MIScnn: A Framework for Medical Image Segmentation with Convolutional Neural Networks and Deep Learning

Shivam Singh[1], Vishesh Yadav[2], Suraj Khokarale[3], Devarsh Patil[4]

[1]*Student at D.Y Patil Institute of Engineering and Technology*
[2]*Student at D.Y Patil Institute of Engineering and Technology*
[3]*Student at D.Y Patil Institute of Engineering and Technology*
[4]*Student at D.Y Patil Institute of Engineering and Technology*

---***---

Abstract - *The excessive ease of use and comfort of current medical imaging has sparked a strong need for automated medical picture segmentation. Current image segmentation systems, however, lack the necessary functionality for a basic medical image segmentation pipeline architecture. Pipelines that have already been enacted are generally independent computer programmes that are optimized for a certain collection of public data. As a result, this article introduces the MIScnn Python module for ASCII text files.*

MIScnn aims to provide an intuitive API that allows for the rapid creation of a medical image segmentation pipeline that includes data I/O, pre-processing, data augmentation, patch-wise analysis, metrics, a library of state-of-the-art deep learning models, and model usage such as training, prediction, and fully automatic evaluation (e.g. cross-validation). Similarly, comprehensive pipeline customization is possible because to strong configurability and many open interfaces.

The Kidney Tumour Segmentation Challenge 2019 data set (multiclass semantic segmentation with 300 CT images) yielded a strong predictor based on the conventional 3D U-Net model after cross-validation using MIScnn.

We were able to demonstrate that the MIScnn framework allows researchers to quickly put up a comprehensive medical image segmentation pipeline using only a few lines of code in this experiment. The MIScnn source code is accessible on

Github at https://github.com/frankkramer-lab/MIScnn.

***Key Words*: Medical image analysis, Computer-aided diagnosis, Biomedical image segmentation, U-Net, Deep learning, Open-source framework.**

1. INTRODUCTION

Because of the widespread use of modern medical imaging in technology, there is a greater demand for automatic medical picture segmentation. Despite this crucial need, existing medical image segmentation technologies lack the capability required to create a basic medical picture segmentation pipeline.

MIScnn is an open-source framework with an intuitive API that lets you easily build medical image segmentation pipelines with Convolutional Neural Networks and Deep Learning models in only a few lines of code. The goal of MIScnn is to provide a framework API that allows for the rapid development of medical image segmentation pipelines, including information I/O, preprocessing, data augmentation, patch-wise analysis, metrics, a library of progressive deep learning models, and model usage such as training, prediction, and automatic analysis.

Convolutional networks are commonly used for classification tasks, with the output to an image being a single category label. However, in a number of visual activities, particularly in the medical specialty picture process, the needed output should include localization, i.e., every element should be assigned a category name. Furthermore, in medical specialty duties, thousands of training photographs are often out of reach. As a result, using an area region (patch) around each element as input, we trained a network in a sliding-window setup to predict the class label of each element. This network will first become localised. Second, the number of patches in the training data is considerably more than the number of Training Images.
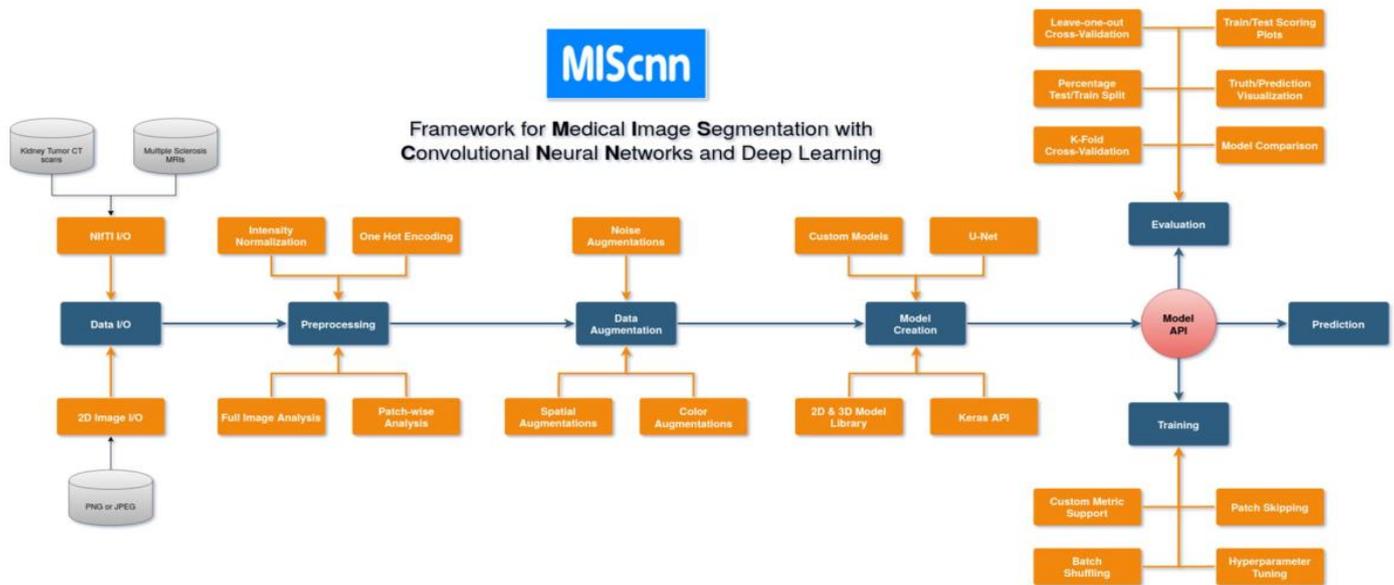
FIG 1 . Flowchart of MIScnn Pipeline

2. Data Input

2.1 NifTi data I/O interface

For loading magnetic resonance imaging and computed axial tomography data into the framework, MIScnn provides a data I/O interface for the Neuroimaging Information Science Technology Initiative (NifTI) file format. This structure was first devised to expedite the event while also enhancing the usability of information science technologies related to neuroimaging. Still, it's currently unremarkably used for sharing public and anonymous magnetic resonance imaging and CT knowledge sets, not only for brain imaging, but also for any other type of human 3D imaging.The 3D image matrix and numerous data, such as the thickness of the magnetic resonance imaging slices, are stored in an NIfTI file.

2.2 Custom data I/O interface

MIScnn allows the use of bespoke knowledge I/O interfaces for various image knowledge formats in addition to the mandated NIfTI I/O interface. This open interface enables MIScnn to handle certain medical specialty imaging choices (for example, MRI slice thickness), without the loss of this information due to a format conversion demand. A specific I/O interface should be dedicated to the preprocessing process, and it should return to the medical picture as a second or 3D matrix for integration into the process.

3. Data Preprocessing

3.1 Pixel intensity normalization

Inconsistent signal intensity ranges in photos will have a significant impact on segmentation methods' performance. Due to completely distinct picture formats, variable hardware/instruments (e.g. completely separate scanners), technical inconsistencies, and easily a biological variance, the signal ranges of medical specialist imaging knowledge are very varied amongst knowledge sets. Moreover, picture segmentation techniques that use machine learning often perform better on alternatives that follow a standard distribution. Scaling and standardizing imaging information is advised to achieve dynamic signal intensity variation uniformity.

3.2 Clipping

Similar to component intensity standardization, it's conjointly common to clip component intensities to an explicit vary. Outside of this range, intensity costs will be reduced to the lowest or most variable value. Even in many scanners, component intensity levels for similar organs or tissue types are assumed to be equal, especially in computer imaging images. Organ-specific component intensity clipping might be used to take advantage of this.

3.3 Resampling

To change the width and/or height of images, the resampling technique is used. This ends up in a brand-new image with a changing range of pixels. resonance or laptop pictorial representation scans will have different slice thicknesses. However, in order to train neural network models, the pictures must have the same slice thickness or voxel spacing, this could be accomplished through resampling. To begin with, downsampling images limits the amount of GPU memory available for training and prediction.

3.4 One hot encoding

MIScnn can handle both binary (background/cancer) and multi-class (background/kidney/liver/lungs) segmentation problems. Using a variable with two states, zero and one, the depiction of a binary segmentation may be readily generated. Except in machine learning techniques, such as deep learning models, where many category segmentation labels are used, it is necessary to translate the categories into an extra mathematical picture. Using the One Hot coding style, this might be accomplished by creating a single binary variable for each segmentation category. One Hot mechanically encodes segmentation labels with more than two categories using MIScnn.

3.5 Patch-wise and full image analysis

In 3D segmentation analysis, the available GPU hardware plays an outsized role depending on the resolution of medical images. Due to the large GPU memory requirements, it is currently unable to fully integrate high-resolution MRIs with an associated example size of 400 512 512 into progressive convolutional neural network models. As a result, 3D medical imaging data is either split into smaller cuboid patches or processed slice by slice, similar to a series of second photographs [10, 11, 23]. MIScnn cuts 3D medical photos into patches with a customizable size (e.g. 128 128 128) by default to fully use the data from all three dimensions. These patches will function in GPUs with RAM capacities of 4–24 GB, which are commonly used in the analysis, according to the model design. The slice-by-slice second analysis is supported and may be used in MIScnn, however, the 3D patch analysis is not. It's also possible to combine the use of complete 3D images in the event that you're studying unusually small medical images or have a large GPU cluster. Second medical images are automatically incorporated into convolutional neural networks and deep learning models. Still, for images with high resolution, a second patch-wise strategy is used.

4. Sampling and Batch Generation

4.1 Skipping blank patches

The well-known problem in medical images is large unbalances between significant segments and the backdrop results in a large number of parts that are strictly classified as background and have no learning information. There's no use in multiplying these blank elements or patches, especially when it comes to knowledge augmentation. As a result, any patches that are completely classified as background are omitted from the patch-wise model training to minimize unnecessary fitting delays.

4.2 Batch Management

Sets of entire photos or patches are packaged into batches after the data preparation and hence the optional data augmentation for training. One batch comprises a large number of ready images that are processed by the model and GPU in a single phase. The neural network adjusts its internal weights in accordance with the specified learning rate for each batch or process step. The number of images that may be stored in a single batch is highly dependent on the amount of GPU memory available, and therefore must be appropriately planned in MIScnn. Each batch is stored in memory so that it may be accessed at any time during the training process. Due to the avoidance of reserve continuous batch preprocessing, this strategy dramatically decreases computation time. MIScnn also permits "on-the-fly" construction of the following batch in memory during the runtime to overcome this limitation.

4.3 Batch Shuffling

At the end of each epoch, the order of batches, that is, the area unit intended to be fitted and processed is shuffled. This strategy lessens the risk of overfitting by reducing the variance of the neural network during fitting over a period of time. It should be emphasized, however, that just the batch process sequence is shuffled, and the knowledge itself is not sorted into a new batch order.

4.4 Multi-CPU and -GPU Support

In addition to GPU computing, MIScnn offers the use of several GPUs and simultaneous central processor batch loading. The storing of already prepared batches on memory, for example, allows for a fast and parallelizable process with central processors and GPU clusters by removing the risk of batch preparation bottlenecks.

5. Deep Learning Model Creation

5.1 Model Architecture

The most important stage in a medical picture segmentation pipeline is choosing a deep learning or convolutional neural network model. There are many distinct model architectures, each with its own set of strengths and limitations. The MIScnn options provide an open model interface that allows you to load and switch between the several progressive convolutional neural network models available, such as the widely used U-Net model. Keras, an ASCII text file neural network framework that provides a straightforward API for commonly used neural network building blocks on top of TensorFlow, is

used to define models. The already established models, like the Optimized High-Resolution Dense-U-Net model, are significantly flexible by a determinable range of neurons, custom input sizes, ex gratia dropout, and batch normalization layers, or enhanced design versions. MIScnn additionally provides 3D architectures, as well as 2D architectures for medical image segmentation. The open model interface enables for bespoke deep learning model implementations and simple integration of these custom models into the MIScnn pipeline, in addition to the freedom in changing between previously imposed models.

5.2 Metrics

MIScnn has a large number of different metrics that may be utilized for loss performance training, figure analysis, and manual performance analysis. One of the most often used metrics for medical picture segmentation is the Dice constant, also known as the Dice similarity index. It calculates the degree of similarity between the expected segmentation and the ground truth. False positives, on the other hand, are penalized. There's a simple and class-wise Dice constant implementation to enjoy the precision metric reckoning on the segmentation categories (binary or multi-class). Unlike the simple approach, which merely counts the number of right and erroneous predictions, the category-wise method accounts for prediction performance for each segmentation class, which is strongly recommended for typically class-unbalanced medical images. The Jaccard Index is another widely accepted statistic. Even though it's similar to the Dice constant, it doesn't focus entirely on exact segmentation. It does, however, punish over-segmentation of understanding. MIScnn, on the other hand, employs the Tversky loss for training. The Tversky loss performance solves knowledge imbalance and is admired by the Dice constant. Even so, it achieves a far better balance of precision and recall. As a result, Tversky loss performance provides smart binary as well as multi-class segmentation performance. In addition, many common metrics used in Keras, such as accuracy and cross-entropy, are used in MIScnn. MIScnn allows you to blend additional metrics for training and analysis in addition to the already imposed metrics or loss functions. As described in Keras, a custom metric is enforced and readily supplied to the deep learning model.

5.3 Model Utilization

The model will now be utilized for training on the information to suit model weights or for prediction by leveraging an already fitted model with the initialized deep learning model and hence the totally preprocessed information. Instead, the model will do AN analysis by using cross-validation with numerous training and prediction calls, as an example. The model API allows you to save and load models so that you may reprocess previously fitted models for prediction or share pre-trained models.

5.4 Training

Various parameters should be established while training a convolutional neural network or deep learning model. The information augmentation options of the data set, which have a significant impact on medical picture segmentation training, should already be described in the pipeline. Following that, the batch management configuration listed the batch size settings as well as the batch shuffling at the end of each epoch. As a result, simply the learning rate and hence the number of epochs must be altered before the training technique can be used. The training rate of a neural network model is determined by the amount to which the neural network model's prior weights are modified in each iteration or epoch. The number of epochs, on the other hand, determines the proportion of times the full knowledge set will be fitted into the model. The resulting fitted model can then be stored in memory. Because of the remaining fitting time, the underlying Keras framework provides insights into the current model performance using specified metrics throughout training. MIScnn also provides the use of a fitting evaluation questioning practicality, in which the fitting scores and metrics are saved in a tab-separated file or immediately plotted as a graph.

5.5 Prediction

Once trained, an associate degree previously fitted neural network model will be utilized directly or it will be imported from a file for segmentation prediction. For each panel, the algorithm estimates a Sigmoid price for each category. This panel's probability assessment for the related label is represented by the Sigmoid price. The argmax of the One Hot encoded category is then born-again to one result variable holding the category with the best Sigmoid pricing, which is known for multi-class segmentation challenges. MIScnn provides two approaches for patches inside the prediction after using the overlapping patch-wise analysis methodology during the training. Either the prediction approach develops discrete patches and regards overlapping patches as solely knowledge enhancement during the training, or overlapping patches are produced for prediction. Due to a lack of prediction capacity at patch edges, generating a second prediction for edge pixels in patches by victimizing associate degree overlap may be an underutilized technique. Within the overlapping section of two patches and with numerous predictions, a merging method for the pixels is needed within the subsequent merge of patches back to the basic medical picture form. MIScnn estimates the mean of the predicted Sigmoid values for each category in each overlapping panel by default. The resulting picture matrix with segmentation prediction, which has the same format as the original medical image, is stored into a file structure using the knowledge I/O interface supplied. The predicted segmentation matrix is stored in NIfTI format with no further information by default when using the NIfTI knowledge I/O interface.

5.6 Evaluation

Multiple automatic analysis techniques are supported by MIScnn to investigate medical image segmentation performance: k-fold cross-validation, leave-one-out cross-validation, percentage-split validation, hold-out sets for checking (data set split into test and plaything with a given percentage), and elaborate validation in which it is frequently given which pictures should be used for training and testing. Apart from extensive validation, sampling is used to create training and testing data sets in all analytical methodologies. The specified measurements and loss performance for the model are automatically planned in figures during the study and recorded in tab-separated files for easy examination. In addition to performance measures, the constituent worth changes, and the frequency of segmentation categories in medical images are frequently examined as part of the MIScnn study. By producing picture representations with segmentation overlays, the resulting forecast is frequently compared directly to the bottom truth. The slices with the segmentation overlays are mechanically depicted inside the Graphics Interchange Format (GIF) for 3D images, such as MRIs.

5.7 Convolutional Neural Network (U-net)

A contracting path (left side) and an expansive path (right side) make up the network design (right side). The convolutional network's contracting route follows the standard architecture. It is made up of two 3x3 convolutions (unpadded convolutions) that are applied repeatedly, each followed by a rectified linear unit (ReLU) and a 2x2 max pooling operation with stride 2 for downsampling. We quadruple the number of feature channels with each downsampling step. An upsampling of the feature map is followed by a 2x2 convolution ("up-convolution") that halves the number of feature channels, a concatenation with the proportionally cropped feature map from the contracting path, and two 3x3 convolutions, each followed by a ReLU in the expanding path. Due to the loss of boundary pixels in every convolution, cropping is required. A 1x1 convolution is employed at the final layer to convert each 64-component feature vector to the appropriate number of classes. The network comprises a total of 23 convolutional layers. To ensure that the output segmentation map tiles seamlessly (see Figure 2), the input tile size should be chosen so that all 2x2 max-pooling operations are applied to a layer with an even x- and y-dimension.
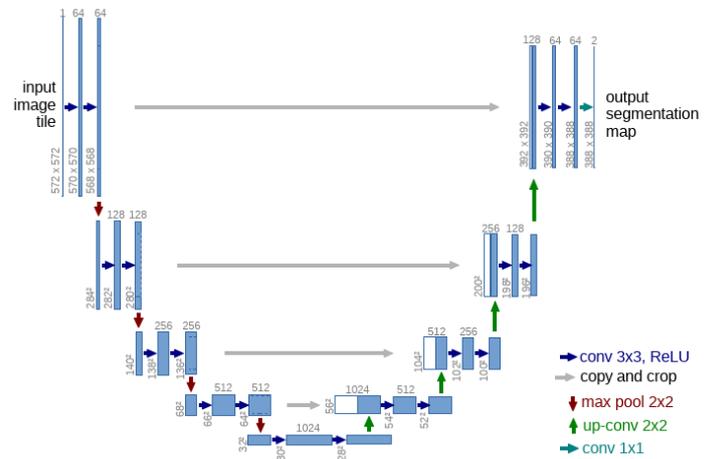
FIG 2 U-Net Architecture

6. Real-Life Use Cases

6.1 Kidney Tumor Segmentation

Computed Tomography scans of urinary organ tumors from the urinary organ tumor Segmentation Challenge 2019 information set showing the urinary organ (red) and tumor (blue) segmentation as overlays. The photographs show the segmentation variations between the bottom truth provided by the KiTS19 challenge and therefore the prediction from the quality 3D U-Net models of our three-fold cross-validation.
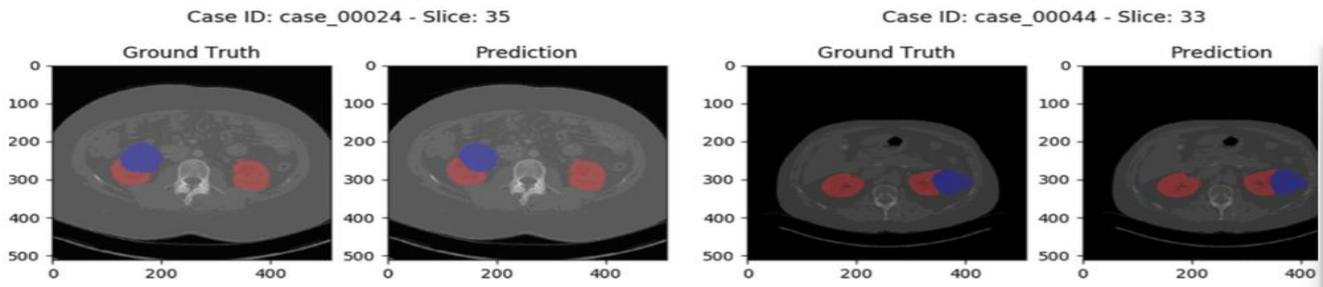
FIG 3 Kidney Tumor Segmentation

6.2 Coronavirus disease 2019(Covid-19)

COVID-19 might be a global pandemic virus that spreads quickly over the planet. RT-PCR (Reverse Transcription enzyme Chain Reaction) is a commonly used test for detecting COVID-19 infection. RT-PCR testing is that the gold standard for COVID-19 testing, RT-PCR is an incredibly complicated, long, and labor-intensive method, sparse availability, and not correct. Chest X-ray may be used for the initial screening of the COVID-19 in places having a shortage of RT-PCR kits and is additional correct at diagnosing. Many researchers have used deep learning to classify if the chest infection is thanks to COVID-19 or different ailments.

6.2.1 PDCOVIDNET

The use of dilation to sight dominant options within the image was explored. The authors planned a parallel expanded CNN model. The expanded module concerned the skipping of pixels throughout the convolution method. Parallel CNN branches square measure proposed with completely different dilation rates. The results of parallel branches were combined and sent into the next convolution layer. The concatenation-convolution process was used to investigate feature relationships in enlarged convolutions, resulting in visually prominent classification alternatives. The model conjointly used Grad-CAM and Grad-CAM++ to highlight the regions of class-discriminative noticeable maps. The performance metrics used were accuracy, precision, recall, F1-score with ROC/AUC, and are 96.58%, 95%, 91%, 93%, and 99.1% severally.

7. Future Scope

Multiple essential aspects are now the focus of active MIScnn development: Adding more knowledge I/O interfaces for the most widely used medical image formats, such as DICOM, expanding preprocessing and knowledge augmentation procedures, and implementing a variety of cost-effective patch skipping approaches rather than ignoring each blank patch (e.g. denoising patch skipping) Implementation of an open interface for bespoke preprocessing algorithms for certain picture types, such as MRIs. The MIScnn road map also contains a model library expansion with a large number of progressive deep learning models for medical image segmentation, in addition to the scheduled feature implementations. Furthermore, an objective comparison of the U-Net model version selection has been made available in order to encourage a lot of insights on alternative model performances using the same pipeline. Contributions from the community in terms of implementations or critique are encouraged and might be included in the assessment. MIScnn currently has a strong pipeline for medical image segmentation; however, it will continue to be upgraded and extended in the future.

8. Results

CNN models are delineated as black boxes and there is a great deal of analysis happening in terms of analyzing and understanding output at each layer. Since medical images square measure concerned, we'd like an associate degree responsible and economical prediction system that ought to even be able to articulate about a call taken. Image captioning is being done by researchers (textual representations of the image). This can change physicians to grasp the perception of the network at each output layer and intermediate levels. Researchers have tried theorem deep learning models that calculate the uncertainty estimates. This would facilitate physicians to assess the model. All of these might help clinicians analyze medical images faster by employing CNNs. Here, we tend to analyze and measure information from the urinary organ tumor Segmentation Challenge 2019 exploitation MIScnn. The main goal of this experiment is to demonstrate MIScnn's 'out-of-the-box' performance without doing extensive and lengthy optimizations on the data set or the medical anomaly. The scripts in the Appendix were used to acquire all of the findings.

9. Conclusion

The ASCII text file Python package MIScnn: A framework for medical image segmentation with convolutional neural networks and deep learning was introduced in this publication. The user-friendly API enables rapid creation of medical image segmentation pipeline, as well as knowledge I/O, preprocessing, knowledge augmentation, patch-wise analysis, metrics, a library of progressive deep learning models, and model usages such as training, prediction, and completely automated analysis (e.g. cross-validation). Users may entirely personalize the pipeline because of its high

configurability and many open interfaces. Researchers may use this framework to replace a complete medical image segmentation process with just a few lines of code. We're going to test the MIScnn practicality by conducting automatic cross-validation on the excretory organ neoplasm Segmentation Challenge 2019 CT knowledge set, which will result in a robust predictor. We expect that it will make it easier to move medical picture segmentation from analytical laboratories to practical applications.

10. Acknowledgement

11. References

1. Aggarwal P, Vig R, Bhadoria S, Dethe CG. Role of segmentation in medical imaging: a comparative study. Int J Comput Appl. 2011;29:54–61.

2. Gibelli D, Cellina M, Gibelli S, Oliva AG, Termine G, Pucciarelli V, et al. Assessing symmetry of zygomatic bone through three-dimensional segmentation on computed tomography scan and "mirroring" procedure: a contribution for reconstructive maxillofacial surgery. J Cranio-Maxillofac Surg. 2018;46:600–4. https ://doi.org/10.1016/j.jcms.2018.02.012.

3. Cellina M, Gibelli D, Cappanella A, Toluian T, Pittino CV, Carlo M, et al. Segmentation procedures for the assessment of paranasal sinuses volumes. Neuroradiol J. 2020. https ://doi.org/10.1177/19714 00920 94663 5.

4. Hu X, Luo W, Hu J, Guo S, Huang W, Scott MR, et al. Brain SegNet: 3D local refinement network for brain lesion segmentation. BMC Med Imaging. 2020;20:17. https ://doi.org/10.1186/s1288 0-020-0409-2.

5. Sun R, Wang K, Guo L, Yang C, Chen J, Ti Y, et al. A potential field segmentation based method for tumor segmentation on multi-parametric MRI of glioma cancer patients. BMC Med Imaging. 2019;19:48. https ://doi. org/10.1186/s1288 0-019-0348-y.

6. Claudia C, Farida C, Guy G, Marie-Claude M, Carl-Eric A. Quantitative evaluation of an automatic segmentation method for 3D reconstruction of intervertebral scoliotic disks from MR images. BMC Med Imaging. 2012;12:26. https ://doi.org/10.1186/1471-2342-12-26.

7. Guo Y, Liu Y, Georgiou T, Lew MS. A review of semantic segmentation using deep neural networks. Int J Multimed Inf Retr. 2018;7:87–93. https ://doi.org/10.1007/s1373 5-017-0141-z.

8. Anwar SM, Majid M, Qayyum A, Awais M, Alnowami M, Khan MK. Medical image analysis using convolutional neural networks: a review. J Med Syst. 2018;42:226. https ://doi.org/10.1007/s1091 6-018-1088-1.

9. Wang G. A perspective on deep imaging. IEEE Access. 2016;4:8914–24.

10. Litjens G, Kooi T, Bejnordi BE, Setio AAA, Ciompi F, Ghafoorian M, et al. A survey on deep learning in medical image analysis. Med Image Anal. 2012;2017(42):60–88.

11. Shen D, Wu G, Suk H-I. Deep learning in medical image analysis. Annu Rev Biomed Eng. 2017;19:221–48. https ://doi.org/10.1146/annur ev-bioen g-07151 6-04444 2.

12. Ronneberger O, Fischer P, Brox T. U-Net: convolutional networks for biomedical image segmentation. Lect Notes Comput Sci (including SubserLect Notes ArtifIntellLect Notes Bioinformatics). 2015;9351:234–41.

13. Zhou Z, Siddiquee MMR, Tajbakhsh N, Liang J. UNet++: a nested U-Net architecture for medical image segmentation. 2018. http://arxiv .org/ abs/1807.10165 . Accessed 19 Jul 2019.

14. Isensee F, Petersen J, Klein A, Zimmerer D, Jaeger PF, Kohl S, et al. nnUNet: self-adapting framework for U-Net-based medical image segmentation. 2018. http://arxiv .org/abs/1809.10486 . Accessed 19 Jul 2019.

15. Kolařík M, Burget R, Uher V, Řiha K, Dutta M. Optimized high resolution 3D dense-U-Net network for brain and spine segmentation. Appl Sci. 2019;9:404.

16. Cicek O, Abdulkadir A, Lienkamp SS, Brox T, Ronneberger O. 3D U-net: learning dense volumetric segmentation from sparse annotation. Lect Notes Comput Sci. 2016;9901:424–32.

17. Lee K, Zung J, Li P, Jain V, Seung HS. Superhuman accuracy on the SNEMI3D connectomics challenge. 2017; Nips:1–11. http://arxiv .org/ abs/1706.00120 .