BIOLOGICALLY INSPIRED ARTIFICIAL INTELLIGENCE

TOPIC: “BRAIN TUMOR DETECTION FROM MRI IMAGES”

Supervisor: mgr inż. Marcin Wierzchanowski

Student: Illia Karpenko

PROJECT OBJECTIVES

- Utilize the U-Net architecture for precise segmentation of images.
- Strive for high performance, Intersection over Union above 0.80 and Dice Coefficient above 0.80 in test results.
- Validate model performance across diverse data samples.
- Address and resolve issues related to GPU processing and numerical stability.
- Find insights into the model's performance and areas for improvement.
- Gain hands-on experience with deep learning frameworks and tools, specifically Keras and TensorFlow.

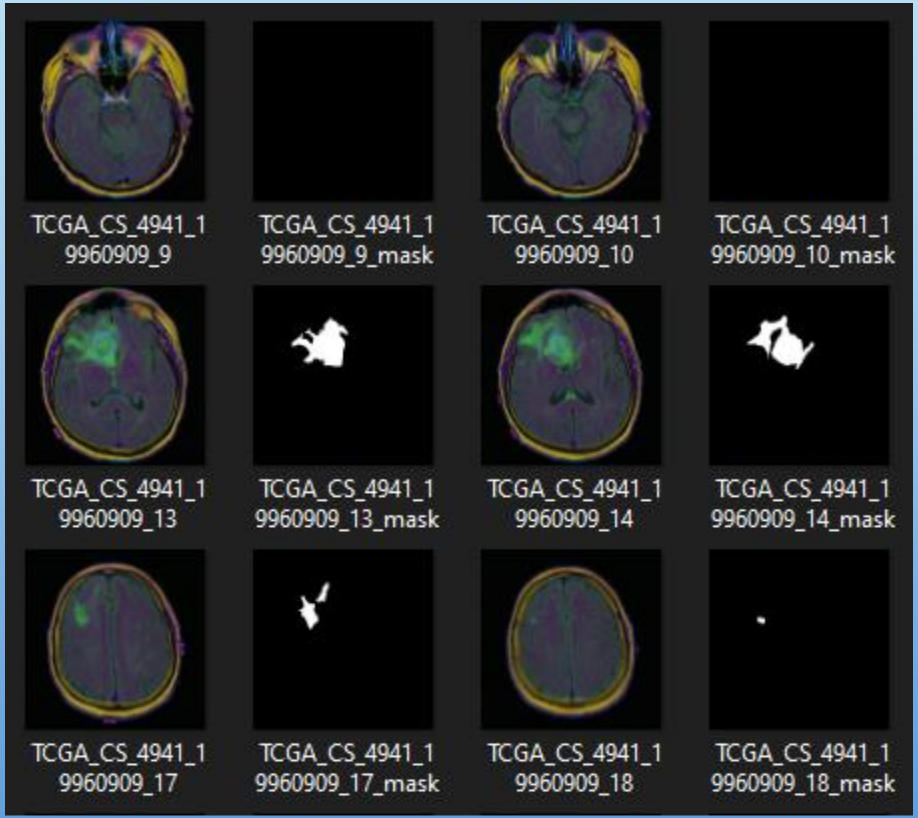
DATA COLLECTION

Data source:

- A publicly available dataset from Kaggle.
- The dataset includes images and corresponding segmentation masks necessary for training and evaluating the model.

Data Preprocessing:

- Images should be normalized.
- To enhance model generalization, data augmentation techniques will be used, such as rotation, width and height shift, shear, and zoom,



Img.1 "Example of the dataset"

MODEL SELECTION

Chosen Architecture: U-Net

U-Net is a convolutional neural network architecture designed for biomedical image segmentation.

Advantages:

- High Performance in Segmentation Tasks: U-Net has consistently shown superior results in medical and other image segmentation tasks.
- Symmetric Architecture: The encoder-decoder structure allows for efficient downsampling and upsampling, preserving spatial information.

Disadvantages:

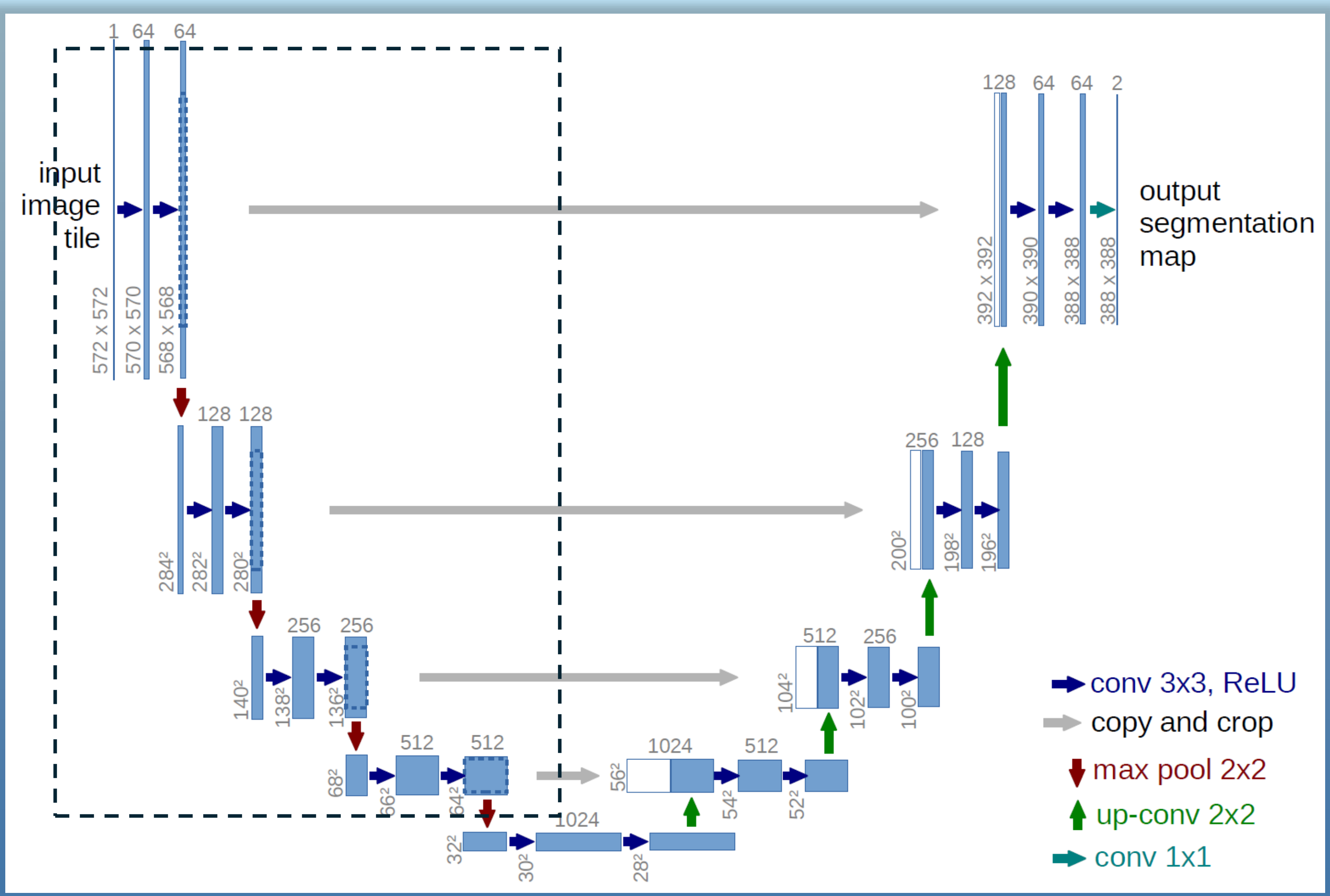
- High Computational Demand: U-Net can be computationally intensive, requiring significant GPU resources.
- Complexity: The architecture is more complex compared to simpler models, which can make it harder to tune and train.

U-NET ARCHITECTURE

Encoder path

Convolutional layers:
The encoder path consists of repeated application of two 3x3 convolutions, each followed by a ReLU activation and a 2x2 max pooling operation with stride 2 for downsampling.

Feature extraction:
This path captures the context of the image by progressively downsampling it, doubling the number of feature channels at each step.



Img.2 "U-Net architecture"

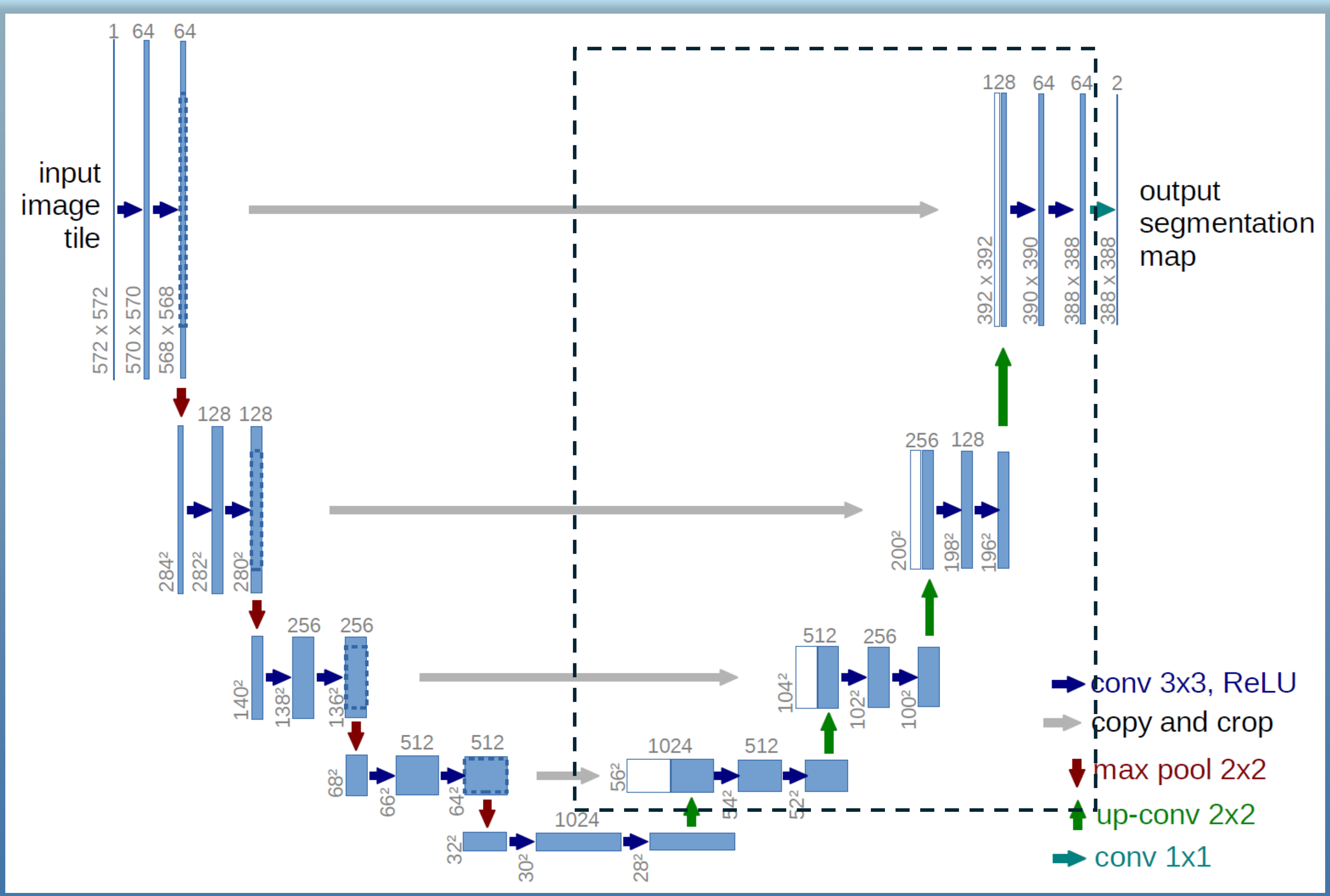
U-NET ARCHITECTURE

Decoder path

Upsampling:
Each step in the decoder path consists of an upsampling of the feature map followed by a 2x2 convolution that halves the number of feature channels.

Skip connections:
The feature map is then concatenated with the corresponding feature map from the encoder path, ensuring that spatial information is preserved.

Convolutions:
Followed by two 3x3 convolutions and ReLU activations.

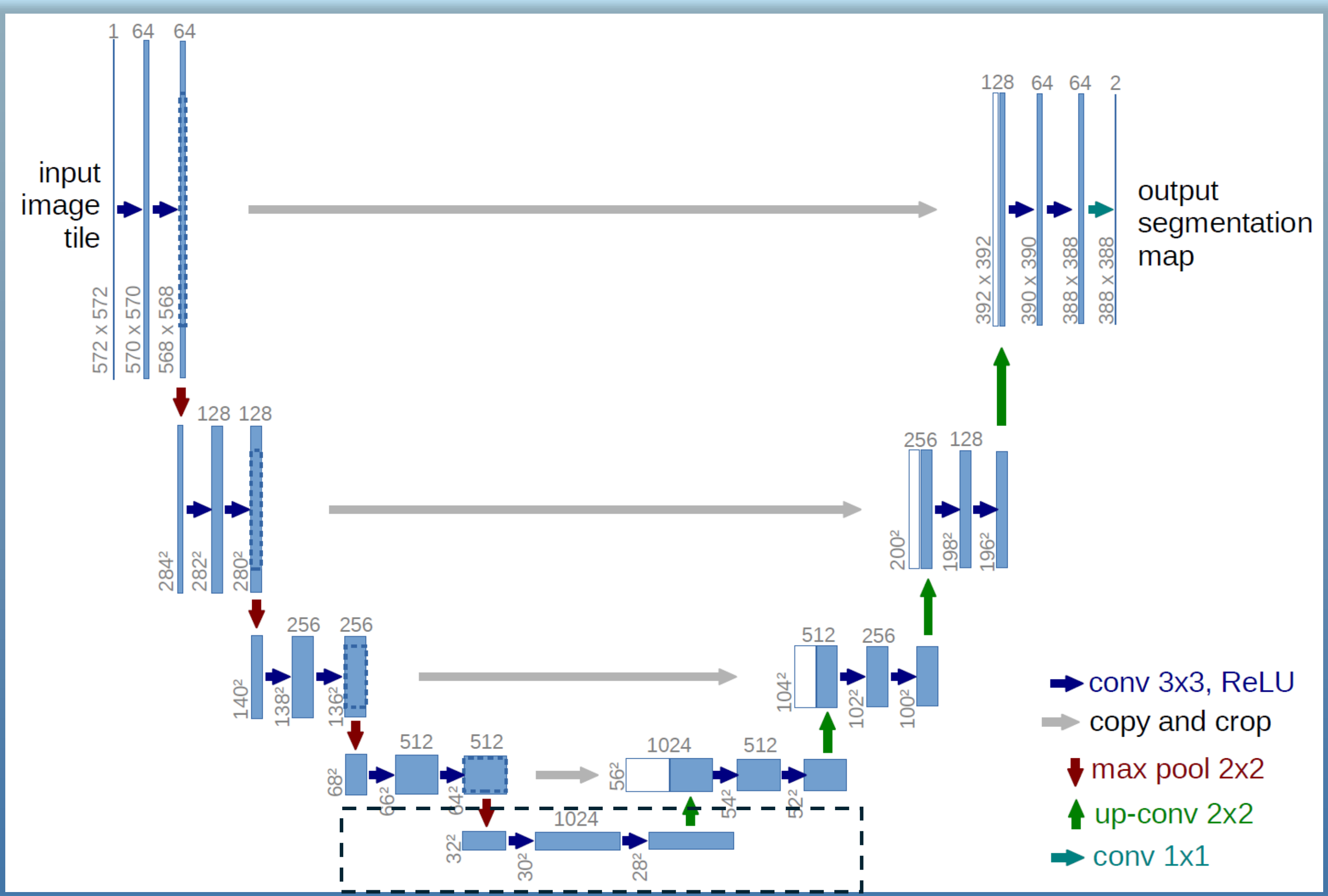


Img.2 "U-Net architecture"

U-NET ARCHITECTURE

Bottleneck:

At the bottom of the U-Net, the bottleneck connects the encoder and decoder paths. It consists of two 3x3 convolutions followed by ReLU activations, without downsampling.

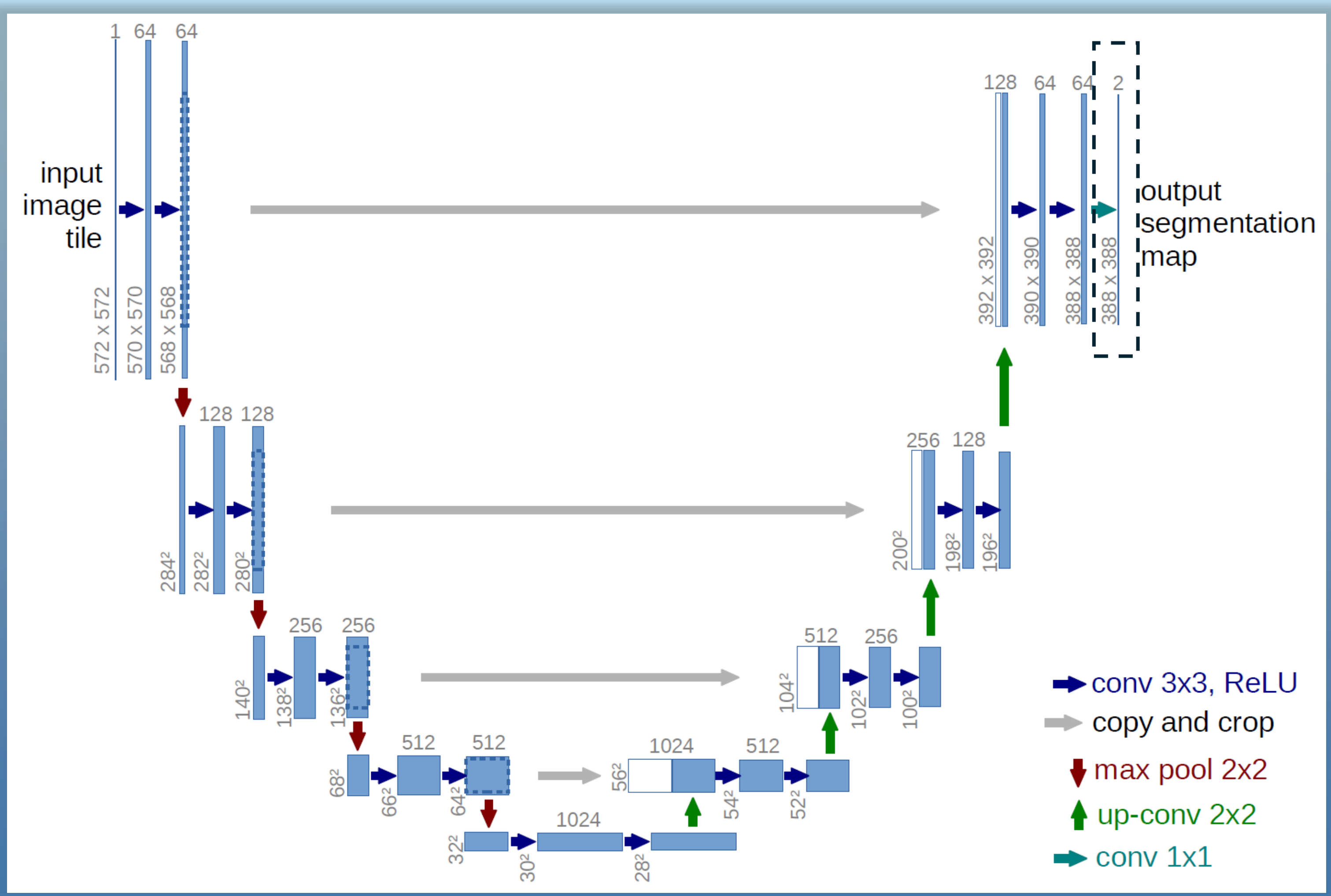


Img.2 "U-Net architecture"

U-NET ARCHITECTURE

Final Convolution:

A 1x1 convolution is applied at the end to map each 64-component feature vector to the desired number of classes (usually binary for segmentation).

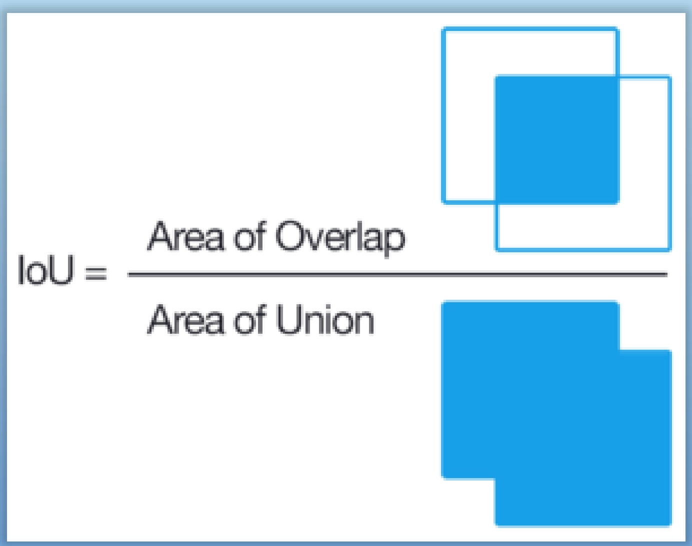


Img.2 "U-Net architecture"

EVALUATION METRICS

Intersection over Union (IoU):

- Measures the overlap between the predicted segmentation mask and the ground truth mask.
- Provides a robust metric for evaluating the accuracy of our segmentation, particularly for object detection tasks.
- Formula: IoU = (True Positives) / (True Positives + False Positives + False Negatives)

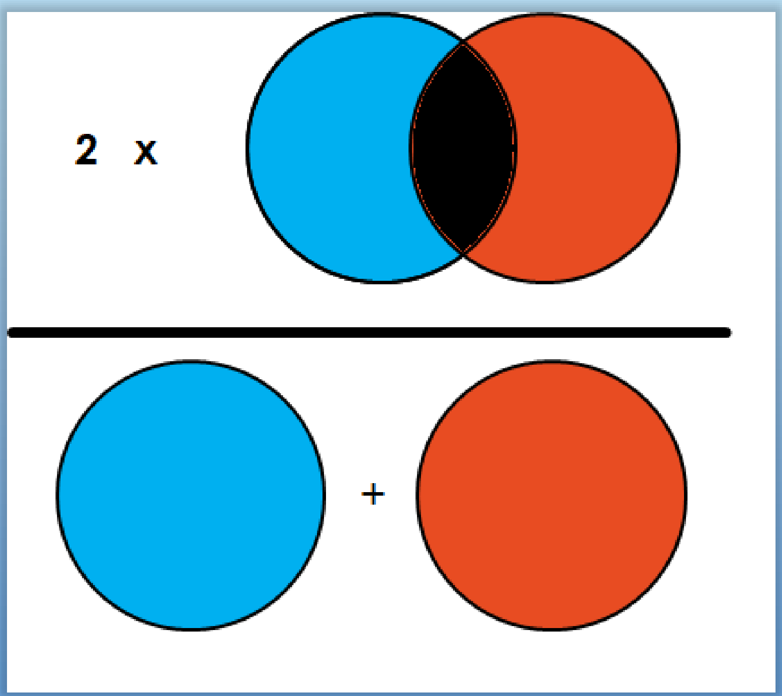


Img.3 "Visualized calculation of IoU"

EVALUATION METRICS

Dice coefficient:

- Measures the similarity between the predicted and ground truth masks, emphasizing correct overlap.
- Useful for medical imaging and other fields where the shape and continuity of the segmented object are critical.
- Formula: Dice = (2 * True Positives) / (2 * True Positives + False Positives + False Negatives)



Img.4 "Visualized calculation of dice coefficient"

CHALLENGES

- Training deep learning models like U-Net is computationally intensive, requiring significant processing power and memory.
- Large datasets required for training can be difficult to manage and process efficiently.
- The risk of overfitting the model to the training data, leading to poor generalization on unseen data.
- Finding the optimal set of hyperparameters (learning rate, batch size, etc.) can be time-consuming.

THANK YOU FOR YOUR ATTENTION!