

# Brain Tumor Segmentation and Classification Using ACGAN with U-Net and Independent CNN-Based Validation

S. Wagle<sup>1,2</sup> and K. Poudel<sup>1,2</sup>

<sup>1</sup>Computational and Data Science, Middle Tennessee State University, Murfreesboro, TN, USA

<sup>2</sup>Computer Science, Middle Tennessee State University, Murfreesboro, TN, USA

sw8k@mtmail.mtsu.edu,  
khem.poudel@mtsu.edu

**Abstract**— This study presents an approach that combines Auxiliary Classifier Generative Adversarial Networks (ACGAN) with U-Net architecture to enhance the brain tumor segmentation and classification of brain tumors using synthetic data. Also, the CNN based classifier was used, thus using classification as a means to verify the accuracy and realism of the synthetic image generated by the ACGAN. This strategy aims to improve AI model training by generating diverse synthetic images, addressing issues like data scarcity, necessity for exact tumor segmentation, validation of synthetic image quality, and ethical concerns related to patient data. ACGAN produces artificial images for four types of brain tumors: meningitis, glioma, pituitary, and healthy. The incorporation of UNet enhances the accuracy of segmentation considerably. An independent CNN-based classifier is used to categorize actual and synthetic images separately in order to verify the accuracy and realism of the synthetic images. The classifier achieved an overall accuracy of 0.84 when trained on combined real and synthetic images, therefore showing the suitability of synthetic images to augment training datasets. The U-Net model performed very well in the generation of accurate segmentation maps, as evidenced by a Dice Coefficient of 76.43% and a MeanIOU of 92.91%, with high sensitivity of 99.49% and specificity of 99.87%. These results demonstrate the great potential of integrating ACGAN with U-Net in the pursuit of developing AI-driven medical imaging solutions.

**Keywords**—Auxiliary Classifier Generative Adversarial Network (ACGAN); U-Net; Segmentation; Glioma; Meningitis; Pituitary; CNN-based classifier; Dice Coefficient and Mean-IOU.

## I. INTRODUCTION

In the medical field, managing brain tumors—which can range from aggressive malignancies to gliomas—presents a significant challenge [1]. High-quality and precise neuroimaging techniques are essential for accurate diagnosis and efficient treatment planning. The most important of these is Magnetic Resonance Imaging (MRI), which offers high-resolution pictures that are essential for determining the position, dimensions, and scope of brain tumors [2]. The introduction of sophisticated MRI sequences, like MR spectroscopy and functional MRI, has improved the capacity to understand the biological behavior of tumors. Despite MRI's advantages, a major obstacle to using it to its full diagnostic potential is a deficiency of annotated data, which is necessary for automated

diagnostic system training and validation [3]. This lack of data is made worse by the rarity of some tumor subtypes, and there are further obstacles due to patient privacy issues and the expense of data curation. Data augmentation, particularly through dataset manipulation techniques like translation, rotation, flipping, cropping, and scaling, is a key strategy to overcome this problem [4]. These methods enrich training sets by introducing pixel-level variations, albeit without creating new images, but Pixel-level alteration can only introduce variations of the original image rather than new ones and can pose a risk of overfitting as well [5]. This issue has given rise to a potent solution in the form of Generative Adversarial Networks (GANs), which can synthesize realistic medical images and grow existing datasets without compromising patient privacy.

The introduction of Generative Adversarial Networks (GANs) has presented a new approach to get around these restrictions. GANs have demonstrated amazing success in producing artificial images that are identical to real ones [6]. GANs have been used in the field of medical imaging to generate realistic and diverse datasets, representing diseases that may not be well-represented in existing corpora [7]. The potential of GANs in medical imaging has been shown in recent studies, especially when it comes to enhancing data for different tumor types [8].

Generative Adversarial Networks (GANs) have changed the field of image synthesis since Goodfellow et al. introduced them in 2014. Adversarial processes were used to train the discriminator and generator neural networks, which together made a GAN [9]. In order to demonstrate the potential of GANs in medical imaging, Bowles et al. (2018) produced synthetic MRI scans of brain tumors in order to greatly increase the diversity of training datasets [10]. According to their findings, limited real datasets could be effectively replaced by artificial data created by GANs, improving the training of AI models. Further in 2018, Han et al. (2018) compared DCGAN and WGAN in generating synthetic multi-sequence brain Magnetic Resonance (MR) images which was successful in passing the preliminary validation test by the expert physician; unable to accurately distinguish the synthetic images from the real samples in the Visual Turing Test [11]. Similarly, the other works

includes the use of multi-scale gradient GAN to synthesize the MRI images with Meningioma disease. The synthesized images were used to augment the training set of a multi-class brain tumor classification problem which resulted in an improvement in the classifier's performance, in terms of the balance accuracy score [12].

Simultaneously, segmentation methods have advanced significantly, especially since the U-Net architecture was introduced by Ronneberger et al. in 2015, it became a fundamental tool in medical image segmentation due to its exceptional performance in capturing detailed features and providing accurate segmentation. In addition, the U-Net architecture has completely changed the field of image segmentation by providing a reliable framework for accurately defining tumor boundaries [13] [14]. In the study named "Automatic brain tumor detection and segmentation using U-Net based fully convolutional networks", the authors proposed a fully automatic method for brain tumor segmentation utilizing U-Net based deep convolutional networks. Their method, evaluated on the Multimodal Brain Tumor Image Segmentation (BRATS 2015) datasets, which include 220 high-grade and 54 low-grade tumor cases, demonstrated promising segmentation efficiency through cross-validation [15]. Referencing to different study that has made potential impact in advancement of the medical image segmentation, Isensee and others developed nnU-Net, an automated and adaptable version of U-Net that achieves state-of-the-art results across various biomedical segmentation tasks. Their results from the Medical Segmentation Decathlon challenge demonstrated that nnU-Net achieved the highest mean Dice scores for multiple tasks, indicating its robust performance and generalizability across segmentation of different biomedical images [16].

The integration of GANs with U-Net has shown substantial promise in brain tumor analysis. This combination leverages the strengths of both architectures: GANs generate realistic synthetic images, while U-Net provides accurate segmentation, resulting in robust AI models. In [17], the authors combined GAN and a U-Net architecture to produce high-quality synthetic medical images. They break down the U-Net into an encoder and generator, adding a Gaussian variable to the latent representation to guarantee that the images produced are diverse. Their findings demonstrate that, on eight different medical imaging datasets, this method achieves superior per-pixel accuracy and Frechet Inception Distance (FID) than both conditional GANs and regular GANs. Similarly, in [18], the study described the combination of GANs with U-Net for medical image segmentation and synthesis. They utilized the Pix2Pix framework, in which a fully convolutional network is used as the discriminator while the generator follows to the U-Net architecture. The synthesized images

detailed feature production and overall coherence are enhanced by this combination. The findings show that, in some tasks, U-Net outperforms conventional GAN architectures in terms of accuracy and realism of the produced medical images when used within the Pix2Pix framework.

Similarly, the study concentrating on comparison of image generation methods, to optimize the quality and applicability of synthetic images, researchers have explored the inclusion of segmentation masks alongside images in the training process. This method provides additional structural information that can enhance the realism of synthetic images. In a study [19], the segmentation maps of COVID-19 CT images were employed in a global-local generator and a multi-resolution discriminator to improve synthetic image production. The discriminator was trained using these down-sampled and re-rendered realistic maps. Compared to other cutting-edge methods, the method produced high-quality synthetic COVID-19 CT images that could be used for semantic segmentation and classification. Also, some authors [20] used a semi-supervised method combining ADC maps and U-Net to generate T2-weighted images. By training with both paired and unpaired data, the model ensures diverse and realistic synthetic images. The results show high-quality synthetic images with meaningful prostate cancer lesions, outperforming existing methods in visual quality and quantitative metrics. The tabular representation of the above studies can be depicted as in Table 1.

Despite significant advancements, current methods often fail to fully evaluate the combined impact of using synthetic image generation with detailed segmentation masks, and the accuracy and realism of these images have not been rigorously validated using independent classifiers. Hence, this study focuses on mitigating these gaps, and the main objectives include:

- Use ACGAN to create synthetic images of brain tumors, including meningitis, glioma, pituitary tumors, and healthy brain tissue to address data scarcity by generating diverse synthetic images.
- Use the U-Net architecture to produce precise segmentation maps for real images.
- Use an independent CNN-based classifier to categorize actual and synthetic images separately, thereby verifying the accuracy and realism of the synthetic images.

By addressing these objectives, the study aims to contribute significantly to the development of reliable AI models for medical imaging through the generation of high-quality synthetic data.

Table 1. Comparison of the model with other state-of-the-art models

Author	Approach	Result
Bowles et al., (2018) [10]	GAN Augmentation for Segmentation	DSC improved by 1-5%.
Han et al., (2018) [11]	DCGAN and WGAN for Synthetic Brain MR Images	Visual Turing Test: 54-77%.
Deepak et al., (2020) [12]	MSG-GAN for Brain MRI Synthesis	Balanced Accuracy: 90.3% to 93.1% with GAN; 86.4% to 88.7% with 35% data + GAN.
Ronneberger et al., (2015) [13]	U-Net for Biomedical Image Segmentation	DSC: 92% (PhC-U373), 77.5% (DIC-HeLa).
Isensee et al., (2018) [16]	nnU-Net	Mean DSC: 84.3%, Highest DSC: 90.5%.
Chen et al., (2021) [17]	Generative Adversarial U-Net	Mean DSC: 0.83, Highest DSC: 0.85.
Kazeminia et al., (2020) [18]	Pix2Pix with U-Net	DSC: 0.84 (Whole), 0.70 (Core), 0.65 (Enhancing); Sensitivity: 0.83 (Whole), 0.74 (Core), 0.72 (Enhancing).
Jiang et al., (2020) [19]	Conditional GAN for COVID-19 CT Image Synthesis	Mean DSC: 0.91.
Wang et al., (2020) [20]	Semi-supervised with ADC maps and U-Net	Balanced Accuracy: 90.3% without GAN, 93.1% with GAN, 86.4% with 35% data, 88.7% with 35% data + GAN.
<b>Our Study</b>	<b>ACGAN with U-Net and CNN-Based Validation</b>	<b>ACGAN Classifier: Accuracy: 84%, U-Net: Accuracy: 99.54%, MeanIOU: 92.91%, Dice Coefficient: 76.43%, Precision: 99.61%, Sensitivity: 99.49%, Specificity: 99.87%..</b>

## II. MATERIALS & METHODS

The generative adversarial network followed by U-Net architecture and the CNN-Based classifier, which is the foundation of the suggested methodology, is covered in detail in this section.

### II-A. Semantic Annotation for Image Segmentation

The training dataset consists of 1849 images, each of which shows a brain tumor in one or more forms or not at all as shown in Fig 1 is denoted by:

$$\mathcal{I} = \{I_1, I_2, \dots, I_{1849}\}.$$

Corresponding segmentation masks are denoted by:

$$\mathcal{M} = \{M_1, M_2, \dots, M_{1849}\}.$$

The aim is to provide the model with detailed semantic information of the MRI images. To increase the realism and semantic accuracy of the images produced by the ACGAN, each images are annotated to specify the tumor types. By combining the segmentation masks with their corresponding images, the ACGAN is trained to better understand and replicate the context and structure of the various tumor features within the images.

Segmentation masks  $M_i$  are pixel-wise matrices where each element  $M_i(p, q)$  assigns a class label to the pixel at position  $(p, q)$  in image  $I_i$  [21]. This detailed semantic information instructs the ACGAN in the generation process, enabling the creation of new images that closely resemble the context and structure as delineated by the segmentation masks [19] [20] [22].

### II-B. Numerical Encoding for Brain Tumor MRI's and its Segmentation Masks

The dataset uses numerical labels in integer format to represent different types of brain tumors. The segmentation masks in the dataset, designated as  $M$ , are 2D arrays of size  $h \times w$  where  $h$  is the mask's height and  $w$  is the width. The study looks at four different kinds of brain tumors ( $C = 4$ ). Class zero represents the non-tumors. Each pixel in  $M$  is assigned a label integer corresponding to its class, as shown in Table 2.

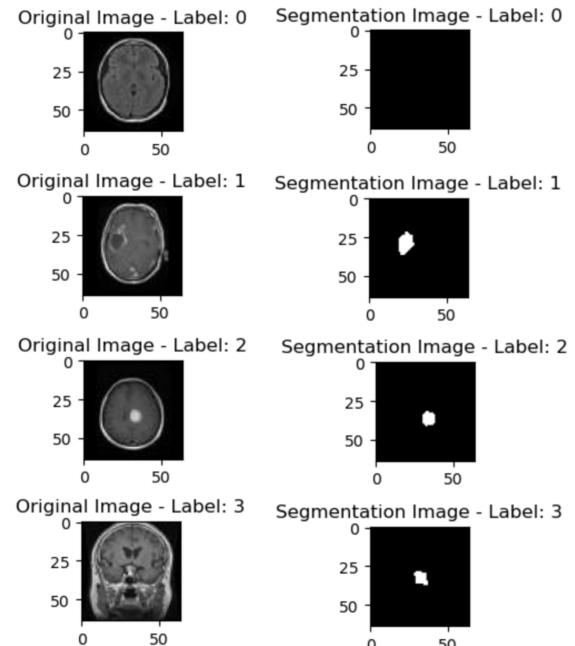


Figure 1. Original Brain MRI Images with Corresponding Segmentation Masks for Different Labels: 0 - Healthy, 1 - Glioma Tumor, 2 - Meningioma Tumor, and 3 - Pituitary Tumor.

Table 2. Label Integer Representation of Brain Tumor Classes

Class Name	Label Integer
Non Tumor	0
Glioma Tumor	1
Meningioma Tumor	2
Pituitary Tumor	3

### II-C. CNN-Based Generator and Discriminator in ACGAN

#### II-C1. Generator

Generator  $G$  shown in Fig 2 takes two types of inputs: a noise vector  $z \in R^d$  and conditional labels  $y$ , where  $d$  is the dimensionality of the noise vector and  $K$  is the

total number of classes ( $K = 4$ ). The architecture uses an embedding layer to transform the class label into a dense vector, subsequently concatenated with the noise vector to guide the process of image generation. An embedding layer is used to embed the class label  $y$  into a dense vector space.

$$e(y) = \text{Flatten}(\text{Embedding}(K, d)(y)) \quad (1)$$

The flatten operation ensures that the output from the embedding layer is a 1D vector. This embedding will enable the generator to learn a continuous representation of the class labels, and it can generate class-specific features.

The noise vector and the embedded label are combined through an element-wise product to ensure that all elements of the noise vector are modulated by the corresponding elements of the class vector. This effectively conditions the noise on the class information.

$$h = z \odot e(y) \quad (2)$$

The combined vector  $h$  is then passed through a dense layer followed by a Leaky ReLU activation function, transforms the combined vector into a higher-dimensional space suitable for subsequent convolutional operations. The weights  $w_1$  and biases  $b_1$  of the dense layer are learned during training.

$$h = \text{LeakyReLU}(W_1 h + b_1) \quad (3)$$

The output of the dense layer is reshaped into a 3D tensor and feeds it as an input into a few transposed convolutional layers, also known as deconvolutional layers, which further upsample the spatial dimensions of the tensor while reducing the number of channels. Each transposed convolutional layer applies a kernel on an input tensor and performs strided convolution to upscale the spatial dimensions, followed by the Leaky ReLU activation function that will introduce nonlinearity in the model in order for it to learn complex features.

$$h = \text{LeakyReLU}(\text{Conv2DTranspose}(128, (4, 4), \text{strides} = 2, \text{padding} = \text{same})(h)) \quad (4)$$

Finally, the output layer uses a  $\tanh$  activation function to the input  $h$  to produce the generated image.  $\tanh$  function ensures that the pixel values of the generated image are in the range  $[-1, 1]$ , which is suitable for image data.

### II-C2. Discriminator

The discriminator  $D$  as in Fig 3 is tasked with distinguishing between real and synthetic data and performing classification. This is achieved using combination of convolutional and fully connected layers. The input to the discriminator is an image  $X$ , of dimensions  $(128, 128, 1)$ . The image passes through a series of convolutional layers designed to extract hierarchical features from the input.

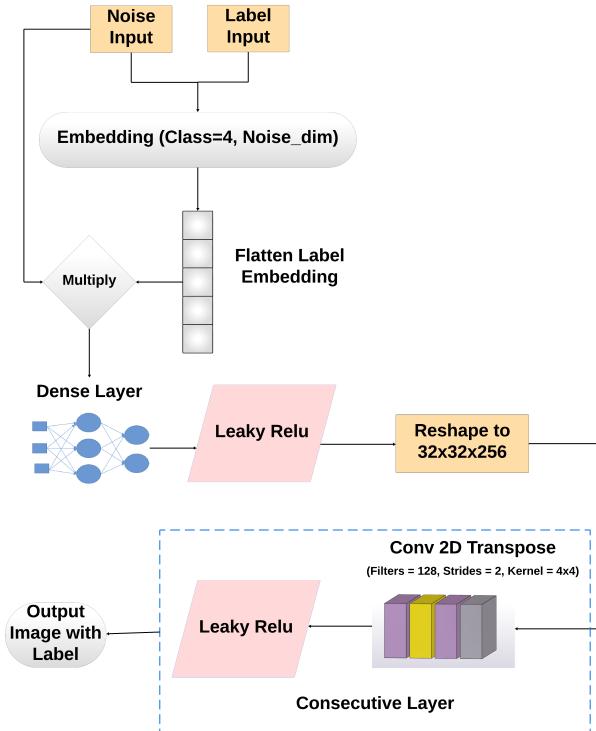


Figure 2. The generator architecture in ACGAN for Brain Tumor Image Generation.

The first convolutional layer applies a set of filters to the input image. This layer extracts low-level features of the image, such as edges and textures. The Leaky ReLU activation function adds non-linearity to the model so that more complex patterns can be learned. The feature maps, after passing through multiple convolution layers, are flattened into a 1D vector. Flattening prepares the feature maps for input into the fully connected layers. A dropout layer is applied to prevent overfitting. Dropout randomly sets a fraction of the input units to zero during training, which helps in regularizing the model and preventing overfitting. The flattened feature vector is fed into two separate dense layers for the validity score and class probabilities. This value will indicate whether the input image is real or fake using sigmoid activation function which outputs a scalar value between 0 and 1.

$$v = \sigma(W_2 h + b_2) \quad (5)$$

This layer's output is passed through a softmax activation function to ensure that this output is a probability distribution over the  $K$  classes. The softmax function ensures that the sum of the output probabilities is equal to 1, making it suitable for multi-class classification tasks.

$$p(c) = \text{softmax}(W_3 h + b_3) \quad (6)$$

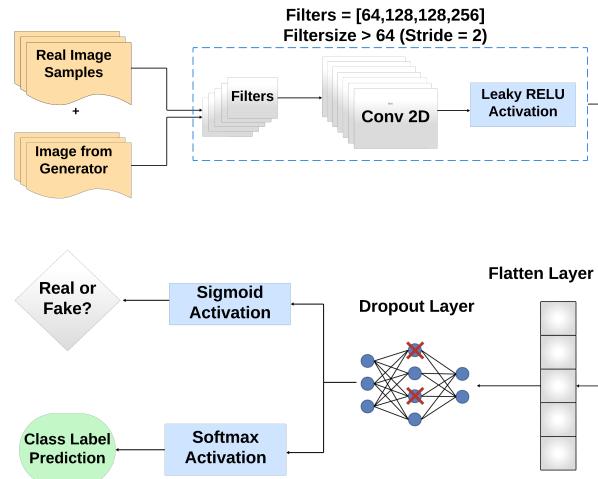


Figure 3. The discriminator architecture in an ACGAN to classify generated images as real or fake and determine their respective classes.

#### II-C3. Activation function

The LeakyReLU activation function is used in both generator and discriminator to enable the network to learn complex pattern over the data and to approximate the continuous function. This overcomes the issue of the dying ReLU by introducing a small non-zero gradient when the input is in the negative range, meaning that all neurons are not shut down at a time. This, therefore, introduces a slight slope in the negative part of the function and is commonly described using [23]:

$$\text{LeakyReLU}(x) = \begin{cases} x & \text{if } x > 0 \\ \alpha x & \text{otherwise} \end{cases} \quad (7)$$

#### II-C4. Loss function

The generator's loss function includes adversarial and auxiliary classification losses. The adversarial loss uses binary cross-entropy, defined as:

$$L_{\text{adv}}(G) = -E_{z \sim p_z(z), y \sim p_y(y)}[\log(D(G(z, y)))] \quad (8)$$

Here,  $G(z, y)$  is the generator's output for noise  $z$  and label  $y$ . The noise  $z$  follows distribution  $p_z(z)$ , and  $y$  is from  $p_y(y)$ .  $D(G(z, y))$  is the discriminator's probability that the sample is real. The generator minimizes this to produce data classified as real by the discriminator [24] [25].

The auxiliary classification loss,  $L_{\text{cls}}$ , complements the adversarial framework by enforcing class-specific generation:

$$L_{\text{cls}}(G) = -E_{z \sim p_z(z), y \sim p_y(y)}[\log P(y|G(z, y))] \quad (9)$$

Here,  $P(y|G(z, y))$  represents the probability that the generated sample belongs to class  $y$ . This encourages the

generator to produce convincing and correctly classified samples, promoting class consistency. The combined generator loss is a sum of the adversarial loss and the auxiliary classification loss, weighted by a hyperparameter  $\lambda$  that balances their relative importance [26].

The discriminator  $D$ , meanwhile, is trained to correctly identify real and synthetic data using its loss function. The loss is the sum of the real image loss  $L_{\text{real}}$  and fake image loss  $L_{\text{fake}}$ :

$$L_{\text{real}} = E_{I \sim p_{\text{data}}}[(1 - D(I_{\text{real}}))^2] \quad (10)$$

$$L_{\text{fake}} = E_{z \sim p_z(z)}[D(G(z, y))^2] \quad (11)$$

Here,  $G(z, y)$  denotes the synthetic image generated from the noise vector  $z$  and label  $y$ . The discriminator aims to assign a probability near zero to fake images, so  $D(G(z, y))$  measures the deviation from this ideal response.

Additionally, the discriminator performs a classification task with an auxiliary classification loss, using categorical cross-entropy to measure accuracy:

$$L_{\text{class}} = -E_{I \sim p_{\text{data}}, y \sim p_y(y)}[y \cdot \log(D_{\text{class}}(I))] \quad (12)$$

Here,  $D_{\text{class}}(I)$  is the discriminator's classification output, predicting the probability that image  $I$  belongs to class  $y$ . The discriminator's total loss is the sum of the real image loss and fake image loss, plus the auxiliary classification loss weighted by a hyperparameter  $\lambda'$  [26].

During training, the Adam optimizer, which is a gradient descent-based algorithm is used to update the weights. With respect to other optimization algorithm, Adam combines the advantages of both adaptive learning rate and momentum for each weight and maintaining running averages of both the gradients and their squared magnitudes [27].

#### II-D. U-Net architecture for Tumor Segmentation Generation

U-Net as in Fig 4 is designed for generating the segmentation mask to predict the tumor region in the real image. It takes an input image of size (128,128,1) to generate a segmentation map using a series of convolutions and pooling steps. In the Contracting path, every block is made up of two convolutions followed by max-pooling [13] [28]. So, for each convolution layer with filters [32, 64, 128, 256, 512], the operation would be:

$$C_i = \text{ReLU}(\text{Conv2D}(\text{filters}[i], 3 \times 3, \text{activation} = \text{relu}, \text{padding} = \text{same})(x_{i-1})) \quad (13)$$

Here,  $x_{i-1}$  is the input feature map from the previous layer, producing output  $C_i$  for the next pooling layer,  $3 \times 3$  is the filter size and ReLU introduces non-linearity within the feature map. The input to the pooling layer is the output  $C_i$  from the convolutional layer, producing

the dimensionally reduced feature map  $x_i$  by taking the maximum value in each  $2 \times 2$  region:

$$x_i = \text{MaxPooling2D}(2 \times 2)(C_i) \quad (14)$$

In the contracting path, the bottleneck layer compresses the spatial dimensions with a filter count of 512 before upsampling.

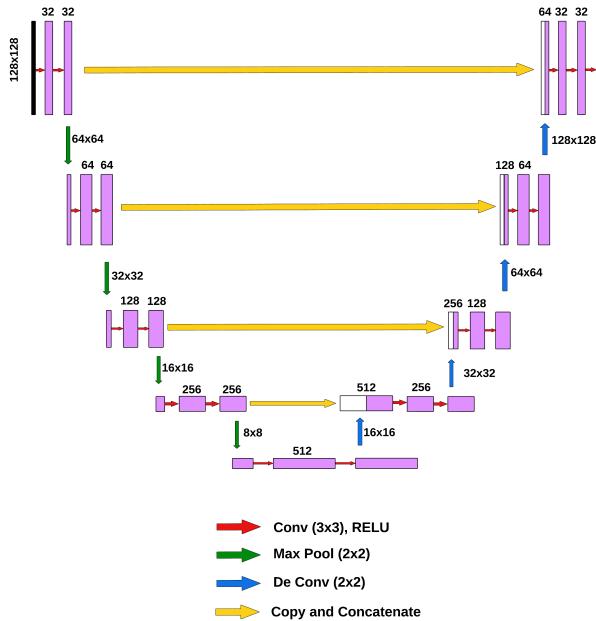


Figure 4. U-Net architecture for brain tumor segmentation of four distinct classes.

In the expansive path (decoder), each block includes a  $2 \times 2$  upsampling layer to double the spatial dimensions of the feature maps, allowing for precise localization in the segmentation map further followed by a skip connection, concatenation, and convolution to refine the feature map [13] [28]. For each layer with filters [256, 128, 64, 32] in the decoder:

$$U_i = \text{UpSampling2D}(2 \times 2)(B_{i-1}) \quad (15)$$

where  $B_{i-1}$  is the output from the previous convolution layer of encoder to produce  $U_i$  which is concatenated with the corresponding encoder output and this concatenated result  $m$  is convolved to produce the next feature map for next upsampling layer. This is explained as:

$$C_{\text{dec}} = \text{ReLU}(\text{Conv2D}(\text{filters}[i], 3 \times 3, \text{padding} = \text{same})(m)) \quad (16)$$

The  $C_{\text{dec}}$  is fed to the final convolutional layer, in which the depth of the feature map is reduced to the number of classes, creating a multi-channel output, that is, segmentation  $S$  where each pixel value shows the probability of belonging to one of the target segments of class label  $y$  using the softmax activation. Therefore, this would be suitable for multi-class segmentation tasks where each pixel is classified as belonging to one of the possible segments.

## II-E. Independent CNN-based Classifier for Evaluating Synthetic Images through classification

This section details the architecture and design for the convolutional neural network-based classifier. In this study, attention is paid to the base model of VGG16 on top of which custom layers are added for the development of a classifier that has the goal of performing image classification tasks on a dataset containing four classes of tumor. The classifier is developed with the purpose to categorize the real and synthetic images separately to verify the accuracy and realism of the synthetic images generated by ACGAN. The goal here is to evaluate the classifier performance based on several classification parameters like accuracy, precision, recall, and F1-score. To understand the classifier's prediction patterns and spot errors, the confusion matrix is also examined. If the classifier performs similarly on both real and synthetic images, it indicates that the synthetic images have successfully captured the key features of the real images. High recall and precision for synthetic images indicate that the classifier is consistently recognizing the correct class labels and classifying the images appropriately. A high F1-score indicates that synthetic images closely mimic real images.

### II-E1. Base model: VGG16

VGG16 is a convolutional neural network comprising 16 weight layers: 13 convolutional layers and 3 dense layers. The network attains a good performance in the learning of complicated features while keeping the parameter size relatively small by using very small ( $3 \times 3$ ) convolution filters. Its depth of 13 convolutional layers allows this network to build a hierarchical representation of the input data, which is very important in attaining high performance in image recognition tasks. The hidden layers in VGG16 use the rectified linear unit (ReLU) activation function. The training of VGG16 involves using stochastic gradient descent (SGD) with a mini-batch size of 256 and momentum of 0.9. Since VGG16 wants input images with 3 channel dimensions, an extra dimension axis was added to the input 128x128x1 images to form an input size of 128x128x3 [29] [30].

### II-E2. Custom layer on top of VGG16

Several custom layers are added on top to adapt the VGG16 base model for the classification task. Additional layers include dropout, dense layers, batch normalization, and activation functions as shown in Fig 5.

Two consecutive layers are added, each containing dropout, a dense layer, batch normalization, and LeakyReLU activation. The first layer contains a dense layer of 128 units, to learn 128 different features. The second layer will contain a dense layer of 32 units, learning 32 different features; all these are based on the output from the VGG16 base model. The output layer consists of 4 units, corresponding to the 4 classes in the

classification task and uses a softmax activation function is used to output a probability distribution over the classes. The complete model is created by connecting the input of the VGG16 base model to the custom layers and compiling it with sparse categorical crossentropy loss function and Adam optimizer.

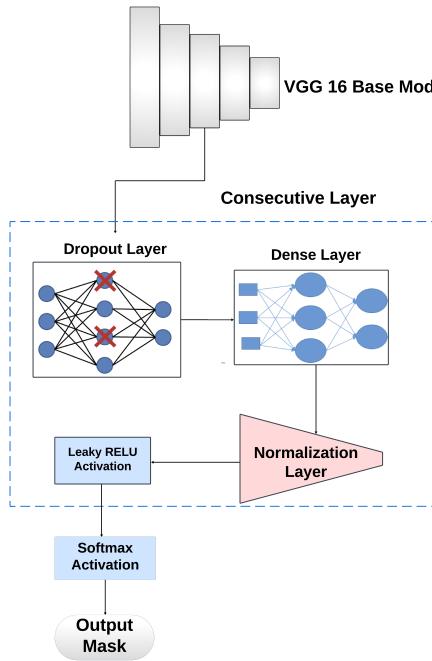


Figure 5. CNN based classifier to classify the real and generated brain tumor samples built on VGG16 base model and added custom layer.

### III. RESULTS

This section presents the findings and analysis of this particular study on brain tumor image synthesis using ACGAN and segmentation using U-Net architecture. The evaluation was based on the two major parts: the analysis of realism and precision of the synthetic images and the evaluation of accuracy with the segmentation maps. Finally, the loss graphs are analyzed to look into the way the ACGAN, U-Net, and the classifier are trained.

#### III-A. Realism and accuracy of Synthetic Images

To assess the realism and accuracy of synthesized brain tumor images generated using ACGAN, this study employs a CNN-based classifier. This classifier categorizes images into healthy tissue and three pathological variants: Glioma, Meningioma, and Pituitary tumors. Initially, the classifier's performance was evaluated using real images. Subsequently, its performance was assessed by combining both real and synthetic images. The classification results were compared using sensitivity, specificity, accuracy, precision, recall, and the F1 score.

Also, we sampled some real images and generated images in Fig 6 and Fig 7. It showcases the quality and similarity of synthetic images with contrast to real ones.

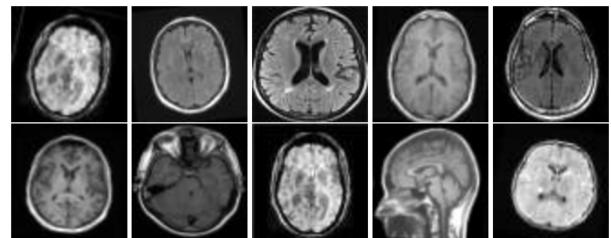


Figure 6. Sample Real Images.

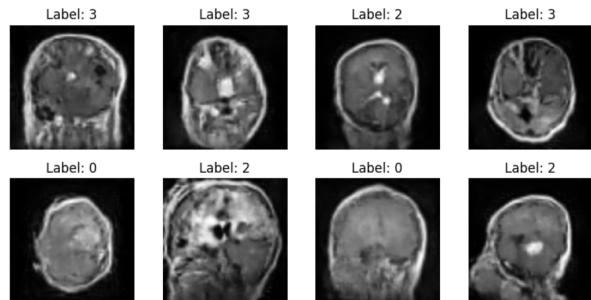


Figure 7. Sample Synthetic Images.

#### III-A1. Classification with Real Images

Table 3 shows the performance for classification using real images only. That is to say, a classifier alone could realize high sensitivity, specificity, precision, recall, and F1 scores for every class in all four classes, thus proving its robust performance on real images with overall accuracy of 0.98.

Table 3. Classification Report for Real Images

Class	Sensitivity	Specificity	Precision	Recall	F1 Score
Healthy	0.99	0.998	0.99	0.99	0.99
Glioma	0.96	0.99	0.97	0.96	0.96
Meningitis	0.97	0.98	0.96	0.97	0.96
Pituitary	0.99	0.996	0.99	0.99	0.99

#### III-A2. Classification with combined Real and Synthetic Images

Table 4 presents the performance of classification using combined real and synthetic images with overall accuracy of 0.84 performed over 5 fold cross validation ( $K = 5$ ). Obtained results shows that the classifier metrics shows a satisfactory result compared to when just only real images are used. Such similarity can indicate that the quality of the synthetic images generated by ACGAN is closer to the quality of the original real images.

The classifier training process can be assessed through the graphs of loss and accuracy as shown in Fig 8 and Fig 9 for both the training and the validation dataset. These charts demonstrated that the model performed

Table 4. Classification Report for Combined Real and Synthetic Images

Class	Sensitivity	Specificity	Precision	Recall	F1 Score
Healthy	0.77	0.92	0.73	0.77	0.75
Glioma	0.81	0.99	0.95	0.81	0.87
Meningitis	0.93	0.93	0.80	0.93	0.86
Pituitary	0.82	0.97	0.89	0.82	0.85

well and generalized effectively across both training and validation sets.

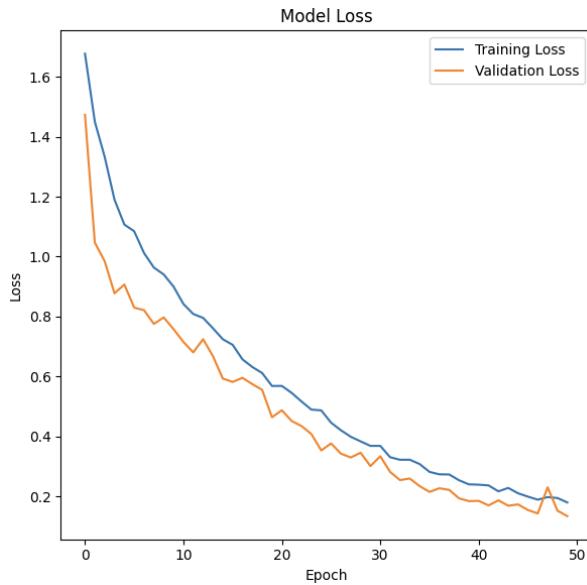


Figure 8. CNN Classifier loss over epochs across training and validation dataset.

In the case of ACGAN, training process was monitored for generator and discriminator for both adversarial and classification losses over the epoch. Fig 10 and Fig 11 demonstrates the loss over epoch for, both for the generator and discriminator.

The loss graphs indicated that both the adversarial and classification losses decreased and conversed overtime for generator and discriminator, suggesting stable and effective training of the ACGAN.

### III-B. Segmentation Accuracy

The Dice Coefficient and Intersection over Union (IoU) metrics were computed to evaluate the accuracy of the generated segmentation maps in identifying tumor regions. These metrics quantify the amount of overlap between the predicted segmentation maps and the ground-truth. Beside Dice Coefficient and IoU, the model's performance on the test set was evaluated using several critical metrics as well which is tabulated as in Table 5.

The training metrics over the training and validation sets as in Fig 12 and Fig 13 can be used to evaluate the

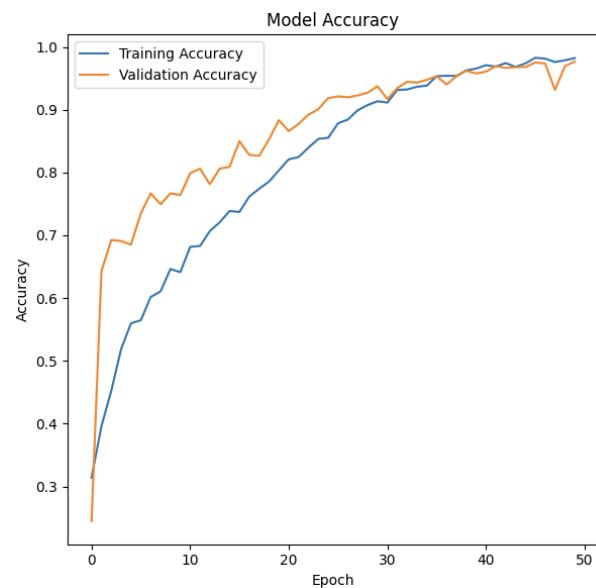


Figure 9. CNN Classifier accuracy over epochs across training and validation dataset.

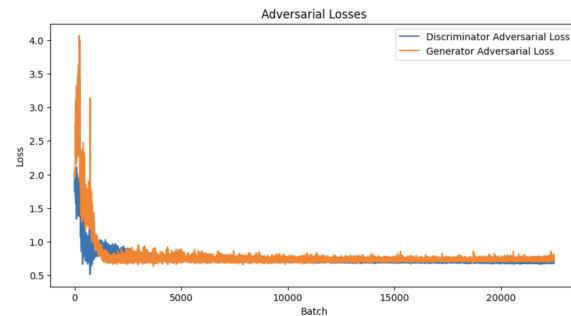


Figure 10. ACGAN Adversarial losses over epochs during training.

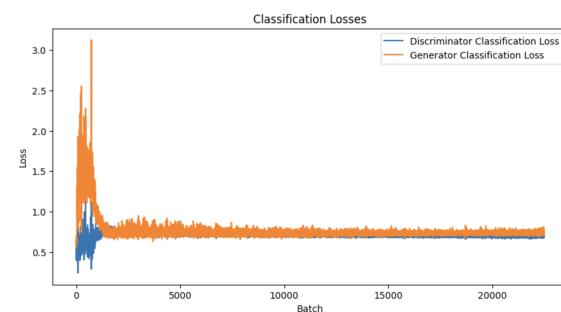


Figure 11. ACGAN Classification losses over epochs during training.

training process of the U-Net architecture. The model shows good performance over all the metrics: Accuracy, Loss, Dice Coefficient and MeanIOU. Also, the comparison between the predicted segmentation with respect to the ground truth segmentation is shown as in Fig 14

Table 5. Performance Metrics for U-Net Model on Test Set

Metric	Value
Loss	0.0142
Accuracy	99.54%
Mean Intersection over Union (MeanIOU)	92.91%
Dice Coefficient	76.43%
Precision	99.61%
Sensitivity (Recall)	99.49%
Specificity	99.87%

which clearly highlights the well performance of the architecture in predicting the tumor regions of multiple classes.

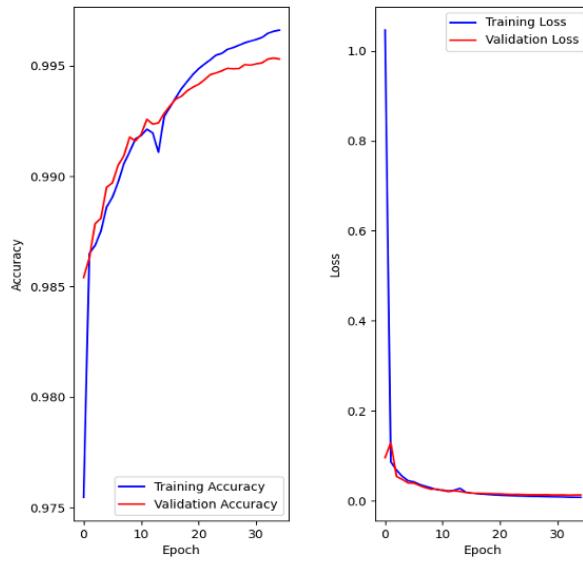


Figure 12. Accuracy and Loss graph over epochs on training and validation sets.

#### IV. DISCUSSION

The ACGAN model presented in this study works well in generating synthetic images of brain tumors to increase dataset variability, very important during the training of reliable AI models. The performance of an independent CNN-based classifier, testing real versus synthetic images, was also shown to be satisfactory, thus proving the good quality and realism of the generated images. In particular, it gave an overall accuracy of 0.84 when trained on combined real and synthetic images, comparable in accuracy to training using only real images. That is to say, the synthetic images represent the original images well with respect to features and classification performance.

Also, the performance of a U-Net architecture for the generation of an accurate segmentation map of brain tumors has been simply phenomenal. This result proves that the model is capable of handling the exact delimitation of the boundaries of the tumor, returning a

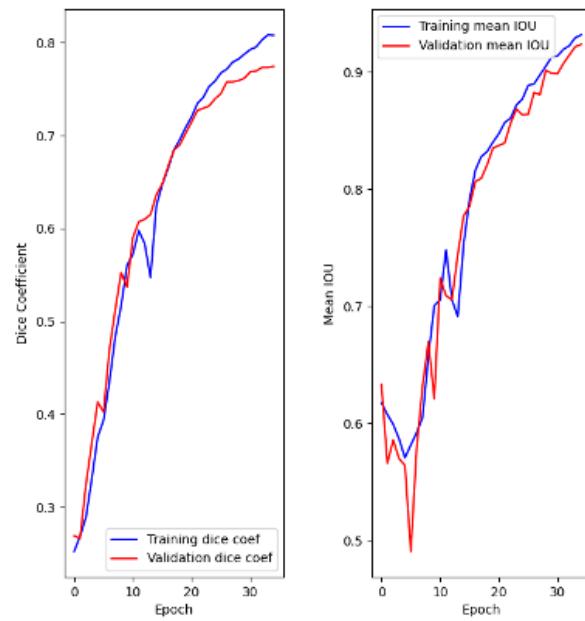


Figure 13. Dice Coefficient and MeanIoU graph over epochs on training and validation sets.

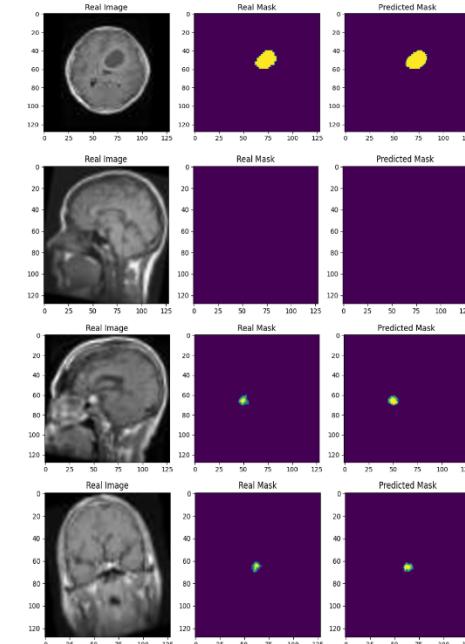


Figure 14. Image sample to compare the predicted segmentation over ground truth for multiple tumor classes.

Dice Coefficient of 76.43% and a MeanIOU of 92.91%. These indicators, when combined with high sensitivity, 99.49%, and specificity, 99.87%, have already proved the model for efficient identification and segmentation of tumor regions. However, there are several areas where further improvements can be made and new research directions can be explored.

- ACGANs can be further enhanced by incorporating dual classification for MRI views (axial, coronal, sagittal) and tumor types. By augmenting the dataset with these labels, we can train the ACGAN to generate synthetic images with both view and tumor type labels, ensuring consistency and improving the quality and accuracy of the generated images.
- Current approach can be enhanced to synthesize and segment images from multiple imaging modalities (e.g. CT, PET) to provide a more comprehensive analysis of brain tumors achieving multi modality image synthesis.
- Exploration with semi-supervised or unsupervised learning could be considered to make use of vast volumes of unlabeled data in addition to synthetic data during training for different varieties of GANs.

## V. CONCLUSIONS

The result presented in this work has proposed a better model that joins the strengths of Auxiliary Classifier Generative Adversarial Networks with U-Net architecture for advanced segmentation and classification of brain tumors. This approach uses synthetic images and segmentation masks of the brain tumor and tries to overcome the problems of data scarcity and ethical constraints. Results also showed a high realism and accuracy of the synthetic images: classifier accuracy was 0.84 while training both on real and synthetic data. The U-Net model achieved a Dice Coefficient of 76.43% with a MeanIOU of 92.91%, indicating very accurate tumor segmentation.

These results show that ACGAN with U-Net has great potential for augmented medical image analysis. This study therefore opens future avenues toward the exploration of dual classification, multi-modality synthesis, and advanced learning techniques to further improve AI models in clinical applications.

## ACKNOWLEDGMENTS

This work was supported by Department of Computational & Data Science, and Computer Science at the Middle Tennessee state University, Murfreesboro, TN, USA.

## REFERENCES

- [1] G. Thenuwara, J. Curtin, and F. Tian, "Advances in diagnostic tools and therapeutic approaches for gliomas: A comprehensive review," *Sensors*, vol. 23, no. 24, p. 9842, 2023.
- [2] M. Martucci, R. Russo, F. Schimperna, G. D'Apolito, M. Panfili, A. Grimaldi, A. Perna, A. M. Ferranti, G. Varcasia, C. Giordano *et al.*, "Magnetic resonance imaging of primary adult brain tumors: state of the art and future perspectives," *Biomedicines*, vol. 11, no. 2, p. 364, 2023.
- [3] D. Guo, X. Liu, D. Wang, X. Tang, and Y. Qin, "Development and clinical validation of deep learning for auto-diagnosis of supraspinatus tears," *Journal of Orthopaedic Surgery and Research*, vol. 18, no. 1, p. 426, 2023.
- [4] M. Frid-Adar, I. Diamant, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan, "Gan-based synthetic medical image augmentation for increased cnn performance in liver lesion classification," *Neurocomputing*, vol. 321, pp. 321–331, 2018.
- [5] J. Nalepa, M. Marcinkiewicz, and M. Kawulok, "Data augmentation for brain-tumor segmentation: a review," *Frontiers in computational neuroscience*, vol. 13, p. 83, 2019.
- [6] M. Park, M. Lee, and S. Yu, "Hrgan: A generative adversarial network producing higher-resolution images than training sets," *Sensors*, vol. 22, no. 4, p. 1435, 2022.
- [7] Y. Skandarani, P.-M. Jodoin, and A. Lalande, "Gans for medical image synthesis: An empirical study," *Journal of Imaging*, vol. 9, no. 3, p. 69, 2023.
- [8] H. Ali, M. R. Biswas, F. Mohsen, U. Shah, A. Alamgir, O. Mousa, and Z. Shah, "The role of generative adversarial networks in brain mri: a scoping review," *Insights into imaging*, vol. 13, no. 1, p. 98, 2022.
- [9] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.
- [10] C. Bowles, L. Chen, R. Guerrero, P. Bentley, R. Gunn, A. Hamers, D. A. Dickie, M. V. Hernández, J. Wardlaw, and D. Rueckert, "Gan augmentation: Augmenting training data using generative adversarial networks," *arXiv preprint arXiv:1810.10863*, 2018.
- [11] C. Han, H. Hayashi, L. Rundo, R. Araki, W. Shimoda, S. Muramatsu, Y. Furukawa, G. Mauri, and H. Nakayama, "Gan-based synthetic brain mr image generation," *2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018)*. IEEE, 2018, pp. 734–738.
- [12] S. Deepak and P. Ameer, "Msg-gan based synthesis of brain mri with meningioma for data augmentation," *2020 IEEE international conference on electronics, computing and communication technologies (CONECCT)*. IEEE, 2020, pp. 1–6.
- [13] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5–9, 2015, proceedings, part III 18*. Springer, 2015, pp. 234–241.
- [14] N. Siddique, S. Paheding, C. P. Elkin, and V. Devabhaktuni, "U-net and its variants for medical image segmentation: A review of theory and applications," *Ieee Access*, vol. 9, pp. 82 031–82 057, 2021.
- [15] H. Dong, G. Yang, F. Liu, Y. Mo, and Y. Guo, "Automatic brain tumor detection and segmentation using u-net based fully convolutional networks," *Medical Image Understanding and Analysis: 21st Annual Conference, MIUA 2017, Edinburgh, UK, July 11–13, 2017, Proceedings 21*. Springer, 2017, pp. 506–517.
- [16] F. Isensee, J. Petersen, A. Klein, D. Zimmerer, P. F. Jaeger, S. Kohl, J. Wasserthal, G. Koehler, T. Norajitra, S. Wirkert *et al.*, "nnu-net: Self-adapting framework for u-net-based medical image segmentation," *arXiv preprint arXiv:1809.10486*, 2018.
- [17] X. Chen, Y. Li, L. Yao, E. Adeli, and Y. Zhang, "Generative adversarial u-net for domain-free medical image augmentation," *arXiv preprint arXiv:2101.04793*, 2021.
- [18] S. Kazeminia, C. Baur, A. Kuijper, B. van Ginneken, N. Navab, S. Albarqouni, and A. Mukhopadhyay, "Gans for medical image analysis," *Artificial Intelligence in Medicine*, vol. 109, p. 101938, 2020.
- [19] Y. Jiang, H. Chen, M. Loew, and H. Ko, "Covid-19 ct image synthesis with a conditional generative adversarial network,"

- IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 2, pp. 441–452, 2020.
- [20] Z. Wang, Y. Lin, K.-T. T. Cheng, and X. Yang, “Semi-supervised mp-mri data synthesis with stitchlayer and auxiliary distance maximization,” *Medical image analysis*, vol. 59, p. 101565, 2020.
  - [21] K. Do, D. Nguyen, N. H. Tran, and V. D. Nguyen, “Pat: Pixel-wise adaptive training for long-tailed segmentation,” *arXiv preprint arXiv:2404.05393*, 2024.
  - [22] Y. Gou, Q. Wu, M. Li, B. Gong, and M. Han, “Segatngan: Text to image generation with segmentation attention,” *arXiv preprint arXiv:2005.12444*, 2020.
  - [23] A. K. Dubey and V. Jain, “Comparative study of convolution neural network’s relu and leaky-relu activation functions,” *Applications of Computing, Automation and Wireless Systems in Electrical Engineering: Proceedings of MARC 2018*. Springer, 2019, pp. 873–880.
  - [24] U. Ruby and V. Yendapalli, “Binary cross entropy with deep learning technique for image classification,” *Int. J. Adv. Trends Comput. Sci. Eng*, vol. 9, no. 10, 2020.
  - [25] H.-W. Dong and Y.-H. Yang, “Towards a deeper understanding of adversarial losses,” *arXiv preprint arXiv:1901.08753*, 2019.
  - [26] Z. Zhang and M. Sabuncu, “Generalized cross entropy loss for training deep neural networks with noisy labels,” *Advances in neural information processing systems*, vol. 31, 2018.
  - [27] S. Y. Şen and N. Özkar, “Convolutional neural network hyper-parameter tuning with adam optimizer for ecg classification,” *2020 innovations in intelligent systems and applications conference (ASYU)*. IEEE, 2020, pp. 1–6.
  - [28] A. H. Alharbi, C. Aravinda, M. Lin, P. Venugopala, P. Reddicherla, and M. A. Shah, “Segmentation and classification of white blood cells using the unet,” *Contrast Media & Molecular Imaging*, vol. 2022, no. 1, p. 5913905, 2022.
  - [29] K. Simonyan, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.
  - [30] S. Mascarenhas and M. Agarwal, “A comparison between vgg16, vgg19 and resnet50 architecture frameworks for image classification,” *2021 International conference on disruptive technologies for multi-disciplinary research and applications (CENTCON)*, vol. 1. IEEE, 2021, pp. 96–99.