Inception-UDet: An Improved U-Net Architecture for Brain Tumor Segmentation

Ilyasse Aboussaleh[1] · Jamal Riffi[1] · Adnane Mohamed Mahraz[1] ·
Hamid Tairi[1]

Abstract

Brain tumor segmentation is an important field and a sensitive task in tumor diagnosis. The treatment research in this area has helped specialists in detecting the tumor's location in order to deal with it in its early stages. Numerous methods based on deep learning, have been proposed, including the symmetric U-Net architectures, which revealed great results in the medical imaging field, precisely brain tumor segmentation. In this paper, we proposed an improved U-Net architecture called Inception U-Det inspired by U-Det. This work aims at employing the inception block instead of the convolution one used in the bi-directional feature pyramid neural (Bi-FPN) network during the skip connection U-Det phase. Furthermore, a comparison study has been performed between our proposed approach and the three known architectures in medical imaging segmentation; U-Net, DC-Unet, and U-Det. Several segmentation metrics have been computed and then taken into account in these methods, by means of the publicly available BraTS datasets. Thus, our obtained results have showed promising results in terms of accuracy, dice similarity coefficient (DSC), and intersection–union ratio (IOU). Moreover, the proposed method has achieved a DSC of 87.9%, 85.5%, and 83.9% on BraTS2020, BraTS2018, and BraTS2017, respectively, calculated from the best fold in fourfold cross-validation employed in the present approach.

Jamal Riffi, Adnane Mohamed and Hamid Tairi have contributed equally to this work.

✉ Ilyasse Aboussaleh
  aboussalehilyasse@gmail.com

  Jamal Riffi
  riffi.jamal@gmail.com

  Adnane Mohamed Mahraz
  adnane1@yahoo.fr

  Hamid Tairi
  hamid.tairi@usmba.ac.ma

[1] LISAC Laboratory, Department of Computer Science, Faculty of Sciences Dhar El Mahraz,
   University Sidi Mohamed Ben Abdellah, 30000 Fez, Morocco

1 Introduction

Data science is a combination of many disciplines which main purpose is extracting knowledge and information from data, both structured or not structured, using various technologies, tools, and methods. It mainly concerns the study of data analysis, statistics, and machine learning, to understand complex phenomena, make predictions and/or informed accurate decisions. Nowadays, data is used in an eminent increasing way. It is provided from various sources such as structured databases, unstructured text documents, sensor readings, medical laboratories, social media platforms, etc. In the meantime, Data mining is considered one of the remarkable subfields of data science. It aims at discovering patterns, relationships, and insights from large data sets. Its techniques provide valuable information and insights considered indispensable in decision-making, forecasting, and optimization. Notably, in [1], Shi et al. have presented a comprehensive and cutting-edge study on big data analysis, as they have introduced some essential skills to deal with real-world big data applications in solving some critical problems. Precisely, they have presented an optimization technique based on data mining. Actually, their work is focused on Support Vector Machines (SVMs) and different versions of Multiple Criteria Programming (MCP), not to mention the recent theoretical progress and real-life applications in various fields.

The importance of data mining is actually illustrated by discovering hidden patterns and relationships in complex big data. It is, in fact, used in many industries such as businesses [2], internet of things [3], banking, consultancy, manufacturing, healthcare, etc. This latter domain possesses several key applications in data science with a significant impact on medical imaging and diagnostic support. Diseases, pandemics and medical phenomena were rigorously treated thanks to automatic machine learning algorithms using many technologies and diagnostics tools. Coronavirus disease of 2019 (COVID-19), for instance, is the recent century pandemic; it appeared for the first time in Wuhan, China, in December 2019. Scientifically, the appearance of this disease has generated a significant number of scientific research articles, constantly increasing on a variety of related topics. Accordingly, Radanliev et al. [4] have utilized data mining and statistical analysis of the most developed countries, universities, and companies. In their study, the authors have used Web of Science data to examine the links between the findings of various scientific studies on COVID-19. On the other hand, the classification of positive and negative COVID-19 patients was genuinely a crucial task, therefore, it required an efficient resolution in order to achieve a high accuracy to, eventually, save people's lives. In this regard, Gada et al. [5] have proposed an approach based on Knuth-Morris-Pratt algorithm. Furthermore, they have analyzed the Covid-19 considered data; medical services and tests, pulse count, body temperature, and the overall effect of age and gender. The accuracy classification has achieved 97.4 %. In addition to COVID-19, cancer is one of the top ten deadliest diseases in the world. Brain and nervous system cancers include a variety of tumor types that can occur in the brain, spinal cord, and other parts of the central nervous

system. Recently, according to the global cancer observatory, over 241000 have been dying of these types of cancer in 2020, and this number keep on rising each year, for which the brain tumors researches have taken the edge on the priority levels in healthcare field. As a matter of fact, Brain tumors are the growth of abnormal cells in the brain. These tumors can be benign or malignant and can start in the brain or spread to the brain from other body parts. Depending on their size, location, and type, brain tumors can cause a range of symptoms and impair brain function, leading to disability or death. However, medical image analysis has helped patients and has saved lives by providing diagnoses using new and safe technologies such as positron emission tomography (PET), computed tomography (CT), and magnetic resonance imaging (MRI). The four MRI imaging modalities are T1-weighted, T2-weighted, T1-enhanced (T1c), and fluid-attenuated inversion recovery (FLAIR). Each pattern is presented as a two-dimensional slice. By putting all the slices together, we obtain a 3D structure of the brain. Semi-automatic and automatic methods have been proposed for various segmentation tasks, especially in brain tumor segmentation. Generally, manual segmentation of brain tumors is notoriously time-consuming, tedious, and error-prone. Therefore, a fully automated and accurate process is required. So far, several automated systems, that actually proved to be a remarkable success and reach accurate results, have been developed using deep learning. Convolution neural networks (CNNs) have been used excessively to get multi-object detection and medical image segmentation as the mentioned methods [6–9] taking the advantage of the automatic features representation of CNNs. Moreover, several deep learning approaches such as [10–12] have opted for auto encoders(AEs) structure, thus, demonstrated a power impact in the field. Inspired from CNNs and AEs networks, U-Net and its extension architectures have been introduced. The latter has, indeed, been an extreme push up in medical imaging segmentation, especially brain tumor segmentation.

In our paper, we have compared three architectures: U-Net, DC-Unet, and U-Det belonging to the U-Net class of approach, to our proposed method, similar to U-Det, called Inception-UDet. In our approach, we replace the convolution block used in U-Det, and precisely in Bidirectional Feature Pyramid Network with an inception block. The purpose of this modification is to enrich the features by applying several filters of different sizes. This block allows the reduction of the dimensions and the execution time helps avoiding the vanishing gradient problem, hence an improvement in the segmentation results.

This paper is organized as follows: in Sect. 2 we introduce some approaches that we used as related work. The proposed methods are presented and described in Sect. 3. In Sect. 4 figures thoroughly the experimental setup. We, then, summarize and discuss the results in Sect. 5. Finally, after the conclusion in Sect. 6, we cared to present our plans for future research.

2 Related Works

Deep learning remains the best path for medical image analysis in order to solve different problems of biomedical image segmentation namely brain tumors, liver, skin lesions, and vessels. U-Net and the U-Net-like architectures are well and truly the

most accurate and successful ones in this field, as they accomplished the majority of brain tumor segmentation tasks. Therefore, U-Net introduced by Ronneberger et al. [13] has employed a symmetric fully convolutional network that contains a contracting path to capture context information and an expanding path to ensure accurate location. Lou et al. [14] have developed an enhanced U-Net architecture called DC-Unet. They used a dual channel block in the encoder-decoder part and a modified skip connection called res-path. This method has been inspired by MultiResUNet architecture which employed MultiResidual block in the encoder-decoder part. Keetha et al. [15] were inspired from the encoder-decoder backbone of U-Net structure, and the feature-enriched Bi-FPN in the skip connection part, to eventually propose an End-to-End deep learning architecture called U-Det. They have also used a Mish activation function and class weights of masks to enhance model training and segmentation efficiency. On the other side, an Attention U-Net segmentation approach has been implemented by Oktay et al. [16], where the attention gate is incorporated into the standard U-Net to highlight salient features that are passed through the skip connections. In addition, Punn et al. [17] have proposed a residual cross-spatial attention-guided inception U-Net for tumor segmentation. They actually replaced the standard convolution with the inception convolution, and they used pooling operations with a hybrid pooling along with the cross-spatial attention filter on long skip connection, to focus on the most relevant features. Moreover, they have employed depth-wise separable convolution operation to minimize the training parameters and the number of multiplications. As for Pravitasari et al. [18], they have proposed a new model based on transfer learning called UNet-VGG16. They have exploited the pre-trained model VGG-16 [19] and fine-tuned it to be used in the segmentation task as a feature extractor in the encoder part. They have frozen the VGG-16 layers to reuse their weights. Based on the same technique of transfer learning and the attention mechanism, in a recent previous work, we have recently proposed, an efficient U-Net [20] architecture employing three different pre-trained models VGG-19 [19], ResNet50 [21] and MobileNetV2 [22] in the encoder part besides an attention decoder to segment different sub-region of brain tumor. Concerning Zhang et al.'s approach [23], they have investigated the effectiveness of a recently released attention model called attentional gate as a novel attention gate with the U-Net model, namely, AGResU-Net. It integrates residual models and attention gates with the original single U-Net architecture, where a series of attention gate units are added to skip connections and emphasize the information of salient features. On the other hand, a three dimensions U-Net-likes architecture has been employed by Cicek et al.'s work [24], where they have proposed the 3D Unet which is one of the earliest proposed 3D fully convolutional neural networks originally proposed for segmenting kidney embryos on Xenopus. Finally, Chen et al. [25] used the separable three dimensions in the encoder-decoder part of U-Net to get a novel framework named Separable 3D U-Net (S3D-Unet) for brain tumor segmentation.

Table 1 shows the performance results of the related works.

Table 1 Summary of related works' performance results

Method	Segmentation problem	Dataset	Accuracy	Performance indicator
U-Net	Lung Nodule	LUNA16	–	0.77 (DSC)
DC-Unet	Endoscopy images	CVC ClinicDB	0.80	–
DC-Unet	Electron microscopy	ISBI-2012	0.92	–
U-Det	Lung Nodule	LUNA16	–	0.82 (DSC)
AttU-Net	Pancreas	CT-150	–	0.84 (DSC)
RCA-IUnet	Tumor in BUSI	BUSI	0.97	0.89 (IoU)
UnetVGG16	Brain Tumor	Random	0.96	0.95 (CCR)
Efficient U-Net	Brain Tumor	BraTS2020	0.99	0.86 (DSC)
3D Unet	Prostate	STBPI	–	0.84 (DSC)
S3D-Unet	Brain Tumor	BraTS2018	–	0.83 (DSC)

"–" means "non", "DSC" means "Dice Similarity Coefficient", "CCR" means "Correct Classification ratio", "BUSI" means "Breast Ultrasound Image", "IoU" means "Intersection over Union", "STBPI" means "SmartTarget Biopsy, Index and the picture"
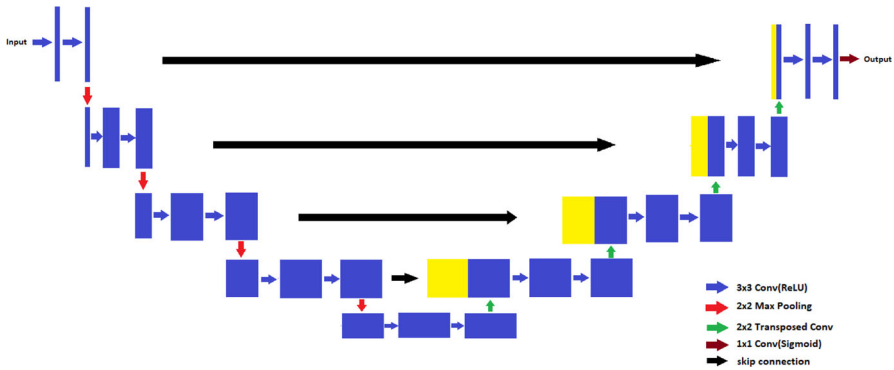
Fig. 1 Architecture of U-Net

3 Methods

In this section, we study three architectures: U-Net, DC-Unet, and U-Det, and apply them to the brain tumor segmentation's problem, then compare the obtained results with the proposed modified U-Det

3.1 U-Net

U-Net is a symmetric network that uses a convolution neural layer. It contains two parts: an encoder and a decoder. The encoder is a convolution network, with four convolution blocks repeated. Each one starts with two 3×3 convolution operations, followed by a max-pooling operation with a pooling size of 2×2 and a stride of 2. While each down-sampling, the number of filters in the convolution is doubled, and to connect the encoder to the decoder part, a progression of two 3×3 convolution operations is utilized. The decoder is the construction part of the segmentation map from the encoder feature. The decoder employs a 2×2 transposed convolution operation to up-sample the feature map and reduce simultaneity the feature channels to half. Then a sequence of two 3×3 convolution operations is performed once again. Similar to the encoder, these series of up-sampling and two convolution operations are repeated four times, reducing the number of filters at each stage by half. Finally, a 1×1 convolution operation is performed to generate the final segmentation map. As a fact, all convolutional layers in U-Net use ReLu (Rectified Linear Unit) as an activation function, except for the last layer which uses a 1×1 convolutional layer and sigmoid activation function. Furthermore, the U-Net architecture introduces skip connections to transfer the output from the encoder to the decoder. Whereas, these skip connections allow the network to retrieve spatial features, those lost during the pooling operation (Fig. 1).
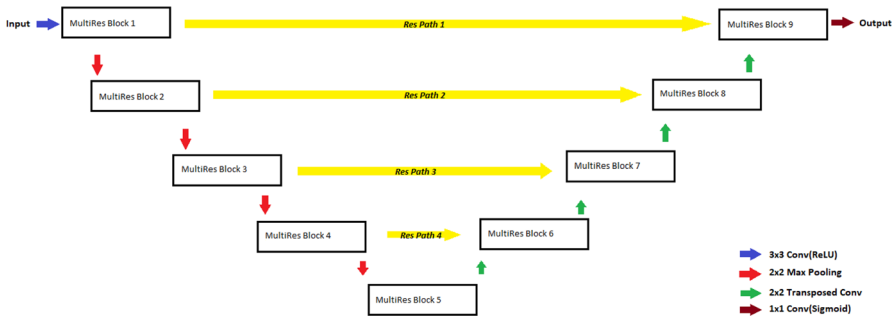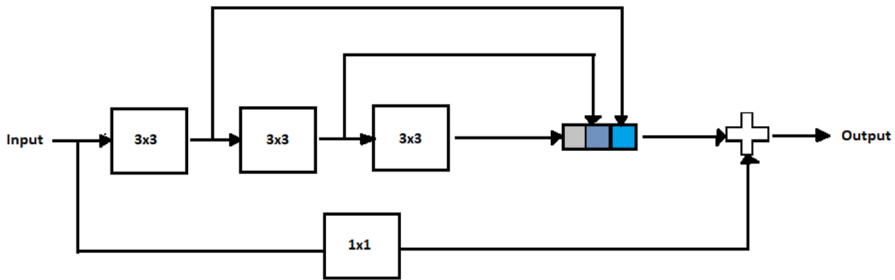
Fig. 2 Architecture of MultiResUNet

Fig. 3 The MultiRes Block employed in MultiResUNet architecture
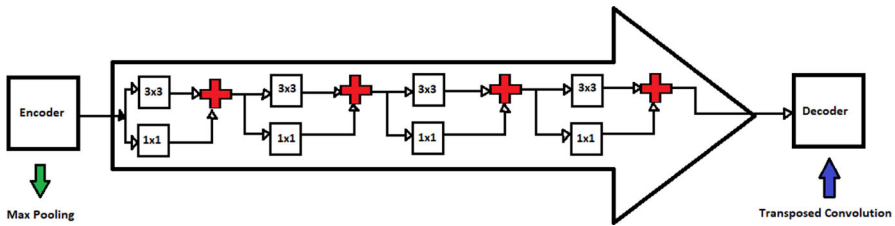

Fig. 4 Illustration of Res-Path

3.2 DC-Unet

DC-Unet is a modified and advanced version of MultiResUNet, therefore MultiResUNet Fig. 2 utilizes Multi-Res blocks Fig. 3 in the encoder and decoder part, those blocks are constructed by adding a residual connection to a succession of 3 × 3 filters of a simple version of inception. Moreover, they have made some modifications in the skip connection between the encoder and decoder called Res-Path Fig. 4. This path is a chain of 3 × 3 convolutional layer with residual connection, where each stage of MultiResUNet contains a precise number of 3 × 3 convolutional layer with a residual connection, starting with 4 and ending with 1 in the last stage. In this architecture, after each convolution layer, a nonlinear activation function ReLu is applied, in addition of a batch normalization, which is used to avoid overfitting, are employed. Then, in the last output layer, a sigmoid function is applied. Table 2 shows the details of the

Table 2 Details of MultiResUNet architecture

Block	Layer	#Filter	Path	Layer	#Filter
	Conv2D(3,3)	17		Conv2D(3,3)	64
MultiRes Block1	Conv2D(3,3)	35		Conv2D(1,1)	64
MultiRes Block9	Conv2D(3,3)	53		Conv2D(3,3)	64
	Conv2D(1,1)	105	Res Path1	Conv2D(1,1)	64
	Conv2D(3,3)	35		Conv2D(3,3)	64
MultiRes Block2	Conv2D(3,3)	71		Conv2D(1,1)	64
MultiRes Block8	Conv2D(3,3)	106		Conv2D(3,3)	64
	Conv2D(1,1)	212		Conv2D(1,1)	64
				Conv2D(3,3)	128
	Conv2D(3,3)	71		Conv2D(1,1)	128
MultiRes Block3	Conv2D(3,3)	142	Res Path2	Conv2D(3,3)	128
MultiRes Block7	Conv2D(3,3)	213		Conv2D(1,1)	128
	Conv2D(3,3)	426		Conv2D(3,3)	128
	Conv2D(3,3)	142		Conv2D(3,3)	256
MultiRes Block4	Conv2D(3,3)	284	Res Path3	Conv2D(1,1)	256
MultiRes Block6	Conv2D(3,3)	427		Conv2D(3,3)	256
	Conv2D(1,1)	853		Conv2D(1,1)	256
	Conv2D(3,3)	285			
MultiRes Block5	Conv2D(3,3)	56	Res Path4	Conv2D(3,3)	512
	Conv2D(3,3)	855		Conv2D(1,1)	512
	Conv2D(1,1)	1709			
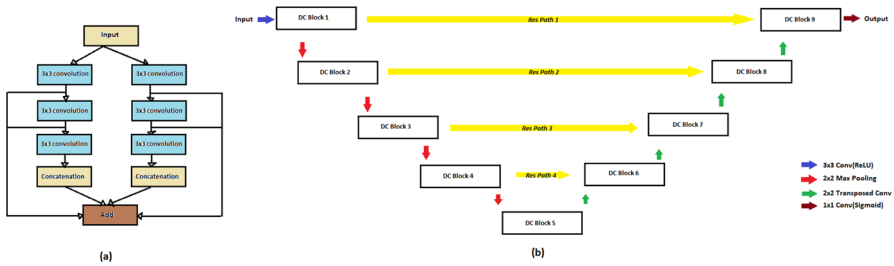
Fig. 5 **a** DC Block, **b** DC-Unet architecture

other parameters used in MultiResUNet architecture. The results illustrate that the residual connection used in MultiResUNet is simple and provides a few additional spatial features that may not be enough for the medical images segmentation task.

To overcome the insufficient spatial features, the authors of those previous architectures have suggested to replace the residual connection in MultiRes blocks with a sequence of three 3×3 convolutional layers to get a new extractor spatial features block named Dual Chanel block as shown in Fig. 5 on one hand. In addition, to save the same connection between encoder and decoder part Res-Path used in MultiResUNet

Table 3 Details of DC-Unet architecture

Block	Layer(left)	#Filter	Layer(right)	#Filter	Path	Layer	#Filter
	Cv2D(3,3)	8	Cv2D(3,3)	8		Cv2D(3,3)	32
DC Block1	Cv2D(3,3)	17	Cv2D(3,3)	17		Cv2D(1,1)	32
DC Block9	Cv2D(3,3)	26	Cv2D(3,3)	26		Cv2D(3,3)	32
					RP1	Cv2D(1,1)	32
	Cv2D(3,3)	17	Cv2D(3,3)	17		Cv2D(3,3)	32
DC Block2	Cv2D(3,3)	35	Cv2D(3,3)	35		Cv2D(1,1)	32
DC Block8	Cv2D(3,3)	53	Cv2D(3,3)	53		Cv2D(3,3)	32
						Cv2D(1,1)	32
						Cv2D(3,3)	64
	Cv2D(3,3)	35	Cv2D(3,3)	35		Cv2D(1,1)	64
DC Block3	Cv2D(3,3)	71	Cv2D(3,3)	71	RP2	Cv2D(3,3)	64
DC Block7	Cv2D(3,3)	106	Cv2D(3,3)	106		Cv2D(1,1)	64
						Cv2D(3,3)	64
						Cv2D(3,3)	64
	Cv2D(3,3)	71	Cv2D(3,3)	71		Cv2D(3,3)	128
DC Block4	Cv2D(3,3)	142	Cv2D(3,3)	142	RP3	Cv2D(1,1)	128
DC Block6	Cv2D(3,3)	213	Cv2D(3,3)	213		Cv2D(3,3)	128
						Cv2D(1,1)	128
	Cv2D(3,3)	142	Cv2D(3,3)	142			
DC Block5	Cv2D(3,3)	284	Cv2D(3,3)	284	RP4	Cv2D(3,3)	256
	Cv2D(3,3)	427	Cv2D(3,3)	427		Cv2D(1,1)	256

"#" means "number", "Cv2D" means "Conv2D", "RP" means "Res-Path"

besides constructing a new U-Net architecture named DC-Unet architecture on the other hand. Figure 5 In Table 3, details of DC-Unet architecture are explicitly figured.

3.3 U-Det

U-Det Fig. 7 is an end-to-end deep learning approach that incorporates a bidirectional feature network (Bi-FPN) between the encoder and decoder to enhance integrating multi-scale feature fusion for efficient feature extraction. Furthermore, it employs the Mish activation function and the class weights of masks to improve the segmentation precision.

Bi-FPN is based on the traditional top-down Feature Pyramid Network Fig (FPN) method [26]. It brings efficient bidirectional cross-scale connections and weighted feature fusion to the model [27]. Meanwhile, multi-scale feature fusion aims at fuse features of different resolutions for efficient feature extraction. The unidirectional flow of information inherently limits traditional top-down FPN. Furthermore, BiFPN does not contain nodes with only one input edge. If a node has only one input and no feature fusion, its contribution to the feature network designed to inject different features will be less relevant. It is based on the traditional top-down FPN (Feature

Fig. 6 Architecture of mish function
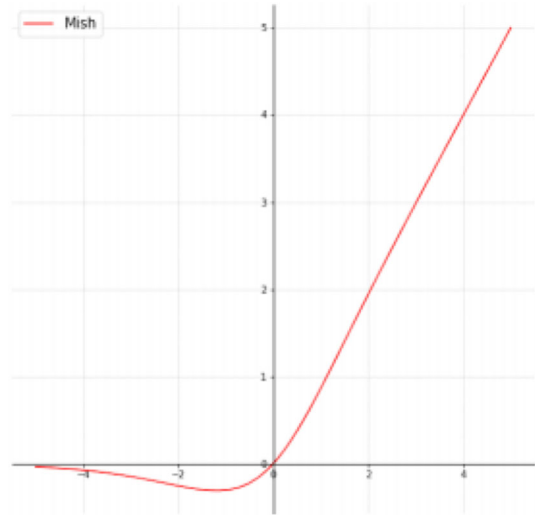
Pyramid Network) method [26]. Bi-FPN also integrates additional weights for each input during feature fusion, allowing the network to learn the importance of specific input features. Fast normalized fusion (one of the methods to include weights during feature fusion) is used for dynamic learning. Furthermore, to improve the model's efficiency, depthwise separable convolution is implemented, followed by batch normalization and nonlinear activation function Rectified Linear Unit(ReLu). In neural networks, activation functions are the gateway to introducing nonlinearities. Its role on training and evaluating deep neural networks is paramount. Activation functions that are most employed are ReLU, Sigmoid, Leaky ReLU, Tan Hyperbolic, and the recently introduced Swish. The proposed method implements the state-of-the-art activation function Mish. In accordance with the carried out results, Mish outperforms ReLU and Swish. Not to mention its simplicity, that allows smooth implementation of neural networks programs. Mish is a non-monotonic and smooth neural network activation function. It is defined as:

$$f(x) = x.tanh(\omega(x)) \tag{1}$$

where $\omega(x)$ is the softplus activation function given by $\ln(1+\exp(x))$ Fig. 6 illustrates the plot of the Mish activation function.

We have exploited the power of the two contributions of the U-Det method, Mish function and Bi-FPN, in order to develop a more accurate one. We, thus, have succeeded on creating a new architecture similar to U-Det based on the inception blocks in the encoder-decoder part. We, then, took the liberty on calling it "Inception U-Det", explicitly addressed in the next subsection (Fig. 7).
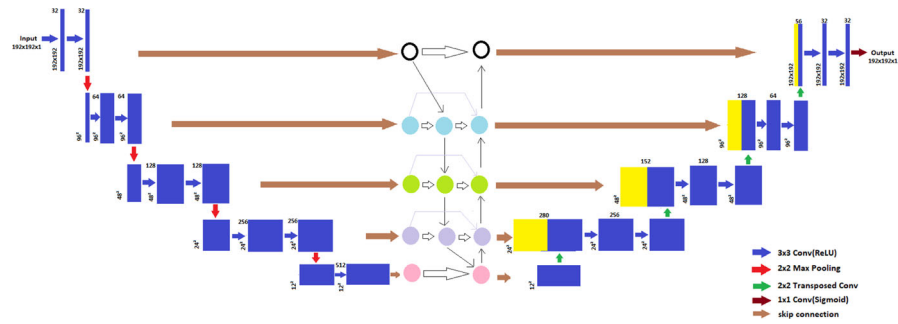
Fig. 7 Architecture of U-Det

3.4 Inception-UDet

Similarly to the encoder and decoder part of U-Net architecture, U-Det architecture consists of two phases: the contraction and the expansion. The contraction path is a simple convolution neural network that contains a repetition of two 3×3 convolutions (with padding='same'), followed by a non-linear Mish activation function and 2×2 max-pooling operation of stride 2 for the downsampling of the input image features. At each downsampling step, the number of features is doubled and the model is regularized by employing a Dropout layer with factor 0.5, after the second 3×3 convolution block at depth 4. The size of features corresponding to each section of the five depths of the contraction is $192 \times 192 \times 64$, $96 \times 96 \times 128$, $48 \times 48 \times 256$, $24 \times 24 \times 512$, $12 \times 12 \times 1024$ where 64, 128, 256, 512 and 1024 are the number of features channels. The convolution operation used at each layer of the model is formulated as:

$$C[m, n] = (I \times k)[m, n] = \sum_i \sum_j k[i, j].I[m - i, n - j] \tag{2}$$

$$Z^{[l]} = W^{[l]}.A^{[l-1]} + b^{[l]} \tag{3}$$

$$A^{[l]} = f^{[l]}(Z^{[l]}) \tag{4}$$

where Eq. 1 represents the kernel convolution and Eqs. 3 and 4 denote the forward process in CNN. In Eq. 2, I and k indicate the input image and the kernel respectively. $A^{[l]}$, $W^{[l]}$, $b^{[l]}$ and $f^{[l]}$ denote the activations, weights, bias, and activation function of layer l respectively.

The features of five depths are input to the feature network (Bi-FPN), and the output feature vector is input to the expansion part. Each step in the expansion path contains upsampling of subsequent feature maps doubles the number of feature channels per depth by 2×2 convolution ("upconvolution"). The resulting feature vectors after upsampling are then concatenated with the corresponding feature vectors from the feature network. The concatenation operation is followed by two 3×3 convolutions ("equal" padding), each followed by a Mish activation function. In the last layer of the network, the feature map of $192 \times 192 \times 64$ is obtained by traversal of Two 3×3
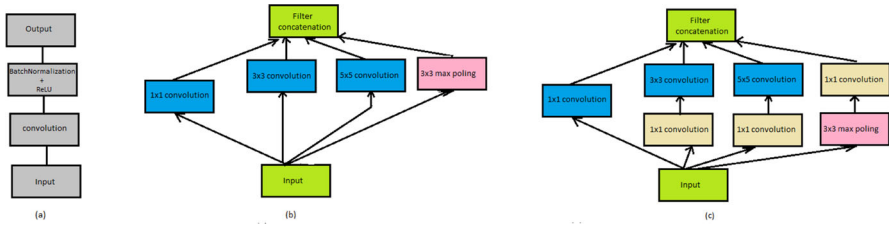
Fig. 8 **a** Convolution block used in U-Det, **b** inception module, naïve version, **c** inception block with dimension reduction used in Inception-UDet

circles. Next is the mish activation function and the final 1×1 convolutional block, and finally the sigmoid activation function. This gives the logits that match with the mask of the input MRI image of shape 192×192. Network training aims to increase the probability of the correct class for each voxel in the mask. To achieve this, a weighted binary cross-entropy loss for each training sample is used. This function is formulated as:

$$Loss = -\frac{1}{output\ size} \sum_{i}^{output\ size} y_i . \log \hat{y}_i + (1 - y_i) . \log(1 - \hat{y}_i) \qquad (5)$$

where $\hat{y}_i$ is the i-th scalar value in the model output, y_i is the corresponding target value, and the output size is the number of scalar values in the model output. To minimize this loss function, we used Adam optimizer with an initial learning rate of $\alpha_0 = 10^{-15}$ and progressively decreased it according to:

$$\alpha = \alpha_0 \times \left(1 - \frac{e}{N_e}\right)^{0.9} \qquad (6)$$

where e is an epoch counter, and N_e is the total number of epochs. In our case the maximum number of epochs $= 200$ and in every epoch, the batch size $= 10$.

Our method "Inception-UDet" Fig. 9 is similar to UDet. This novel architecture's work consists in replacing the convolution block with the Inception one to get pertinent and significant features. The inception block illustrated in Fig. 8, which process is, from one hand, the contraction path's output features are, in fact, the Bi-FPN's input, then the output features of Bi-FPN are generally the input of the expansion part, on the other hand.

Inception Models are used in Convolution Neural Networks to achieve more efficient computation and deeper networks by stacking 11 convolutions. Since neural networks process a large number of images, the presented image content varies widely, also known as salient parts, and they must be designed properly. By building a CNN to perform its folding at the same level, the network gradually becomes wider, not deeper. To make the process less computationally intensive, the neural network can be designed to add an extra 1×1 convolution before the 3×3 and 5×5 layers. This limits the number of input channels and makes 1×1 convolutions much cheaper
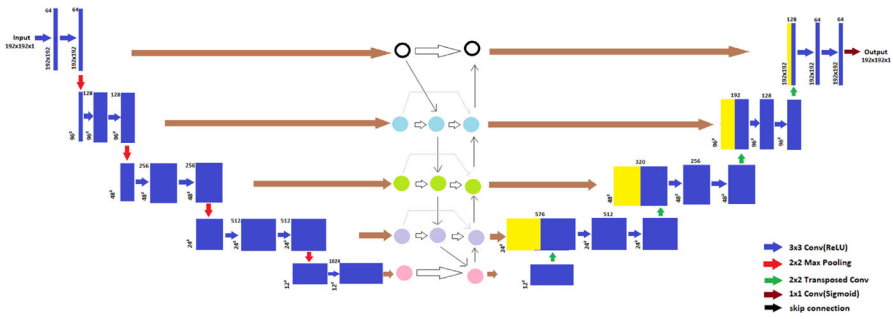
Fig. 9 Inception-UDet Architecture

Table 4 Number of kernels for each Inception block

Depth	Input size	Number of kernels of inception block
1	$192 \times 192 \times 64$	(192, 96, 208, 16, 48, 64)
2	$96 \times 96 \times 128$	(160, 112, 224, 24, 64, 64)
3	$48 \times 48 \times 256$	(128, 128, 256, 24, 64, 64)
4	$24 \times 24 \times 512$	(160, 112, 224, 24, 64, 64)
5	$12 \times 12 \times 1024$	(192, 96, 208, 16, 48, 64)

than 5×5 convolutions (Fig. 9). At each depth, an inception block is applied before entering the features to Bi-FPN, and the number of kernels at each one is shown in Table 4 below: We summarize all the U-net like architectures analyzed in this paper including our proposed one Inception-UDet, in Fig. 10, and show all the blocks and operations used for each approach.

4 Data and Experiment

In this section, we present data, implementation data, and evaluation metrics.

4.1 Data

BraTS2020 [28–30] contest provides a large training set of 369 MRI scans and a validation set of 125 scans. The BraTS2018 dataset consists of 285 training scans (HGGs and LGGs) and 66 scans, while the BraTs2017 contains 285 training scans(HGGs and LGGs), 46 validation scans, and 146 test scans. Each MRI scan was $240 \times 240 \times 155$ in size, and each case had FLAIR, T1, T1 extension, and T2 volumes. The dataset is co-registered, re-sampled to $1 \times 1 \times 1 \, mm^3$, and skull-stripped. Segment brain tumors, including necrosis, edema, non-enhancing, and enhancing tumors. The ground truth of the training set is only obtained by manual segmentation results given by experts.
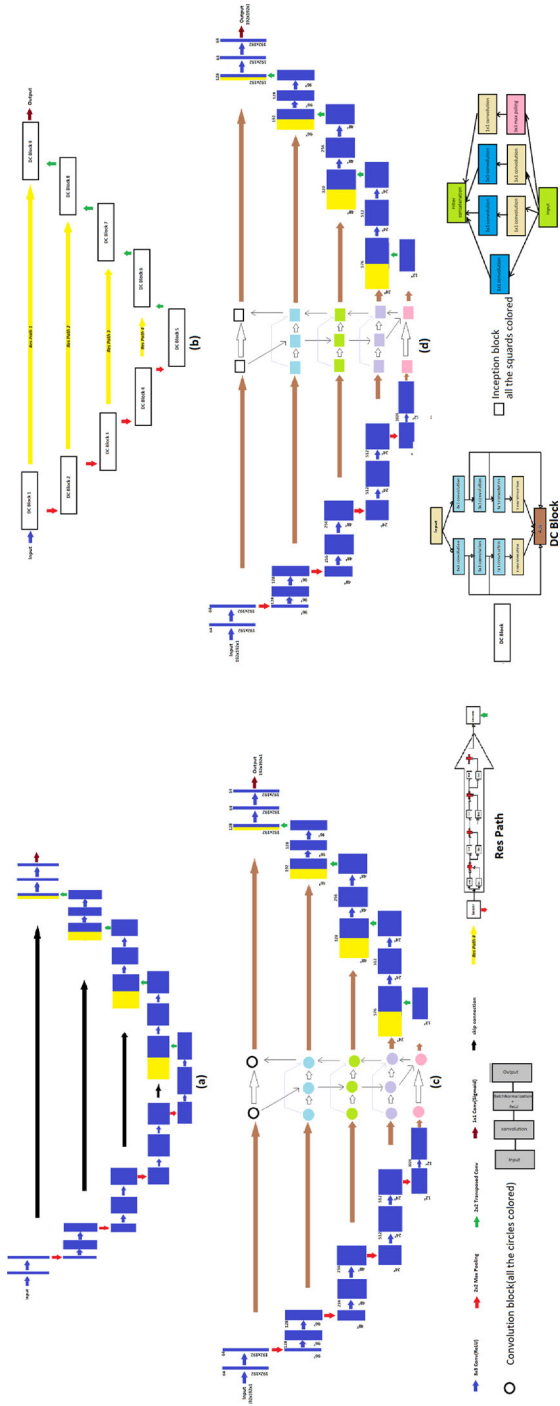
Fig. 10 Illustration of all the different methods, blocks, and operations used for each one. **a** U-Net, **b** DC-Unet, **c** U-Det, **d** Inception-UDet

4.1.1 Data Preprocessing

To make the features of the tumor more obvious, we normalize the input image i, to improve the accuracy of the segmentation, by subtracting the mean value μ and dividing by the standard deviation σ to get the output i_0 as:

$$i_0 = \frac{i - \mu}{\sigma} \tag{7}$$

4.1.2 Data Augmentation

Data augmentation improves network performance, reduces the occurrence of over-fitting, and generates more training data from the original data. Indeed, in this paper, we apply the augmentation methods with simple transformations such as flipping, rotating, adding noises, and translating.

4.2 Implementation Details

In this experiment, we used SIMPLTIK an open-source multi-dimensional image analysis in Python for image registration and segmentation to read MRI images from the BraTS2017, BraTS2018, and BraTS2020 datasets with NIFTI format type. Since we are interested in the segmentation of the whole tumor only in two dimensions, the best extension to choose is FLAIR one and the significant slice which contains more features is the 90th slice of 155. Before data preprocessing (Sect. 4.1.1) and data augmentation (Sect. 4.1.2) steps, we cropped each image and saved their size as (192, 192, 3) instead of (240, 240, 3). Furthermore, data augmentation is implemented in the three BraTS training set to improve the robustness of the model. An early termination training strategy is actually required to prevent the model from overfitting; that is when the model's performance stops improving. The training dataset was divided randomly into training and testing sets with 80:20 ratios, and a k-fold cross-validation was employed to get more performance. We have tested several numbers of fold ($k = 3, k = 4, k = 5, k = 10$) and eventually found that $k = 4$ was the best choice. The inception block kernels were initialized with a constant value equal to 0.2 and bias values set to zero. The best parameters to choose for all the methods are shown in Figs. 1, 7, 9 and Tables 3, 4. The experiment is implemented in the Kaggle platform using Keras (Version 2.6.0) library based on Tensorflow (Version 2.6.2) and Python (Version 3.7.12) as the used coding language. The experiment was carried out on the Kaggle platform on a virtual instance equipped with CPUs, 13GB memory, and an HDD drive of 73 GB. During the training of the model, acceleration was executed on Tesla (P100-PCIE-16GB) GPU (16GB video memory). Well, it took 6 h to converge.

4.3 Evaluation Metrics

The experimental results have been evaluated using different types of performance indicators: Accuracy, Dice Similarity Coefficient (DSC), and intersection over union (IoU) for tumor segmentation:

- Accuracy: Formally, accuracy has the following definition:

$$Accuracy = \frac{True positive + True Negative}{Total} \tag{8}$$

- The DSC represents the overlapping of predicted segmentation with the manually segmented output label and is computed as:

$$DSC = 2 \times \frac{|G \cap S|}{|G| + |S|} \tag{9}$$

where G and S stand for output label and predicted segmentation, respectively.
- The IoU is used when calculating Mean Average Precision (mAP). It specifies the amount of overlap between the predicted and ground truth and is computed as:

$$IoU = 2 \times \frac{Area\, of\, Overlap}{Area\, of\, Union} \tag{10}$$

5 Results and Discussion

This section covers the detailed results of all methods, their analysis, an experimental comparison, and a visualization.

The braTS2020 training dataset was applied to train all the methods cited in this manuscript, this dataset was divided randomly into two subsets: training and validation (80:20 ratios) and a fourfold cross-validation was implemented into this data. On the other hand, BraTS2018 and 2017 training datasets were utilized to perform our proposed model with the same number of folds. Tables 5 and 6 summarize the results obtained with and without data augmentations. Effective results are illustrated in Table 5 which concerns the training subset, where we denote the high accuracy reached by all methods in all the folds and an interesting difference in terms of other metrics; DSC and IoU. U-Det and Inception-UDet show a significant improvement over U-Net and maintain good robustness, however the other metrics decrease, which shows the effect of the Bi-FPN and the Mish activation function. Hence, the use of the Inception block in our method to extract more pertinent features in order to get high performance, is considered paramount. Our architecture compiled from the fourth fold has achieved 99.9%, 95.8% and 93.3% in terms of accuracy, DSC and IoU respectively, without data augmentation and 98.9%, 97.4% and 94% respectively with data-augmentation.

Table 6 presents the metrics of the validation subset, the best fold of our Inception-UDet method achieved 98.8%, 86.8% and 77.7% in terms of accuracy, DSC and IoU

Table 5 Whole tumor segmentation results with and without data augmentation per fold on the BraTS2020 training subset

| Method | f1 | | | | | | f2 | | | | | |
| | N-DA | | | DA | | | N-DA | | | DA | | |
	Acc	DSC	IoU	Acc	DSC	IoU	Acc	DSC	IoU	Acc	DSC	IoU
U-Net	95.4	70.9	59.3	96	72.8	60.5	96.2	71.1	62.7	97	74	66.1
DC-Unet	96	55.1	33.3	96.4	57.8	35.1	96.9	57.8	36.9	97.1	67.2	40.3
U-Det	97.8	79	70.5	98.5	80.6	71.7	97.6	77.3	68.2	98.7	81.8	70
I-UDet	98.1	85.1	82.9	98.9	85.7	85.8	98.5	88.8	86.3	99.1	89.2	87.6

| Method | f3 | | | | | | f4 | | | | | |
| | N-DA | | | DA | | | N-DA | | | DA | | |
	Acc	DSC	IoU	Acc	DSC	IoU	Acc	DSC	IoU	Acc	DSC	IoU
U-Net	99.1	77.8	67.3	99.5	83.8	71	97	84.8	68.5	98.3	86.2	72.3
DC-Unet	98.9	66.6	42.1	99	70.1	44.5	98.9	68.8	47.4	99.5	70.2	50.5
U-Det	97.8	84	72.5	98.9	85.8	75	99.5	85.1	76.7	99.8	87.9	78.1
I-UDet	99.9	94.9	92.9	97.9	96.4	93.9	99.9	95.8	93.3	98.9	97.4	94

"*fk*" means "fold *k* in k-fold cross-validation", "N-DA" means "Without data augmentation", "DA" means "With data augmentation", "I-UDet" means "Inception-UDet", "avg" means "average of all folds"

Table 6 Whole tumor segmentation results with and without data augmentation per fold on the BraTS2020 validation subset

	f1									f2		
	NDA			DA			NDA			DA		
Method	Acc	DSC	IoU	Acc	DSC	IoU	Acc	DSC	IoU	Acc	DSC	IoU
U-Net	96.5	65.1	60.1	97.2	68.1	61.9	96.4	69.2	61.1	98.3	71.6	63.8
DC-Unet	99.1	14.9	20.7	99.3	25.8	15	97.6	35.9	22.9	99.4	39.7	24.8
U-Det	97.2	74	66.8	97.7	76.1	67.1	97.6	75.4	65.5	98.8	77.2	67.9
I-UDet	98	78.9	65.2	98.9	80	66.7	97.8	82.6	70.4	98.9	84.1	72.5

	f3									f4		
	NDA			DA			NDA			DA		
Method	Acc	DSC	IoU	Acc	DSC	IoU	Acc	DSC	IoU	Acc	DSC	IoU
U-Net	99.9	81.5	69.2	99.9	82.1	73.8	99.6	79.5	70.1	99.8	80.4	71.9
DC-Unet	99.3	46.7	42.4	99.2	57.5	39.2	98.3	51.9	37.9	99.4	58.8	42
U-Det	98.9	84.8	75.9	99.3	86.6	77.6	99.1	80.9	74.9	99.7	82	75.9
I-UDet	98.3	84.9	75.3	99.1	86.4	76.1	98.8	86.8	77.7	99.3	87.9	78.4

"fk" means "fold k in k-fold cross-validation", "N-DA" means "Without data augmentation"DA" means "With data augmentation", "I-UDet" means "Inception-UDet", "avg" means "average of all folds"

Table 7 DSC results of our method on BraTS2017 and BraTS2018 validation datasets per fold

Data	Fold 1	Fold 2	Fold 3	Fold 4	Avg
BraTS2017	75.8	79.9	83.9	76.2	78.9
BraTS2018	79.5	80	84	85.5	82.4

"avg" means "average of all folds"

Table 8 Comparison study of a whole tumor performance between our proposed method and different supervised and non-supervised approaches on different BraTS datasets

| Methods	Data	Performance
Aboussaleh et al. [9]	BraTS2017	DSC 82.35 %
Single-Path MLDeepMedic [31]	BraTS2017	DSC 79.73%
ResNets [32]	BraTS2017	DSC 85%
Myronrnko et al. [10]	BraTS2018	DSC 81%
Fang et al. [33]	BraTS2018	DSC 85.60%
Chen et al. [25]	BraTS2018	DSC 83.60%
AGResU-Net [23]	BraTS2019	DSC 87%
Efficient U-Net [20]	BraTS2020	DSC 86.89%
Inception-UDet (best fold performance)	BraTS2020	DSC 87.9%
	BraTS2018	DSC 85.5%
	BraTS2017	DSC 83.9%

respectively without data-augmentation and 99.3%, 87.9% and 78.4% respectively with data-augmentation.

By observation of all the results, our method is proved more performant than UDet in a close manner, which justifies the crucial impact of using the Inception block instead of the convolution one that UDet has employed.

Data augmentation and the k-fold cross-validation are still helpful and useful to improve the performance, as shown in the table of results, where all the results are increasing.

Table 7 demonstrates that fold 3 and fold 4 of the cross-validation used in BraTS 2017 and 2018 respectively have achieved a great DSC of 83.9% and 85.5%.

Table 8 demonstrates a comparison study between our proposed method and some of the state of art methods. Therefore, the U-Net-like architecture [20, 23, 25, 33] in fact, tops the rankings. It conferred a high performance in terms of DSC on the brain tumor segmented. Besides, the attention mechanism used in methods [20, 23] has showed a significant effect on the results. All of these methods have exceeded 86% and been close to the top score that we have obtained. It is obvious that our method overcomes all the unsupervised ones [9, 31], and the approach belongs to the FCNNs [32] and AEs [10]. Additionally, we notice that the obtained results based on BraTS2020 using our proposed approach skip all the methods cited in terms of DSC, and thus, get an acceptable score employing the other version of the BraTS dataset (2017 and 2018). This can only ensure the performance and the impact of our contributions.
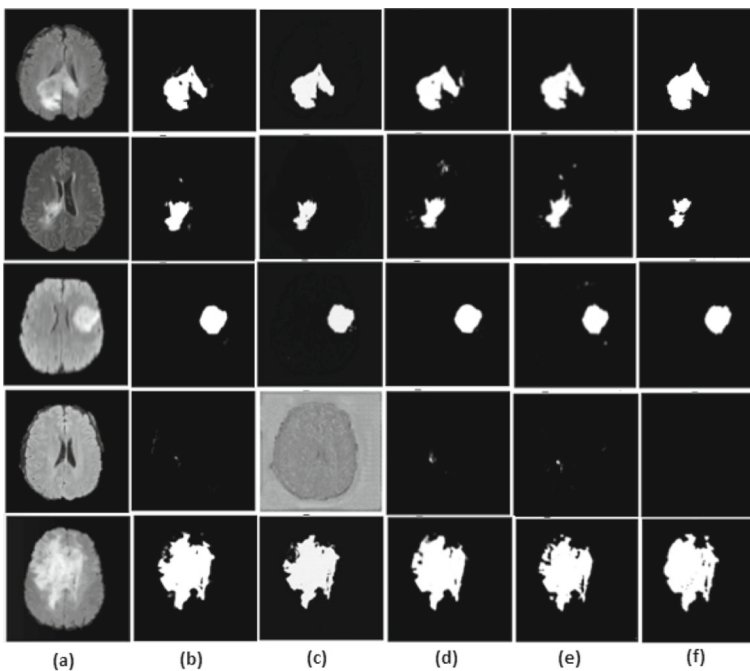
Fig. 11 Visual results of stat of art methods on some BraTS2020 validation subset images. **a** original image, **b** U-Net, **c** DC-Unet, **d** U-Det, **e** Inception-UDet, **f** Ground Truth
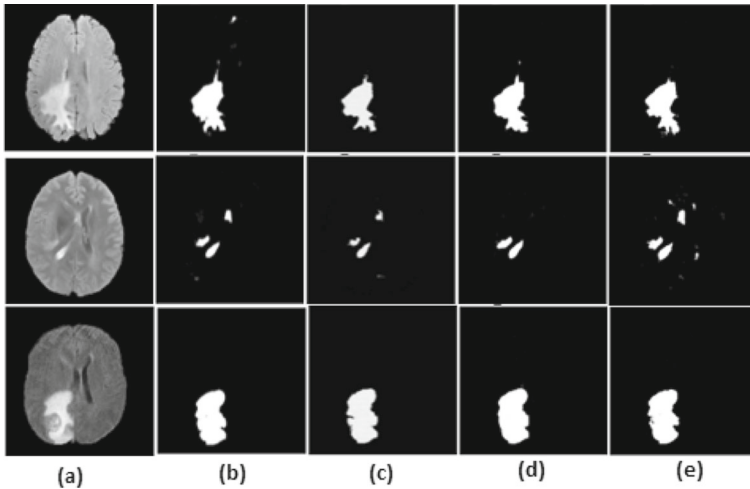
Fig. 12 Visual results of stat of art methods on some BraTS2020 validation set images. **a** original image, **b** U-Net, **c** DC-Unet, **d** U-Det, **e** Inception-UDet

Figure 11 illustrates the qualitative results of the tumor on some images from the BraTS2020 validation subset of the proposed architectures Unet, DC-Unet, U-Det, and Inception-UDet. Our method shows good performance compared to the others, it almost reaches the true label. Although, the DC-Unet method cannot assign clear segmentation to non-tumor images. Consequently, its performance keep on dropping. On the other hand, U-Net and UDet neglect some pixels core of the tumor.

The visual results of some Brats2020 validation sets are summarized in Fig. 12. All the methods show an efficient performance in some different levels, particularly in the second sample that proves the power of our method.

6 Conclusion

Above all, this work represent a concrete proof of existence of a tight link between the improvement of scientific research in medicine and data mining. Precisely, Brain tumor segmentation is a dainty field, that, through and through, requires a rigorous kind of treatment. Therefore, in this manuscript, we meticulously tried to improve the Brain tumor segmentation architectures, by introducing an improved U-Net architecture with an Inception block for brain tumor segmentation. Our model's structure consists concisely in the following steps: as after the pre-processing and data-augmentation phases, our proposed method keeps the structure of U-Det and changes the convolution block by the inception one in each depth in order to get more features. This modification genuinely increases the evaluation metrics. As a matter of fact, we have trained and evaluated our model on BraTS2020, BraTS2018, and BraTS2017 datasets using ground truths (extracted by medical experts). Afterwards, we have compared our results with the state of the art works mentioned earlier. Eventually, the experimental results concretely demonstrate the high capacity and performance of our architecture

in segmentation tasks. Last but not least, our perspective work will focus on improving these previous results, segmenting the other type of tumor core and enhancing edema, as well as using deeper architectures to improve the performance of segmentation outputs.

Author Contributions IA and JR are the contributors, IA is the coder and the writer, JR, MAM, and HT are the reviewers.

Data Availability Data is available on the Kaggle platform.

Code Availability Not applicable.

Declarations

Conflicts of interest The authors declare they have no financial interests.

Ethical Approval All authors approved this article.

Consent to Participate All the authors Consent to participate.

Consent for Publication All the authors Consent for publication.

References

1. Shi Y (2022) Advances in big data analytics. Springer, Singapore
2. Olson DL, Shi Y, Shi Y (2007) Introduction to business data mining, vol 10. McGraw-Hill/Irwin, New York
3. Tien JM (2017) Internet of things, real-time decision making, and artificial intelligence. Ann Data Sci 4:149–178
4. Radanliev P, De Roure D, Walton R, Van Kleek M, Santos O, Maddox L (2022) What country, university, or research institute, performed the best on Covid-19 during the first wave of the pandemic? bibliometric analysis of scientific literature-analysing a 'snapshot in time'of the first wave of Covid-19. Ann Data Sci 9(5):1049–1067
5. Gada V, Shegaonkar M, Inamdar M, Dinesh S, Sapariya D, Konde V, Warang M, Mehendale N (2022) Data analysis of Covid-19 hospital records using contextual patient classification system. Ann Data Sci 9(5):945–965
6. Pereira S, Pinto A, Alves V, Silva CA (2016) Brain tumor segmentation using convolutional neural networks in MRI images. IEEE Trans Med Imaging 35(5):1240–1251
7. Iqbal S, Ghani MU, Saba T, Rehman A (2018) Brain tumor segmentation in multi-spectral MRI using convolutional neural networks (CNN). Microsc Res Tech 81(4):419–427
8. Wang G, Li W, Ourselin S, Vercauteren T (2018) Automatic brain tumor segmentation using cascaded anisotropic convolutional neural networks. In: Brainlesion: glioma, multiple sclerosis, stroke and traumatic brain injuries: third international workshop, BrainLes 2017, Held in Conjunction with MICCAI 2017, Quebec City, QC, Canada, September 14, 2017, Revised Selected Papers 3, Springer, pp 178–190
9. Aboussaleh I, Riffi J, Mahraz AM, Tairi H (2021) Brain tumor segmentation based on deep learning's feature representation. Journal of Imaging 7(12):269
10. Myronenko A (2019) 3D MRI brain tumor segmentation using autoencoder regularization. In: Brainlesion: glioma, multiple sclerosis, stroke and traumatic brain injuries: 4th international workshop, BrainLes 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, Revised Selected Papers, Part II 4, Springer, pp 311–320
11. Vaidhya K, Thirunavukkarasu S, Alex V, Krishnamurthi G (2016) Multi-modal brain tumor segmentation using stacked denoising autoencoders. In: Brainlesion: glioma, multiple sclerosis, stroke and

traumatic brain injuries: first international workshop, Brainles 2015, Held in Conjunction with MICCAI 2015, Munich, Germany, October 5, 2015, Revised Selected Papers 1, Springer, pp 181–194

12. Amin J, Sharif M, Gul N, Raza M, Anjum MA, Nisar MW, Bukhari SAC (2020) Brain tumor detection by using stacked autoencoders in deep learning. J Med Syst 44:1–12

13. Ronneberger O, Fischer P, Brox T (2015) U-Net: convolutional networks for biomedical image segmentation. In: Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5–9, 2015, Proceedings, Part III 18, Springer, pp 234–241

14. Lou A, Guan S, Loew M (2021) DC-UNet: rethinking the U-Net architecture with dual channel efficient CNN for medical image segmentation. In: Medical imaging 2021: image processing, vol 11596. SPIE, pp 758–768

15. Keetha NV, Annavarapu CSR et al (2020) U-Det: A modified U-Net architecture with bidirectional feature network for lung nodule segmentation. arXiv preprint arXiv:2003.09293

16. Oktay O, Schlemper J, Folgoc LL, Lee M, Heinrich M, Misawa K, Mori K, McDonagh S, Hammerla NY, Kainz B et al (2018) Attention U-Net: learning where to look for the pancreas. arXiv preprint arXiv:1804.03999

17. Punn NS, Agarwal S (2022) RCA-IUnet: a residual cross-spatial attention-guided inception u-net model for tumor segmentation in breast ultrasound imaging. Mach Vis Appl 33(2):27

18. Pravitasari AA, Iriawan N, Almuhayar M, Azmi T, Irhamah I, Fithriasari K, Purnami SW, Ferriastuti W (2020) UNet-VGG16 with transfer learning for MRI-based brain tumor segmentation. TELKOMNIKA (Telecommun Comput Electron Control) 18(3):1310–1318

19. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556

20. Aboussaleh I, Riffi J, Fazazy KE, Mahraz MA, Tairi H (2023) Efficient U-Net architecture with multiple encoders and attention mechanism decoders for brain tumor segmentation. Diagnostics 13(5):872

21. Huang G, Liu Z, Van Der Maaten L, Weinberger KQ (2017) Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4700–4708

22. Sandler M, Howard A, Zhu M, Zhmoginov A, Chen L-C (2018) Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4510–4520

23. Zhang J, Jiang Z, Dong J, Hou Y, Liu B (2020) Attention gate resU-Net for automatic MRI brain tumor segmentation. IEEE Access 8:58533–58545

24. Çiçek Ö, Abdulkadir A, Lienkamp SS, Brox T, Ronneberger O (2016) 3d U-Net: learning dense volumetric segmentation from sparse annotation. In: Medical image computing and computer-assisted intervention–MICCAI 2016: 19th international conference, Athens, Greece, October 17–21, 2016, Proceedings, Part II 19, Springer, pp 424–432

25. Chen W, Liu B, Peng S, Sun J, Qiao X (2019) S3d-Unet: separable 3d U-Net for brain tumor segmentation. In: Brainlesion: glioma, multiple sclerosis, stroke and traumatic brain injuries: 4th international workshop, BrainLes 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, Revised Selected Papers, Part II 4, Springer, pp 358–368

26. Lin T-Y, Dollár P, Girshick R, He K, Hariharan B, Belongie S (2017) Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2117–2125

27. Tan M, Le Q (2019) Efficientnet: rethinking model scaling for convolutional neural networks. In: International conference on machine learning, PMLR, pp 6105–6114

28. Menze J, Masuch H, Bachert P et al (2009) A comparison of random forest and its Gini importance with standard chemometric methods for the feature selection and classification of spectral data. BMC Bioinform 10(1):1–16

29. Lloyd CT, Sorichetta A, Tatem AJ (2017) High resolution global gridded data for use in population studies. Sci Data 4(1):1–17

30. Bakas S, Reyes M, Jakab A, Bauer S, Rempfler M, Crimi A, Shinohara RT, Berger C, Ha SM, Rozycki M et al (2018) Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the brats challenge. arXiv preprint arXiv:1811.02629

31. Chen S, Ding C, Liu M (2019) Dual-force convolutional neural networks for accurate brain tumor segmentation. Pattern Recogn 88:90–100

32. Shehab LH, Fahmy OM, Gasser SM, El-Mahallawy MS (2021) An efficient brain tumor image segmentation based on deep residual networks (resnets). J King Saud Univ Eng Sci 33(6):404–412

33. Fang L, He H (2018) Three pathways U-Net for brain tumor segmentation. In: Pre-conference Proceedings of the 7th medical image computing and computer-assisted interventions (MICCAI) BraTS Challenge, vol 2018, pp 119–126