


Article

Semantic Segmentation of Extraocular Muscles on Computed Tomography Images Using Convolutional Neural Networks

Ramkumar Rajabathar Babu Jai Shanker¹, Michael H. Zhang¹ and Daniel T. Ginat^{2,*} 

¹ Department of Radiology, University of Chicago, Chicago, IL 60615, USA; rbramkumar@gmail.com (R.R.B.J.S.); michael.zhang@uchospitals.edu (M.H.Z.)

² Department of Radiology, Section of Neuroradiology, University of Chicago, Chicago, IL 60615, USA

* Correspondence: ginatd01@gmail.com; Tel.: +1-(773)-702-6039

Abstract: Computed tomography (CT) imaging of the orbit with measurement of extraocular muscle size can be useful for diagnosing and monitoring conditions that affect extraocular muscles. However, the manual measurement of extraocular muscle size can be time-consuming and tedious. The purpose of this study is to evaluate the effectiveness of deep learning algorithms in segmenting extraocular muscles and measuring muscle sizes from CT images. Consecutive CT scans of orbits from 210 patients between 1 January 2010 and 31 December 2019 were used. Extraocular muscles were manually annotated in the studies, which were then used to train the deep learning algorithms. The proposed U-net algorithm can segment extraocular muscles on coronal slices of 32 test samples with an average dice score of 0.92. The thickness and area measurements from predicted segmentations had a mean absolute error (MAE) of 0.35 mm and 3.87 mm², respectively, with a corresponding mean absolute percentage error (MAPE) of 7 and 9%, respectively. On qualitative analysis of 32 test samples, 30 predicted segmentations from the U-net algorithm were accepted while 2 were rejected. Based on the results from quantitative and qualitative evaluation, this study demonstrates that CNN-based deep learning algorithms are effective at segmenting extraocular muscles and measuring muscles sizes.

Keywords: CT; semantic segmentation; extraocular muscles; deep learning; convolutional neural networks; dice coefficient



Citation: Shanker, R.R.B.J.; Zhang, M.H.; Ginat, D.T. Semantic Segmentation of Extraocular Muscles on Computed Tomography Images Using Convolutional Neural Networks. *Diagnostics* **2022**, *12*, 1553. <https://doi.org/10.3390/diagnostics12071553>

Academic Editor: Henk A. Marquering

Received: 16 May 2022

Accepted: 24 June 2022

Published: 26 June 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Computed tomography (CT) imaging of the orbit with measurement of extraocular muscle size can be useful for diagnosing and monitoring thyroid eye disease and other conditions that may affect the extraocular muscles. To assess the size of the extraocular muscles, parameters such as the muscle diameter, cross-sectional area, and muscle volume have been measured on CT and MRI images [1]. However, the manual measurement of extraocular muscle size is time-consuming and tedious. Automated techniques for segmenting extraocular muscles and estimating muscle sizes can provide a reliable and accurate method for clinical use.

Several techniques to carry out the automated segmentation of extraocular muscles have been developed. Some of the earlier works either relied on operators for manual inputs [2] or required the scan template to be aligned in a rigid manner [3,4]. Xing et al. [5] proposed carrying out segmentation using super pixels, which are groups of pixels with coherent intensities and spatial locations. This approach relies on specific spatial connections and prior knowledge which may not be generalizable. Furthermore, all the above methods [2–5] were developed and evaluated on magnetic resonance (MR) images. Thus, it is of interest to develop fully automated and generalizable segmentation techniques as an aid to radiologic diagnosis from computed tomography (CT) images.

Machine learning can be used as an aid for detecting abnormalities in imaging, but there is limited medical literature regarding the use of deep learning to achieve clinically

applicable segmentation of extraocular muscles in humans. In recent years, deep learning, the subfield of machine learning that uses multilayered neural networks, has shown promising results in many cognitive tasks including the semantic segmentation of medical imaging datasets. The deep learning algorithm proposed by Ronneberger et al. [6] has been widely adopted and improved the effectiveness of convolutional neural networks (CNNs) for semantic segmentation tasks in medical imaging. Milletari et al. [7] proposed the V-net, which is based on a volumetric convolutional neural network. The V-net performs segmentation on three-dimensional image volumes instead of two-dimensional image slices and thereby benefits from utilizing information across slices. Further convolutional neural network architectures have been proposed for the segmentation of other organs and tissues such as lungs, brain regions, and tumors [8–11].

For extraocular muscle segmentation, Zhu et al. [12] proposed a three-dimensional volumetric convolutional neural network that is based on V-net architecture. This proposed CNN model inputs a volume of thirty-two adjacent slices with cropped region-of-interest, which is localized to the area of either the left or right orbit, of size 256 by 256 pixels. However, this was developed and evaluated only on orbital images acquired without contrast enhancement. Furthermore, depending on the window settings used, the boundaries of extraocular muscles can be subjective and vary between studies. Hanai et al. [13] proposed multiple CNN models where the first CNN segments the globe from the coronal CT image and the second CNN performs the segmentation and trimming of the orbital area.

Thicknesses, cross-sectional areas, and volumes of extraocular muscles can be useful in assessing their size for enlargement and monitoring size differences from progression or response to therapy. These size parameters can vary based on the settings and methods used including window settings and the plane of measurement [14,15]. Since the superior rectus and superior levator palpebrae muscles could not be reliably distinguished from each other, they were measured together as a single muscle group, namely the superior muscle group. To measure the thickness, the horizontal diameters of the lateral and medial rectus, and vertical diameters of the superior group and inferior rectus muscle are used. While the vertical diameters of the superior muscle group and inferior rectus muscle were measured on the coronal plane, the horizontal diameters of the medial and lateral rectus muscles were measured on either the coronal or axial plane. Since the cross-sections of the medial and lateral rectus muscles can be at an angle to the coronal plane, the horizontal diameters as measured on the axial plane may be different from those measured on the coronal plane. On the other hand, cross-sectional areas and volumes are computed directly from the outlined segmentations [1,16]. To compute the cross-sectional areas of the extraocular muscles, the outlined muscle boundaries on the coronal slice and the enclosed pixel sizes are used. Similarly, the muscle volume is computed by adding the previously identified cross-sectional areas and multiplying them with the slice thickness.

Despite the promise deep learning offers, challenges remain in developing clinically applicable algorithms to segment and measure the size of extraocular muscles on CT in an automated and generalizable manner. In particular, the CNN algorithms should be able to carry out the segmentation and muscle size measurements on the overall CT of the orbit including slices and sites that may or may not contain extraocular muscles. Since there can normally be slight asymmetries between the left and right side, radiologists evaluate and record measurements of extraocular muscles separately for left and right orbital areas within their reports and findings. The automated algorithms therefore need to provide segmentations and size results specific to left and right sides, respectively. In the clinical setting, these specific measurements can potentially help radiologists while drafting impressions and findings within their reports.

Here we address these challenges with a deep learning approach that can perform a fully automated segmentation and size measurement of extraocular muscles on CT images of the orbit without any manual inputs from a radiologist. We achieve this by (i) training a convolutional neural network to segment extraocular muscles from orbit CTs, (ii) algorithmically calculating the two-dimensional parameters for muscle size such as thickness

and cross-sectional area, and (iii) providing the segmentations and size measurements for left and right side separately. We evaluate the effectiveness of predicted segmentation and measurements by comparing them against their ground truths using both quantitative and qualitative evaluations.

2. Materials and Methods

2.1. Convolutional Neural Network (CNN)

Convolutional neural networks are a class of neural networks that perform well with data that has a grid-like topology, which in our case is a multidimensional pixel array of CT intensity values [17]. It is composed of multiple sequentially applied convolution operations with each operation expressed as,

$$s = (x * w), \quad (1)$$

where, in CNN terminology, x is referred to as the input, w as the kernel, and the output s as the feature map. For a 2-dimensional input image I and 2-dimensional kernel K (with dimensions m and n), the convolution operation resulting in output matrix Z , implemented mathematically using a cross-correlation operation, can be formulated as,

$$Z(i, j) = (K * I)(i, j) = \sum_m \sum_n I(i + m, j + n)K(m, n) \quad (2)$$

where i and j represent the element in the i th row and j th column of the matrix.

A typical layer in a convolutional network as shown in Figure 1 comprises three operations: the convolution operation, the activation function followed by the pooling function. The convolution operation applies many kernels (K_i) so that many different feature maps are extracted at each layer. The activation function inputs the feature map (Z) from the convolution operation and outputs the non-linear activation (A) thereby introducing non-linearity in the layer. Most recent CNNs use the rectified linear unit (ReLU) as an activation function, which is an element-wise operation on the feature map Z , expressed as $a = \max(0, s)$. This is followed by a pooling function which replaces the output of the network at a certain location with a summary statistic of the nearby outputs, which helps make the pooling output (P) approximately invariant to minor translations or scale of the input.

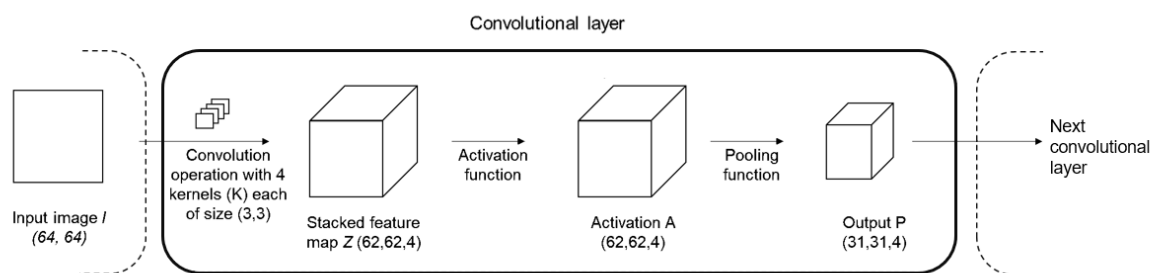


Figure 1. Typical convolutional layer comprising of convolution, activation, and pooling operations.

Convolutional layers progressively extract higher dimensional image representations (P^l —output P at layer l). With enough layers and training, a deep convolutional network can yield robust features that help perform the cognitive task. The aim of training is to arrive at the optimized set of values, referred to as parameters, within the kernels K^l (kernel K at layer l). These parameters can successfully transform the original CT slice/volume (I), with values in Hounsfield units (HUs) into the regions-of-interest, i.e., segmentation masks for extraocular muscles. Arriving at this set of optimized network parameters is an optimization task that is carried out using gradient-based algorithms such as gradient descent, which iteratively updates the network parameters to arrive at the final network weights.

The semantic segmentation of images involves assigning a class label to each pixel in the image [18]. In the context of medical images, it can be used to segment anatomical tissues which can later be analyzed for diagnosis purposes. Prior to deep learning, semantic segmentation was carried out using pixel-wise classifiers such as random forests [19], where the prediction for a specific pixel was made using the pixel intensities around that pixel. Several convolutional neural network architectures have been shown to be useful in medical image segmentation in recent years. While there have been many individual architectures proposed, the existing CNN based medical image segmentation architectures can be classified into three categories: fully convolutional neural networks, U-net, and generative adversarial networks [20].

Fully convolutional networks (FCNs), proposed by Long et al. [21], was one of the first deep learning works for semantic segmentation that used only convolutional layers. This CNN architecture takes an image of any size and applies a series of successive convolutional operations and produces the output segmentation map with the same size as the input image. While a typical convolution operation would result in an output feature map that is of lower size than input image, the final two layers use a deconvolution layer that up-samples the feature map from the previous layer and outputs the resulting image with same size as the original image. This architecture also uses skip connections where the output from initial layers of the model is combined with the inputs to the final prediction layer to provide higher-level semantic information, which results in better predictions that respect global structure. However, the results from the up-sampling layers in FCN were still relatively fuzzy and insensitive to the details. These shortcomings were addressed in the subsequent CNN architectures such as the DeepLab v1, DeepLab v2, DeepLab v3, and DeepLab v3+ [22–25] which resulted in better segmentation boundaries and at multiple scales and made use of conditional random fields (CRF) [26]. SegNet [27] used the encoder-decoder architecture with the up-sampling operation performed by a trainable convolution layer. FCN networks have been used to segment multiple organs and tissues such as brain tumors [28–30], eye [31,32], chest [33], liver [34], and left and right ventricles of the heart [35].

The U-net architecture, proposed by Ronneberger et al. [6], is based on the encoder-decoder architecture where the encoder module (contracting path) captures context, and the expanding decoder module (expanding path) enables precise localization. The 3D U-net, proposed by Çiçek et al. [36], realizes 3D image segmentation by inputting a continuous sequence of 2D images. The V-net [7] can perform segmentation on 3D volumes by using 3D convolution kernels in place of 2D convolution kernels. Further works improved on the U-net by adding an attention mechanism that helped the network localize better [37]. U-net and its variants have been used to segment multiple organs and tissues including retinal vessels [38], chest [39], and heart [40].

Generative adversarial networks (GANs) are a class of neural networks in which two networks, the generator module and discriminator module, compete against each other. While the generator network uses random noise to generate an image, the discriminator network judges whether the image is “real” or not. As iterations progress, the generator network gets better at generating images that look more real and the discriminator network becomes better at judging the generated images. In the work proposed by Luc et al. [41], the generator network generated segmentation maps, and the discriminator network judged whether the segmentation maps were coming from the ground truth or the generator. GANs have been shown to successfully segment the brain [42], retinal vessels [43], and spines [44] from medical images.

Due to its excellent performance, the U-net and its variants have been widely used in various fields of computer vision. The U-net was chosen for this implementation because of its ability to capture global context and precise localization. As shown in Figure 2, we use the U-net architecture with convolutional layers as displayed in the legend (bottom-right). The network is comprised mainly of down-sampling layers, i.e., convolutional layers that reduce the feature map size, and up-sampling layers, i.e., convolutional layers that increase

the feature map size, and skip connections between them to provide a direct flow of feature maps from an early layer in the network to a later one. Skip connections are realized by either concatenating the feature maps of the early layer to those of the later one or by applying an element-wise summation.

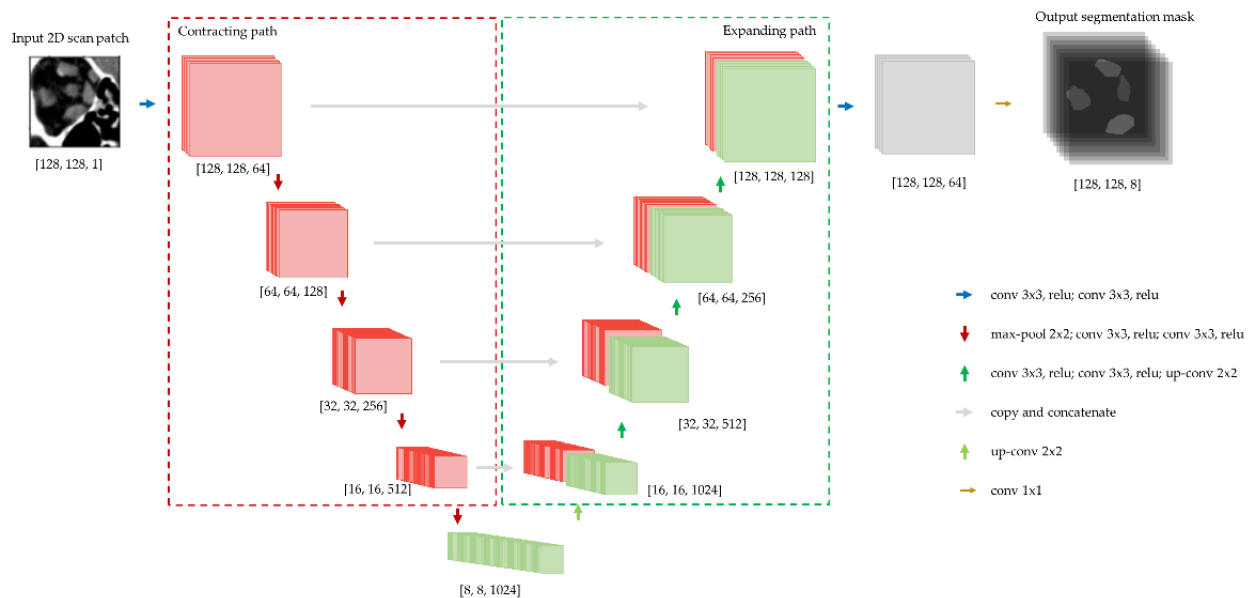


Figure 2. Schematic of U-net network architecture [7] with image/feature map size on bottom of the box (adapted to input patch size of 128×128) and convolutional layers depicted using arrows.

2.2. Dataset

For this retrospective study, we analyzed coronal CT images of the orbit acquired from 215 patients between 1 January 2010 and 31 December 2019. After excluding patients with facial trauma and/or image artifacts, the final 210 patients were randomly split into training and test sets with 178 and 32 patients, respectively. The training set was used to develop the model and the test set was used to evaluate the performance of the model on scans previously unseen to the model. The model's performance on a test set is important since it gives an estimate of its ability to generalize predictions to unseen scans.

Each scan was annotated using 3D slicer (version 4.11) to create masks (ground truths) of extraocular muscles from the DICOM files [45]. A multiclass mask was created with muscles classes (L-medial rectus, L-lateral rectus, L-superior group (including L-superior rectus and L-superior levator palpebrae), L-inferior rectus, R-medial rectus, R-lateral rectus, R-superior group (including R-superior rectus and R-superior levator palpebrae), R-inferior rectus) and background class.

The ground truths for extraocular muscle segmentations were analyzed by a board-certified radiologist with a certificate of added qualification in neuroradiology. These ground truths were used to train the CNNs in a supervised manner. Figure 3 shows an example of (L-R) coronal plane, axial plane, and sagittal plane with the original DICOM image (left) and EOM masks highlighted.

The comparison (Table 1) between the training and test set on baseline patient characteristics (age, sex, and ground truth EOM thickness measurement) shows that there are no significant differences between the training set versus the test set.

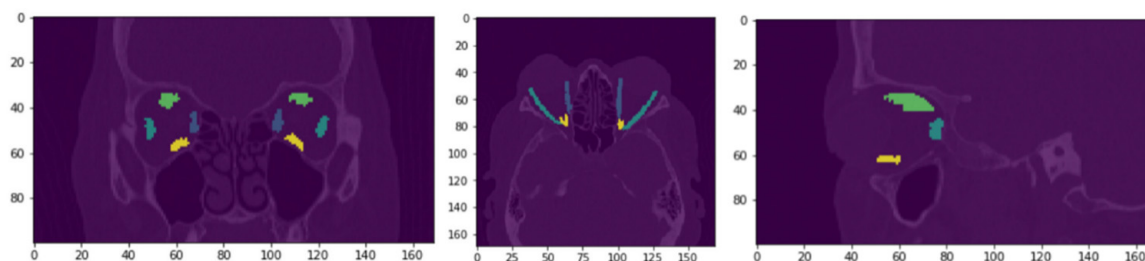


Figure 3. Label-wise annotation of EOMs with EOM masks highlighted on (L-R) coronal, axial, and sagittal CT images.

Table 1. Baseline patient characteristics of the training and test groups with mean (and standard deviation) thickness and area in mm and mm² respectively.

	Train	Test	<i>p</i> -Value
N	178	32	
Sex = M (%)	53 (30%)	9 (28%)	1
Age	46.67 (17.49)	50.97 (19.7)	0.21
Thickness—L-Medial Rectus	4.87 (0.84)	4.85 (0.7)	0.9
Thickness—L-Lateral Rectus	5.5 (1.13)	5.38 (1.18)	0.58
Thickness—L-Superior group	4.79 (0.93)	4.9 (0.77)	0.53
Thickness—L-Inferior Rectus	5.23 (1.07)	5.19 (1.02)	0.84
Thickness—R-Medial Rectus	4.74 (0.66)	4.66 (0.85)	0.55
Thickness—R-Lateral Rectus	5.62 (1.41)	5.87 (1.33)	0.35
Thickness—R-Superior group	4.85 (1.04)	4.99 (0.91)	0.48
Thickness—R-Inferior Rectus	5.13 (1.08)	4.97 (0.98)	0.44
Area—L-Medial Rectus	38.93 (8.54)	39.16 (6.55)	0.89
Area—L-Lateral Rectus	46.03 (10.03)	46.17 (12.05)	0.89
Area—L-Superior group	38.29 (9.66)	40.28 (8.04)	0.27
Area—L-Inferior Rectus	41.57 (11.78)	41.17 (9.35)	0.86
Area—R-Medial Rectus	38.14 (6.88)	38.56 (7.01)	0.75
Area—R-Lateral Rectus	47.2 (14.29)	49.81 (12.33)	0.33
Area—R-Superior group	40.1 (13.79)	41.39 (9.62)	0.61
Area—R-Inferior Rectus	42.38 (14.24)	41.14 (10.8)	0.64

2.3. Image Acquisition

The orbit scans were performed using 65 mL of Omnipaque 350 (injection rate of 1.2 mL/s with a delay of 55 s). Image acquisition was performed with field of view of 200 mm, collimation of 64 by 0.625 mm, source slice thickness of 0.9 mm with an increment of 0.45 mm, 120 kV, 200 mAs, and 3 mm soft-tissue reconstructions in axial, coronal, and sagittal planes.

2.4. Data Preprocessing

Since different studies may have varying different pixel spacing values, they are first isometrically resampled to pixels of size 1 × 1 mm with the aid of the PixelSpacing DICOM attribute and spline interpolation of order three. To facilitate a standard input size for the network, we used a patch-based input as used in the original implementation of U-net. For example, using this method, multiple patches of size (64, 64 pixels) are drawn from random areas of an original CT slice of size (512, 512 pixels). This method also acts as a data augmentation process and can alleviate the challenges of using a small dataset. Since the patches can be drawn from any area of the overall image, the network learns to be translation invariant and can effectively segment extraocular muscles in any localized area of an input slice. Further data augmentation methods, which improve the robustness of the model such as rotation, size scaling and noise addition are applied. Flipping images horizontally from left right was not performed in this work. This was to enable the network to learn to differentiate extraocular muscles on the left orbit from

those on the right. Noise addition to CT images that can simulate low-dose acquisition settings require access to the raw scanning data [46,47]. Since the raw sinogram data from the scanners was not available at the time of training, Gaussian noise (of mean = 0 and standard deviation = 10 Hounsfield Units) was added to the CT slice intensities. The scans were then windowed to highlight extraocular muscles using level and width settings of 50 and 250 Hounsfield Units, respectively [14]. The scan inputs were then provided as a stack of 2-dimensional patches, which were then fed into the network. Half of the patches in this stack were drawn from the orbit area, and the other half was drawn from non-orbital areas.

2.5. Architecture

We employed the 2-dimensional implementation of U-net as our CNN architecture. Since the coronal plane is the only plane in which all rectus muscles can be visualized on a 2-dimensional slice, we trained a 2-dimensional U-net network that could predict all four rectus muscles (left and right) using only coronal slices.

2.6. Loss Functions

Loss functions, which measure the dissimilarity between actual and predicted segmentations, are important because they guide the network to learn meaningful predictions. They also govern how the network should learn from mistakes (false positives, false negatives, segmentation boundaries vs. volume, hard vs. easy scenarios). Loss functions can be formulated to measure the mismatch in distribution, region, boundary, or a combination of these [48]. Distribution-based loss functions such as weighted cross entropy (WCE) train the network to minimize dissimilarity between the predicted and ground truth distributions. Region-based loss functions such as the Dice similarity coefficient loss [49], Jaccard (Intersection over Union) loss [50], and Focal Tversky loss [51] aim to minimize the mismatch or maximize the overlap regions between the predicted and ground truth segmentations. Boundary-based losses such as the boundary loss [52], surface Dice similarity coefficient [10] aim to minimize the difference between the contours of predicted and ground truth segmentations.

Weighted cross entropy (WCE) loss is defined as the measure of difference between two distributions and is given mathematically defined as

$$WCE\ loss = -\frac{1}{N} \sum_{c=1}^C \sum_{i=1}^N w_c g_i^c \log s_i^c, \quad (3)$$

where g_i^c is the ground truth value, s_i^c is the corresponding predicted segmentation probability and w_c is the weight for each class c , and N is the total number of pixels.

Dice similarity coefficient (DSC) loss, also known as the overlap index, is used to compare the similarity between predicted and ground truth segmentations. For binary labels of ground truth $g_{ic} \in \{0,1\}$ and predicted probability of class label $p_{ic} \in [0,1]$ with total number of pixels N , the DSC loss is expressed as

$$DSC\ loss = 1 - \frac{\sum_{i=1}^N p_{ic} g_{ic} + \epsilon}{\sum_{i=1}^N p_{ic} + g_{ic} + \epsilon}, \quad (4)$$

where ϵ is a smoothing parameter with a value close to zero, which provides numerical stability to prevent division by zero. Lower values of DSC indicate better overlap between the predicted and ground truth segmentations.

Jaccard (Intersection over Union) loss measures the extent of overlap and penalizes regions that do not overlap with the ground truth. For binary labels of ground truth

$g_{ic} \in \{0,1\}$ and predicted probability of class label $p_{ic} \in [0,1]$ with total number of pixels N , the *IOU loss* is expressed as

$$IOU\ loss = 1 - \frac{\sum_{c=1}^C \sum_{i=1}^N g_i^c p_i^c}{\sum_{c=1}^C \sum_{i=1}^N g_i^c + p_i^c - g_i^c p_i^c}, \quad (5)$$

Focal Tversky loss is an extension of the dice loss and addresses some of the issues with Dice loss where the regions-of-interest are small. The Tversky index, in which generalization of Dice loss that allows for flexibility in balancing false-positives and false-negatives., is combined with the γ parameter, which controls for easy background and hard ROI training examples. The Focal Tversky loss is given by,

$$FTL_c = \sum_c (1 - TI_c)^{\frac{1}{\gamma}}, \quad (6)$$

where TI_c is the Tversky Index, and parameter γ varies in the range [1,3].

Boundary loss measures the distances between two boundaries. The integral framework, which is differentiable and can be used as a loss function, is formulated as

$$L_{BD} = \sum_{\Omega} \phi_G(p) s_{\theta}(p), \quad (7)$$

where ϕ_G is the level set representation of the ground truth boundary and $s_{\theta}(p)$ is the SoftMax probability outputs from the trained network.

Compound loss functions use a combination of the above losses that are tailored for a specific application. For example, if the objective of the segmentation is to arrive at only volume or area measurements, region-based loss functions would be well suited to train the CNN. However, if the clinical objective is to identify organ contours for radiation therapy or a thickness measurement which depends on the contours identified, distance-based loss functions would be well suited. Boundary-based losses would need to be used jointly with a region-based loss to deliver improved segmentation results. Compound losses have also been shown to be more robust loss functions [48].

It is worth noting that, in our application with extraocular muscles, we measure not only the area and volume of the predicted segmentation but also its thickness, which is measured from muscle boundary contours. For this reason, we train the U-net using not only individual loss functions but also compound loss functions and pick the best performing model.

2.7. Training & Experiment Design

To train our model, we used an Adam optimizer [53]. Adam is an optimization algorithm that is used to arrive at the final network weights from a set of randomly initialized weights by updating the network weights in an iterative manner. The networks were allowed to train for a maximum of one hundred epochs on a Nvidia Tesla K80 GPU (maximum 8 h).

Hyperparameters are model parameters that are set to control the learning process. While some hyperparameters were chosen based on default values, other values were chosen after preliminary performance evaluation.

Learning rate: Learning rates control how much the model weights are updated at the end of each iteration. A large learning rate helps the algorithm learn fast, but it may also make the training process unstable. On the other hand, a small learning rate will require many updates before reaching the optimal solution and may make the training process take too long to converge. Since Adam is an adaptive optimization algorithm, the learning rates are computed individually for different parameters from estimates of first (mean) and second (uncentered variance) moments of the gradients. After a preliminary evaluation of the default settings, we set the value for learning rate of 0.0001, exponential decay rate for the 1st moment (mean) estimates as 0.9, and exponential decay rate for the 2nd moment (uncentered variance) estimates as 0.999.

Image patch size: Patch size is the size of each image input to the CNN during the training process. While larger patches contain more information, they also take more memory resources during training. The U-net architecture applies a series of four pooling operations, each of which reduce the image size by a factor of two. Therefore, the minimum input image size would be 16. However, image patches of size 16×16 pixels would not contain enough spatial and contextual information for the network to train on. On the other hand, the maximum patch size is constrained by the maximum orbital scan size in the coronal plane. In this work, a patch size of 128×128 pixels was used to facilitate the patches generated to have at least one orbit fully with adjoining areas.

Batch size: Batch size is the number of patches that are input to the CNN at each step of the training iteration. Using smaller batch sizes makes the network more resilient to noise but also increases the training time significantly. To make efficient use of the GPU memory, we used a batch size of 20 patches drawn from 4 training images for each step within the training iteration.

Loss functions: In this work, we evaluated the performance for individual loss functions and compound loss functions below: (i) weighted cross entropy, (ii) Dice similarity coefficient loss, (iii) weighted cross entropy + Dice similarity coefficient loss, (iv) Focal Tversky loss, (v) Dice similarity coefficient loss + boundary loss.

Dropout: Dropout [54] is a technique that approximates training many networks with different architectures in parallel. This is implemented during training by randomly setting a portion of the network to zero, thus having the effect of making the architecture be treated as a layer with different numbers of nodes. This also helps prevent overfitting by making the training process noisier and breaking-up co-adaptation situations where hidden units may change in a way that they fix up the mistakes of other units. Dropout has a tunable hyperparameter p , which is the probability of retaining a unit in the network. This hyperparameter controls the intensity of dropout, where higher values of p ($p \approx 1$) correspond to lower dropout and lower values ($p \approx 0$) correspond to more dropout. In this work, we set the dropout hyperparameter p as 0.8.

Weight initialization: Initializers are used to define the way to set the initial random weights of CNN layers. Initialization can have a significant impact on the training process, convergence, and final performance. While a network initialized with high initial weights may lead to exploding gradients, a network initialized with too low initial weights may lead to vanishing gradients. In this work, we used Glorot normal initialization, also known as Xavier initialization [55], where the weights are initialized by drawing samples from a truncated normal distribution centered on zero and standard deviation, which is computed dynamically as $\text{sqrt}(2/(fan_in + fan_out))$, where fan_in is the number of input units in the weight tensor and fan_out is the number of output units in the weight tensor.

Cross-validation is a resampling method that uses different sections of the data to train and validate the model. As illustrated in Figure 4, we performed k -fold cross validation, where the original training sample of 178 studies is partitioned into k equal sized subsamples. Of the k subsamples, a single subsample is chosen as validation set for evaluating the model performance, and the remaining $k-1$ subsamples are used as training set. This process is repeated k times, with each subsample used exactly once as the validation set. In our experiments, we set k as 10. The validation results were then aggregated to produce a single estimation. Using this estimate, we could compare the performance across parameters, and select the best performing model. After this step, the best parameters were then used to train the model on the training data and then the final evaluation was carried out on test data.

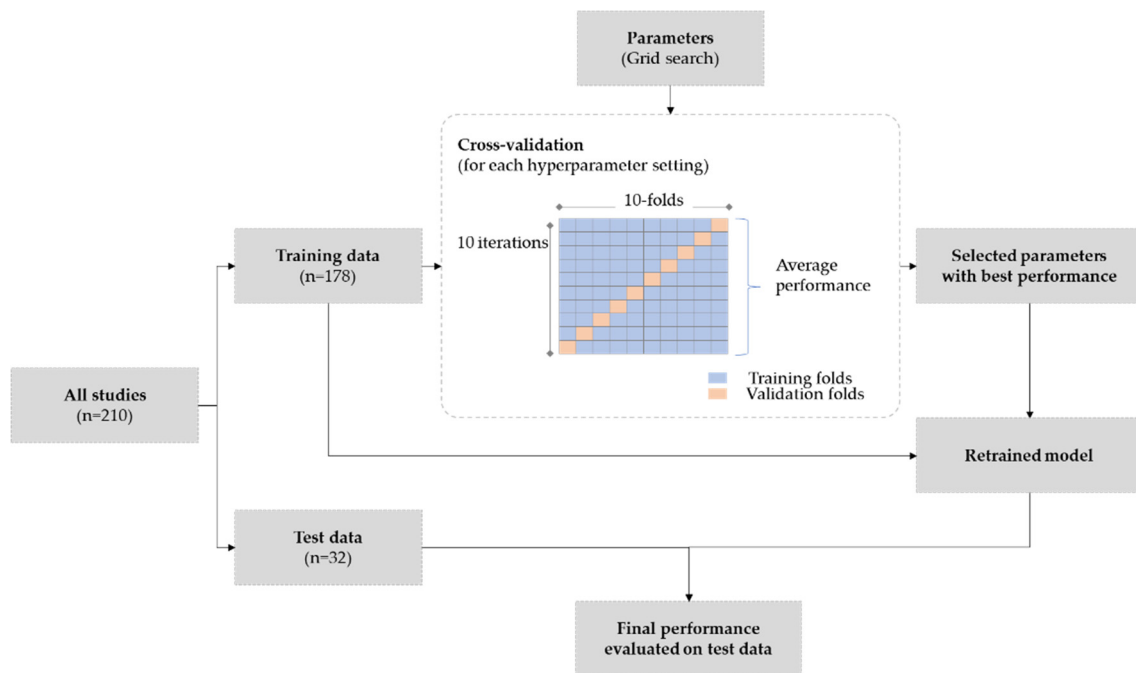


Figure 4. Experimental setup for model training, cross-validation, and final evaluation.

2.8. Muscle Size Measurement

For a given coronal slice i , the thickness of the extraocular muscles is measured by fitting a rotated rectangle of minimum area that completely encloses the segmented muscle contour. Since the cross-sections of extraocular muscles (in the mid-orbital region) on the coronal plane are nearly ellipsoids, the length and width of the bounding rectangle are the long axis diameter and short axis diameter (or thickness), respectively, of the muscle cross-section.

As illustrated in Figure 5, the thickness (t) for each muscle m on coronal slice i , is calculated as

$$t_m^i = \text{width}(R_m^i), \tag{8}$$

where R_m^i is the bounding rectangle with minimum area of muscle m on slice i . The maximum thickness across all coronal slices is taken as the overall thickness of the muscle.

Similarly, the cross-sectional area (A) of muscle m on coronal slice i is given by

$$A_m^i = N_m^i, \tag{9}$$

where N_m^i is the number of pixels within the segmentation of muscle m on coronal slice i . The maximum cross-sectional area across all coronal slices is taken as the cross-sectional area of the entire muscle.

2.9. Evaluation

To perform quantitative evaluation, the Dice coefficient and intersection-over-union (IOU) metrics were used. The extraocular muscles originate from the common tendinous ring located at the apex of the orbit and insert onto the sides of the eyeball. The cross-sections of the extraocular muscles become small as they crowd together towards their ligamentous origin. Therefore, it is especially challenging for automated algorithms to reliably segment the rectus muscles from each other near the apex of the orbit. Similarly, at the anterior aspect of the orbit, it is hard to distinguish extraocular muscles from the tendons inserting into the globe. For this reason, we split the coronal slices of the extraocular muscles into three sections—(i) near insertion, (ii) central part, and (iii) near tendinous origin. As shown in Figure 6, this was achieved by splitting the coronal slices in each study

into three equal parts to form the three regions. We present an evaluation of the model performance specific to each region.

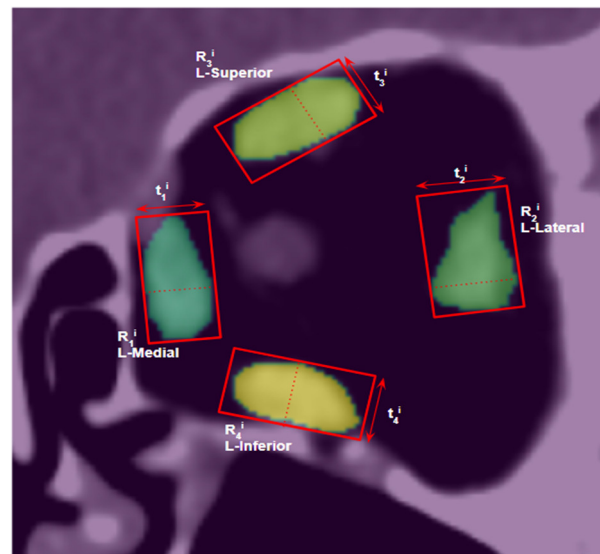


Figure 5. Illustration of thickness measurements ($t_1^i, t_2^i, t_3^i, t_4^i$) on a coronal slice i , using rotated rectangles of the minimum area ($R_1^i, R_2^i, R_3^i, R_4^i$) enclosing the segmented muscle contours (L-medial, L-lateral, L-superior, L-inferior), respectively.

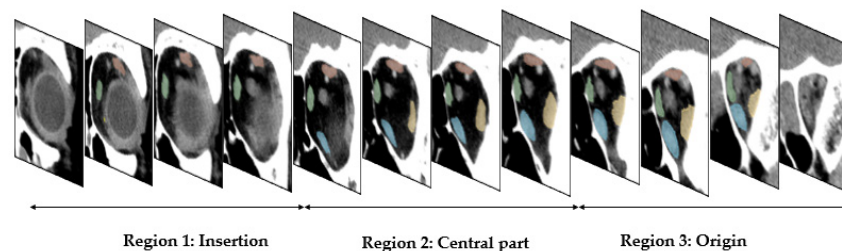


Figure 6. Illustration that shows the coronal slices containing extraocular muscles being split equally into three regions that include the muscle insertion, central part, and origin, respectively.

We also compared the thickness and cross-sectional area measurements of extraocular muscles from ground truth segmentation to those from predicted segmentations using the two metrics—mean absolute error (MAE) and mean absolute percentage error (MAPE), given by

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |x_i^{pr} - x_i^{gt}| \quad (10)$$

$$\text{MAPE} = \frac{1}{n} \sum_{i=1}^n \left| \frac{x_i^{pr} - x_i^{gt}}{x_i^{gt}} \right| \quad (11)$$

where x_i^{pr} is the predicted value, x_i^{gt} is the ground truth value of the i th sample, and n is the number of samples. The MAE and MAPE together can be used to evaluate how closely the predicted measurements from the model align with the ground truth measurements.

To perform qualitative evaluation, the predicted segmentations from the models were analyzed visually by the radiologist for similarity of contours and other subjective assessments. An accept/reject decision was provided with rejected segmentations accompanied by a reason.

3. Results

3.1. Quantitative Evaluation

To evaluate the performance on the validation and test data, the same data preprocessing steps were applied during model training, i.e., isometric resampling, drawing image patches, and CT windowing, were applied to the test dataset as well. The final predicted slice was then reconstructed from the individual patch predictions using sliding window predictions and majority voting for each pixel to facilitate thickness and area measurements.

3.1.1. Model Performance

The results from 10-fold cross validation training process are summarized in Table 2. The model trained using WCE+Dice compound loss function had the best overall performance on cross-validation data with a mean Dice score of 0.92 for all eight extraocular muscle classes with a standard deviation of 0.03. This was followed by the U-net model trained using Dice+Boundary compound loss function with a mean dice score of 0.91 and a standard deviation of 0.04. While the Dice+Boundary compound loss function had a better Dice score for L-medial rectus and L-lateral rectus, the WCE+Dice compound loss function consistently outperformed the other loss functions in the remaining classes. For this reason, we selected the model trained using WCE+Dice compound loss as the final model.

Table 2. Results from training and evaluation using 10-fold cross-validation. Values indicate mean \pm standard deviation of Dice score and IOU score from 10 cross-validation iterations. Values in bold represent the loss function setting that provides the best performance for a specific muscle class.

Evaluation Metric	Muscle	Loss Function				
		WCE	Dice	WCE + Dice	FTL	Dice + Boundary
Dice similarity coefficient (DSC) score	L-medial rectus	0.90 \pm 0.01	0.91 \pm 0.03	0.93 \pm 0.02	0.90 \pm 0.05	0.94 \pm 0.01
	L-lateral rectus	0.90 \pm 0.00	0.91 \pm 0.04	0.91 \pm 0.03	0.90 \pm 0.05	0.93 \pm 0.01
	L-superior group	0.84 \pm 0.03	0.90 \pm 0.03	0.91 \pm 0.02	0.87 \pm 0.06	0.87 \pm 0.05
	L-inferior rectus	0.90 \pm 0.02	0.92 \pm 0.03	0.94 \pm 0.02	0.90 \pm 0.03	0.93 \pm 0.02
	R-Medial rectus	0.90 \pm 0.00	0.93 \pm 0.02	0.94 \pm 0.01	0.91 \pm 0.02	0.93 \pm 0.01
	R-lateral rectus	0.88 \pm 0.01	0.91 \pm 0.04	0.91 \pm 0.04	0.88 \pm 0.06	0.90 \pm 0.05
	R-superior group	0.85 \pm 0.01	0.89 \pm 0.02	0.91 \pm 0.02	0.87 \pm 0.03	0.88 \pm 0.03
	R-inferior rectus	0.91 \pm 0.01	0.90 \pm 0.04	0.92 \pm 0.02	0.90 \pm 0.05	0.92 \pm 0.03
	All	0.89 \pm 0.03	0.91 \pm 0.03	0.92 \pm 0.03	0.89 \pm 0.05	0.91 \pm 0.04
Jaccard (IOU) score	L-medial rectus	0.81 \pm 0.02	0.86 \pm 0.04	0.88 \pm 0.03	0.83 \pm 0.07	0.89 \pm 0.02
	L-lateral rectus	0.83 \pm 0.00	0.85 \pm 0.05	0.86 \pm 0.04	0.84 \pm 0.06	0.87 \pm 0.01
	L-superior group	0.73 \pm 0.04	0.82 \pm 0.04	0.85 \pm 0.03	0.79 \pm 0.07	0.80 \pm 0.05
	L-inferior rectus	0.82 \pm 0.03	0.86 \pm 0.04	0.88 \pm 0.03	0.83 \pm 0.04	0.87 \pm 0.04
	R-medial rectus	0.82 \pm 0.00	0.87 \pm 0.03	0.89 \pm 0.01	0.84 \pm 0.03	0.88 \pm 0.02
	R-lateral rectus	0.79 \pm 0.02	0.85 \pm 0.05	0.85 \pm 0.05	0.81 \pm 0.07	0.84 \pm 0.06
	R-superior group	0.75 \pm 0.01	0.82 \pm 0.02	0.84 \pm 0.03	0.79 \pm 0.03	0.80 \pm 0.03
	R-inferior rectus	0.83 \pm 0.02	0.84 \pm 0.04	0.87 \pm 0.03	0.83 \pm 0.07	0.87 \pm 0.04
	All	0.80 \pm 0.04	0.85 \pm 0.04	0.87 \pm 0.04	0.82 \pm 0.06	0.85 \pm 0.05

The performance of the selected U-net model was evaluated using test data and the results are summarized in Table 3. The selected model that was trained using WCE+Dice loss function could segment the extraocular muscles with a mean Dice score of 0.92 and a standard deviation of 0.02, which corresponds to an IOU score of 0.87 and a standard deviation of 0.03.

Model performance was also evaluated on different regions of the extraocular muscles near their origin, central part and near their insertion and the results summarized in Table 4. We observed that the model performed better on the central region than the regions near the origin and insertion but the difference in performance was not statistically significant (p -value = 0.1763).

Table 3. Performance of selected U-net model (trained using WCE+Dice loss) on test data. Values indicate mean \pm standard deviation of Dice score and IOU score across 32 test scans.

Muscle	DSC Score	IOU Score
L-medial rectus	0.94 \pm 0.07	0.90 \pm 0.09
L-lateral rectus	0.93 \pm 0.09	0.88 \pm 0.10
L-superior group	0.90 \pm 0.12	0.83 \pm 0.13
L-inferior rectus	0.94 \pm 0.08	0.90 \pm 0.08
R-medial rectus	0.92 \pm 0.16	0.88 \pm 0.17
R-lateral rectus	0.93 \pm 0.04	0.88 \pm 0.06
R-superior group	0.87 \pm 0.14	0.80 \pm 0.15
R-inferior rectus	0.93 \pm 0.09	0.88 \pm 0.11
All	0.92 \pm 0.02	0.87 \pm 0.03

Table 4. Performance of selected U-net on test data split by extraocular muscle regions (near insertion, muscle belly, and near origin).

Muscle	Region 1: Insertion		Region 2: Central Part		Region 3: Origin	
L-medial rectus	0.89 \pm 0.13	0.82 \pm 0.16	0.97 \pm 0.01	0.94 \pm 0.02	0.91 \pm 0.10	0.85 \pm 0.13
L-lateral rectus	0.88 \pm 0.15	0.81 \pm 0.15	0.95 \pm 0.02	0.90 \pm 0.03	0.94 \pm 0.08	0.89 \pm 0.09
L-superior group	0.79 \pm 0.26	0.71 \pm 0.25	0.92 \pm 0.06	0.86 \pm 0.07	0.92 \pm 0.05	0.85 \pm 0.08
L-inferior rectus	0.93 \pm 0.04	0.88 \pm 0.06	0.94 \pm 0.07	0.89 \pm 0.08	0.95 \pm 0.02	0.90 \pm 0.04
R-medial rectus	0.91 \pm 0.16	0.85 \pm 0.16	0.79 \pm 0.34	0.75 \pm 0.35	0.83 \pm 0.25	0.77 \pm 0.25
R-lateral rectus	0.77 \pm 0.20	0.66 \pm 0.21	0.93 \pm 0.03	0.87 \pm 0.05	0.94 \pm 0.04	0.89 \pm 0.06
R-superior group	0.78 \pm 0.29	0.70 \pm 0.28	0.91 \pm 0.04	0.84 \pm 0.06	0.89 \pm 0.04	0.80 \pm 0.07
R-inferior rectus	0.89 \pm 0.16	0.83 \pm 0.16	0.95 \pm 0.07	0.90 \pm 0.08	0.94 \pm 0.03	0.88 \pm 0.06
All	0.86 \pm 0.20	0.78 \pm 0.20	0.92 \pm 0.14	0.87 \pm 0.15	0.91 \pm 0.11	0.86 \pm 0.12

3.1.2. Comparison of Muscle Size Measurements

The results from thickness and area measurements are summarized in Table 5. The thicknesses and areas measured from the predicted segmentations, when compared with the ground truth segmentations, had a mean absolute error of 0.35 mm and 3.87 mm², respectively. The corresponding mean absolute percentage errors in thickness and area were 7 and 9%, respectively.

Table 5. Performance of selected U-net model (trained using WCE+Dice loss) on test data. Values indicate mean \pm standard deviation of Dice score and IOU score on cross-validation data.

Muscle	MAE Thickness (mm)	MAPE Thickness	MAE Area (mm ²)	MAPE Area
L-medial rectus	0.24	5%	1.99	6%
L-lateral rectus	0.35	7%	6.53	14%
L-superior group	0.37	8%	3.15	8%
L-inferior rectus	0.26	6%	4.2	10%
R-medial rectus	0.41	7%	3.93	8%
R-lateral rectus	0.46	9%	3.85	10%
R-superior group	0.33	7%	4.09	10%
R-inferior rectus	0.36	8%	3.18	9%
All	0.35	7%	3.87	9%

3.1.3. Model Performance on Noisy Images

To assess its robustness, the trained model was evaluated with inputs with varying degrees of noise. During the training stage, the model was trained using images added with a Gaussian noise with zero mean and a standard deviation of 10 HUs. During the

testing stage, the input images were added with a Gaussian noise with standard deviations of 5 and 10 HUs, respectively. The performance of the trained U-net on noisy images is shown in Table 6. There was no significant reduction in performance (p -value = 0.8913), evaluated using the Dice score, of the model on noisy images with Gaussian noise up to ($\mu = 0, \sigma = 10$). This can be explained by the fact that the U-net was trained on similar noisy images during the training stage. Figure 7, shows a sample coronal patch from the test data with and without noise, and its corresponding predictions by the trained model.

Table 6. Average performance of the trained U-net on 32 test cases where input images were added with different levels of Gaussian noise. Values indicate mean DSC (and standard deviation) and mean IOU (and standard deviation), respectively.

	Without Added Noise	With Added Noise ($\mu = 0, \sigma = 5$)	With Added Noise ($\mu = 0, \sigma = 10$)
L-medial rectus	0.94 ± 0.07	0.94 ± 0.08	0.93 ± 0.09
L-lateral rectus	0.93 ± 0.09	0.93 ± 0.07	0.92 ± 0.09
L-superior group	0.90 ± 0.12	0.89 ± 0.15	0.90 ± 0.13
L-inferior rectus	0.94 ± 0.08	0.94 ± 0.04	0.94 ± 0.09
R-medial rectus	0.92 ± 0.16	0.93 ± 0.10	0.92 ± 0.14
R-lateral rectus	0.93 ± 0.04	0.92 ± 0.08	0.92 ± 0.07
R-superior group	0.87 ± 0.14	0.86 ± 0.16	0.86 ± 0.17
R-inferior rectus	0.93 ± 0.09	0.93 ± 0.08	0.93 ± 0.07
All	0.92 ± 0.02	0.92 ± 0.12	0.92 ± 0.12

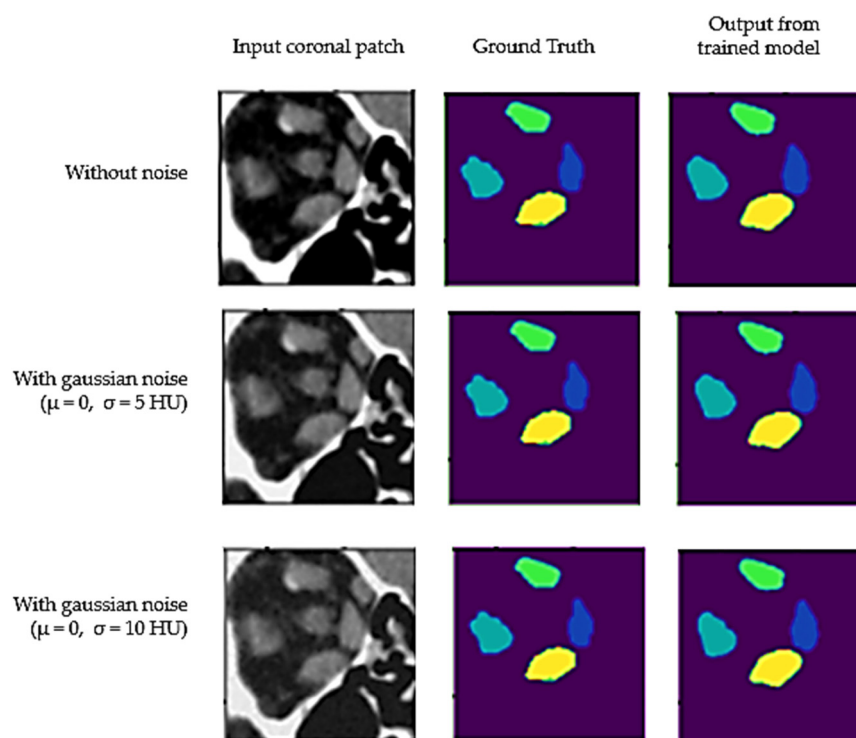


Figure 7. Output from the trained U-net on sample coronal patch where input image has been added with different levels of Gaussian noise (with mean μ and standard deviation σ).

3.1.4. Performance Comparison with Traditional Segmentation Methods

Prior works that used traditional segmentation methods to segment extraocular muscles were built and evaluated on MR images [2–5]. Hanai et al. [13] proposed a deep-learning method to detect enlarged muscles and therefore only provided classification accuracy of their model in detecting enlarged extraocular muscles and not segmentation evaluation results. The U-net based CNN model proposed by Zhu et al. [12] was built

and evaluated on CT images. However, the imaging studies and ground truths that were used varied between the studies. Since there are no established benchmarks for extraocular muscle segmentation, we show the results from our model alongside previous works. In our proposed model, the superior rectus and superior levator palpebrae muscles were measured together as a single muscle group, namely, the superior muscle group. We also consider extraocular muscles from left and right as different classes. The analysis of the evaluation metric (intersection over union) is summarized in Table 7.

Table 7. Regional IOU score of our model and previously published CNN models for extraocular muscle segmentation on CT. Values indicate mean IOU \pm standard deviation. SU-Net and SV-net proposed by Zhu et al. [12] was trained and evaluated using images from 97 subjects without contrast enhancement and our model was trained and evaluated using images from 210 subjects with contrast enhancement.

Muscle	SU-Net	SV-Net	2D Coronal U-Net
Medial rectus	$0.82 \pm 2.83 \times 10^{-5}$	$0.84 \pm 3.62 \times 10^{-5}$	0.91 ± 0.12
Lateral rectus	$0.80 \pm 5.83 \times 10^{-5}$	$0.82 \pm 3.56 \times 10^{-5}$	0.89 ± 0.04
Superior rectus	$0.73 \pm 9.73 \times 10^{-5}$	$0.74 \pm 7.84 \times 10^{-5}$	-
Superior muscle group	-	-	0.84 ± 0.09
Inferior rectus	$0.82 \pm 2.83 \times 10^{-5}$	$0.84 \pm 3.39 \times 10^{-5}$	0.89 ± 0.06
Optic nerve	$0.81 \pm 1.77 \times 10^{-4}$	$0.82 \pm 9.96 \times 10^{-5}$	-
Total	$0.80 \pm 2.56 \times 10^{-5}$	$0.82 \pm 3.22 \times 10^{-5}$	0.88 ± 0.09

3.2. Qualitative Evaluation

Among the thirty-two test samples, thirty predicted segmentations from the U-net algorithm were accepted while two were rejected. Example coronal slices from the two rejected segmentations are presented in Figure 8. The predicted segmentation on Study ID 2G04345 was rejected because the L-lateral rectus also included other orbital structures. The predicted segmentation on Study ID 2G04323 was rejected because the L-lateral rectus included areas of bone.

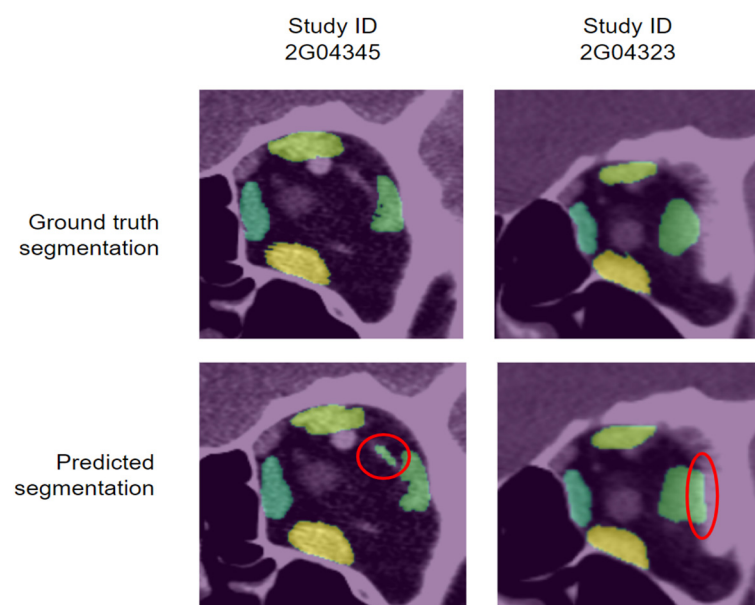


Figure 8. Coronal slice examples from the two test samples whose predicted segmentations were rejected after qualitative evaluation. Red circles outline the areas erroneously predicted by the U-net algorithm.

4. Discussion

We have developed and evaluated a 2D CNN-based deep learning algorithm that can perform the automated segmentation of extraocular muscles and provide measurements of two-dimensional parameters for muscle size, such as thickness and cross-sectional area. The proposed algorithm provides a method to carry out automated segmentation in a computationally efficient way using only images in the coronal plane of a CT scan.

To improve the segmentation accuracy further, postprocessing steps such as thresholding, erosion and dilation can be applied. The segmentations can therefore be further refined to exclude bone and other orbital structures that are not extraocular muscles. Furthermore, it is worth noting that the thickness errors (in mm) between predicted and ground truth measurements were in the same range as the pixel sizes (0.3–0.4 mm) of the CT images. This could be due to the data preprocessing (isometric resampling) step where we downsampled all CT images to a constant pixel size of 1 mm × 1 mm. Since the CNN algorithm reads and makes the prediction on resampled images, a single-pixel misclassification along the short-axis of extraocular muscles can result in a MAPE of up to 10% during reconstruction back to original pixel size. The downsampling step could potentially result in the loss of granular information and therefore drive the errors in thickness and area measurements. A training methodology that uses the original pixel intensities without isometric resampling and various postprocessing techniques will be explored as part of future work.

Neural networks perform best and generalize successfully when input data at the time of inference has a similar data distribution to that of the training data used. The proposed model was developed using CT images of the orbit and of extraocular muscles (EOM) with and without enlargement for male and female cases. Thus, it may not perform well with scans of other modalities, such as MRI. Furthermore, the model needs to be trained using other types of scans that might not include EOM for it to learn the other organs that do not constitute EOM. Single institution training and testing data were used for this study. Generalizability to other institutions and patient populations should be evaluated in the future.

In this work, we evaluate one type of neural network architecture, i.e., the encoder-decoder architecture to carry out semantic segmentation. As next steps, we could carry out further evaluations using other neural network architectures that have shown promise to work well for semantic segmentation, such as residual networks (ResNets) and region-proposal networks.

5. Conclusions

Based on the results from the quantitative and qualitative evaluations, this study demonstrates that CNN-based deep learning algorithms are effective at segmenting extraocular muscles and measuring muscle sizes on CT images without any manual inputs from a radiologist.

Author Contributions: Conceptualization, R.R.B.J.S. and D.T.G.; methodology, R.R.B.J.S.; software, R.R.B.J.S.; validation, R.R.B.J.S.; formal analysis, R.R.B.J.S.; investigation, R.R.B.J.S. and D.T.G.; resources, D.T.G.; data curation, R.R.B.J.S., M.H.Z. and D.T.G.; writing—original draft preparation, R.R.B.J.S., M.H.Z. and D.T.G.; writing—review and editing, R.R.B.J.S., M.H.Z. and D.T.G.; visualization, R.R.B.J.S.; supervision, D.T.G.; project administration, R.R.B.J.S. and D.T.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: The study was conducted in accordance with the Declaration of Helsinki and approved by the Institutional Review Board (or Ethics Committee) of the University of Chicago (IRB18-1247, approved on 30 March 2019).” for studies involving humans.

Informed Consent Statement: Patient consent was waived due to the retrospective nature of the study with anonymized data.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Acknowledgments: This work was completed in part with resources provided by the University of Chicago Research Computing Center.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Szucs-Farkas, Z.; Toth, J.; Balazs, E.; Galuska, L.; Burman, K.D.; Karanyi, Z.; Leovey, A.; Nagy, E.V. Using morphologic parameters of extraocular muscles for diagnosis and follow-up of Graves' ophthalmopathy: Diameters, areas, or volumes? *AJR Am. J. Roentgenol.* **2002**, *179*, 1005–1010. [[CrossRef](#)] [[PubMed](#)]
2. Firbank, M.J.; Harrison, R.M.; Williams, E.D.; Coulthard, A. Measuring extraocular muscle volume using dynamic contours. *Magn. Reson. Imaging* **2001**, *19*, 257–265. [[CrossRef](#)]
3. Lv, B.; Wu, T.; Lu, K.; Xie, Y. Automatic Segmentation of Extraocular Muscle Using Level Sets Methods with Shape Prior. *IFMBE Proc.* **2013**, *39*, 904–907. [[CrossRef](#)]
4. Wei, Q.; Sueda, S.; Miller, J.M.; Demer, J.L.; Pai, D.K. Template-based reconstruction of human extraocular muscles from magnetic resonance images. In Proceedings of the 2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro, Boston, MA, USA, 28 June–1 July 2009; pp. 105–108. [[CrossRef](#)]
5. Xing, Q.; Li, Y.; Wiggins, B.; Demer, J.; Wei, Q. *Automatic Segmentation of Extraocular Muscles Using Superpixel and Normalized Cuts. Advances in Visual Computing*; ISVC 2015. Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2015; Volume 9474. [[CrossRef](#)]
6. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*; Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F., Eds.; Springer: Cham, Switzerland, 2015; Volume 9351, pp. 234–241. [[CrossRef](#)]
7. Milletari, F.; Navab, N.; Ahmadi, S. V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation. In Proceedings of the 2016 Fourth International Conference on 3D Vision (3DV), Stanford, CA, USA, 25–28 October 2016; pp. 565–571. [[CrossRef](#)]
8. Jalali, Y.; Fateh, M.; Rezvani, M.; Abolghasemi, V.; Anisi, M.H. ResBCDU-Net: A Deep Learning Framework for Lung CT Image Segmentation. *Sensors* **2021**, *21*, 268. [[CrossRef](#)]
9. Wang, T.; Xing, H.; Li, Y.; Wang, S.; Liu, L.; Li, F.; Jing, H. Deep learning-based automated segmentation of eight brain anatomical regions using head CT images in PET/CT. *BMC Med. Imaging* **2022**, *22*, 99. [[CrossRef](#)]
10. Nikolov, S.; Blackwell, S.; Zverovitch, A.; Mendes, R.; Livne, M.; De Fauw, J.; Patel, Y.; Meyer, C.; Askham, H.; Romera-Paredes, B.; et al. Clinically Applicable Segmentation of Head and Neck Anatomy for Radiotherapy: Deep Learning Algorithm Development and Validation Study. *J. Med. Internet Res.* **2021**, *23*, e26151. [[CrossRef](#)]
11. Kamnitsas, K.; Ledig, C.; Newcombe, V.F.; Simpson, J.P.; Kane, A.D.; Menon, D.K.; Rueckert, D.; Glocker, B. Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. *Med. Image Anal.* **2017**, *36*, 61–78. [[CrossRef](#)]
12. Zhu, F.; Gao, Z.; Zhao, C.; Zhu, Z.; Tang, J.; Liu, Y.; Tang, S.; Jiang, C.; Li, X.; Zhao, M.; et al. Semantic segmentation using deep learning to extract total extraocular muscles and optic nerve from orbital computed tomography images. *Optik* **2021**, *244*, 167551. [[CrossRef](#)]
13. Hanai, K.; Tabuchi, H.; Nagasato, D.; Tanabe, M.; Masumoto, H.; Nishio, S.; Nishio, N.; Nakamura, H.; Hashimoto, M. Automated Detection of Enlarged Extraocular Muscle In Graves' Ophthalmopathy With Computed Tomography And Deep Neural Network. *Res. Sq.* **2022**. [[CrossRef](#)]
14. Ozgen, A.; Ariyurek, M. Normative Measurements of Orbital Structures Using CT. *AJR Am. J. Roentgenol.* **1998**, *170*, 1093–1096. [[CrossRef](#)]
15. Xu, L.; Li, L.; Xie, C.; Guan, M.; Xue, Y. Thickness of Extraocular Muscle and Orbital Fat in MRI Predicts Response to Glucocorticoid Therapy in Graves' Ophthalmopathy. *Int. J. Endocrinol.* **2017**, *2017*, 3196059. [[CrossRef](#)] [[PubMed](#)]
16. Firbank, M.J.; Coulthard, A. Evaluation of a technique for estimation of extraocular muscle volume using 2D MRI. *Br. J. Radiol.* **2000**, *73*, 1282–1289. [[CrossRef](#)] [[PubMed](#)]
17. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.
18. Szeliski, R. *Computer Vision: Algorithms and Applications*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2010. [[CrossRef](#)]
19. Geremia, E.; Menze, B.H.; Clatz, O.; Konukoglu, E.; Criminisi, A.; Ayache, N. Spatial decision forests for MS lesion segmentation in multi-channel MR images. In *Medical Image Computing and Computer-Assisted Intervention—MICCAI*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 111–118. [[CrossRef](#)]
20. Liu, X.; Song, L.; Liu, S.; Zhang, Y. A Review of Deep-Learning-Based Medical Image Segmentation Methods. *Sustainability* **2021**, *13*, 1224. [[CrossRef](#)]
21. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.

22. Chen, L.; Papandreou, G.; Kokkinos, I.; Murphy, K.P.; Yuille, A.L. Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs. *arXiv* **2015**, arXiv:1412.7062.
23. Chen, L.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 834–848. [[CrossRef](#)]
24. Chen, L.; Papandreou, G.; Schroff, F.; Adam, H. Rethinking Atrous Convolution for Semantic Image Segmentation. *arXiv* **2017**, arXiv:abs/1706.05587.
25. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 801–818.
26. Krähenbühl, P.; Koltun, V. Efficient Inference in Fully Connected CRFs with Gaussian Edge Potentials. In Proceedings of the Advances in Neural Information Processing Systems, Granada, Spain, 12–15 December 2011.
27. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [[CrossRef](#)]
28. Myronenko, A. 3D MRI brain tumor segmentation using autoencoder regularization. In Proceedings of the International MICCAI Brainlesion Workshop, Shenzhen, China, 17 October 2018; pp. 311–320.
29. Nie, D.; Wang, L.; Adeli, E.; Lao, C.; Lin, W.; Shen, D. 3-D fully convolutional networks for multimodal iso-intense infant brain image segmentation. *IEEE Trans. Cybern.* **2019**, *49*, 1123–1136. [[CrossRef](#)]
30. Wang, S.; Yi, L.; Chen, Q.; Meng, Z.; Dong, H.; He, Z. Edge-aware Fully Convolutional Network with CRF-RNN Layer for Hippocampus Segmentation. In Proceedings of the 2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC), Chongqing, China, 24–26 May 2019; pp. 803–806.
31. Edupuganti, V.G.; Chawla, A.; Amit, K. Automatic optic disk and cup segmentation of fundus images using deep learning. In Proceedings of the 25th IEEE International Conference on Image Processing (ICIP), Athens, Greece, 7–10 October 2018; pp. 2227–2231.
32. Shankaranarayana, S.M.; Ram, K.; Mitra, K.; Sivaprakasam, M. Joint optic disc and cup segmentation using fully convolutional and adversarial networks. In *Fetal, Infant and Ophthalmic Medical Image Analysis*; Springer: Cham, Switzerland, 2017; pp. 168–176.
33. Anthimopoulos, M.M.; Christodoulidis, S.; Ebner, L.; Geiser, T.; Christe, A.; Mougiakakou, S. Semantic Segmentation of Pathological Lung Tissue with Dilated Fully Convolutional Networks. *IEEE J. Biomed. Health Inform.* **2019**, *23*, 714–722. [[CrossRef](#)]
34. Christ, P.F.; Ettliger, F.; Grün, F.; Elshaera, M.E.A.; Lipkova, J.; Schlecht, S.; Ahmaddy, F.; Tatavarty, S.; Bickel, M.; Bilic, P.; et al. Automatic liver and tumor segmentation of CT and MRI volumes using cascaded fully convolutional neural networks. *arXiv* **2017**, arXiv:1702.05970.
35. Tran, P.V. A fully convolutional neural network for cardiac segmentation in short-axis MRI. *arXiv* **2016**, arXiv:1604.00494.
36. Çiçek, Ö.; Abdulkadir, A.; Lienkamp, S.S.; Brox, T.; Ronneberger, O. 3D U-Net: Learning dense volumetric segmentation from sparse annotation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Athens, Greece, 17–21 October 2016; pp. 424–432.
37. Oktay, O.; Schlemper, J.; Folgoc, L.L.; Lee, M.; Heinrich, M.; Misawa, K.; Mori, K.; McDonagh, S.; Hammerla, N.Y.; Kainz, B.; et al. Attention u-net: Learning where to look for the pancreas. *arXiv* **2018**, arXiv:1804.03999.
38. Zhang, Y.; Chung, A.C.S. Deep supervision with additional labels for retinal vessel segmentation task. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Granada, Spain, 16–20 September 2018; pp. 83–91.
39. Novikov, A.A.; Lenis, D.; Major, D.; Uvka, J.H.; Wimmer, M.; Bühler, K. Fully convolutional architectures for multiclass segmentation in chest radiographs. *IEEE Trans. Med. Imaging* **2018**, *37*, 1865–1876. [[CrossRef](#)] [[PubMed](#)]
40. Ye, C.; Wang, W.; Zhang, S.; Wang, K. Multi-depth fusion network for whole-heart CT image segmentation. *IEEE Access* **2019**, *7*, 23421–23429. [[CrossRef](#)]
41. Luc, P.; Couprie, C.; Chintala, S.; Verbeek, J. Semantic segmentation using adversarial networks. *arXiv* **2016**, arXiv:1611.08408.
42. Moeskops, P.; Veta, M.; Lafarge, M.W.; Eppenhof, K.A.J.; Pluim, J.P.W. Adversarial training and dilated convolutions for brain MRI segmentation. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*; Springer: Cham, Switzerland, 2017; pp. 56–64.
43. Son, J.; Park, S.J.; Jung, K.H. Retinal vessel segmentation in fundoscopic images with generative adversarial networks. *arXiv* **2017**, arXiv:1706.09318.
44. Han, Z.; Wei, B.; Mercado, A.; Leung, S.; Li, S. Spine-GAN: Semantic segmentation of multiple spinal structures. *Med. Image Anal.* **2018**, *50*, 23–35. [[CrossRef](#)]
45. Fedorov, A.; Beichel, R.; Kalpathy-Cramer, J.; Finet, J.; Fillion-Robin, J.C.; Pujol, S.; Bauer, C.; Jennings, D.; Fennessy, F.; Sonka, M.; et al. 3D Slicer as an image computing platform for the Quantitative Imaging Network. *Magn. Reson. Imaging* **2012**, *30*, 1323–1341. [[CrossRef](#)]
46. Frush, D.P.; Slack, C.C.; Hollingsworth, C.L.; Bisset, G.S.; Donnelly, L.F.; Hsieh, J.; Lavin-Wensell, T.; Mayo, J.R. Computer-simulated radiation dose reduction for abdominal multidetector CT of pediatric patients. *AJR Am. J. Roentgenol.* **2002**, *179*, 1107–1113. [[CrossRef](#)]
47. Zeng, D.; Huang, J.; Bian, Z.; Niu, S.; Zhang, H.; Feng, Q.; Liang, Z.; Ma, J. A Simple Low-dose X-ray CT Simulation from High-dose Scan. *IEEE Trans. Nucl. Sci.* **2015**, *62*, 2226–2233. [[CrossRef](#)]

48. Ma, J.; Chen, J.; Ng, M.; Huang, R.; Li, Y.; Li, C.; Yang, X.; Martel, A.L. Loss odyssey in medical image segmentation. *Med. Image Anal.* **2021**, *71*, 102035. [[CrossRef](#)] [[PubMed](#)]
49. Dice, L.R. Measures of the Amount of Ecologic Association Between Species. *Ecology* **1945**, *26*, 297–302. [[CrossRef](#)]
50. Rahman, M.A.; Wang, Y. *Optimizing Intersection-Over-Union in Deep Neural Networks for Image Segmentation*. *Advances in Visual Computing*; ISVC 2016. Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2016; Volume 10072. [[CrossRef](#)]
51. Abraham, N.; Khan, N.M. A Novel Focal Tversky Loss Function With Improved Attention U-Net for Lesion Segmentation. In Proceedings of the 2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019), Venice, Italy, 8–11 April 2019; pp. 683–687.
52. Kervadec, H.; Bouchtiba, J.; Desrosiers, C.; Granger, E.; Dolz, J.; Ayed, I.B. Boundary loss for highly unbalanced segmentation. *Med. Image Anal.* **2021**, *67*, 101851. [[CrossRef](#)] [[PubMed](#)]
53. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980. [[CrossRef](#)]
54. Srivastava, N.; Hinton, G.E.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.
55. Glorot, X.; Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. *Proc. Thirteen. Int. Conf. Artif. Intell. Stat. Proc. Mach. Learn. Res.* **2010**, *9*, 249–256.