Efficient Segmentation of Abdominal Organs using Skip Residual Block UNet Model

Abdul Qayyum[1], Abdesslam BENZINOU[1] and Moona Mazher[2],

[1] ENIB, UMR CNRS 6285 LabSTICC, Brest, 29238, France
[2] Department of Computer Engineering and Mathematics, University Rovira i Virgili, Spain
qayyum@enib.fr

Abstract. The deep learning segmentation model has been widely used in medical image segmentation. Fast and Low GPU memory Abdominal Organ segmentation (FLARE 2021) challenge provide a very big and diverse dataset that are collected from multicenter for evaluation of deep learning models for segmentation of abdominal organs such as liver, kidney, spleen, and pancreas. In this paper, we proposed a simple and efficient solution for abdominal multiorgan segmentation based on deep learning model. The simple and lightweight 2D convolutional layers' blocks before the proposed residual block have been presented for FLARE. The residual blocks have been introduced at skip connection at each decoder block after the 2D upsampling layer for efficiently improving the sementation maps. Initial results are satisfactory for big organs and provide optimal solution for multi-organ abdominal segmentation task.

Keywords: Abdominal segmentation, Residual block, UNet deep learning segmentation model.

1 Introduction

The availability of a large and relevant dataset claims for the study of a machine learning approach and more particularly for a deep learning one. Indeed, in the past few years, deep learning has become a breakthrough technology in a wide range of problems such as image classification, action recognition, automatic labeling of an image, image segmentation to name a few. In many computer vision areas, deep learning-based models achieved state-of-the-art performances and started to catch the attention and effort in the context of medical imaging.

However, most of the existing abdominal datasets only contain single-center, single-phase, single-vendor, and single-disease cases, which makes it unclear that if the performance obtained on these datasets can generalize well on more diverse datasets. Most deep learning models are based on encoder-decoder type architectures and these architectures have been used widely for numerous medical image segmentation applications.

The main contribution in this paper is as follows:

1. The simple and lightweight 2d convolutional layers' blocks before the proposed residual block have been presented for abdominal segmentation.

2. The residual blocks have been introduced at skip connection and also used at each decoder block after the 2D upsampling layer for efficiently improving the segmentation maps.

The detailed description of the proposed model is shown in Figure. 1 and Figure. 2.

2 Methods

2.1 Proposed Method

A framework of the proposed model is presented as an encoder, a decoder, and a baseline module. The 1x1 convolutional layer with softmax function has been used at the end of the proposed model. The 2D maxpooling layer has been used to reduce the input image spatial size. The convolutional block consists of convolutional layers with Batch-Normalization and ReLU activation function to extract the different feature maps from each block in the encoder side. In the encoder block, the spatial input size has been reduced with an increasing number of feature maps and on the decoder side, the input image spatial size will increase using a 2D upsampling layer with a bilinear upsampling method. Furthermore, after the 2D upsampling layer the convolutional efficient module has been used to enhance the receptive field and extract the contextual information to minimize the semantic gap between feature maps. The input features' maps that are obtained from every encoder block are concatenated with every decoder block feature map to reconstruct the semantic information represented as a red dotted line in Figure. 1. The convolutional (3x3conv-BN-ReLu) layer used the input feature maps extracted from every convolutional block in the encoder side and further passed these feature maps into the proposed residual module. The spatial size doubled at every decoder block and feature maps are halved at each decoder stage of the proposed model. The feature concatenation has been done at every encoder and decoder block except the last 1x1 convolutional layer.

The detail of the residual module is shown in Figure. 2. The Residual network (ResNets) has been widely used for deep learning classification and segmentation and this model is built on hundreds of layers. Each residual block comprised of two paths, the first path either used identity mapping or used 1x1 convolutional layer with BN, and the second path consisted of a series of layers such as ReLU, convolutional, and Batch Normalization (BN), these two paths are summed together to get the final output from the residual block. The residual block has three advantages as compared to traditional CNN-based models. The first advantage is that the gradient can flow continuously, permitting the parameters to be updated in very deep networks, and second, the operations applied by a single layer are only a small modification to identity operation. Third, ResNets are robust to layers permutation suggesting that neighboring layers perform similar operations.

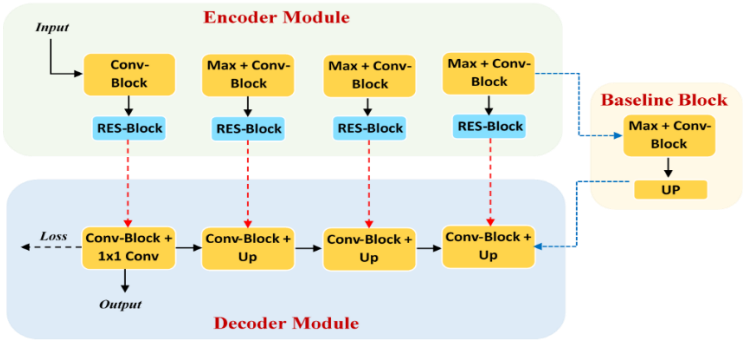


Figure.1. proposed model for liver, kidney, spleen, and pancreas segmentation.

In this paper, the 1x1 Conv with BN has been used as a skip connection with series of several layers for boundary and structural information preservation in the residual block. The main purpose of residual blocks is the preservation of the feature maps within convolutional layer blocks that are used before every encoder block which helps bridge the semantic gap between the encoder and decoder while maintaining the same or little increment in the computational overhead for providing the accurate segmentation map. The structural information for feature maps could be restored by the addition of the residual blocks that aimed to preserve the fine-grained structures that would be useful and play an important role in medical image segmentation.

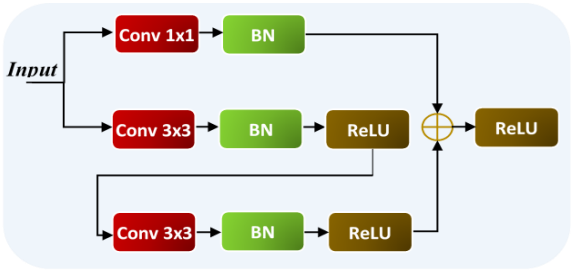


Figure. 2. The proposed residual block for liver, kidney, spleen, and pancreas segmentation.

2.2 Pre-processing

We have used the following preprocessing steps for data cleaning:

- Cropping strategy: None
- Resampling Method for anisotropic data:
  Nearest neighbor interpolation method has been applied for resampling.
- Intensity Normalization method:
  The dataset has been normalized using z-score method based on mean and standard deviation.

2.3 Post processing

In post processing, we did not use any post-processing step except to resize the prediction mask to original input size of each slice in the input volume. The bilinear interpolation method in resize function is to use to make the original size.

2.4 Model Parameters.

The total number of trainable parameters used in our proposed model is 23,889,221 and the total FLOPS is 319815680. The detail description of training and validation parameters is explained in implementation detail section.

3 Dataset and Evaluation Metrics

3.1 Dataset

The dataset used of FLARE2021 is adapted from MSD [1] (Liver [2], Spleen, Pancreas), NIH Pancreas [3, 4, 5], KiTS [6, 7], and Nanjing University under the license permits. The distribution of data for training, validation, and testing datasets is given in Table 1. For more detailed information on the dataset, please refer to the challenge website and [8]. The total number of cases is 511. An approximate 70%/10%/20% train/validation/testing split is employed resulting in 361 training cases, 50 validation cases, and 100 testing cases. The detail information is presented in Table 1.

Table 1. Data splits of FLARE2021.

Data Split	Center	Phase	#Num.
Training (361 cases)	The National Institutes of Health Clinical Center	portal venous phase	80
	Memorial Sloan Kettering Cancer Center	portal venous phase	281
Validation (50 cases)	Memorial Sloan Kettering Cancer Center	portal venous phase	5
	University of Minnesota	late arterial phase	25
	7 Medical Centers	various phases	20
Testing (100 cases)	Memorial Sloan Kettering Cancer Center	portal venous phase	5
	University of Minnesota	late arterial phase	25
	7 Medical Centers	various phases	20
	Nanjing University	various phases	50

For training and optimization of our proposed model used 80 % (288 cases) for training and 20 % (73 cases) for validation. The 50 cases have been used for testing or validation of our proposed model. The 288 cases consisted of 38990 numbers of 2D images and the same number of 2D annotations that used for training the 2D model. 73 internal validation cases that are comprised of 7300 number of 2D images with annotations used for internal testing or validation of our proposed model.

3.2 Evaluation Metrics

In the evaluation, both accuracy and efficiency of the model will be considered to rank the challenge participants using the following metrics:
• Dice Similarity Coefficient (DSC)
• Normalized Surface Distance (NSD)
• Running time
• Maximum used GPU memory (when the inference is stable)

4 Implementation Details

4.1 Environments and requirements

The proposed deep learning model is implementd in PyTorch and other libraries based on python are used for preporcessing and aanalysis of the datasets. The Numpy used to process the input images and volume and SimpleITK used for reading and writing the nifity dataformat. The ITK-SNAP used for data visualization. The OpenCV and skimage used for reading and converting the NumPy array into 2D images
The environments and requirements of the proposed method is shown in Table 2.

Table 2. Environments and requirements.

Windows/Ubuntu version	Ubuntu 18.04.5 LTS
CPU	Intel(R) Core (TM) i9-7900X CPU@3.30GHz
RAM	16×4GB
GPU	Nvidia V100
CUDA version	11
Programming language	Python3.7
Deep learning framework	Pytorch (Torch 1.7.0, torchvision 0.2.2)
Specification of dependencies	SimpleITK, Numpy, OpenCV, Skimage, Scipy, Nibabel, ITK-SNAP
(Optional) code is publicly available at	https://github.com/RspectKnowledge/FLARE_21_Segmentation_DL[1]

[1] https://github.com/RespectKnowledge/FLARE_21_Segmentation_DL

4.2 Training protocols

The learning rate of 0.0004 with Adam optimizer has been for training the proposed model. The binary cross-entropy function is used as a loss function between the output of the model and the ground-truth sample. 48 batch-size with 10000 number of epochs has been used with 20 early stopping steps. The best model weights have been saved for prediction in the validation phase. The 256x256 input image size was used for training and prediction resample with original input size at prediction using nearest-neighbor interpolation method. The Pytorch library is used for model development, training, optimization, and testing. The V100 tesla NVidia-GPU machine is used for training and testing the proposed model. In the initial results, there is no data augmentation method used for training the proposed model except to convert into torch tensor. Later, the data augmentation methods are used to further improved the results. The dataset has been taken from a different vendor and a different center. The dataset cases have different intensity ranges. The dataset is normalized between 0 and 1 using the max and min intensity normalization method. The detail of training protocol is shown in Table.3

Table 3. Training protocols.

Data augmentation methods	HorizontalFlip (p=0.5), VerticalFlip (p=0.5), RandomGamma (p=0.8)
Initialization of the network	"he" normal initialization
Patch sampling strategy	None
Batch size	48
Patch size	256x256
Total epochs	1000
Optimizer	Adam
Initial learning rate	0.0004
Learning rate decay schedule	None
Stopping criteria, and optimal model selection criteria	Stopping criterion is reaching the maximum number of epoch (1000).
Training time	48.5 hours
CO_2eq[1]	None

4.3 Testing protocols

The same preprocessing has been applied at testing time. The training size of each image is fixed (256x256) and used linear interpolation method to resample the prediction mask to original shape for each validation slice. The prediction mask produced by our proposed model has been resampled such that it has the same size and spacing as the original image and copy all of the meta-data, i.e., origin, direction, orientation, etc.,

5 Results

5.1 Quantitative Results on Validation Set

The quantitative scores are evaluated between predicted and ground truth segmentation masks 50 validation dataset provided by challenge organizer. The average dice and HD for all validation scores also showed in Table 4.
The average Dice score (DSC) of liver, kidney, spleen and Pancreas are 91.65%, 78.69%, 86:30% and 57.44, respectively, while the pancreas has the lowest DSC of 57.44%. The normalized surface distance (NSD) between predicted and segmentation volume is shown in Table.2. NSD for liver, kidney, spleen and pancreas are 54.07%,58.36%, 68.63% and 42.51 %. Again, the pancreas produced lower NSD value as compared to other organs. Our proposed solution did not produce better results especially in case of Kidney and Pancreas.

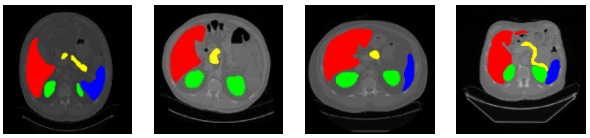
Table 4. The dice coefficient and HD for liver, kidney, spleen, and pancreas.

	Liver	**Kidney**	**Spleen**	**Pancreas**	**Mean DSC and NSD**
Dice (DC)	91.65 ± 5.94	78.69 ± 19.49	86.30 ± 16.87	57.44 ± 20.97	78.52
NSD	54.07 ± 13.54	58.36 ± 16.72	68.63 ± 20.90	42.51 ± 17.65	55.89

5.2 Qualitative Results

The challenge organizers provided four easy and four hard cases for qualitative evaluation of our proposed model. We have selected one 2D slice from each easy and hard cases for evaluation is shown in Figure3 and Figure 4. The first row shows ground truth segmentation mask and second row shows prediction mask. The color coding for each segmented organ are the liver (red), the kidneys (green), the spleen (blue) and the pancreas (yellow). The worst prediction mask produced by proposed model performance in hard cases is shown in Figure4. The segmentation mask produced by proposed model for easy cases are very close to the ground truth segmentation except at the boundary of the spleen, and pancreas.

Figure 3. Qualitative analysis between ground truth and prediction using easy cases (case_15, case_35, case_46, case_47).

Ground-Truth 

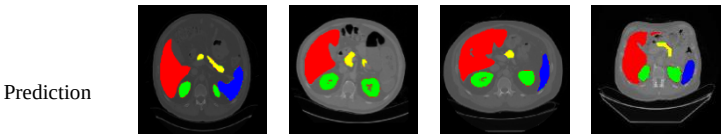
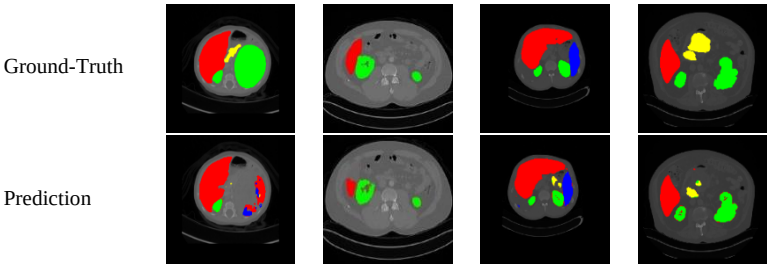


Figure 4. Qualitative analysis between ground truth and prediction using hard cases (case_10, case_11, case_23, case_25).



6 Discussion and Conclusion

Few slices have been chosen to see the qualitative information about each class of dataset. The proposed model could not estimate the boundary of each class either in easy and hard cases. The boundary of each organ has an error and overestimate the segmentation masks. The proposed model totally failed to produce kidney and pancreas as shown in second row and first column in Figure 4. The model is completely missing pixel value of kidney and pancreas tumors. Similarly, model also failed to produce better segmentation map in second row and fourth column for hard case especially for pancreas. The proposed model overestimate in case23 and produced extra pixels values for pancreas labels in third column in Figure 4. The dataset may have imbalance pixel values and different contrast region produced different results. Some hard cases, we have observed there is no contrast enhanced between organs and boundary and model did not perform good in that particular region. We did not use any data augmentation method and maybe in future, we need a more sophisticated data augmentation strategy for better performance. The large tumors of individual organs are only present in a small number of images which can lead to lacking segmentation predictions for unseen images. The number of cases provided by challenge organizers have different spatial dimension with varying number of frames in each input volume, different pixel spacing and various contrast are the reasons to fail the segmentation in some validation cases.
The proposed model produced better results in big organs and not satisfactory in small lesions. We will further improve the results using different training and optimization tricks to improve the score. The model produced little overestimation in some cases especially at the boundary of each organ. Test time augmentation and cross-validation with some data augmentation techniques may improve the results. The post ensemble

would provide a better option to produce the optimal prediction mask. All these tricks should investigate in future to obtain the better performance.

Acknowledgment

The authors of this paper declare that the segmentation method they implemented for participation in the FLARE2021 challenge has not been used any pre-trained models nor additional datasets other than those provided by the organizers.

References

1. A. L. Simpson, M. Antonelli, S. Bakas, M. Bilello, K. Farahani, B. Van Ginneken, A. Kopp-Schneider, B. A. Landman, G. Litjens, B. Menze et al., "A large annotated medical image dataset for the development and evaluation of segmentation algorithms," arXiv preprint arXiv:1902.09063, 2019. 2
2. P. Bilic, P. F. Christ, E. Vorontsov, G. Chlebus, H. Chen, Q. Dou, C.-W. Fu, X. Han, P.-A. Heng, J. Hesser et al., "The liver tumor segmentation benchmark (lits)," arXiv preprint arXiv:1901.04056, 2019. 2
3. H. Roth, A. Farag, E. Turkbey, L. Lu, J. Liu, and R. Summers, "Data from pancreas-ct. the cancer imaging archive (2016)." 2
4. H. R. Roth, L. Lu, A. Farag, H.-C. Shin, J. Liu, E. B. Turkbey, and R. M. Summers, "Deeporgan: Multi-level deep convolutional networks for automated pancreas segmentation," in international conference on medical image computing and computer-assisted intervention. Springer, 2015, pp. 556– 564. 2
5. K. Clark, B. Vendt, K. Smith, J. Freymann, J. Kirby, P. Koppel, S. Moore, S. Phillips, D. Maffitt, M. Pringle et al., "The cancer imaging archive (tcia): maintaining and operating a public information repository," Journal of digital imaging, vol. 26, no. 6, pp. 1045–1057, 2013. 2
6. N. Heller, F. Isensee, K. H. Maier-Hein, X. Hou, C. Xie, F. Li, Y. Nan, G. Mu, Z. Lin, M. Han et al., "The state of the art in kidney and kidney tumor segmentation in contrastenhanced ct imaging: Results of the kits19 challenge," Medical Image Analysis, vol. 67, p. 101821, 2021. 2
7. N. Heller, S. McSweeney, M. T. Peterson, S. Peterson, J. Rickman, B. Stai, R. Tejpaul, M. Oestreich, P. Blake, J. Rosenberg et al., "An international challenge to use artificial intelligence to define the state-of-the-art in kidney and kidney tumor segmentation in ct imaging." American Society of Clinical Oncology, vol. 38, no. 6, pp. 626–626, 2020. 2
8. J. Ma, Y. Zhang, S. Gu, C. Zhu, C. Ge, Y. Zhang, X. An, C. Wang, Q. Wang, X. Liu, S. Cao, Q. Zhang, S. Liu, Y. Wang, Y. Li, J. He, and X. Yang, "Abdomenct-1k: Is abdominal organ segmentation a solved problem?" IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021. 2, 3, 4