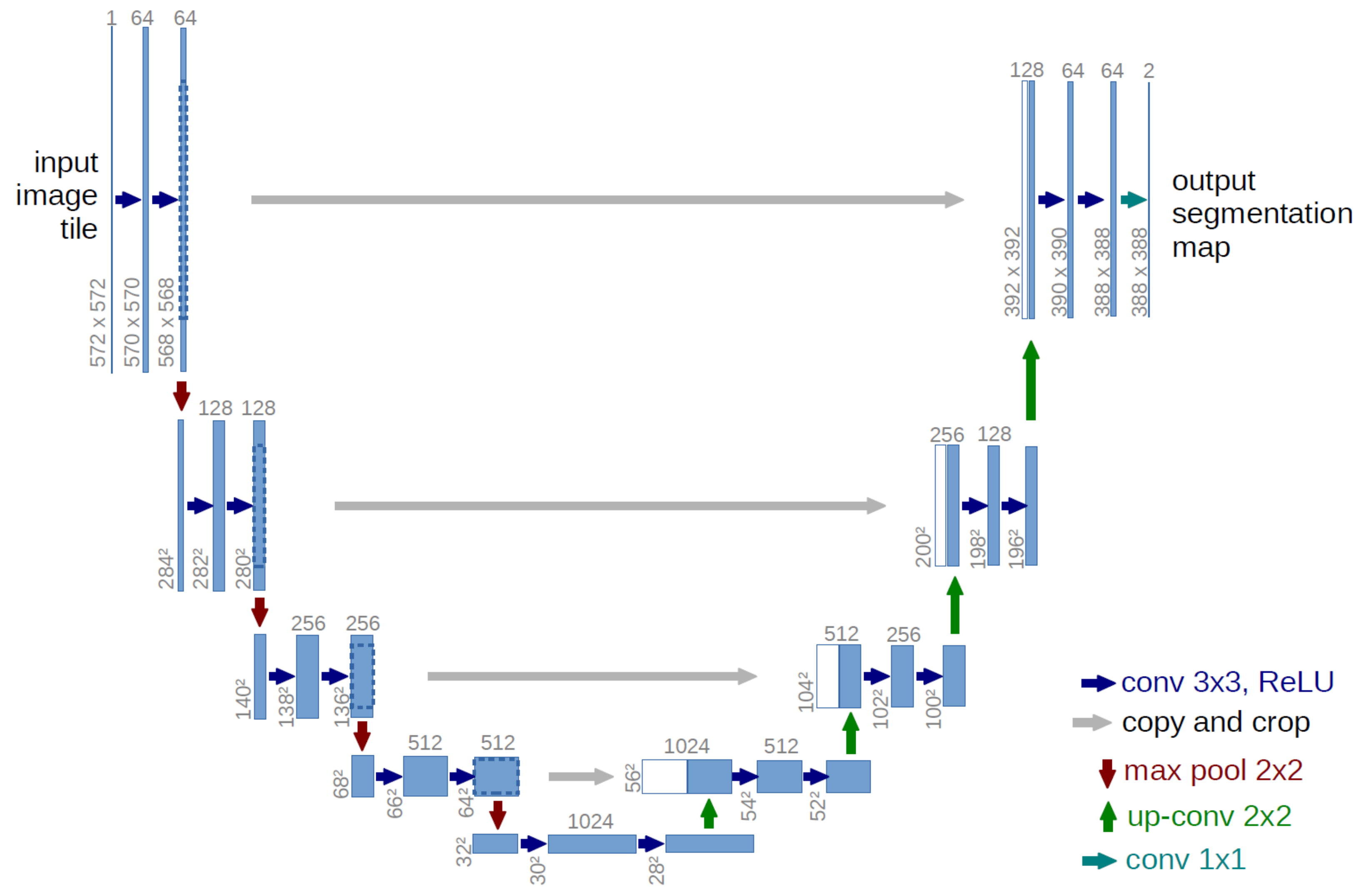
PROPOSED ARCHITECTURES AND ALGORITHMS

U-Net Architecture :

Downsampling Path: Capture semantic/contextual information(low level features)

Upsampling Path: Recover spatial information(high level features)

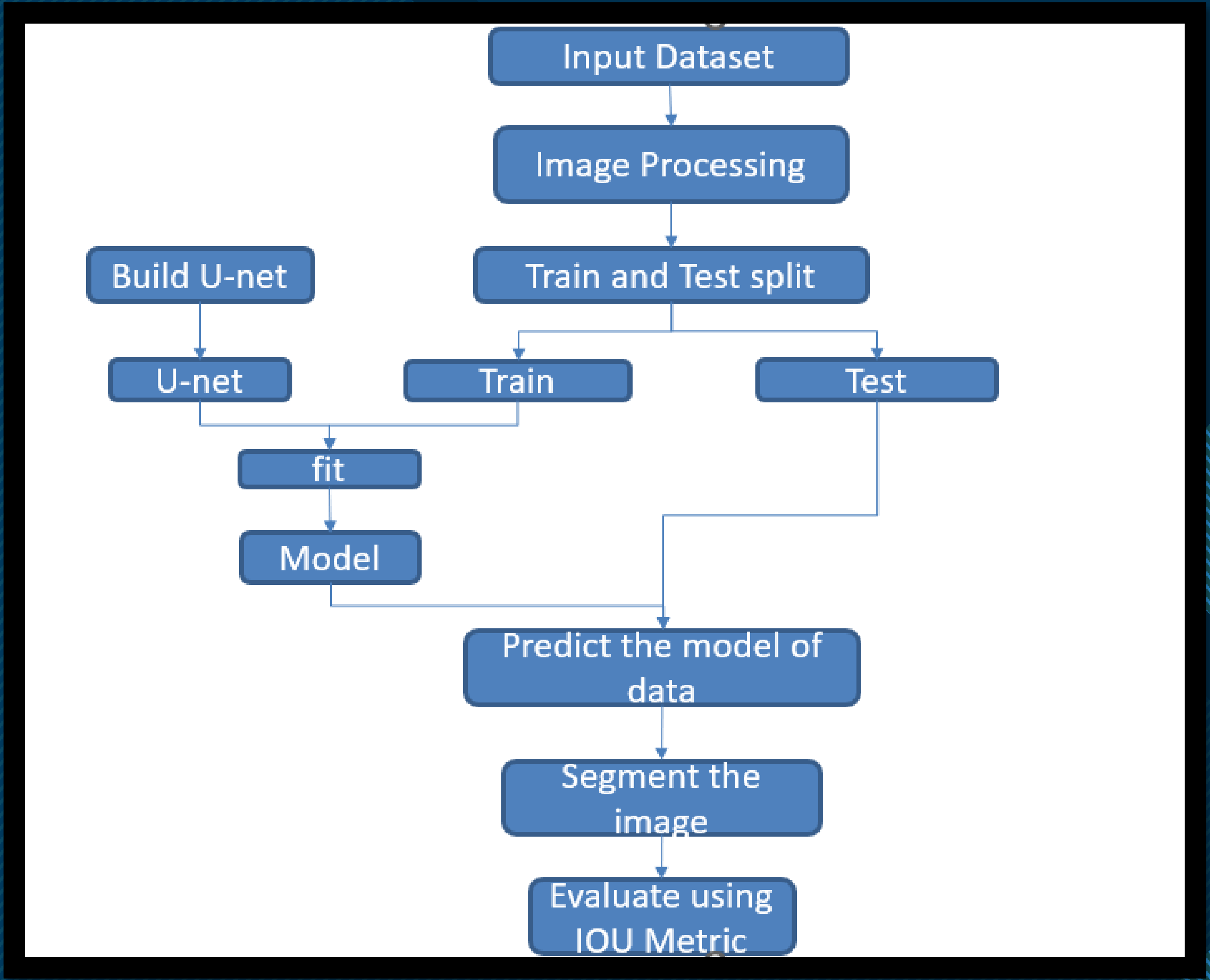
Bottleneck: Feature Mapping



Major Libraries Used

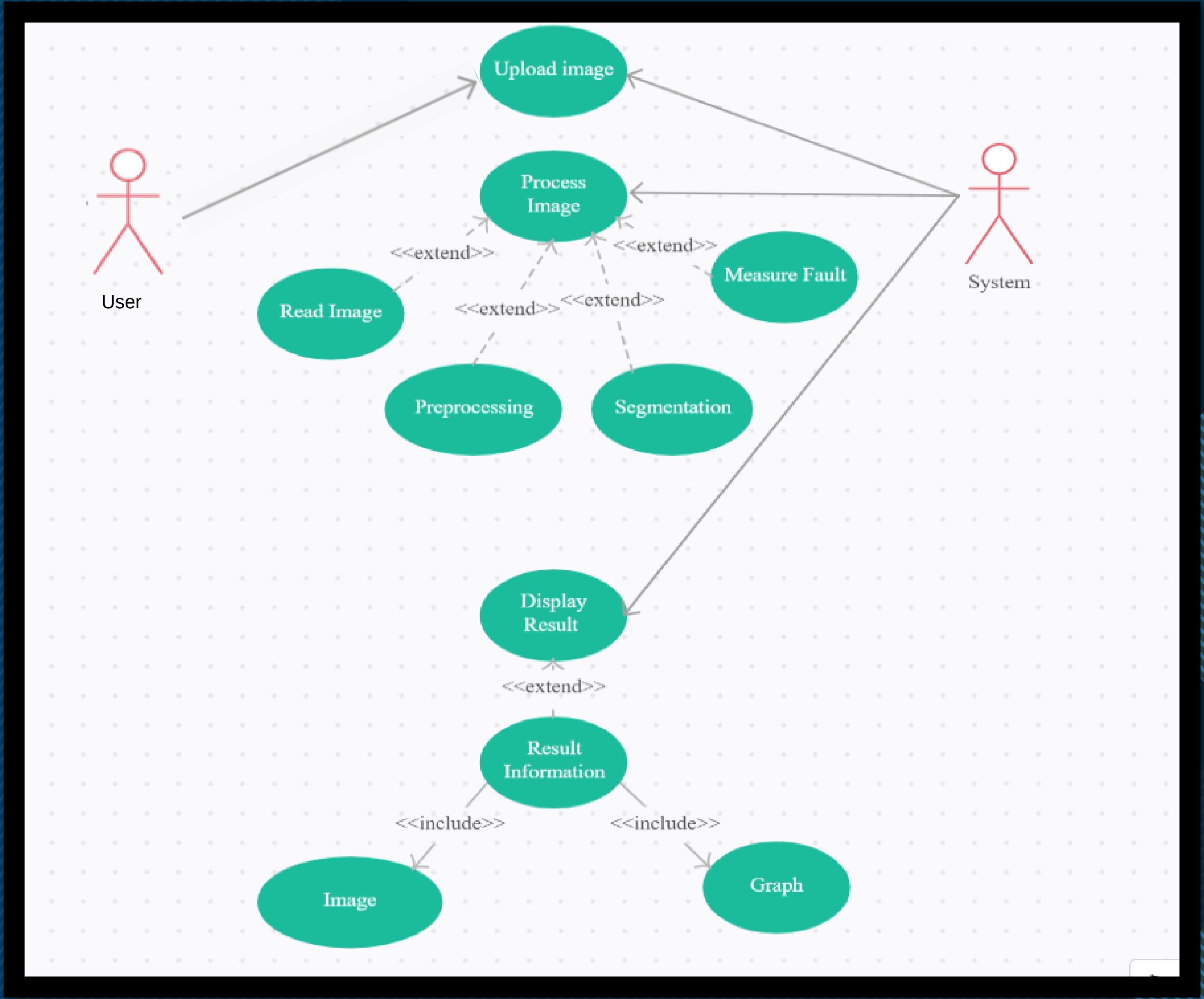
- *TensorFlow: A deep learning framework used to build and train neural networks for image segmentation.*
- *Keras: A high-level API built on top of TensorFlow, providing a user-friendly interface for developing neural networks.*
- *NumPy: A package for numerical computing in Python, used to manipulate and process the input image data.*
- *Matplotlib: A data visualization package, used to display the input image and segmentation results.*
- *Pickle: A module used to save and load trained models, allowing for easy reuse and sharing of trained models.*
- *Scikit Image: A collection of image processing algorithms, used to preprocess the input data for better results.*

PROCESS MODEL



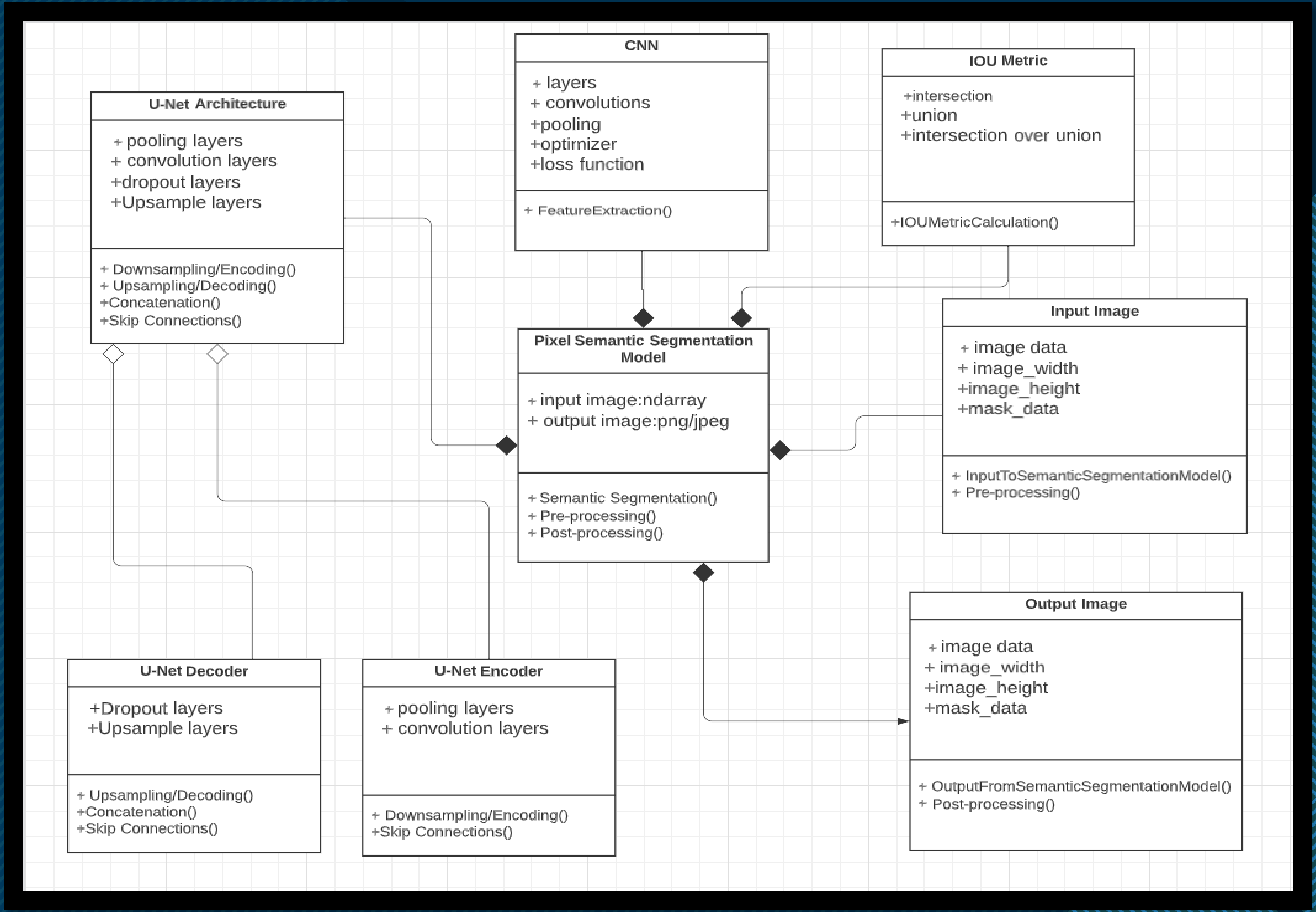
DESIGN PHASE

USE CASE DIAGRAM:



DESIGN PHASE

CLASS DIAGRAM:



CNN

+ layers
+ convolutions
+pooling
+optimizer
+loss function

+ FeatureExtraction()

IOU Metric

+intersection
+union
+intersection over union

+IOUMetricCalculation()

U-Net Architecture

+ pooling layers
+ convolution layers
+dropout layers
+Upsample layers

+ Downsampling/Encoding()
+ Upsampling/Decoding()
+Concatenation()
+Skip Connections()

Pixel Semantic Segmentation Model

+ input image:ndarray
+ output image:png/jpeg

+ Semantic Segmentation()
+ Pre-processing()
+ Post-processing()

Input Image

+ image data
+ image_width
+image_height
+mask_data

+ InputToSemanticSegmentationModel()
+ Pre-processing()

U-Net Decoder

+Dropout layers
+Upsample layers

+ Upsampling/Decoding()
+Concatenation()
+Skip Connections()

U-Net Encoder

+ pooling layers
+ convolution layers

+ Downsampling/Encoding()
+Skip Connections()

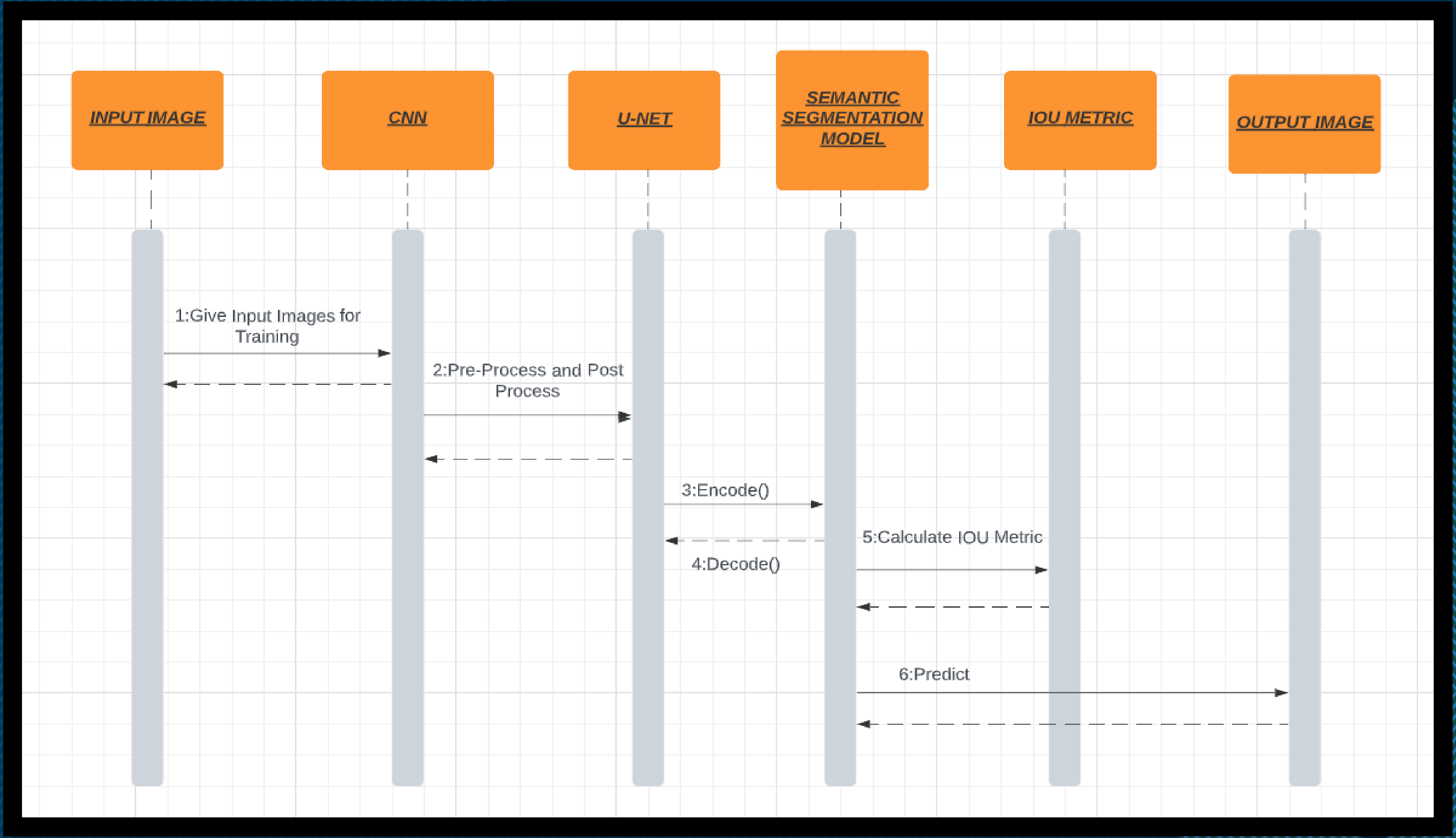
Output Image

+ image data
+ image_width
+image_height
+mask_data

+ OutputFromSemanticSegmentationModel()
+ Post-processing()

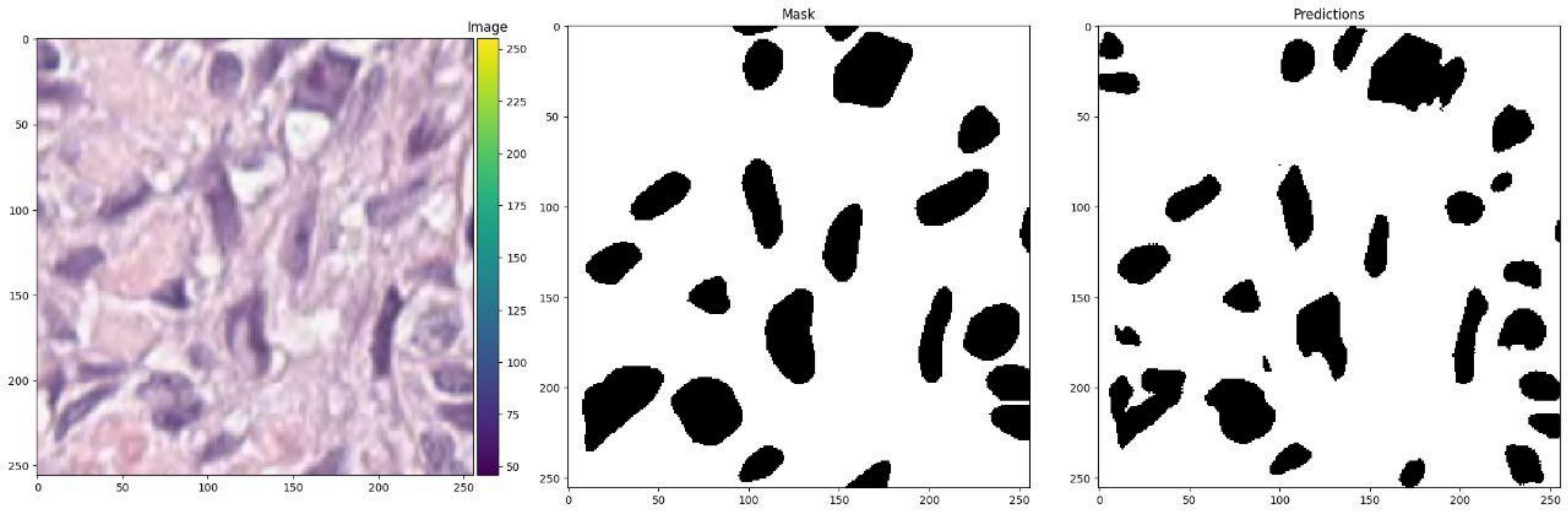
DESIGN PHASE

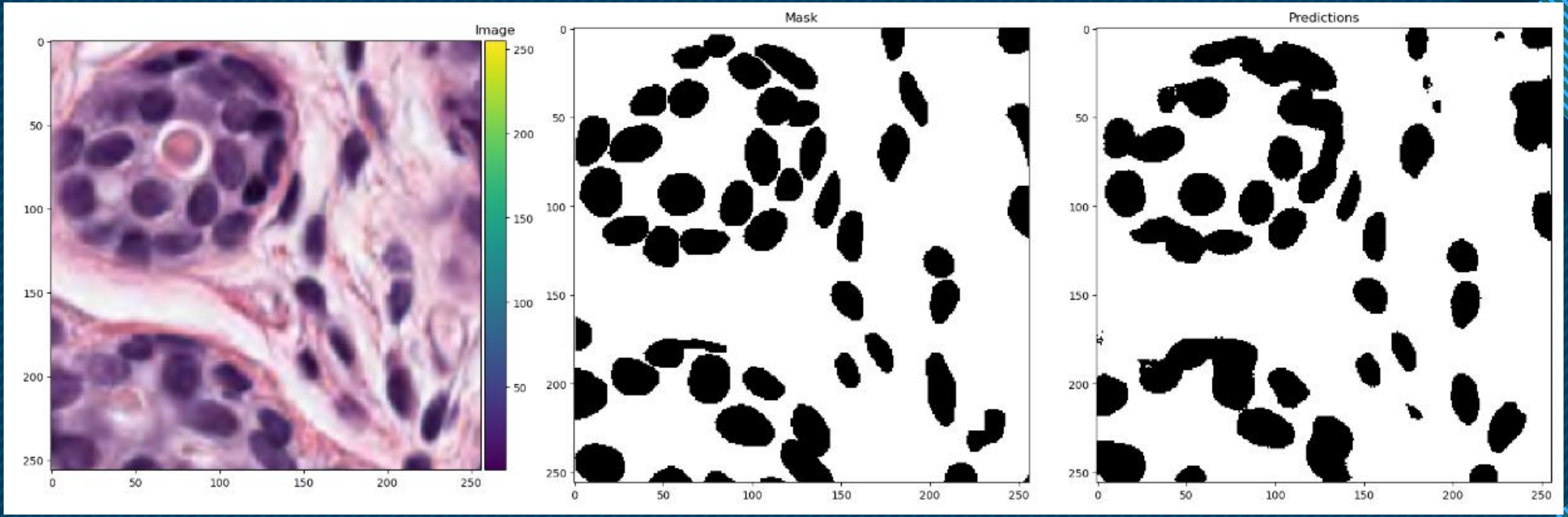
SEQUENCE DIAGRAM:



INPUT OUTPUT SPECIFICATION

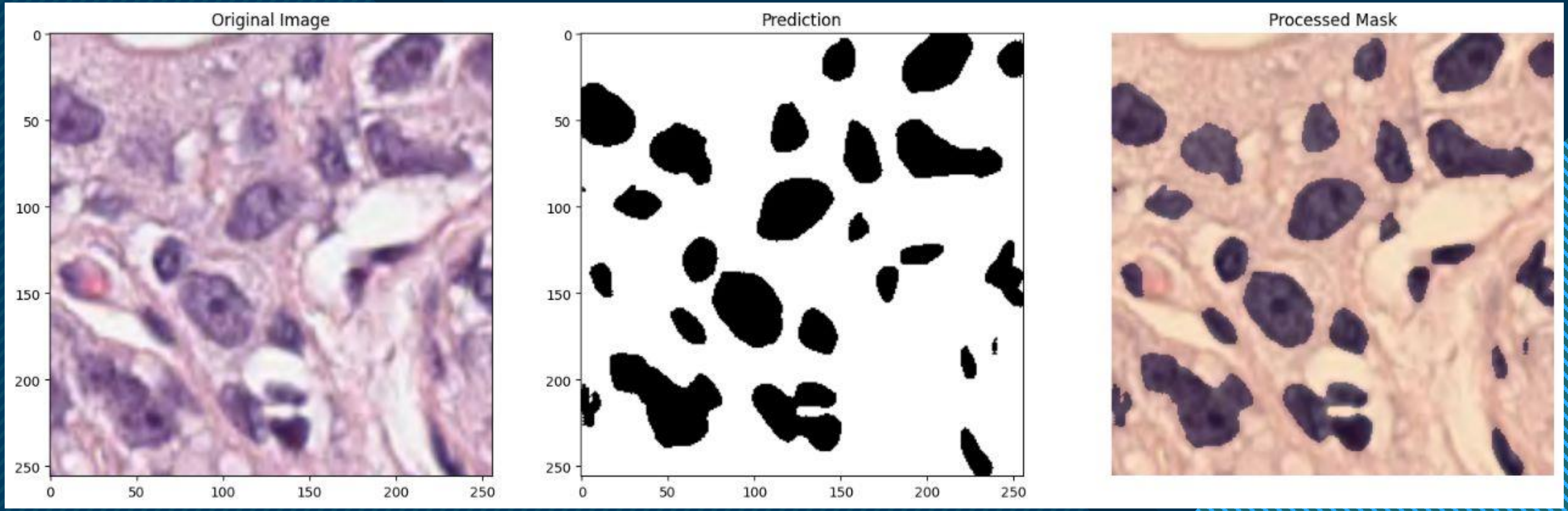
TRAINING PHASE

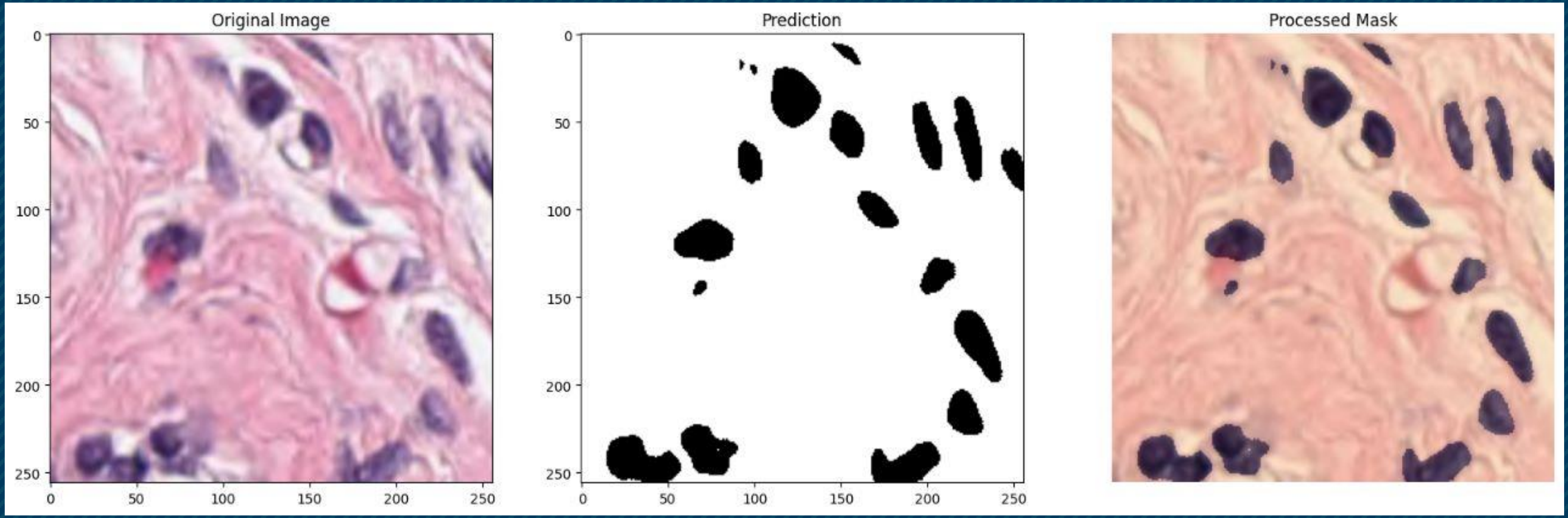




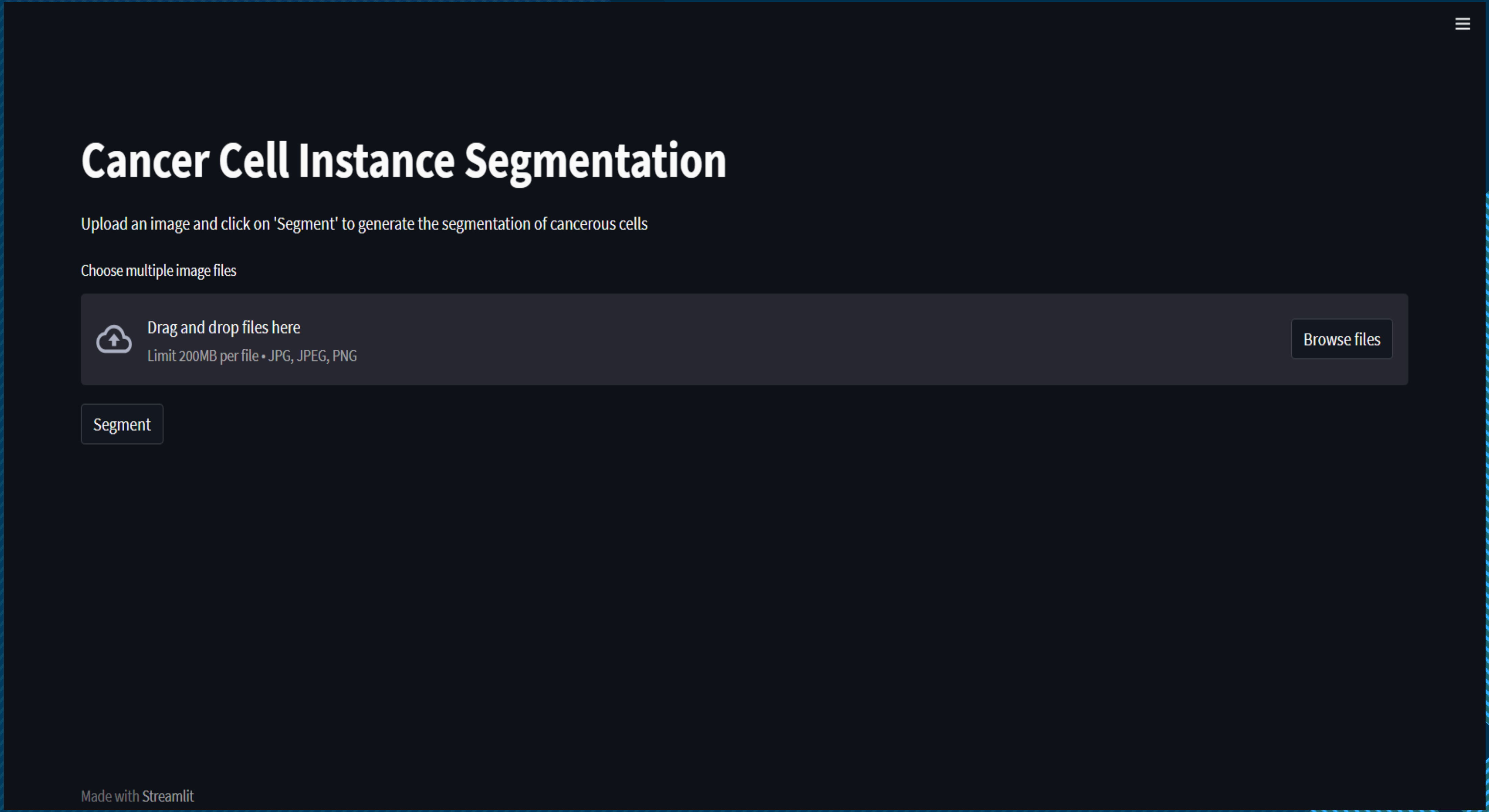
INPUT OUTPUT SPECIFICATION

TESTING PHASE





INTERFACE



Cancer Cell Instance Segmentation

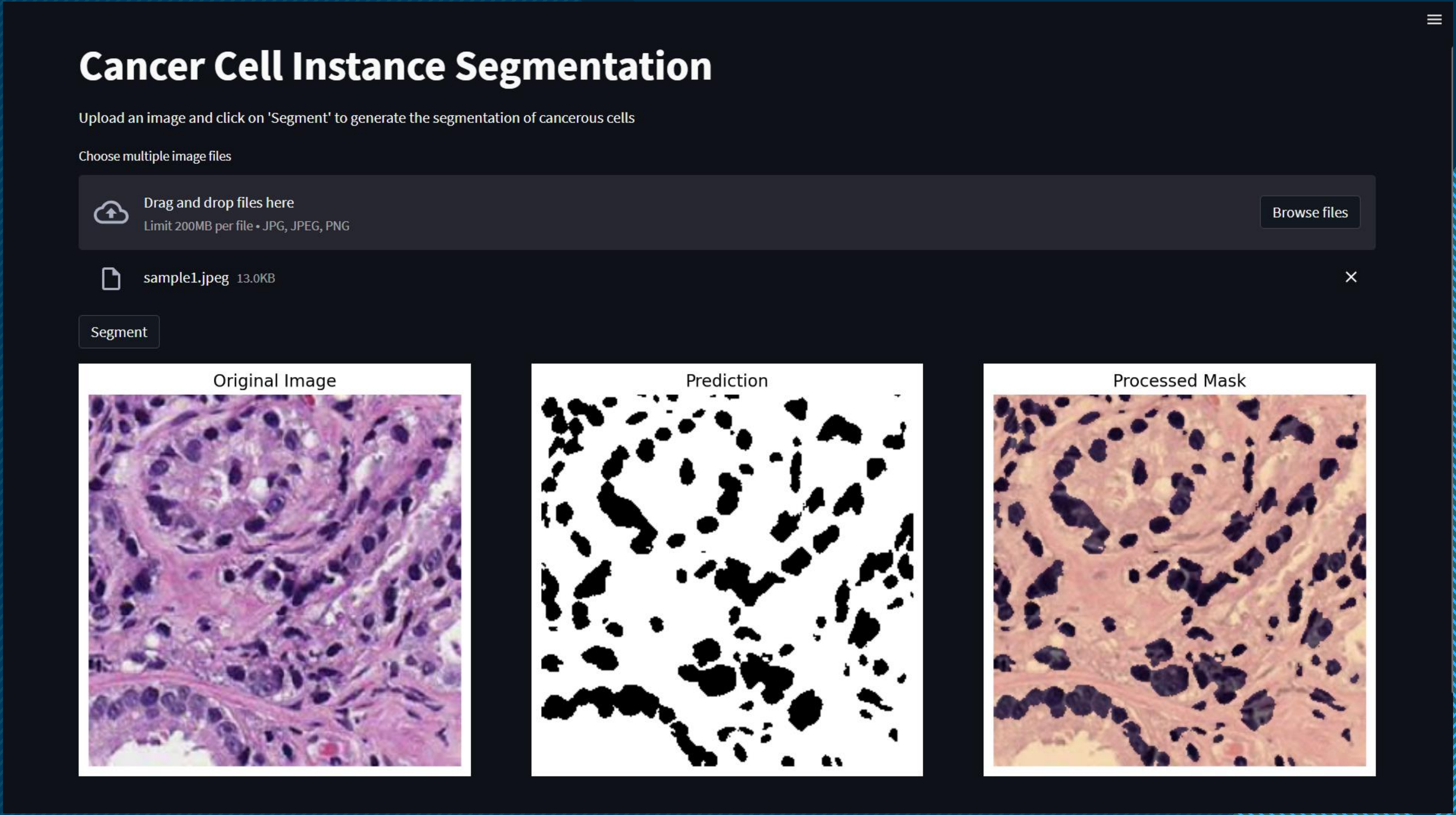
Upload an image and click on 'Segment' to generate the segmentation of cancerous cells

Choose multiple image files

Drag and drop files here

Limit 200MB per file • JPG, JPEG, PNG

INTERFACE



Cancer Cell Instance Segmentation

Upload an image and click on 'Segment' to generate the segmentation of cancerous cells

Choose multiple image files

Drag and drop files here

Limit 200MB per file • JPG, JPEG, PNG

sample1.jpeg 13.0KB

FUTURE ENHANCEMENTS

- *Improved model architectures tailored for pixel semantic segmentation tasks.*
- *Achieving higher-resolution segmentation for fine-grained details and accurate object boundary delineation.*
- *Real-time segmentation capabilities for time-sensitive applications like autonomous vehicles*
- *Incorporating contextual reasoning and scene understanding into segmentation models.*
- *Interactive and incremental segmentation methods for user guidance and continuous learning.*
- *Robustness techniques to handle noise and adversarial attacks.*

THANK YOU