ARTIFICIAL INTELLIGENCE AND COGNITIVE COMPUTING

NEURON SEMANTIC SEGMENTATION

Cognitive Computing and Artificial Intelligence
University of Catania, Italy

2023-2024

TEAM

Lorenzo Basile
1000055691

Antonio Santo Buzzone
1000055698

Angelo Cocuzza
1000055700

NEURON SEMANTIC SEGMENTATION

OUTLINE

- Introduction
- Dataset
- Model
- Results
- Conclusions

Introduction

The objective of this project is to train a deep learning network for semantic segmentation of 2-photon calcium imaging data. Semantic segmentation involves classifying each pixel in the images to identify and delineate biological structures such as neurons.

2-photon calcium imaging provides high-resolution fluorescence images of neuronal activity, which are complex and time-consuming to analyze manually. By developing an automated segmentation model, we aim to streamline this process, making it more efficient and accurate.

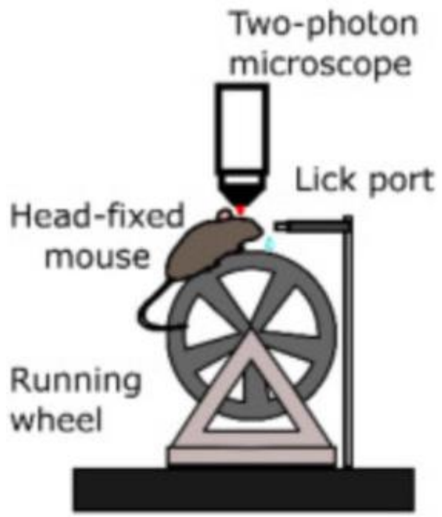
The project includes:

Data Preparation: Preprocessing and augmenting the imaging data.

Model Training: Training the neural network and evaluating it using metrics like loss, accuracy, and Intersection over Union (IoU).

Evaluation: Testing the model on unseen data and visualizing predictions.

This approach will improve the analysis of neuronal activity and contribute to advancing neuroscience research.



Dataset

Our dataset consists of 1376 .npy files, each containing:

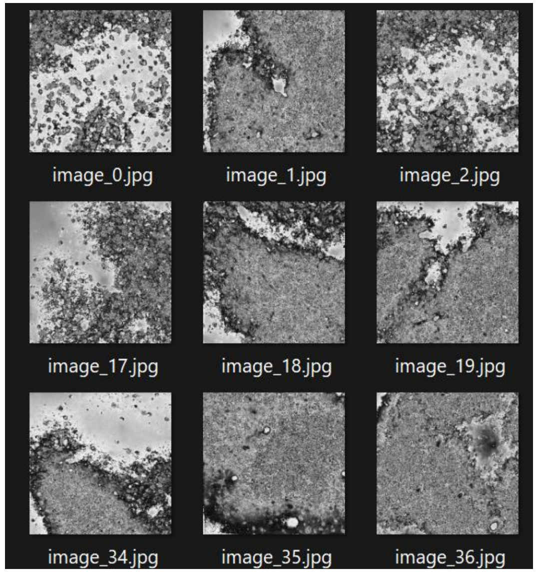
- **Image Identifier** *'id'*
- **Metadata**
- **Maximum Projection Image** *'max_projection'* **Type**:

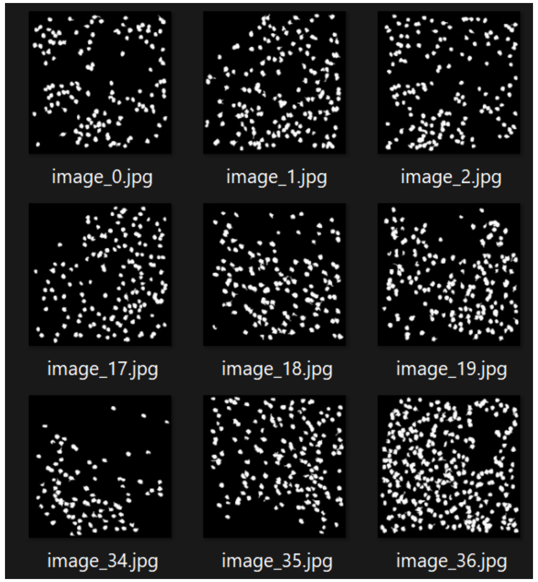
  NumPy Array (512x512)

  Description: Maximum projection image used for segmentation.
- **Neuron IDs** *'roi_ids'*
- **Neuron Mask Array** *'roi_mask_array'* **Type:**

  NumPy Array (N x 512 x 512)

  Description: Binary masks for each neuron. The masks have been combined to create a comprehensive segmentation mask.





Dataset

Dataset operations

- **Bitwise OR between masks**

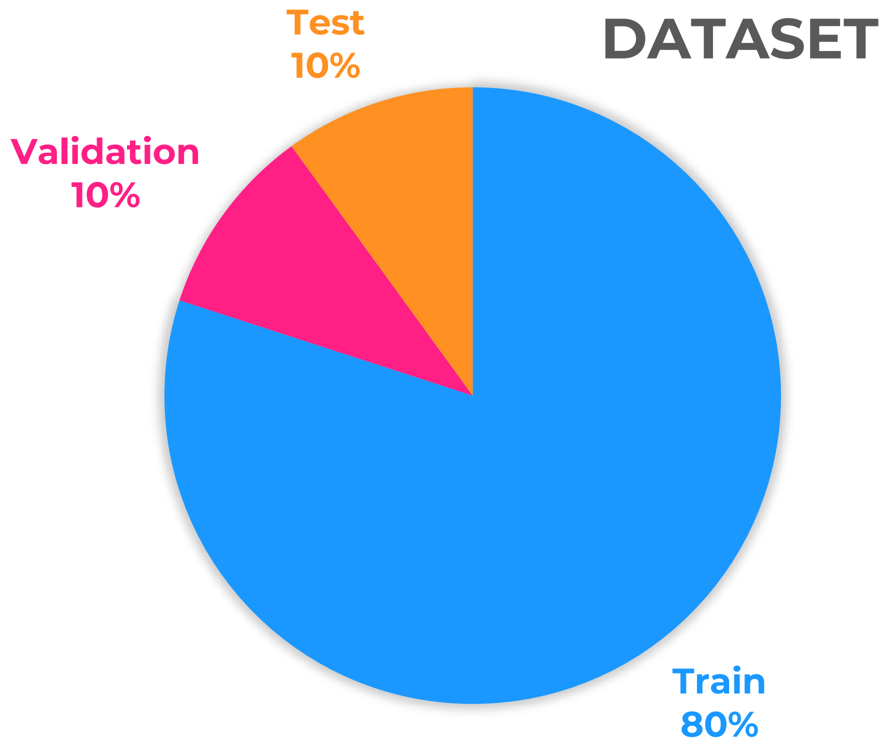
  Combine all neuron masks into a single mask using a bitwise OR operation.

- **Trasformation images and masks in file.jpg**
- **Images and masks crop from 512x512 to 256x256**

  The cropping operation is performed to enlarge the dataset by generating additional samples, resulting in a dataset four times larger than the original. It is also useful for focusing on specific areas of interest in applications such as segmentation and helps reduce overfitting.

- **Dataset Division**

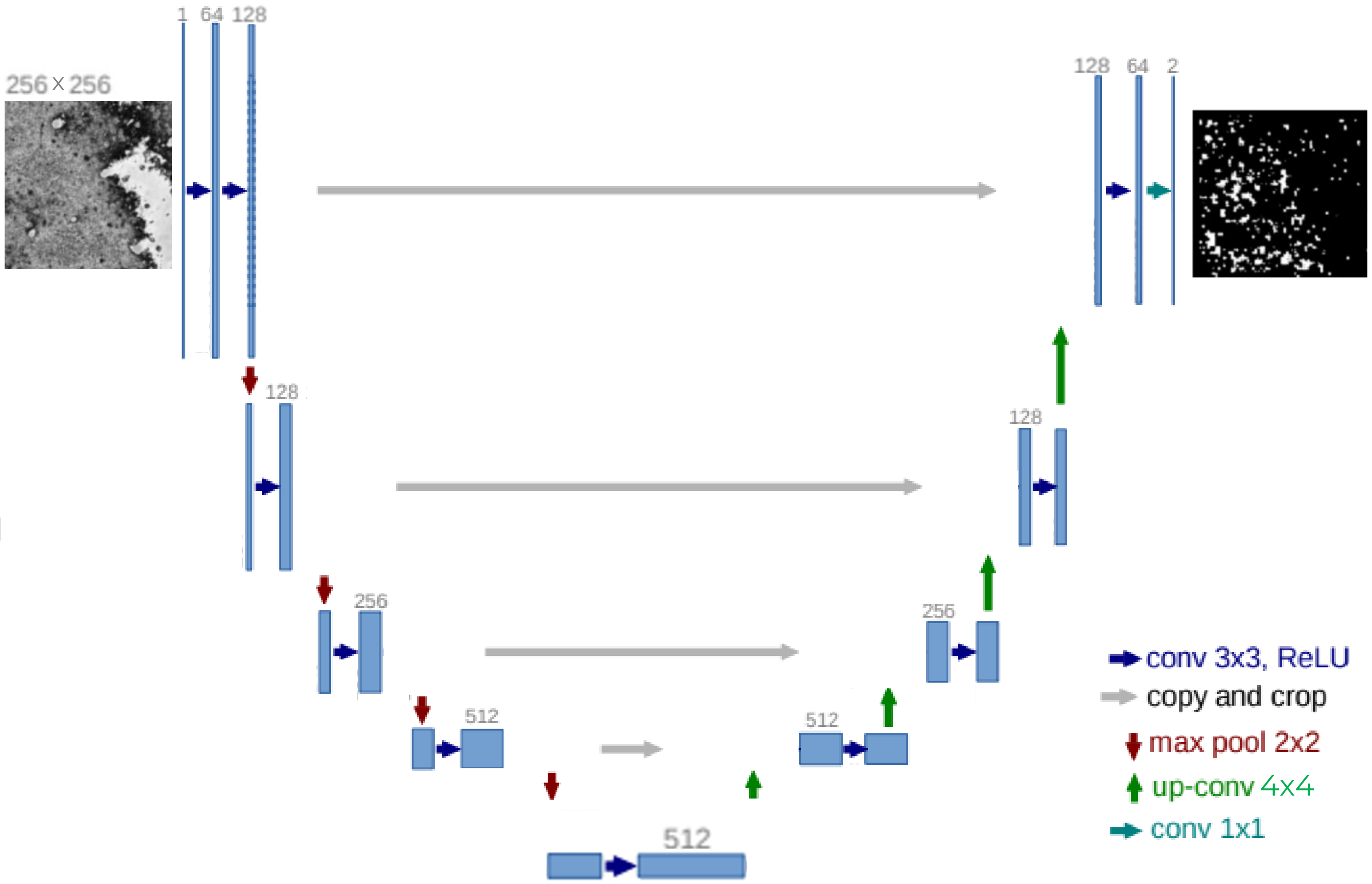
  After the crop the dataset, consisting of 4716 images, divided as follows: 3772 images for training 472 images for validation, and 472 images for testing.



Model

The defined model is a convolutional neural network with a U-Net architecture, used for image segmentation.

This network includes downsampling and upsampling paths, with the addition of dropout to improve generalization and reduce overfitting.

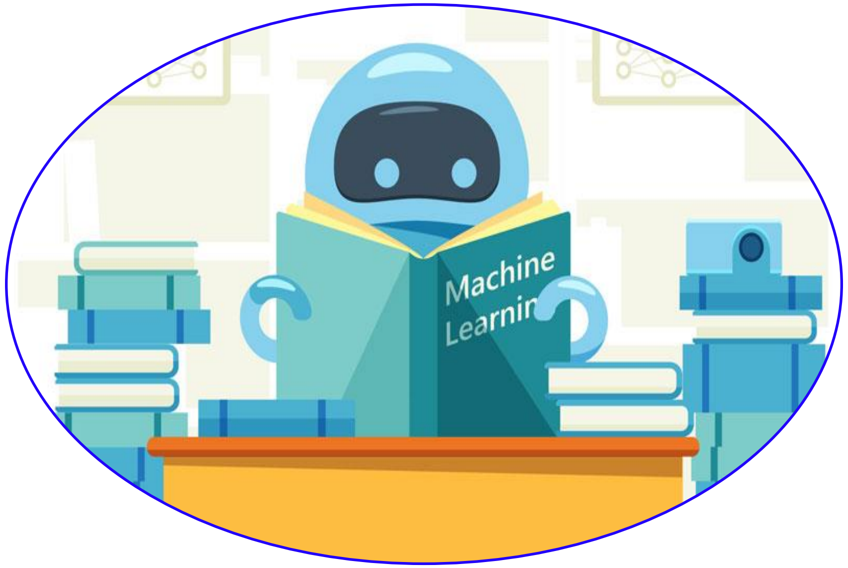


Model

Training the model

For the training phase, we set the following hyperparameters:

- Data used: training data
- Optimizer: adam
- Loss function: crossentropyloss
- Dropout probability= 0.1
- Epochs=18
- Batch size=16
- Learning rate=0.0001



Model

Validate the trained model:

- Data used: validation data

During training, for each epoch, the validation loss and validation accuracy are calculated to monitor overfitting and to make any necessary adjustments to the hyperparameters.

Evaluate the trained model:

At the end of training, we evaluate the model using:

•Loss and accuracy metrics

•Intersection over Union (IoU)

•Confusion matrix

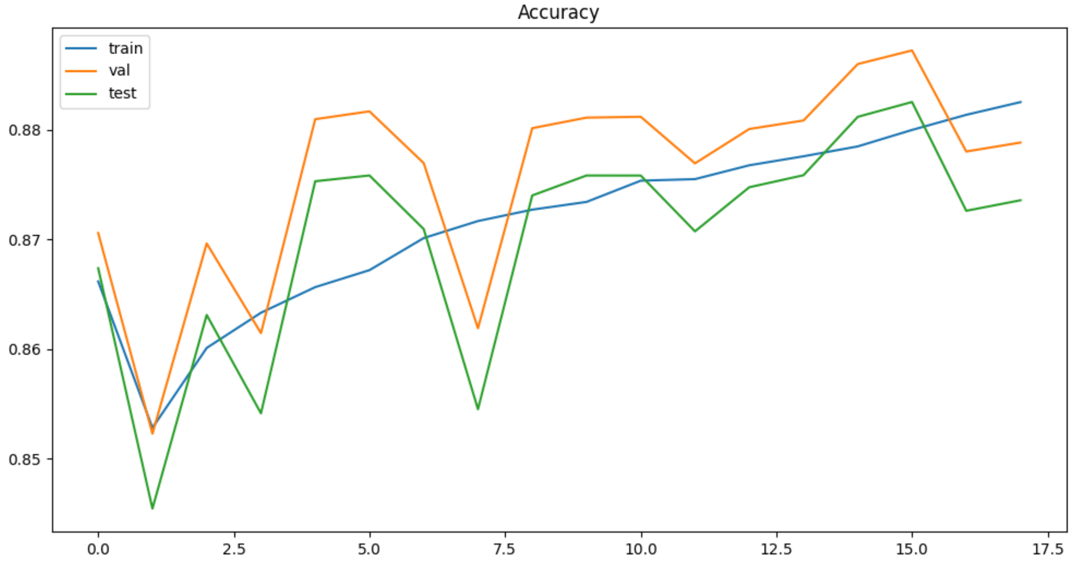
Testing the model on unseen data and visualizing predictions

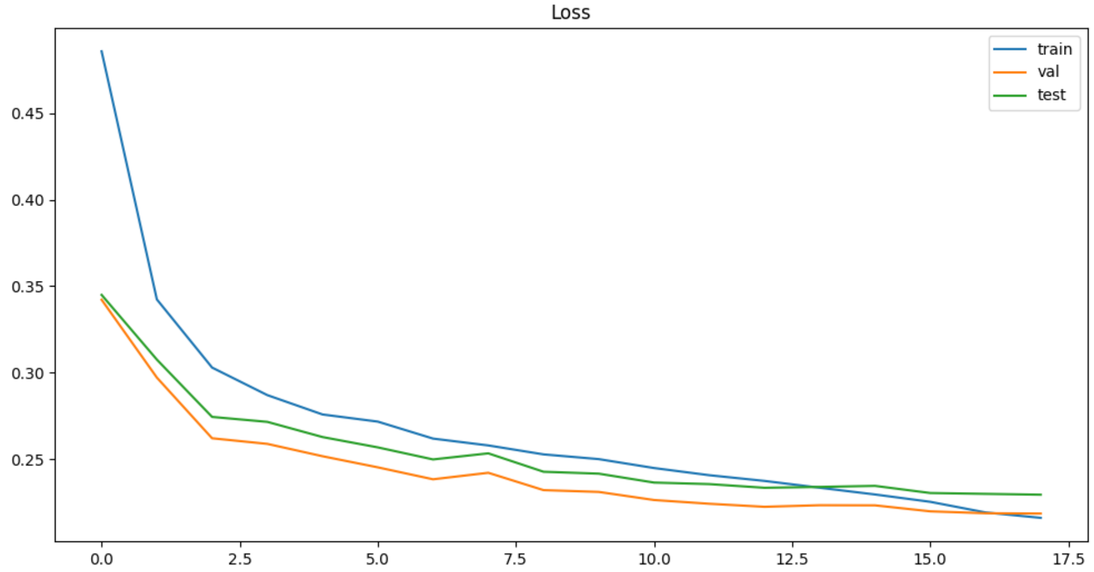
Results

At the end of the training phase, we obtained the following results. This phase was interrupted at epoch 18 as the first signs of overfitting were observed.

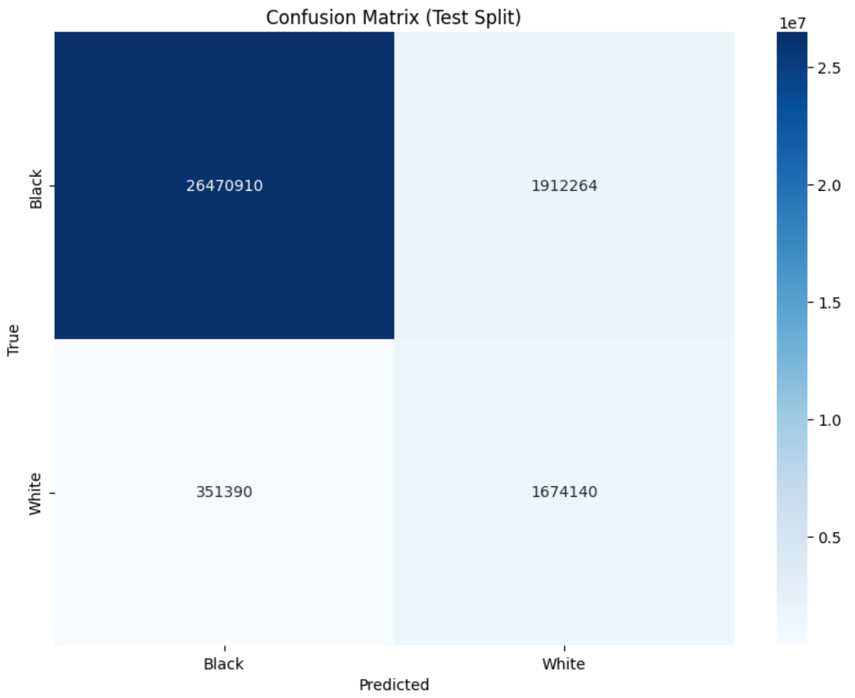
```
Epoch 9: TrL=0.2528, TrA=0.8727, TrIoU=0.6630, VL=0.2321, VA=0.8802, VIoU=0.6749, TeL=0.2427, TeA=0.8740, TeIoU=0.6654,
Epoch 10: TrL=0.2500, TrA=0.8734, TrIoU=0.6653, VL=0.2311, VA=0.8811, VIoU=0.6770, TeL=0.2416, TeA=0.8758, TeIoU=0.6690,
Epoch 11: TrL=0.2449, TrA=0.8754, TrIoU=0.6714, VL=0.2264, VA=0.8812, VIoU=0.6804, TeL=0.2365, TeA=0.8758, TeIoU=0.6723,
Epoch 12: TrL=0.2407, TrA=0.8755, TrIoU=0.6735, VL=0.2242, VA=0.8770, VIoU=0.6749, TeL=0.2356, TeA=0.8708, TeIoU=0.6655,
Epoch 13: TrL=0.2374, TrA=0.8768, TrIoU=0.6778, VL=0.2225, VA=0.8801, VIoU=0.6807, TeL=0.2335, TeA=0.8748, TeIoU=0.6722,
Epoch 13: TrL=0.2374, TrA=0.8768, TrIoU=0.6778, VL=0.2225, VA=0.8801, VIoU=0.6807, TeL=0.2335, TeA=0.8748, TeIoU=0.6722,
Epoch 14: TrL=0.2335, TrA=0.8776, TrIoU=0.6816, VL=0.2234, VA=0.8809, VIoU=0.6821, TeL=0.2340, TeA=0.8759, TeIoU=0.6744,
Epoch 15: TrL=0.2296, TrA=0.8785, TrIoU=0.6849, VL=0.2233, VA=0.8860, VIoU=0.6904, TeL=0.2346, TeA=0.8812, TeIoU=0.6827,
Epoch 16: TrL=0.2254, TrA=0.8800, TrIoU=0.6897, VL=0.2199, VA=0.8873, VIoU=0.6954, TeL=0.2305, TeA=0.8826, TeIoU=0.6875,
Epoch 17: TrL=0.2192, TrA=0.8814, TrIoU=0.6952, VL=0.2188, VA=0.8780, VIoU=0.6793, TeL=0.2300, TeA=0.8726, TeIoU=0.6714,
Epoch 18: TrL=0.2161, TrA=0.8825, TrIoU=0.6996, VL=0.2186, VA=0.8789, VIoU=0.6810, TeL=0.2295, TeA=0.8736, TeIoU=0.6732,
```

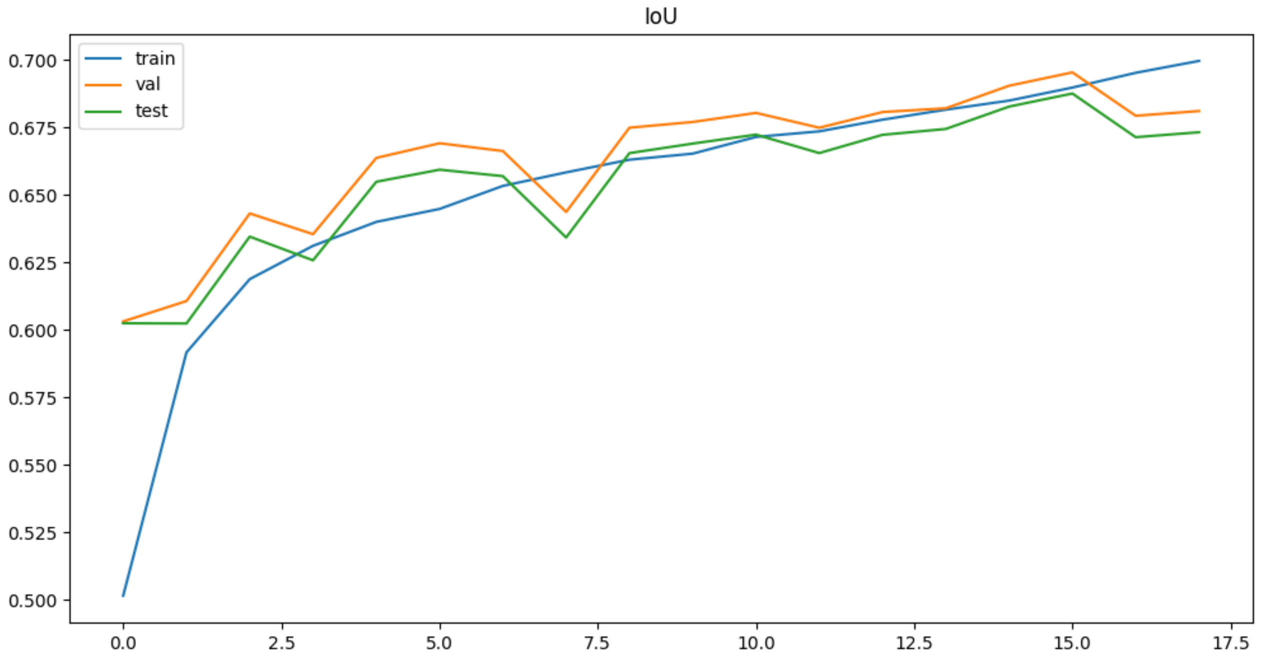
Results


Accuracy
train
val
test
0.88
0.87
0.86
0.85
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5


Loss
train
val
test
0.45
0.40
0.35
0.30
0.25
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5

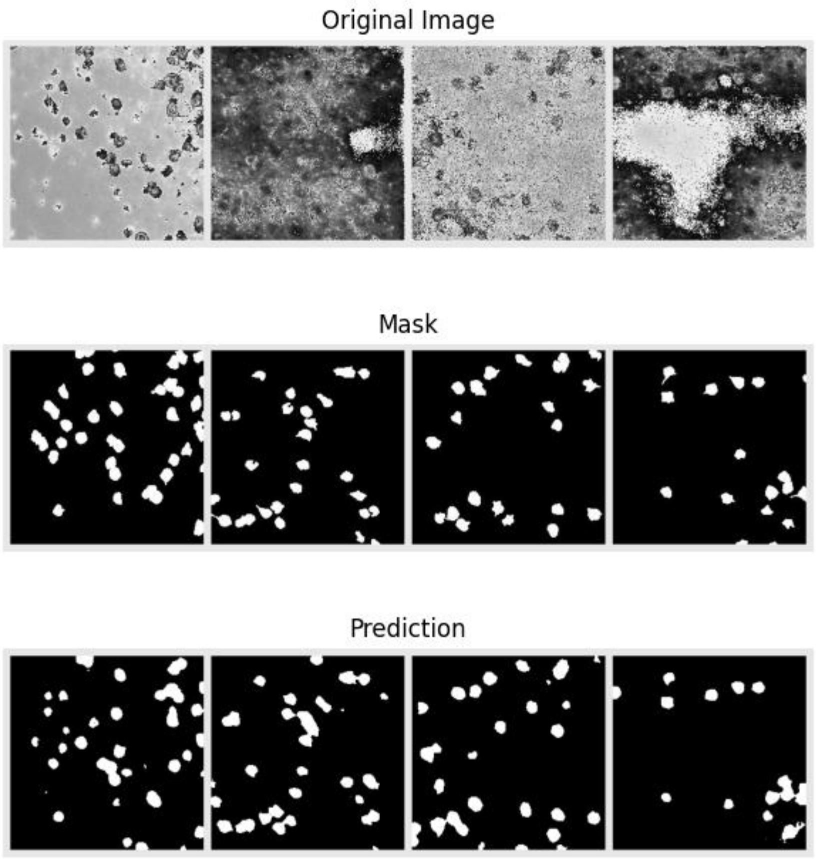
Results

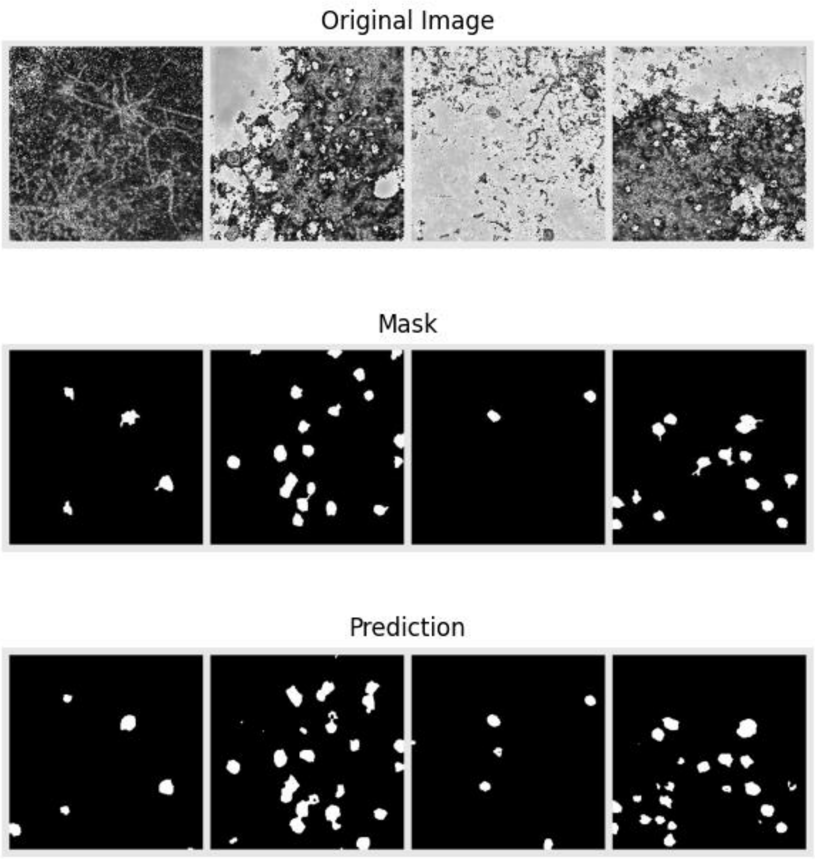


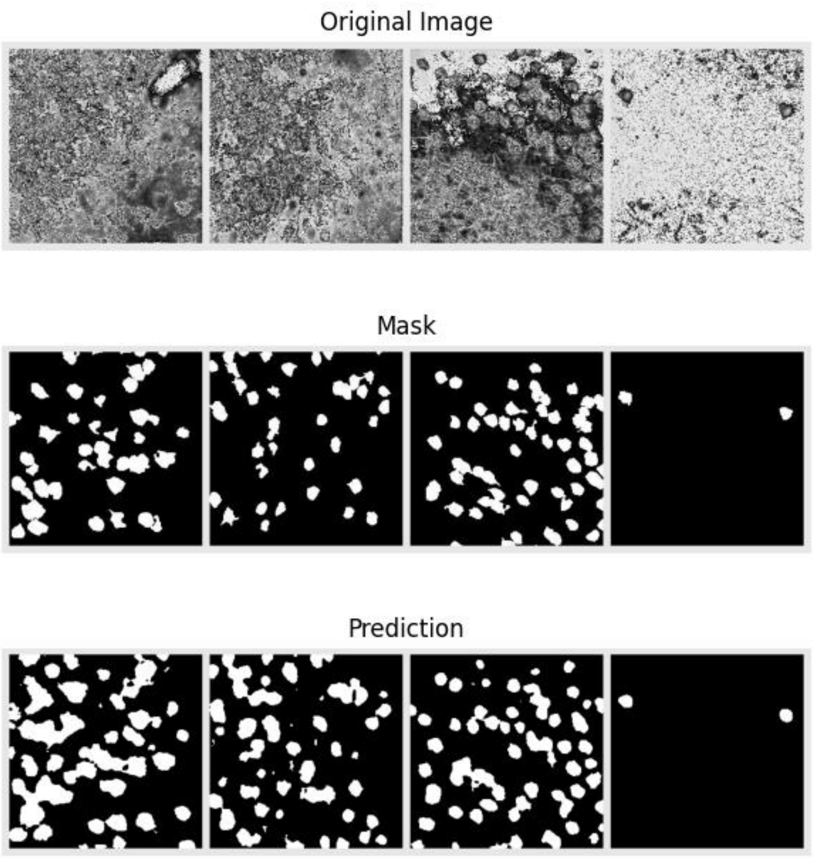


$$IoU = \frac{TP}{(TP + FP + FN)}$$

Results


Original Image
Mask
Prediction


Original Image
Mask
Prediction


Original Image
Mask
Prediction

Conclusions

The results were obtained through several fine-tuning operations.

Various combinations of hyperparameters were tested, and a pre-trained network was also used; however, significant overfitting issues were encountered.

One of the most effective improvements was the use of dropout along with accurate weight balancing, which allowed for less noisy predictions.

NEURON SEMANTIC SEGMENTATION

THANK YOU

Lorenzo Basile

Angelo Cocuzza

Antonio Santo Buzzone