EE 469

IMAGE PROCESSING

COURSE PROJECT: PORTRAIT SEGMENTATION

Ali Baki TURKOZ

Faculty of Engineering

Department of Electrical and Electronics Engineering

Istanbul, Fall 2023

Yeditepe University

Table Of Contents

ABSTRACT

PORTRAIT SEGMENTATION

A portrait is a drawing or photograph depicting an individual, particularly focusing on the face, head, and shoulders. In the realm of portrait segmentation, a pivotal step involves the meticulous segmentation process where semantically related pixels, encompassing hair, face, body, and background, are intricately delineated. In the digital domain, a person's portrait is achieved by the inclusion of the individual in the image, with the adjustment of the background to create a blurred effect (Figure 1). Each pixel in the image is assigned a class value, and pixels with the same class value are segmented together due to their shared characteristics, such as depicted in the image [1]. However, due to variations in hair shape, colour, lighting, angle, and background, this process poses significant challenges.

This study encompasses various portraits and their segmented images. Portrait segmentation has been applied to images featuring diverse angles, lighting conditions, and orientations. Numerous segmentation methods exist in the literature, with the primary drawbacks of current methods being low segmentation accuracy and excessively large network structures [1]. In this study, the U-Net deep learning model is employed for portrait segmentation. The proposed model is implemented due to its ability to achieve rapid and highly accurate portrait segmentation. Algorithms are evaluated on invisible portrait examples using F-measures with precision and recall, and the execution times of the selected method are considered.

Keywords: Portrait Segmentation, U-Net, Deep Learning, F-Measure, Semantic segmentation

Figure 1:Real-time Portrait Segmentation

LIST OF FİGURES

LIST OF SYMBOLS/ABBREVIATIONS

DL	Deep Learning
CNN	Convolutional Neural Network
AR	Augmented Reality
FCN	Fully Convolutional Networks
ROI	Region Of Interest
GT	Ground Truth
GPU	Graphics Processing Unit

1. INTRODUCTION

Image segmentation has become one of the most challenging problems in the fields of image processing and computer vision in recent years. Unlike image classification algorithms, which are tasked with classifying objects with specific labels, image segmentation goes beyond this by requiring an ideal algorithm to delineate even unknown objects into distinct segments. The literature is replete with numerous image segmentation algorithms, some of which have been tested and elucidated in the Literature Review section. Traditional image segmentation algorithms generally rely on clustering technologies.

Among image segmentation algorithms, Deep Learning-based image segmentation has ushered in a new era, giving rise to next-generation models that, in many aspects, surpass traditional segmentation methods [2,3]. Deep Learning-based image segmentation aims to predict category labels for each pixel in every image, a task of paramount importance but one that poses significant challenges. Semantic image segmentation plays a central role in a wide range of applications, including medical image analysis, autonomous vehicles, video surveillance, and Augmented Reality (AR) [4]. Portrait segmentation, a subset of semantic image segmentation, is often employed to meticulously segment the upper torso of an individual in an image. In portrait segmentation, achieving precise segmentation of the human body is crucial but inherently challenging.

Given the substantial increase in the performance and accuracy of semantic image segmentation attributed to Deep Learning technology, a significant number of Deep Learning-based portrait segmentation approaches have been developed [5]. This project aims to identify the optimal approach by testing various Deep Learning-based portrait segmentation models, with the experimental results aiming to highlight the capability of the proposed approach to deliver high-performance outcomes.

2. LITERATURE REVIEW

Deep learning, a state-of-the-art artificial learning paradigm, has garnered significant attention from researchers due to its high-precision classification capabilities. Convolutional Neural Networks (CNN) are particularly prominent among current methodologies, owing to their ease of use, high matching capabilities, and GPU-supported parallel processing features in tasks such as image classification, object recognition, and detection.

A segmentation model learns the desired task from given inputs to produce the expected output. In this project, deep learning-based algorithms will be explored, specifically utilising Convolutional Neural Networks (CNN). CNNs have played a pivotal role in approaching human accuracy in recognition and localization tasks. This is attributed to the network's possession of numerous deep layers, enabling it to learn various features from an image. With the advent of large datasets and novel training approaches, the network has become highly sensitive in detecting the regions of interest. The Fully Convolutional Network (FCN) consists of two sections: convolution and deconvolution. Convolutional layers facilitate the extraction of diverse features and the positioning of the region of interest in a smaller dimension. In this case, reducing the dimensions of the 800x600pixel dataset allows for faster and more accurate processing (Figure 2).

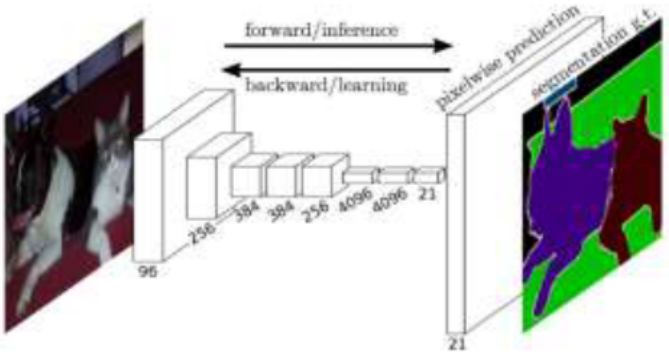

Figure 2: A Fully Connected Layer

In recent literature, a considerable number of deep learning-based portrait segmentation networks have been developed. Song-Hai Zhang et al. proposed a novel semantic segmentation network named PortraitNet, specifically designed for real-time portrait segmentation on mobile devices with limited computational power. Their experimental results demonstrated both high accuracy and efficiency [6]. The Fully Convolutional Network (FCN), introduced by Jonathan Long and others, adapts a deep learning algorithm initially proposed for Full Convolutional Contour Detection.

The objective of this project is to apply segmentation to images in a portrait dataset, which includes images with different backgrounds, lighting, angles, and noise challenges. The goal is to separate the background from the portrait by applying portrait segmentation. There are various AI-based segmentation models for this process, including semantic segmentation, UNet, Mask R-CNN, and Fast R-CNN, all of which are deep learning-based algorithms.

The process began with an understanding of portrait segmentation and a review of previous studies. After examining object detection studies through Mathworks, the pre-trained ResNet-50 model was applied. However, this pre-trained model proved insufficient for processing a large number of images in the dataset.

Subsequently, using the Image Labeler app, portrait and background were successfully separated. However, this manual method proved impractical due to the large dataset.

Thus, based on the results and more detailed research, it was determined that the most effective method is deep learning-based algorithms. The advantage of deep learning-based algorithms is their ability to classify each pixel in an image, ultimately resulting in an image segmented according to classes.

The first approach includes semantic segmentation and the UNet model. The disadvantages of semantic segmentation include the prolonged training and testing phases. Additionally, the model struggled to adapt to different lighting and angles.

The "Semantic Segmentation Using Dilated Convolutions" technique was applied through Mathworks. However, upon examining the processed images, it was observed that the semantic segmentation model did not yield satisfactory accuracy rates due to various

angles, lighting, and noises. Therefore, the Semantic Segmentation model was deemed inappropriate.

Consequently, it was determined that the research findings through Mathworks were insufficient. Following further research, the U-Net model was chosen for experimentation. The U-Net model is a compact and efficient model containing 7 million parameters. Its low parameter count facilitates faster and more efficient training from scratch. The portrait images trained in this project have two class labels (portrait and background), making a model with fewer parameters sufficient.

3. PROPOSED WORK

U-Net is an architecture for semantic segmentation. It consists of a contracting path and an expansive path. The contracting path follows the typical architecture of a convolutional network. It consists of the repeated application of two 3x3 convolutions (unpadded convolutions), each followed by a rectified linear unit (ReLU) and a 2x2 max pooling operation with stride 2 for downsampling. At each downsampling step we double the number of feature channels. Every step in the expansive path consists of an upsampling of the feature map followed by a 2x2 convolution ("up-convolution") that halves the number of feature channels, a concatenation with the correspondingly cropped feature map from the contracting path, and two 3x3 convolutions, each followed by a ReLU. The cropping is necessary due to the loss of border pixels in every convolution. At the final layer a 1x1 convolution is used to map each 64-component feature vector to the desired number of classes. In total the network has 23 convolutional layers.The general U-Net architecture is shown below.(Figure 3) [7]

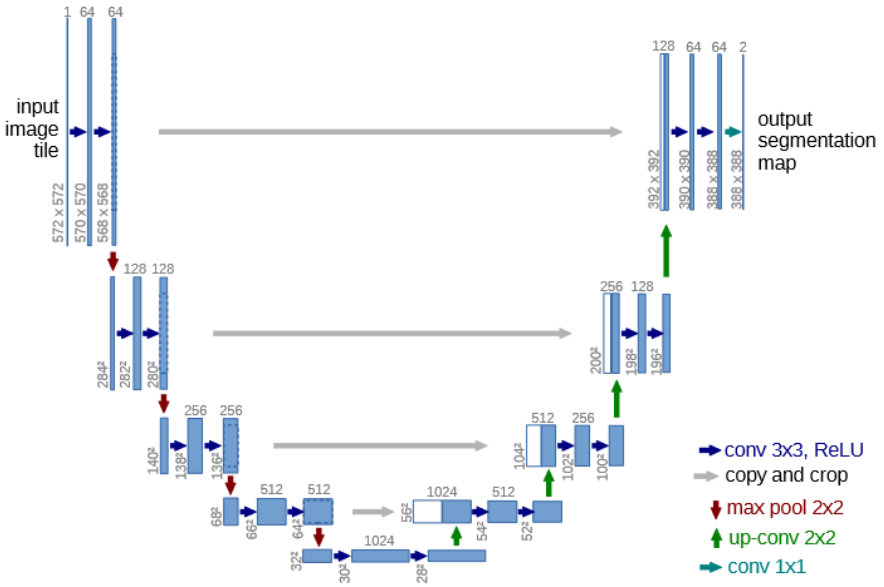

Figure 3: U-Net model general architecture

U-Net is an encoder-decoder network architecture shaped like a 'U', comprising four encoder blocks and four decoder blocks connected through a bridge. The encoder network reduces the dimensions of images by half and doubles the number of filters. Conversely, the decoder network doubles the dimensions of images while halving the number of filters. The encoder involves convolution, max pooling, and dropout operations, while the decoder includes conv2DTranspose, concatenate, and dropout operations.

In the field of image segmentation, two primary types exist:

Semantic segmentation, which involves classifying each pixel with a label, and Instance segmentation, which aims to classify each pixel while distinguishing individual object instances.

Portrait segmentation methods can be classified into three groups based on various visuals: Manual, semi-automatic, and fully automatic segmentation.[7]

Manual segmentation involves manually outlining the boundaries of portrait images and representing different structures in the image with distinct markers. Manual annotation requires software tools with advanced graphical user interfaces to facilitate the drawing and representation of regions. Semi-automatic portrait segmentation necessitates human intervention in three stages. These stages involve initiating the segmentation method, verifying the accuracy of the result, and even manually correcting the segmentation output [8]. In fully automatic methods, portrait segmentation is performed by computers without any human interaction. These methods typically combine human intelligence and prior knowledge within algorithms [9].

The simplified U-Net model is an architecture designed by Ronneberger et al. specifically for biomedical image segmentation [10]. This architecture consists of two paths: the encoder and the decoder. The encoder, the first path, involves a downsampling path used to capture features in the image. It comprises traditional convolutional layers, rectified linear unit (ReLU) applied after each convolutional layer, and max-pooling layers. The second path, the decoder, is a symmetric expansive path used for precise localization by applying transpose convolutions. It is solely composed of convolutional layers without any fully connected layers, making it an end-to-end fully convolutional network. The original U-Net

model encompasses a total of 23 convolutional layers. The architectural depiction of the U-Net is illustrated in Figure 2. It is named 'U-Net' due to its resemblance to the letter 'U', where the left side represents the contracting path, and the right side illustrates the expansive path. The contracting path consists solely of convolutional neural networks. The input image undergoes convolution with 64 filters of size 3x3. Feature maps resulting from each convolution pass through the ReLU activation function, followed by 2x2 max-pooling for downsampling. This sequence of convolution+ReLU+max-pooling is repeated three more times. With each downsampling step in the contracting path, the number of filters doubles. The intent behind the initial reduction in size in the first part of the model is to increase it in the second part, i.e., the expansive path, aiming to enhance the output resolution. To achieve this, a 2x2 up-convolution is applied in the expansive path, halving the number of filters. Subsequently, the feature map suitably cropped from the contracting path is concatenated with the output obtained from the expansive path. This concatenated result undergoes two consecutive 3x3 convolutions. Similarly, after each convolution operation in this segment, the ReLU activation function is applied, and these sequential operations are repeated thrice. In the final layer, a 1x1 convolution operation is employed to map the 64-dimensional feature vector to the desired number of classes [10].

Google Colab was employed due to inadequate graphical processing capacity on the PC. The implementation involved the utilisation of libraries including Keras, TensorFlow, and NumPy.

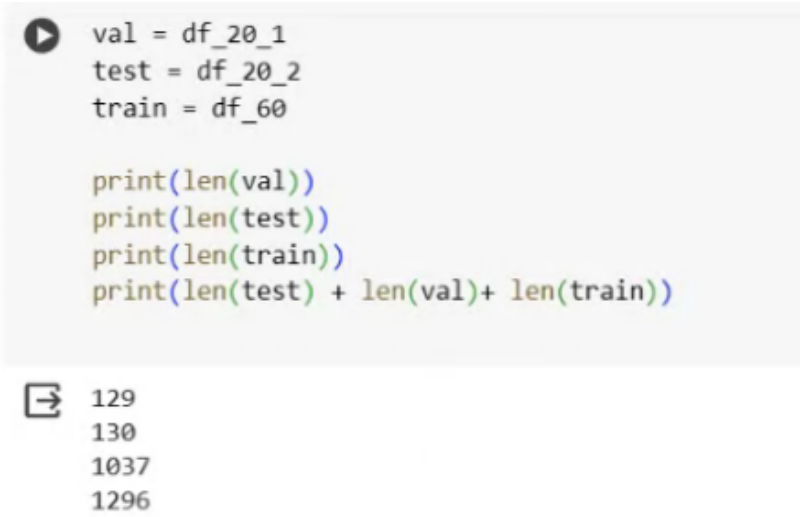

Figure 4: Data Splitting via Colab

During code development, the “imgaug” library was employed. The “imgaug” library serves as a Python module utilised for augmenting image data. Data augmentation, a technique commonly used to enhance model training when the dataset is limited, involves modifying image data through operations such as rotation, scaling, cropping, colour adjustments, and similar transformations, all of which are included in the augmenters module of the “imgaug” library.

The cv2 library represents the Python implementation of OpenCV, an open-source computer vision library renowned for providing a broad range of tools and algorithms used in developing computer vision applications.

The image of the layer structure created is given below(Figure5).

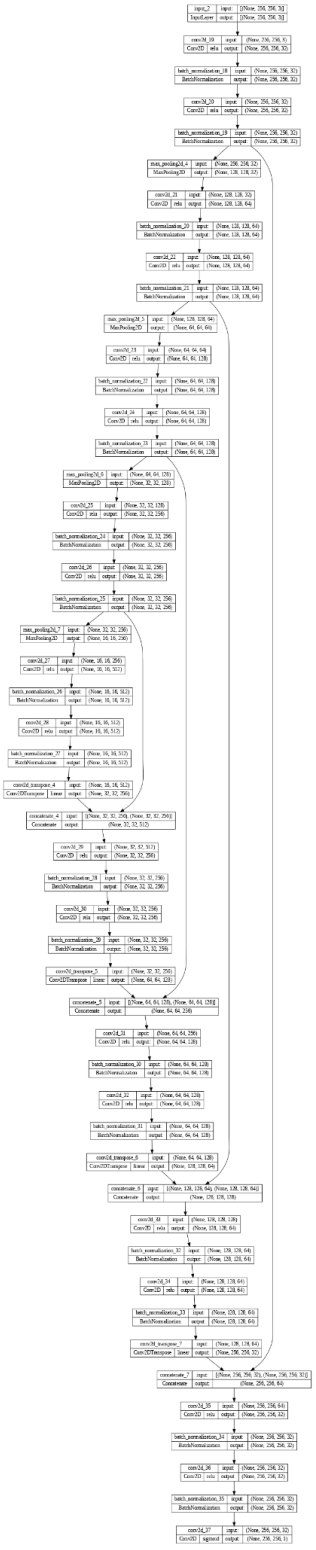

Figure 5: U-Net model architecture of our dataset

3.1. LAYERS

Convolution Layers: This layer applies several convolution kernels to the previous layer. The convolution kernels are trained to extract important features from the images such as edges, corners or other informative region representations.[11]

ReLU Layers: The ReLU is a nonlinear activation to the input. The function is f(x) = max(0, x). This nonlinearity helps the network compute nontrivial solutions on the training data.[11]

Pooling Layers: These layers compute the max or average value of a particular feature over a region in order to reduce the feature's spatial variance.[11]

Deconvolution Layers Deconvolution layers: learn kernels to upsample the previous layers. This layer is central in making the output of the network match the size of the input image after previous pooling layers have downsampled the layer size.[11]

Loss Layer: This layer is used during training to measure the error (Equation 1) between the output of the network and the ground truth. For a segmentation labelling task, the loss layer is computed by the softmax function. Weights for these layers are learned by backpropagation using a stochastic gradient descent (SGD) solver. [11]

Total params: 7771873 (29.65 MB)

Trainable params: 7765985 (29.62 MB)

Non-trainable params: 5888 (23.00 KB)

Non-trainable parameters typically refer to the parameters in a machine learning model that are fixed or not updated during the training process.

In portrait segmentation models, fixed parameters defining the region of interest (ROI), such as the face region, can be utilised. These parameters are generally employed to determine the position and size of the face and remain unchanged during training. Fundamental features like facial edges, the location of the eyes, or specific points defining facial contours can fall into this category. Such features are typically manually defined and remain constant throughout the training process.

In Python, multiprocessing is a technique enabling the execution of a program using multiple processes simultaneously. This capability significantly enhances the performance of computationally intensive tasks. For instance, a single-core processor (CPU) can handle only one task at a time, but modern computers usually possess multi-core processors. Multiprocessing divides the program to run independently on each core, allowing each core to undertake different computation tasks, resulting in improved overall performance.

In Python, the 'multiprocessing' standard library can be used for implementing multiprocessing techniques. This library offers functions for tasks such as process creation, resource sharing, synchronisation, and other significant operations.

PIL (Python Imaging Library) serves as a library within the Python language specifically designed for image processing duties. Essentially, PIL encompasses a broad spectrum of functionalities facilitating tasks like opening, editing, saving, and processing image files.

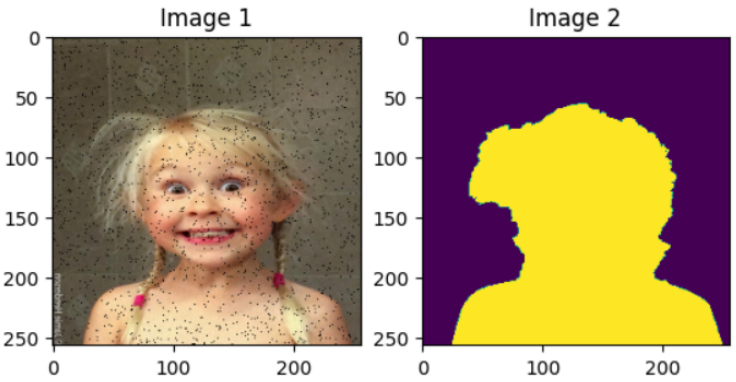

Figure 6: First augmentation segmentation

The augmentation model aids in diversifying the existing training dataset, contributing to the model's enhanced performance by enabling more generalised learning.Image segmentation involves the identification and isolation of specific objects or regions within an image. Within the realm of image segmentation, each pixel in an image is either assigned to a particular class or subjected to a specific classification scheme to determine its association with a particular region or class (Figure 6).

In the application of image segmentation, the exponential learning rate method can assist the model in achieving better results. Specifically, during the initial stages of the training process, it can help the model progress rapidly with large steps towards the minimum, while as the training progresses, smaller steps can be taken to approach a more precise position. However, it is crucial to adjust and implement exponential learning rates accurately. Otherwise, the learning rate may decrease too rapidly or not rapidly enough, diminishing the effectiveness of the training process (Figure 7).

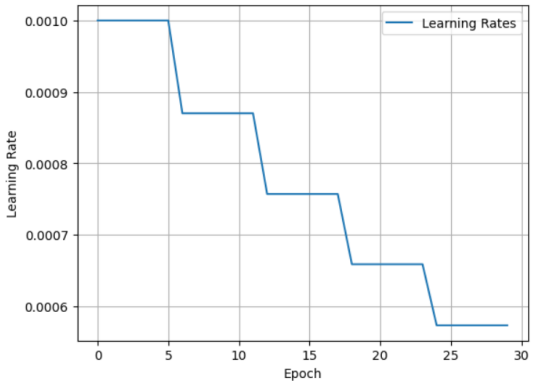

Figure 7: Exponential learning rate

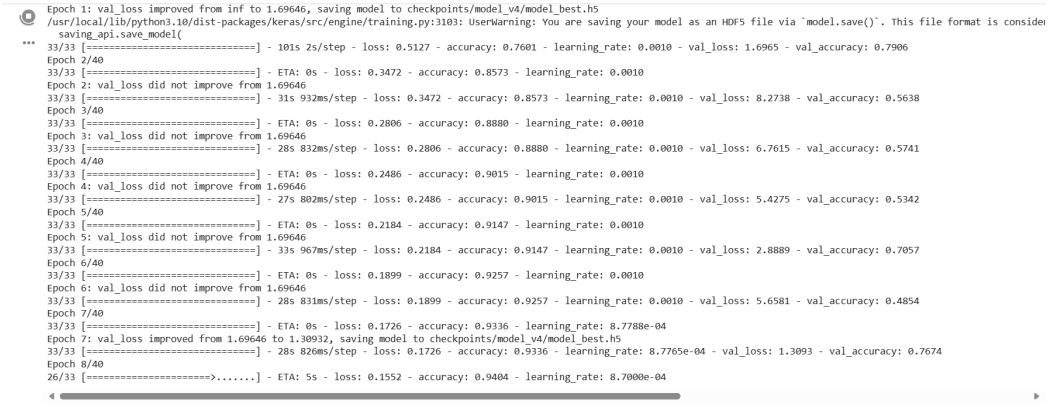

Figure 8: All maximum epochs

Image segmentation is the process of classifying and isolating specific objects or regions within an image. Deep learning models used for such tasks are typically trained over a specific number of epochs on training data. An epoch refers to one complete pass of the entire training dataset through the model. Max epoch determines how many times a particular training cycle will repeat. The maximum number of epochs used for training should be regulated to reach the model's optimal performance. However, increasing the max epoch count in some cases can elevate the risk of overfitting. Overfitting occurs when the model excessively fits the training dataset and fails to be applied to new data in a generalised manner. Setting the max epoch to an excessively high value might lead the model to overly adapt to the training data, potentially diminishing its generalisation capabilities.

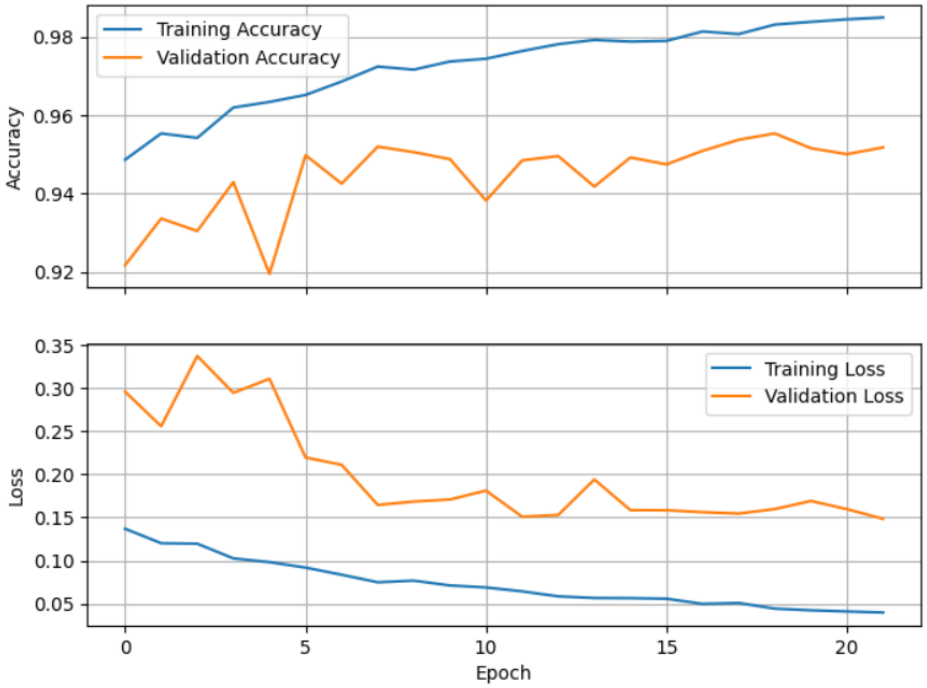

Figure 9: The metric used during training

These metrics are utilised to monitor the performance of the model during training and assess its accuracy. Continuous tracking of these metrics throughout the model's training process is essential to observe the risk of overfitting and, if necessary, make adjustments to the model.

Training Accuracy: This metric, computed at the conclusion of each epoch, is commonly employed to assess the model's performance during the training phase.

Validation Accuracy: It denotes the model's accuracy when evaluated on a designated separate validation dataset during the training process, illustrating its proficiency in making accurate predictions on unseen data.

Training Loss: Throughout the training phase, this metric quantifies the disparity between the model's predictions on the training dataset and the actual values.

Validation Loss: It evaluates the model's performance on the validation dataset, indicating its capacity to generalise and perform well on new data. A substantial reduction in this metric during training could be indicative of overfitting.

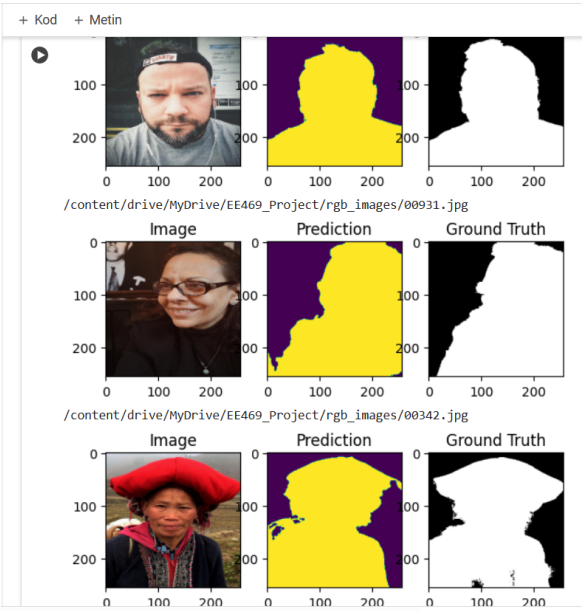

Figure 10: U-Net image segmentation results

UNET serves as a deep learning model employed in image segmentation tasks to predict the classification or region assignment of pixels within an image.

Prediction: The output produced by a model such as UNET, given an input image, determines the assigned classes or regions of corresponding pixels. Typically presented as either a probability or classification map, each pixel is linked to a likelihood of belonging to a specific class or region or is labelled with a classified class value as determined by the model.

Ground Truth (GT): Ground truth signifies the precisely labelled versions of images within the training dataset. For every pixel, the actual class or region label is derived from manually designated or accurately labelled values. These genuine labels are crucial during the model's training phase, facilitating learning and comparison with predictions.

Throughout model training, the model-generated predictions are contrasted with the ground truth to calculate loss. This loss metric is then utilised in backpropagation, enabling the adjustment of the model's weights. Model accuracy is appraised by assessing the agreement between its predictions and the ground truth labels. As a result, the model is trained by aligning with accurate labels and learning to make precise predictions.

3.2. TECHNİCAL CONSTRUCTIONS

We performed our experiments using:

- https://colab.google/
- TensorFlow (Python, C, Java, Go, JavaScript, Swift): https://www.tensorflow.org/
- Keras (Python): https://keras.io/
- Pandas: https://pandas.pydata.org/
- numpy: https://numpy.org/
- PIL: https://pypi.org/project/Pillow/
- OpenCv: https://docs.opencv.org/
- Nvidia GTX 1660ti 6GB RAM

3.3. CHALLENGES

Remarkable progress has been achieved in deep learning techniques like U-Net over the past decade. The project's feasibility necessitates algorithms operating with minimal error. However, reducing this error in deep learning techniques is mostly constrained by computational power. Training robust deep learning algorithms demands more time, making them less feasible. U-Net algorithms have employed transfer learning as a solution to alleviate this challenge [12]. Another issue lies in the limited availability of annotated data for training. Finally, deep learning models suffer from being a 'black box'; while understanding the network's input and output, the behaviour of internal hidden layers remains elusive. This leads to researchers often facing difficulty in understanding how to rectify network errors or determining the crucial layers or filters for the task. Furthermore, correctly interpreting these black boxes is challenging, and replicating their features poses complexities [13].

4. EXPERIMENTAL RESULTS

4.1. DATASET

The utilised images in the project constitute 1296 portrait images, each with dimensions of 800x600 pixels, comprising both RGB and segmented images. To expedite the Unet model's learning pace, the input images were downscaled to 256x256 pixels. The dataset contained RGB and segmented images. Following this, a threshold was applied to the model's output, resulting in binary images composed of 0s and 1s. These binary images were subsequently resized back to the original dimensions of the input images, namely 800x600 pixels, and saved.

4.2. RESULTS

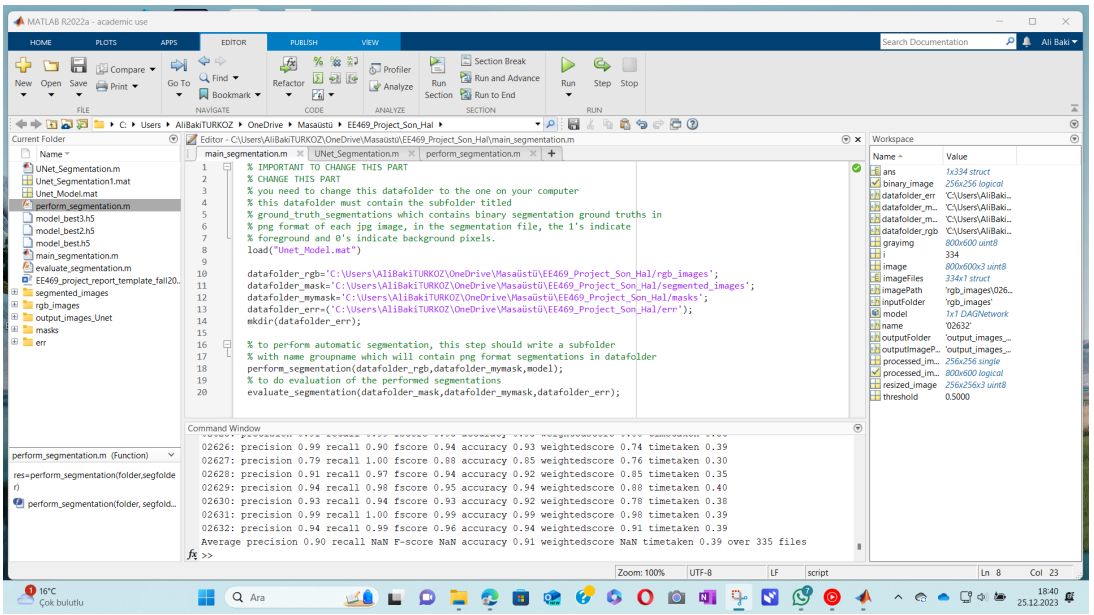

Figure 11: U-Net segmentation results and F-score with faulty image

Some of the metrics used to assess the performance of an image segmentation model like UNET are as follows (Figure 11):

Average Precision: Average Precision measures the accuracy rate of the model and classification performance on the validation set. This metric represents the average precision values for each class.

F1-Score: F1-score balances the precision and recall of the model. This metric is commonly used to measure the performance of a classifier or segmentation model. The F1-score is the harmonic mean of precision and recall, indicating the balance between these two metrics. Precision refers to the ratio of true positive predictions to the total positive predictions made by the model, while recall represents the ratio of true positive predictions to the total actual positives.

These metrics are utilised when evaluating the performance of the model, particularly in tasks like image segmentation, to assess its ability to correctly classify and isolate objects or regions.

The U-Net model used in image segmentation demonstrated successful results in metrics used to evaluate the model's performance (such as accuracy rate and F-score). These results showcased the model's ability to accurately classify images and isolate specific regions.

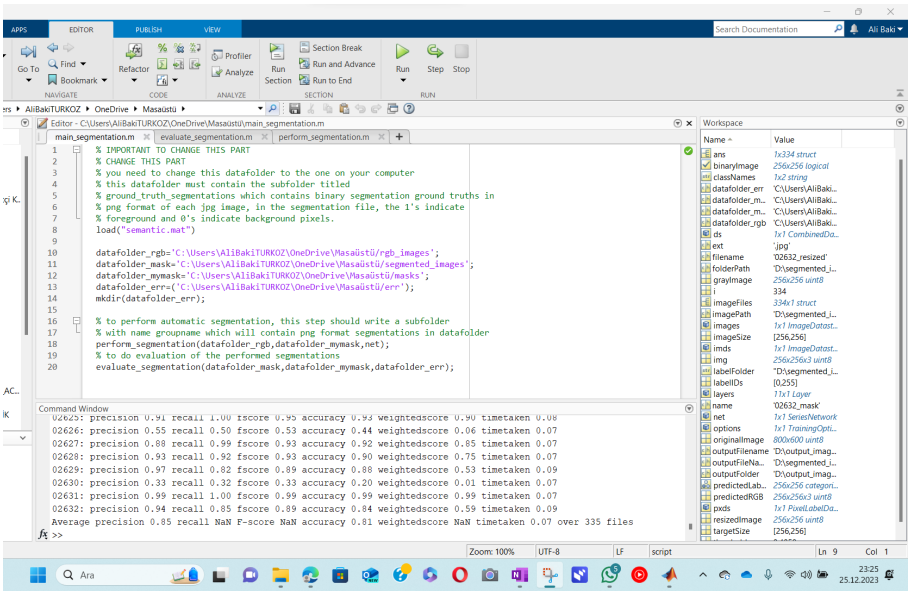

Figure 12: Semantic segmentation results

Low accuracy rate and F-score could imply that the model did not achieve the expected performance in the segmentation task (Figure 12). Semantic segmentation is a crucial area in visual processing, and studies conducted in MATLAB sometimes result in failures due to lengthy training times and various factors in the images. In some cases, factors such as clutter in the background, lighting disparities, or variations in the positions of individuals in portraits can make it challenging for the model to succeed in the segmentation task. While image segmentation requires clear boundaries and defined regions, the complexity or diversity of images in certain instances may hinder the model's accurate segmentation.

For instance, filtering techniques can be used to reduce noise in the background or colour balance methods to rectify lighting imbalances. Additionally, gathering more diverse data or diversifying the existing dataset might assist the model in better generalisation.

Segmentation tasks in images demand precise delineation, and dealing with the complexity of images or variations could pose challenges for accurate segmentation. Techniques such as filtering for noise reduction in the background or employing colour balancing methods to rectify lighting discrepancies can be used. Furthermore, acquiring a more extensive and varied dataset or diversifying the existing one may aid in the model's improved generalisation.

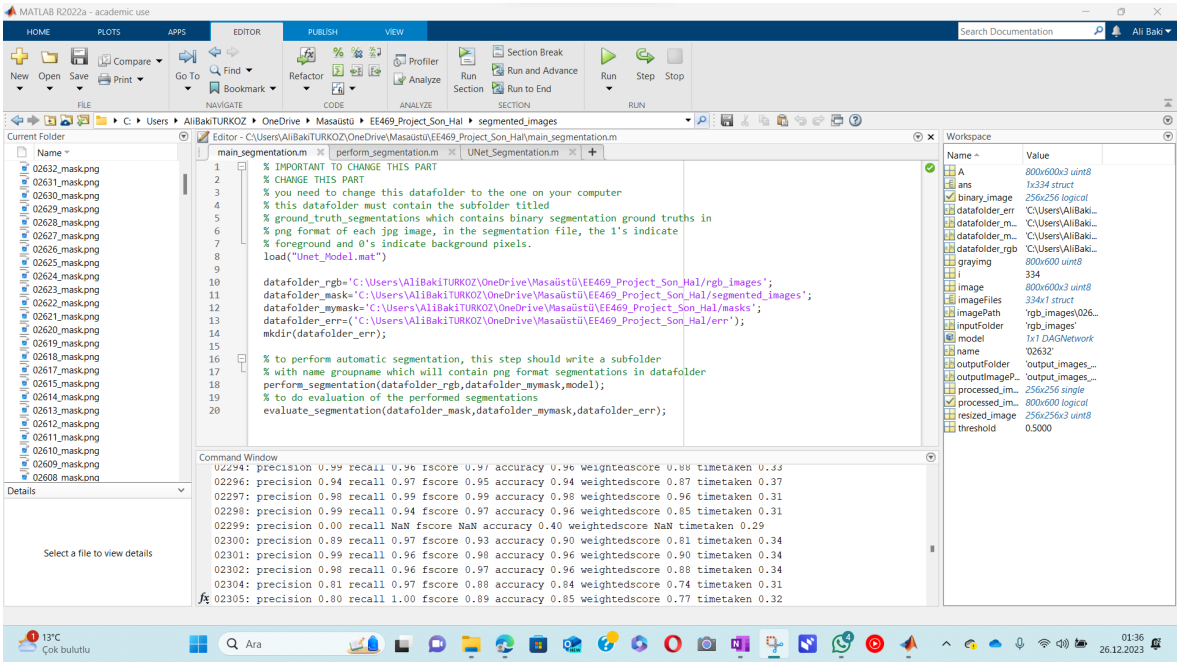

Figure 13: Unet Segmentation Result with Faulty Image 02299

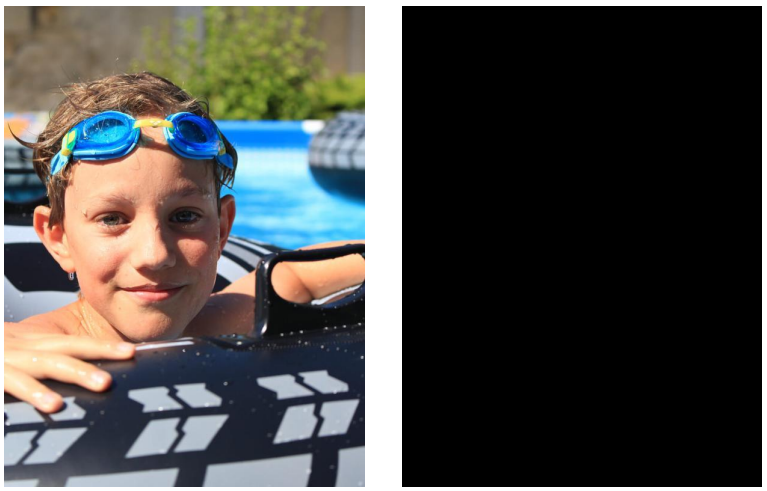

Figure 14: 02299.jpg İmage and its Faulty Segmented Version

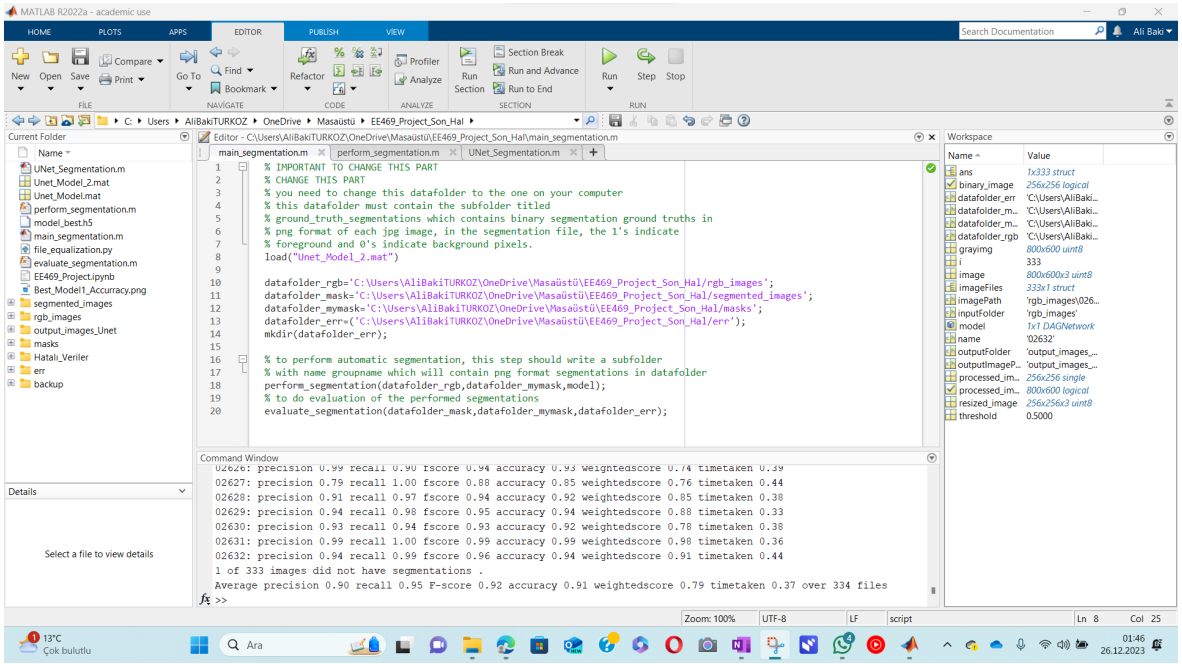

Figure 15: Unet Segmentation Result without Faulty Image

In the dataset provided for testing, the number of segmented images did not match the number of RGB images. To address this discrepancy, unmatched images between these two sets were filtered out using Python, ensuring the equalization of file sizes. The unmatched images were saved in a backup folder. During the testing phase, it was observed that F-score, Recall, and weighted score results were NaN. Subsequent investigation revealed that the issue stemmed from an error in the image 02299_mask.png (Figure 13). The error was attributed to incorrect segmentation. The image 02299_mask.png is a 800x600 image

consisting of zeros only (Figure 14). Consequently, it was identified as a flawed segmentation and removed from the dataset. After cleaning the dataset from erroneous data, the model was re-run, and the results were saved in a .mat file. Upon testing the results, an Accuracy of 0.91, F-score of 0.92, and Avg. Precision of 0.90 was obtained, as shown in Figure 15.

Models	**Accuracy**	**F-score**	**Avg. Precision**	**Recall**	**Weighted score**	**Timetaken**
Semantic segmentation	%81	NaN	%85	NaN	NaN	0.07
U-Net with faulty image	%91	NaN	%90	NaN	NaN	0.39
U-Net without faulty image	%91	%92	%90	%95	%79	0.37

5. CONCLUSION AND FEATURE WORK

In this project, a powerful deep learning model commonly used for portrait segmentation, U-Net, was implemented. To achieve this, various versions of U-Net and its applications in different image modalities were explored. Indeed, the U-Net-based architecture has been groundbreaking and valuable in medical image analysis. The growing prominence of U-Net papers since 2017 instil confidence in its status as a leading deep learning technique in medical image diagnostics. Therefore, despite the ongoing challenges in deep learning-based image analysis, we expect U-Net to remain one of the most significant pathways forward.

A novel end-to-end portrait segmentation architecture with unique cross-scale categorical attention and boundary refinement mechanisms was employed. Semantic attention imposition and increased boundary awareness in a multi-scale feature hierarchy resolved semantic inconsistencies and weak boundary delineation issues.

Architectures employing sharp segmentation boundaries, achieving working speeds through deeply separable convolutions, and using activation layers with the 'ReLU' activation function were incorporated to enhance U-Net-based performance and simplify training.

To achieve better results in image segmentation, careful attention to numerous factors, from dataset preparation to model selection and the training process, might be necessary.

5.1. ADVANTAGES AND DİSADVANTAGES

5.1.1. ADVANTAGES

- U-Net exhibits a symmetric U-shaped architecture composed of an encoder and a decoder. This architectural design enables the model to learn features within the encoder while allowing segmentation by bringing these features back to their original dimensions in the decoder.
- U-Net can deliver good results even with a small number of labelled examples. It has the potential to achieve satisfactory outcomes, particularly with limited datasets. This can be advantageous when dealing with restricted datasets.
- Moreover, U-Net adapts well to data augmentation techniques, enabling the model to generate more generalised results with less data.

5.1.2. DİSADVANTAGES

- Deep and complex models like U-Net require more memory and computational power. This disadvantage becomes more pronounced when working with large datasets and high-resolution images.
- They might face challenges in accurately segmenting small-sized objects or details. This issue can be particularly evident in cases where small details are prominent or when dealing with limited datasets.
- In scenarios with imbalanced datasets, where some classes are less represented than others, the performance might be lower. It can be challenging to accurately segment under-represented classes

Despite the many challenges in deep learning-based image analysis, U-Net is expected to be one of the most promising pathways going forward.

REFERENCES

[1] Sadeleştirilmiş U-Net mimarisi ile beyin tümörü segmentasyonu Ö. Polat (Sivas,2022)

[2] Garcia-Garcia, A.; Orts-Escolano, S.; Oprea, S.; Villena-Martinez, V.;Garcia-Rodriguez, J. A Review on Deep Learning Techniques Applied to Semantic Segmentation. arXiv 2017, arXiv:1704.06857.

[3] Miao, J.; Sun, K.; Liao, X.; Leng, L.; Chu, J. Human Segmentation Based on Compressed Deep Convolutional Neural Network.IEEE Access 2020, 8, 167585–167595. [CrossRef]

[4] Liu, X.; Deng, Z.; Yang, Y. Recent progress in semantic image segmentation. Artif. Intell. Rev. 2019, 52, 1089–1106. [CrossRef]

[5]. Chen, X.; Qi, D.; Shen, J. Boundary-Aware Network for Fast and High-Accuracy Portrait Segmentation. arXiv 2019,arXiv:1901.03814. Available online: https://arxiv.org/abs/1901.03814 (accessed on 22 October 2020).

[6] Zhang, S.H.; Dong, X.; Li, H.; Li, R.; Yang, Y.L. PortraitNet: Real-time portrait segmentation network for mobile device. Comput. Graph. 2019, 80, 104–113. [CrossRef]

[7] Convolutional Networks for Biomedical Image Segmentation O. Ronneberg, P. Fischer, T. Brox (Freiburg, 2015)

[8] J.L. Foo, A survey of user interaction and automation in medical image segmentation methods. Iowa State University Human Computer Interaction Technical Report ISU-HCI-2006-02, 2006.

[9] N. Gordillo, E. Montseny ve P.Sobrevilla, State of the art survey on MRI brain tumor segmentation. Magnetic Resonance Imaging, 31 (8), 1426-1438, 2013. https://doi.org/10.1016/j.mri.2013.05.002.

[10] O. Ronneberger, P. Fischer, ve T. Brox, U-Net: Convolutional Networks for Biomedical Image Segmentation. In: Navab, N., Hornegger, J., Wells, W., Frangi, A. (eds) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. MICCAI 2015. Lecture Notes in Computer Science, 9351. Springer, Cham.

[11] Portrait Segmentation Using Deep Learning, J. G. Berna, S. V. Datar. (The University of Texas at Arlington, 2022)

[12] M. Frid-Adar, A. Ben-Cohen, R. Amer and H. Greenspan, "Improving the segmentation of anatomical structures in chest radiographs using U-Net with an ImageNet pre-trained encoder" in Image Analysis for Moving Organ Breast and Thoracic Images, Cham, Switzerland:Springer, pp. 159-168, 2018.

[13] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik and A. Swami, "Practical black-box attacks against machine learning", *Proc. ACM Asia Conf. Comput. Commun. Secur.*, pp. 506-519, 2017

[14] Portrait Instance Segmentation for Mobile Devices, Lingyu Zhu; Tinghuai Wang; Emre Aksu; Joni-Kristian Kamarainen, (IEEE International Conference on Multimedia and Expo (ICME,2019)

https://www.mathworks.com/help/vision/ug/getting-started-with-object-detection-using-deep-learning.html

https://www.mathworks.com/help/vision/ug/get-started-with-the-image-labeler.html

https://www.mathworks.com/help/vision/ug/semantic-segmentation-using-dilated-convolutions.html

https://dergipark.org.tr/en/download/article-file/833620

https://dergipark.org.tr/tr/download/article-file/394923

https://dergipark.org.tr/tr/download/article-file/380999

https://arxiv.org/ftp/arxiv/papers/2202/2202.02705.pdf

https://dergipark.org.tr/en/download/article-file/1109740

https://ieeexplore.ieee.org/document/9446143/metrics#metrics

https://colab.google/

https://www.tensorflow.org/

https://keras.io/

https://pandas.pydata.org/

https://numpy.org/

https://pypi.org/project/Pillow/

https://docs.opencv.org/