*DOI: 10.24996/ijs.2023.64.4.37*

**ISSN: 0067-2904**

# Residual Network with Attention to Neural Cells Segmentation

**Rabab Farhan Abbas*, Matheel Emaduldeen Abdulmunim**
*Department of Computer Science, University of Technology, Baghdad, Iraq*

Received: 24/2/2022 Accepted: 18/8/2022 Published: 30/4/2023

**Abstract**

Many neuroscience applications, including understanding the evolution of the brain, rely on neural cell instance segmentation, which seeks to integrate the identification and segmentation of neuronal cells in microscopic imagery. However, the task is complicated by cell adhesion, deformation, vague cell outlines, low-contrast cell protrusion structures, and background imperfections. On the other hand, existing segmentation approaches frequently produce inaccurate findings. As a result, an effective strategy for using the residual network with attention to segment cells is suggested in this paper. The segmentation mask of neural cells may be accurately predicted. This method is built on U-net, with EfficientNet serving as the encoder's backbone. The attention approach is employed in the detection and segmentation modules to guide the model's attention to the most valuable features. A massive collection of neural cell microscopic images tests the proposed method. According to the findings of the experiments, this technology can accurately detect and segment neuronal cell occurrences with an intersection over the union IoU of **95.47** and a Dice-Coeff of **98.34**, which is superior to current state-of-the-art approaches.

**Keywords:** Neural network, segmentation, image processing.

# تحديد الخلايا العصبية باستعمال الشبكات الالتفافية الارتجاعية

**رباب فرحان عباس*, مثيل عمادالدين عبدالمنعم**

قسم علوم الحاسوب, الجامعة التكنولوجية, بغداد, العراق

**الخلاصة**

تعتمد العديد من تطبيقات علم الأعصاب بما في ذلك فهم تطور الدماغ على تحديد اجزاء الخلايا العصبية وعزلها في الصور المجهرية. وعلى الرغم من اهمية الموضوع، الانها تبقى مهمة معقدة بسبب التصاق اجسام الخلايا والتغييرات غير المنظمة في جسم الخلايا وعدم وضوح حدود جسم الخلايا والتباين في الالوان بالاضافة الى العيوب الموجودة في خلفية الصور المجهرية بسب قلة دقة بعض الاجهزة. في هذا البحث نقدم شبكة لتحديد اجسام الخلايا العصبية باستعمال تقنيات The residual network with attention. استعملنا لهذا الغرض شبكة U-Net كهيكل لبناء الشبكة المطلوبة مع استعمال Efficient Net في جزء استخراج صفات الصورة المهمة للاستدلال على حدود الخلية العصبية. تم اختبار الطريقة المقترحة على مجموعة ضخمة من صور الخلايا العصبية وكانت النتائج متفوقة على العديد من الطرق الحديثة المقترحة.

*Email: rabab.f.abbas@uotechnology.edu.iq

## 1. Introduction

Numerous neurological disorders result in mortality and disability worldwide. Around 100 million Americans were diagnosed with one of the 1000 neurological illnesses in 2011. These disorders equate to a total cost of $765 billion for the most frequent conditions, including Alzheimer's and other dementias, chronic low back pain, stroke, migraine, epilepsy, traumatic brain injury, and Parkinson's disease [1]. Therefore, precision statistics on the incidence, prevalence, mortality, and disability-related neurological illnesses and their trends are critical for evidence-based health care planning and resource allocation. Additionally, assessing the effectiveness of therapy for specific illnesses is complicated [2]. One of the most prevalent methods for studying neural cells is light microscopy [3]. This technique is simple and non-invasive. However, separating individual neural cells in microscopic images may be time-consuming and complex [4].

On the other hand, the development in computer vision and image processing helps in solving many similar problems in the segmentation of the brain, lunge, and different types of tumors or normal parts [5]. Precisely segmenting neural cells into groups may help discover new and effective medications to treat the millions who suffer [6]. Existing approaches, particularly for neuronal cells, are not precise since they have a distinct, irregular, and concave structure, making it difficult to distinguish them from standard mask heads [7].

The process of identifying each pixel in an image with an index corresponding to a different item from a collection of preset object classes is known as segmentation [8, 9]. The identification of cells in microscopic images, primarily when utilized for quantitative analysis, is a frequent example of a segmentation challenge in biomedical imaging [10]. While current cell segmentation approaches have improved their pixel precision to the point that they are now suitable for many imaging settings, detection accuracy remains a significant issue [11]. Accurate object identification and realistic object shape recovery are essential to the biomedical technology field. Unfortunately, many instance segmentation approaches employ one unique object index per pixel, which only corresponds to the foreground object, which leads to a partial capture of partially overlaid objects and, as a result, a misinterpretation of their form. In addition, this might impair shape-sensitive applications like morphological cell analysis. Therefore, it is necessary to employ instance segmentation techniques that precisely define object boundaries to address these concerns.

Semantic segmentation has become a focus for researchers working with images spanning from biological to natural scene datasets [12–15]. Due to the difficulty of manually segmenting medical images, it is becoming increasingly necessary to divide them automatically. The U-Net [16] architecture is a critical component of convolutional neural network CNNs' segmenting biological images. It is commonly employed in biomedical image segmentation due to its effective performance with little labeled training data. Numerous visual tests have demonstrated the effectiveness of U-Net versions. Thus far, it has been employed in conjunction with pixel-wise regression and pansharpening [17]. TernausNet [18] begins the architecture's encoder path using weights from an ImageNet [19] trained VGG11 [20] model. Attention U-Net [21] augments the standard U-Net with a medical imaging attention gate model that automatically hunts for target structures of varying shapes and sizes. Deep learning solutions are rapidly employed on mobile devices, embedded systems, and any computer with a limited processing capacity, which is not accessible due to CNNs' excessive parameterization, requiring more processing power and storage space for training and inference. Researchers have developed many techniques for decreasing or restricting the weights of models trained on large image datasets [22]. Others have sought to build compact models from the ground up by decomposing standard convolution layers into depth-separated layers, which results in a significant increase in computing speed [23]. Similar to how these compact architectures, also known as EffecientNet architectures, train the U-Net model with fewer parameters, reduced computational requirements, and faster inference.

This article discusses how to train the U-Net model with fewer parameters, less storage space, reduced computational requirements, and faster inference. With LinkNet [24], the same architecture minimizes network configurations, resulting in the best outcomes and the shortest operation time. The proposed technique is evaluated using data from the Sartorius challenge.
The following are the paper's key contributions:
Two network architectures are suggested using U-Net and LinkNet with pre-trained EffecientNet for the encoder part. The LinkNet-based method achieved high results with fewer parameters (about 40% fewer).
1- The attention unit is used to select the most valuable features, increasing segmentation accuracy.
2- The proposed model is more efficient and accurate than the current work, with 95.47 and 98.34 for IoU and Dice-Coeff, respectively, demonstrating great potential for neural cell analysis.
The remaining sections of this paper are organized as follows: The related work for cell segmentation is shown in Section 2. The method employed in this study is described in Section 3. Section 4 explains the experiments. Section 5 discusses the results. Finally, section 6 concludes.

## 2. Related works

This section discusses several cell segmentation and detection methodologies, emphasizing the application of deep learning to the recommended method. Finally, the attention units are introduced.

### *2.1 Cells detection*

CNNs, fully convolutional networks (FCNs), and stacked autoencoders (SAEs) have been used to locate objects in microscope images, and their positions are typically represented by single dots around the object's center, referred to as seeds or markers. Object detection may be considered a problem of pixel-by-pixel categorization. Firstly, the network constructs a probability map for the testing image. Then, the probability of that pixel serving as a seed for each pixel value may be determined. Finally, the target objects can theoretically be located by looking for local maxima in the probability map. On the other hand, non-maximum suppression is frequently utilized to enhance work performance [24].

Ciresan et al. [25] used CNNs to detect mitosis in histological images of breast cancer stained with hematoxylin and eosin. Data augmentation is performed by rotating and mirroring training images in non-consistent ways. In the test phase, the outputs of CNNs that analyze rotated and mirrored images are pooled to build final probability maps. In addition, kernel smoothing minimizes extra noise, allowing for faster detection of mitotic centroids through local maxima (with non-maximum suppression), using the same previous method to predict the nuclei for pancreatic neuroendocrine, brain, and breast cancer by CNNs [26–28], respectively. Phase-contrast microscopy images also predict circulating tumor cells in the blood [29]. Wang et al. [30] used an eight-layer CNN to make neutrophil candidates in images of inflammatory bowel disease. In [31], an SVM classifier is used to identify cell regions, and then a trained CNN classifies them as cells or image backgrounds. It demonstrates that this method outperforms CNN’s prediction on a pixel-by-pixel basis. SAEs are utilized to identify nuclei in images of breast carcinoma [31]. Unsupervised pretraining and supervised fine-tuning are used to ensure that the network learns effectively. Initially, an SAE is trained on raw image data. The SAE is then fine-tuned using a softmax layer, with image patches aligned to nuclei being considered positive samples and those misaligned to nuclei being considered negative samples. Finally, the testing step performs model inference using the sliding window approach. This approach benefits from supervised training for fine-tuning the model's parameters, resulting in increased detection accuracy.

### *2.2. Cells segmentation*

Cell segmentation was used as the foundation for various imaging analyses [32], including cellular morphology computation, characteristic value quantification, and cell identification. In order to construct dense neuronal networks, efficient segmentation of neural structures is also necessary. As a result, reliable segmentation is required for microscope image analysis. Deep neural networks have recently been effective at segmenting microscope images and performing well. Typically, CNNs represent it as a pixel-by-pixel classification approach, producing probability maps from input images and thresholding image segmentation. End-to-end trained FCNs, on the other hand, may immediately generate probability maps with the exact dimensions as the input images, greatly enhancing computing efficiency.

U-Net is utilized in neuronal membranes segmentation [33], phase-contrast images of glioblastoma-astrocytoma cells, and differential interference contrast images of HeLa cells [34].

In order to develop deeper networks for the segmentation of neural structures, Chen et al. [35] modified the fully convolutional network by integrating multilayer contextual information and auxiliary supervised classifiers. Similarly, convolution and pooling methods are also used in the contraction path to categorize semantic data. Furthermore, the expansion route incorporates numerous convolutional and deconvolutional layers at various stages. Finally, the hierarchical context information is aggregated and sent to a softmax layer, which resolves the vanishing gradient problem and enhances the discriminative performance of intermediate layers or auxiliary classifiers [35].

### *2.3. Attention units*

Inspired by the human visual system, attention units can aid CNN models in focusing more effectively on crucial features. Attention models have been extensively studied and applied in various applications, including object recognition [37], language translation [38], semantic segmentation [39], and video classification [40]. Additionally, attention models effectively establish long-range interactions across channels and physical locations.

## 3. Methods

The overall layout of the proposed method is shown in Figure 1, where the model is trained using the input image with its mask containing the cells' borders. The training algorithm is described in Algorithm 1 as follows.

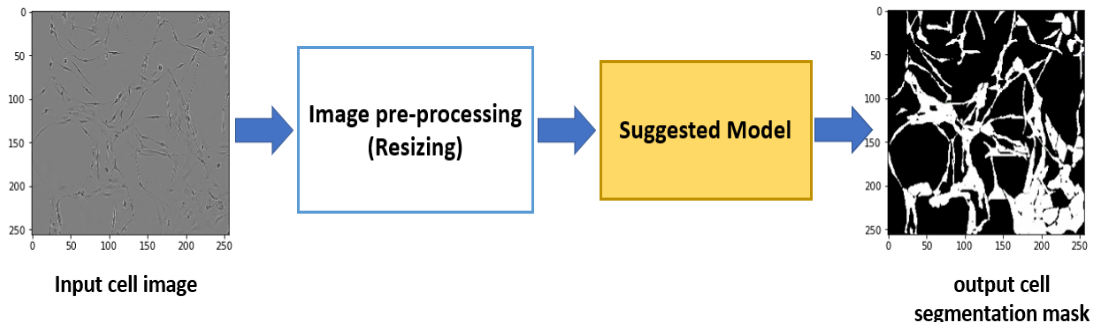


**Figure 1:** The overall steps of the suggested method.

| **Algorithm1**: The training process of the suggested model | |
|---|---|
| *Input* | Input image, mask, E number of epochs |
| *Output* | Trained model |
| *Steps* | **While E ≥ 1 do:**<br>**Read** each image with its corresponding mask<br>**Extract features** using the EffecientNet encoder with the attention mechanism<br>**Build** the mask using the decoder<br>**Calculate the loss**<br>**Re-assign** the model weights<br>**Test** the model using the test data |

Figure 2 illustrates the suggested end-to-end trainable model, which performs cell image segmentation. First, the image input size is set to 256 × 256 before being fed into the network. Then, a series of CNN layers for down-sampling and up-sampling with skip connections and attention units are used to train the model to produce image segmentation masks for neural cells. Below the suggested architecture of the network is introduced.

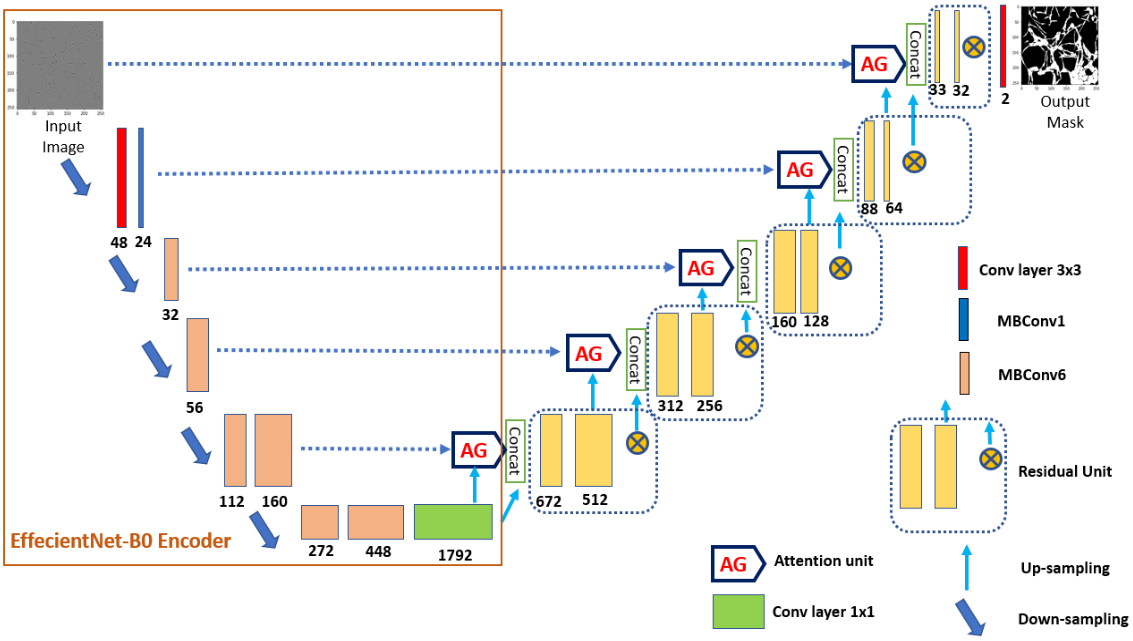


**Figure 2:** The model architecture.

### 3.1 EfficientNet

The EfficientNet structure is used as the encoder part of the proposed model. Since it can balance the model's depth, width, and image resolution, EfficientNet has gotten a lot of attention and is widely utilized in image segmentation and classification. Furthermore, the family of EfficientNet models is efficient and provides better results with considerably fewer parameters. Table 1 shows the number of parameters for some pre-trained models on the ImageNet dataset.

**Table 1:** Number of parameters for some pre-trained models.

| Model | No. parameters in Millions |
|---|---|
| **EfficientNet-B0** | 5.3 |
| **ResNet-152** | 60 |
| **DenseNet-264** | 34 |
| **Inception-v3** | 24 |
| **Xception** | 23 |
| **Inception-resnet-v2** | 56 |

### 3.2 Attention based Residual U-Net

The reason behind using the attention mechanism with the U-Net model is the nature of the U-Net model since it is a multi-stage cascaded convolutional neural network architecture based on encoder-decoders. Additionally, U-Net considers the feature representation from the encoder/down-sampling path when configuring the decoder/up-sampling path. Then a dense layer is employed for prediction. However, the feature vectors at the start of the down-sampling process are not robust, and employing them in conjunction with same-level up-sampling does not yield significant improvement. Therefore, attention units may be employed to solve the previously mentioned issues. Attention takes the same level of down-sampling feature map and the "feature map from one level below" and sends them via an attention gate, which aids in the extraction of a more accurate feature map for contact with the corresponding up-sampling level. Additionally, it aids in focusing on the critical portion of the significant image, which reduces the time per epoch.

### 3.3 Residual/ skip connections

Many efficient deep learning models consist of many stacked CNN layers, which may produce the vanishing gradient problem because, during the backpropagation process, the weight values become extremely diminutive and approach zero. The skip connection has garnered much attention due to its ability to resolve this issue. In addition, employing skip connections in the segmentation task can improve accuracy. Figure 3 shows the difference between CNN layers with and without skip connection.

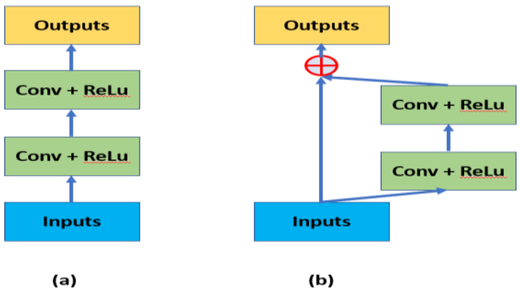


**Figure 3:** (a) Convolutional layers without a skip connection. (b) Convolutional layers with a skip connection.

### 3.4 Model Architecture

Figure 2 illustrates that the presented architecture comprises an encoder and a decoder. The EfficientNet-B0 model was chosen for the encoder part since it has the least number of parameters, reducing the need for high computational power and decreasing the running and prediction time.

Nine stages are stacked to build the encoder (as in Table 2): a 3×3 CNN, 32 mobile reversed bottleneck convolutional MBConv and a 1×1 CNN layer. While the decoder consists of 5 up-sampling and a series of convolution operations, the segmentation results are achieved after the encoder's features are restored to the original image size. To make the segmentation results more accurate, the attention gates are added to each skip connection, reducing the noise by focusing on the important features learned by the attention mechanism and making the model focus more on essential features. Furthermore, batch normalization is applied to accelerate the model convergence, and then the ReLu activation function is applied after each convolution.

**Table 2:** The encoder layers

| Layer number | Size | Layer type |
|---|---|---|
| **1** | 256×256 | Conv 3×3 |
| **2** | 128×128 | Mobile bottleneck conv1, kernel 3×3 |
| **4** | 128×128 | Mobile bottleneck conv6, kernel 3×3 |
| **4** | 64×64 | Mobile bottleneck conv6, kernel 5×5 |
| **6** | 32×32 | Mobile bottleneck conv6, kernel 3×3 |
| **6** | 16×16 | Mobile bottleneck conv6, kernel 5×5 |
| **8** | 16×16 | Mobile bottleneck conv6, kernel 5×5 |
| **2** | 8×8 | Mobile bottleneck conv6, kernel 3×3 |
| **1** | 8×8 | Conv 1×1 |

Each mobile bottleneck convolutional (MBConv) structure is explained in Figure 4. Furthermore, as mentioned earlier, the attention mechanism is used across attention gates AG to focus on essential features. The structure of each AG is illustrated in Figure 5. Firstly, decoding matrix g and encoding matrix x pass in a 1×1 Conv operation parallelly, then apply the ReLu function on the product of addition. Then, a convolutional kernel of size 1x1 and a sigmoid function are applied sequentially. After the resampling process, the attention coefficient α (a hyperparameter which is tuned by the training process to achieve the best results) is obtained. Lastly, the final output is obtained by multiplying the input encoding matrix x by α.

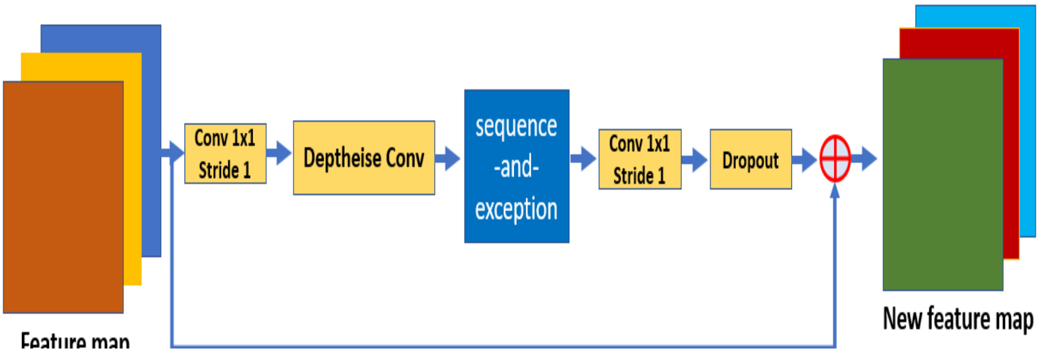


**Figure 4:** MBConv structure.

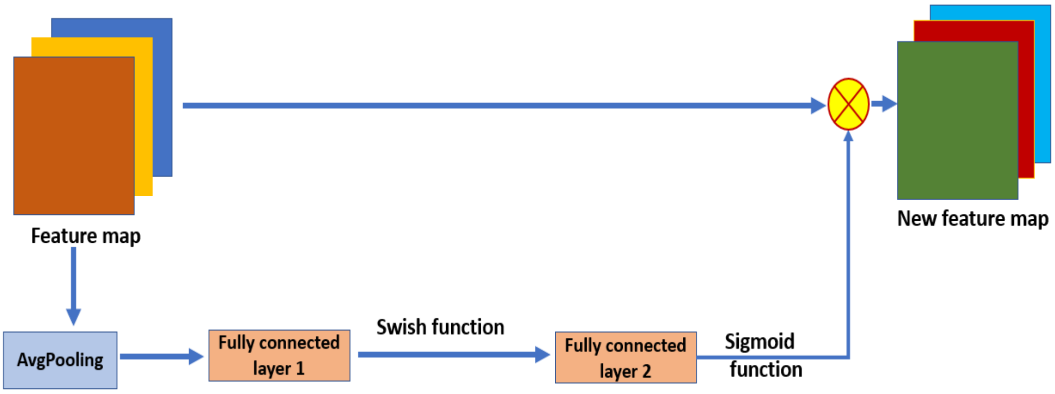


**Figure 5:** Attention gate AG structure.

### *3.5 LinkNet model*

LinkNet adopted the same U-Net structure. The main difference is using add instead of the concatenating operation. Using the same previously described architecture, the network parameters decrease significantly, reducing the training and prediction time as explained in Table 3.

### *3.6 Loss function*

The combo loss function is used in this work to estimate the loss. This function weights the summation of the dice loss and cross-entropy loss, and it exploits the Dice loss flexibility for imbalance and smooths the curves using cross-entropy. It can be calculated as follows:

$$L_{D-BCE} = -\frac{1}{N}\sum_i \beta\,(y - \log(\hat{y})) + (1-\beta)(1-y)\log(1-\hat{y}) \qquad (1) \qquad [40]$$

$$L_{combo}(y,\hat{y}) = \partial\ L_{D-BCE} - (1-\partial)D_{loss}(y,\hat{y}) \qquad (2) \qquad [41]$$

Where, $N$ is the number of samples, $\partial$ is a hyperparameter with value [0,1], y is the ground truth, $\hat{y}$ is the predicted value, and $\beta$ is used for tunning the false negative and false positive. For instance, when $\beta > 1$, the number of false negative is decreased, and when $\beta < 1$, the number of false positive is decreased.

$D_{loss}$ is the dice loss function can be calculated as in Eq. (3):

$$D_{loss}(y,\hat{y}) = 1 - \frac{2y\hat{y}+1}{y+\hat{y}+1} \qquad (3) \qquad [41]$$

## 4. Experiments

### *4.1 Training*

The training was conducted using Keras with a Tensorflow backend as the deep learning framework on Intel(R) Core (TM) i7-10750H CPU @ 2.60GHz with 2.59 GHz, 32.0 Gigabyte of RAM, and a display adapter from NVIDIA GeForce GTX 1650 Ti. Adam optimization algorithm is used to train the network with an initial learning rate of 0.001. Early stopping technique with a patience value of 3 and reduced learning rate with a patience of 2 (a reducing rate of 0.5) are used to avoid overfitting and stop training when there is no improvement.

### *4.2 Dataset*

A dataset from the Kaggle Sartorius – Cell Segmentation competition is used to evaluate the suggested method in this paper. The dataset contains 704x540 pixel 606 images. Although there are few images, the number of annotated objects is reasonably large.

The images are resized to 256 x 256 to decrease the required computational power. Furthermore, the data augmentation techniques are applied to generate more data, improve model generalization ability, and avoid overfitting. The techniques applied include zoom-out, horizontal and vertical flipping, and height and width shifting. Finally, the data was split into 80% for training and 20% for testing.

### *4.3 Evaluation matrices*

Two evaluation matrices were used to evaluate the performance of this method: intersection over union IoU and dice coefficient Dice-Coeff.

The Jaccard Index, often known as the IoU, estimates the proportion of overlap between the ground truth area and the predicted area and is calculated as follows:

$$IoU = \frac{G_{truth} \cap P}{G_{truth} \cup P} \quad (4) \quad [41]$$

Where, $G_{truth}$ is the ground truth area, and $P$ is the predicted area.

Dice-Coeff quantifies the frequency with which the ground truth and predicted areas overlap, and it can be calculated as follows:

$$\text{Dice} - \text{Coeff} = \frac{2|G_{truth} \cap P|}{|G_{truth}|+|P|} \quad (5) \quad [41]$$

## 5. Results and discussion

This section extensively discusses qualitative and quantitative analysis with inference speed and segmentation evaluation.

### *5.1 Computational power comparison*

As mentioned earlier, two architectures are tested in this study: U-Net and LinkNet with attention units. The number of parameters, floating-point operations per second (FLOPs), inference speed, and the required memory storage are used to compare the performance in terms of computing power. Furthermore, U-Net architecture without attention units AU was tested to make this comparison. The LinkNet model with AU reduced the number of parameters by about 40% compared to U-Net with AU and about 6 times less than U-Net without AU, significantly reducing inference time and memory storage requirements. These results make this model suitable for mobile devices or devices with low computational power.

**Table 3:** Performance comparison for the suggested models with U-Net model without attention units

| Network | No. parameters in Millions | FLOPs in Millions | Inference speed | Memory storage |
|---|---|---|---|---|
| **U-Net without AU** | 32 | 64.08 | 205ms | 370.8MB |
| **U-Net with AU** | 10.15 | 24.3 | 57ms | 127.3MB |
| **LinkNet with AU** | **6.09** | **13.7** | **38ms** | **77.6MB** |

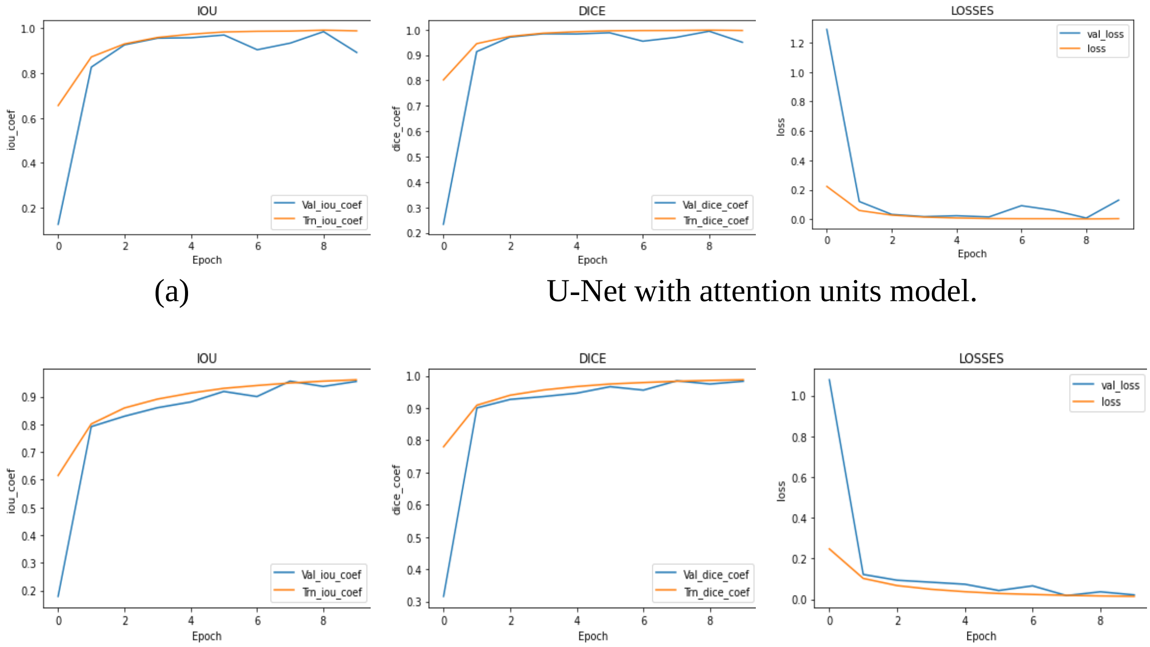
### *5.2 Results on the dataset*

The threshold used to measure both IoU and Dice-Coeff is 0.5. The same dataset is used to compare similar works under the same computing environment. As Table 4 shows, the performance of the suggested method outperforms all the other methods.

**Table 4:** Quantitative results comparison for suggested models with similar works

| Model | IoU | Dice-Coeff | Loss |
|---|---|---|---|
| **U-Net without AU** | 88.2 | 97.2 | 0.054 |
| **[7]** | 78.80 | 85.7 | 0.067 |
| **U-Net with AU (proposed)** | 89.24 | 95.02 | 0.08 |
| **LinkNet with AU (proposed)** | **95.47** | **98.34** | **0.0209** |

Figure 6 shows the training curves for both the proposed models. It is noticed that U-Net without an AU model needs fewer epochs to converge (about 2 epochs), while the LinkNet model is more stable and achieves higher results.

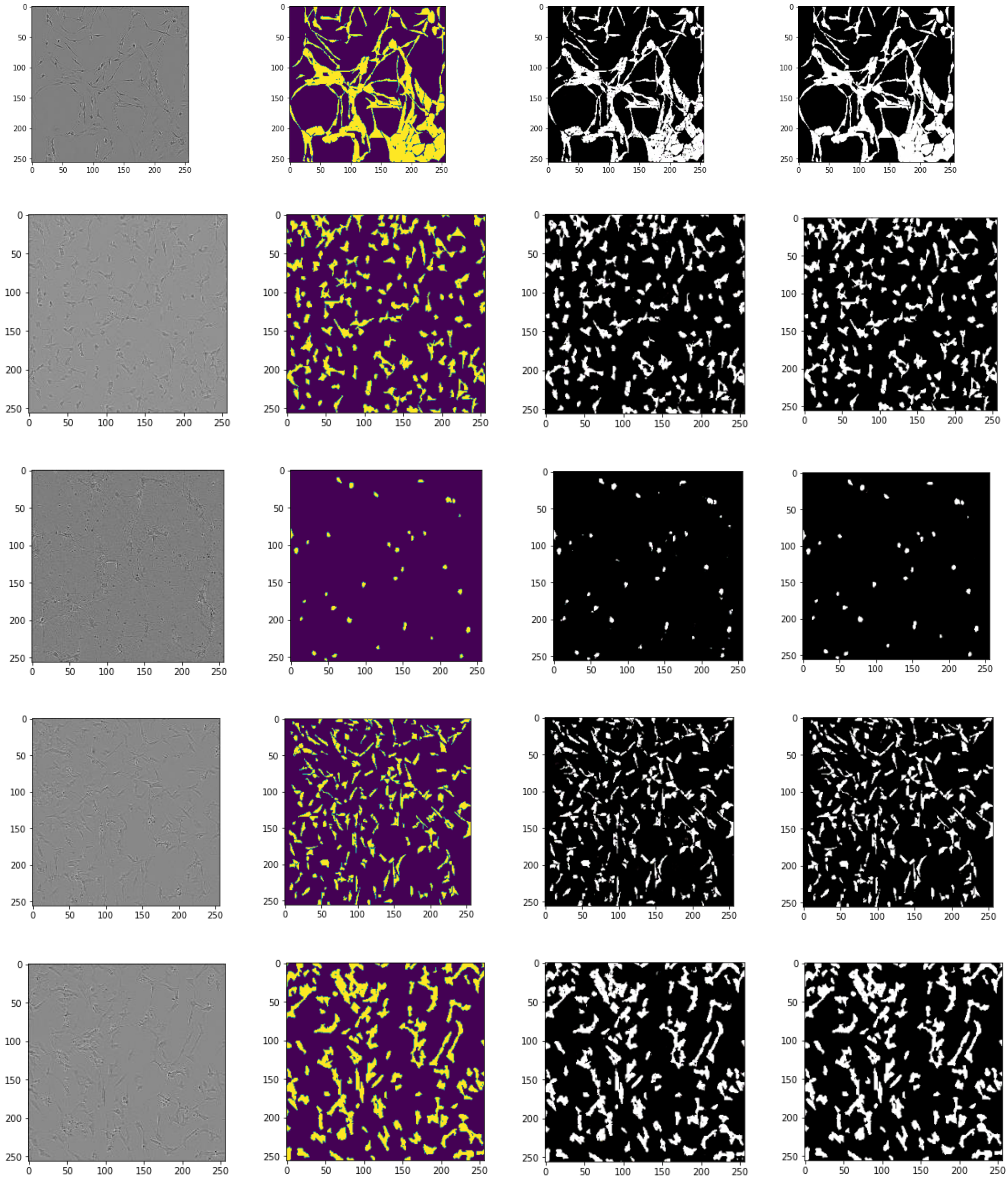


(a) U-Net with attention units model.

(b) LinkNet with attention units model.

**Figure 6:** Training curves for the suggested models.

A qualitative comparison of the proposed methods is shown in Figure 7. The original cell images are illustrated with the ground truth mask and the prediction segmentation mask.



(a) Cell image (b) Ground truth mask (c) U-Net prediction (d) LinkNet prediction

**Figure 7:** Qualitative segmentation results.

## 7. Conclusion

In this work, a neural cell segmentation model is presented. This approach can recalibrate the in-depth features and guide the model to segment the cells accurately with the skip connections and attention units. Furthermore, the proposed model is efficient in inference with low computation power and storage memory, making it suitable for mobile devices. These properties imply that this approach might be well applicable to and beneficial for researching neural cells. It is suggested to test it in the other kinds of cell segmentation for future work.

## References

**[1]** C. L. Gooch, E. Pracht, and A. R. Borenstein, “The burden of neurological disease in the United States: A summary report and call to action,” Annals of Neurology, vol. 81, no. 4. 2017. doi: 10.1002/ana.24897.

**[2]** K. Nishimura, D. F. E. Ker, and R. Bise, “Weakly supervised cell instance segmentation by propagating from detection response,” in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2019, vol. 11764 LNCS. doi: 10.1007/978-3-030-32239-7_72.

**[3]** J. Sun, A. Tárnok, and X. Su, “Deep Learning-Based Single-Cell Optical Image Studies,” Cytometry Part A, vol. 97, no. 3. 2020. doi: 10.1002/cyto.a.23973.

**[4]** J. Yi, P. Wu, Q. Huang, H. Qu, D. J. Hoeppner, and D. N. Metaxas, “Context-refined neural cell instance segmentation,” in Proceedings - International Symposium on Biomedical Imaging, 2019, vol. 2019-April. doi: 10.1109/ISBI.2019.8759204.

**[5]** A. Esteva et al., “Deep learning-enabled medical computer vision,” npj Digital Medicine, vol. 4, no. 1. 2021. doi: 10.1038/s41746-020-00376-2.

**[6]** J. Yi, P. Wu, D. J. Hoeppner, and D. Metaxas, “Pixel-wise neural cell instance segmentation,” in Proceedings - International Symposium on Biomedical Imaging, 2018, vol. 2018-April. doi: 10.1109/ISBI.2018.8363596.

**[7]** J. Yi, P. Wu, M. Jiang, Q. Huang, D. J. Hoeppner, and D. N. Metaxas, “Attentive neural cell instance segmentation,” Medical Image Analysis, vol. 55, 2019, doi: 10.1016/j.media.2019.05.004.

**[8]** K. He, G. Gkioxari, P. Dollar, and R. Girshick, “Mask R-CNN,” in Proceedings of the IEEE International Conference on Computer Vision, 2017, vol. 2017-October. doi: 10.1109/ICCV.2017.322.

**[9]** T. Y. Lin et al., “Microsoft COCO: Common objects in context,” in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2014, vol. 8693 LNCS, no. PART 5. doi: 10.1007/978-3-319-10602-1_48.

**[10]** J. C. Caicedo et al., “Nucleus segmentation across imaging experiments: the 2018 Data Science Bowl,” Nature Methods, vol. 16, no. 12, 2019, doi: 10.1038/s41592-019-0612-7.

**[11]** P. K. Gadosey et al., “SD-UNET: Stripping down U-net for segmentation of biomedical images on platforms with low computational budgets,” Diagnostics, vol. 10, no. 2, 2020, doi: 10.3390/diagnostics10020110.

**[12]** W. Sun and R. Wang, “Fully Convolutional Networks for Semantic Segmentation of Very High Resolution Remotely Sensed Images Combined with DSM,” IEEE Geoscience and Remote Sensing Letters, vol. 15, no. 3, 2018, doi: 10.1109/LGRS.2018.2795531.

**[13]** R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” 2014. doi: 10.1109/CVPR.2014.81.

**[14]** V. Badrinarayanan, A. Kendall, and R. Cipolla, “SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 12, 2017, doi: 10.1109/TPAMI.2016.2644615.

**[15]** E. Shelhamer, J. Long, and T. Darrell, “Fully Convolutional Networks for Semantic Segmentation,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 4, 2017, doi: 10.1109/TPAMI.2016.2572683.

**[16]** O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2015, vol. 9351. doi: 10.1007/978-3-319-24574-4_28.

**[17]** W. Yao, Z. Zeng, C. Lian, and H. Tang, "Pixel-wise regression using U-Net and its application on pansharpening," Neurocomputing, vol. 312, 2018, doi: 10.1016/j.neucom.2018.05.103.

**[18]** V. I. Iglovikov and A. A. Shvets, "TernausNet," in Computer-Aided Analysis of Gastrointestinal Videos, 2021. doi: 10.1007/978-3-030-64340-9_15.

**[19]** O. Russakovsky et al., "ImageNet Large Scale Visual Recognition Challenge," International Journal of Computer Vision, vol. 115, no. 3, 2015, doi: 10.1007/s11263-015-0816-y.

**[20]** Simonyan, K. and Zisserman, A., "Very deep convolutional networks for large-scale image recognition," The 3rd International Conference on Learning Representations (ICLR2015), (2015). https://arxiv.org/abs/1409.1556

**[21]** O. Oktay et al., "Attention U-Net: Learning Where to Look for the Pancreas," Arxiv.org, Apr. 2018. https://arxiv.org/pdf/1804.03999.pdf

**[22]** M. Denil, B. Shakibi, L. Dinh, M. Ranzato, and N. de Freitas, "Predicting parameters in deep learning," Advances in Neural Information Processing Systems 26 (NIPS 2013), 2013.

**[23]** X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices," 2018. doi: 10.1109/CVPR.2018.00716.

**[24]** N. Ahmed, A. Yigit, Z. Isik, and A. Alpkocak, "Identification of leukemia subtypes from microscopic images using convolutional neural network," Diagnostics, vol. 9, no. 3, 2019, doi: 10.3390/diagnostics9030104.

**[25]** D. C. Cireşan, A. Giusti, L. M. Gambardella, and J. Schmidhuber, "Mitosis detection in breast cancer histology images with deep neural networks," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2013, vol. 8150 LNCS, no. PART 2. doi: 10.1007/978-3-642-40763-5_51.

**[26]** F. Xing, X. Shi, Z. Zhang, J. Z. Cai, Y. Xie, and L. Yang, "Transfer shape modeling towards high-throughput microscopy image segmentation," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2016, vol. 9902 LNCS. doi: 10.1007/978-3-319-46726-9_22.

**[27]** F. Xing and L. Yang, "Fast cell segmentation using scalable sparse manifold learning and affine transform-approximated active contour," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2015, vol. 9351. doi: 10.1007/978-3-319-24574-4_40.

**[28]** F. Xing, Y. Xie, and L. Yang, "An automatic learning-based framework for robust nucleus segmentation," IEEE Transactions on Medical Imaging, vol. 35, no. 2, 2016, doi: 10.1109/TMI.2015.2481436.

**[29]** Y. Mao, Z. Yin, and J. M. Schober, "Iteratively training classifiers for circulating tumor cell detection," in Proceedings - International Symposium on Biomedical Imaging, IEEE, 2015, vol. 2015-July. doi: 10.1109/ISBI.2015.7163847.

**[30]** J. Wang, J. D. MacKenzie, R. Ramachandran, and D. Z. Chen, "Neutrophils identification by deep learning and Voronoi diagram of clusters," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2015, vol. 9351. doi: 10.1007/978-3-319-24574-4_27.

**[31]** E. Meijering, "Cell segmentation: 50 Years down the road [life Sciences]," IEEE Signal Processing Magazine, vol. 29, no. 5, 2012, doi: 10.1109/MSP.2012.2204190.

**[32]** I. A. Carreras et al., "Crowdsourcing the creation of image segmentation algorithms for connectomics," Frontiers in Neuroanatomy, vol. 9, no. November, 2015, doi: 10.3389/fnana.2015.00142.

**[33]** M. Maška et al., "A benchmark for comparison of cell tracking algorithms," Bioinformatics, vol. 30, no. 11, 2014, doi: 10.1093/bioinformatics/btu080.

**[34]** H. Chen, X. Qi, J. Z. Cheng, and P. A. Heng, "Deep contextual networks for neuronal structure segmentation," , Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, 2016.

**[35]** Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," 2007. doi: 10.7551/mitpress/7503.003.0024.

**[36]** H. Hu, J. Gu, Z. Zhang, J. Dai, and Y. Wei, "Relation Networks for Object Detection," 2018. doi: 10.1109/CVPR.2018.00378.

**[37]** A. Vaswani et al., "Attention is all you need," in Advances in Neural Information Processing Systems, 2017, vol. 2017-December.

**[38]** L. C. Chen, Y. Yang, J. Wang, W. Xu, and A. L. Yuille, "Attention to Scale: Scale-Aware Semantic Image Segmentation," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2016, vol. 2016-December. doi: 10.1109/CVPR.2016.396.

**[39]** X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local Neural Networks," 2018. doi: 10.1109/CVPR.2018.00813.

**[40]** F. van Beers, A. Lindström, E. Okafor, and M. A. Wiering, "Deep neural networks with intersection over union loss for binary image segmentation," 2019. doi: 10.5220/0007347504380445.

**[41]** J. Ma et al., "Loss odyssey in medical image segmentation," Medical Image Analysis, vol. 71, 2021, doi: 10.1016/j.media.2021.102035.