IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Ingénierie des Systèmes d'Information
Vol. 31, No. 5, May, 2026, pp. 1511-1523
Journal homepage: http://iieta.org/journals/isi

Deep O-Net Model for Medical Image Segmentation and its FPGA Implementation

Ula T. Salim*, Shefa A. Dawwd

Computer Engineering Department, Engineering College, Mosul University, Mosul 41002, Iraq

Corresponding Author Email: ula.tariq@uomosul.edu.iq

https://doi.org/10.18280/isi.310510

Received: 19 December 2025
Revised: 27 April 2026
Accepted: 13 May 2026
Available online: 31 May 2026

Keywords:
medical image segmentation, O-shaped Network, Particle Swarm Optimization, Random Search, Field-Programmable Gate Array acceleration, COVID-19 CT

ABSTRACT

The demand for an efficient, accurate, and fast segmentation system for medical images has increased in many clinical and AI-based medical domains. This work presents a hardware-oriented segmentation framework based on an optimized O-shaped Network (O-Net) architecture, with emphasis on its efficient implementation on Field-Programmable Gate Array (FPGA) devices. The main property of the model is to segment the lung and infections simultaneously, using two low-cost, connected, symmetric custom C-shaped networks (C-Nets) with one layer per level. The suggested model is optimized using Particle Swarm Optimization (PSO), then refined with Random Search (RS) to match the dataset size. The performance is analyzed by applying the optimized O-Net model on the public dataset of CT scans. The assessment results demonstrate that the proposed model is competitive with prior work, achieving high accuracy (0.99) and a good Dice coefficient (0.9913). The optimized O-Net achieved a 24% reduction in model parameters when embedding PSO and RS techniques. Furthermore, based on PSO and RS, the inference time is reduced to 694 ms on the FPGA Zynq board, compared with 1050 ms on a PC with an Intel(R) i7 Graphics Processing Unit (GPU)at 2.9 GHz and an advanced RTX 4060 Ti GPU. The proposed model offers an accurate, fast, and automated mechanism to complement clinical expertise in patient monitoring.

1. INTRODUCTION

In the domain of healthcare, the medical image is an essential tool in controlling the quality of segmentation, and it depends on high-quality medical scanning units. Computed tomography (CT), and X-ray are useful medical imaging types that provide high-quality details to track early the development of diseases and pathological cases [1].

Lung disease is one of the health problems that individuals suffer from and leading to a malfunction in the functionality of the lungs. Segmentation is considered as the first procedure in understanding and interpreting the common lung disease types, such as water density [2], pulmonary fibrosis, and pneumonia [3].

In the last years, the COVID-19 pandemic has swept the world, and its repercussions are still ongoing until now. COVID-19 is a respiratory infection that can be recognized through segmentation of CT lung imaging.

Lung segmentation uses some mechanism to transform a lung image into several meaningful regions or sub-images. It detects the class for every pixel in the lung scan image.

In the real world, the lung regions are delineated visually using medical images by qualified doctors. However, manual segmentation is tedious, not scalable for large image data, and a time-consuming process. Additionally, some errors may appear because the results of delineation details are related to low image contrast, which has an unfair distribution of gray levels and creates noise issues. Moreover, lung scanning is influenced by neighboring tissues and organs, and it is difficult to take multiple medical images due to the effects and harms of the types of radiation. To limit and cope with the spread of viruses and infections, several conventional and automated experiments and solutions have been considered and presented. The conventional methods or extraction features are employed with the help of the developer. Usually, these methods include several procedures related to the threshold, region of interest (ROI), and the convex hull method [4]. Thus, it can be difficult to be general when dealing with multisource images. Today, modern accurate systems are dependent on the integration of Deep learning models. They have as much imitation in data processing as processing takes place in the structure of the human brain. A limited dataset cannot emphasize accurate assessment results, and may not cover all situations. However, the deep segmentation architectures are often constructed using encoding and decoding units with variant scaling levels, like the U-Net and its modified versions [5, 6], SegNet [7], SP-V-Net [8], Anam-Net [9], GFNet [10], DW-UNet [11]. The promise solution merges deep learning models and image algorithms, as addressed in the work [12, 13]. Thereby, cost design is a common issue facing performance; the running speed will be low when applying deep levels. Additionally, the mismatch between the cost design and the dataset's size will contribute to creating an overfitting state and false errors. Consequently, the developments and scientific research have gone towards Hyperparameter Optimization (HPO) strategies that support the functionality of Training Parameters (TPs) together with Design Parameters (DPs). Usually, TPs minimize the score of

the loss function for trial design by utilizing one of the numerical Gradient methods like Adam. Another approach improves accuracy by integrating two or more types of losses, but this will require more inclusive refining of the hyperparameters. To resolve this issue, model-free paradigms like Random Search (RS) are leveraged. In return, the DPs can be solved by building light designs based on human tuning settings [14], but this degrades accuracy. Thus, some of the HPOs apply bioinspired natural algorithms to determine the structure components, such as Particle Swarm Optimization (PSO) [15]. These algorithms mitigated the cost of computations compared to traditional concepts of RS.

In general, the nature optimization concepts add a delay to the running time and may result in deep structure parameters. To complete the implementation, some general works utilized high-performance computing, for example, Graphics Processing Units (GPUs) and Field-Programmable Gate Arrays (FPGAs) [16-18]. The time performance of the diagnosis COVID-19 is compared over several heterogeneous processors, involving multicore CPU, GPU, and FPGA [19].

This work focuses on segmenting the lung and infections in parallel. The major contributions that is to be achieved in this work can be summarized as follows:

1. O-shaped Network (O-Net) attempts to collect lung and infection data by applying the symmetric concept with the encoder-decoder approach.
2. The main property of the suggested O-Net extracts the lung and infection feature from the input images using the same number of layers per level as the previous works based U-Net model. The infrastructure uses absolute difference operations to find the feature maps of the decoding blocks along all levels, instead of using the concatenation operator. These modifications enable the O-Net to offer comparable results; at the same time, the number of parameters will be reduced compared to the hybrid systems.
3. The suggested O-Net is built using two identical new C-shaped networks called C-Nets, then implemented according to the trial method for finding the model and training parameters.
4. RS and PSO are embedded to enhance the accuracy and shorten the selection time of hyperparameters (number of filters, learning rate, kernel size, epochs) of the O-Net infrastructure.
5. The inference for O-Net will be accelerated by conducting it with the help of a high-performance computing FPGA Zynq board and GPU.

This paper is structured into 6 sections: Section 1 gives an introduction to the application of lung segmentation. Section 2 reviews how the researchers deal with the challenges. Section 3 demonstrates the explanations of the suggested methodology. Section 4 describes the experimental details. Section 5 presents the practical results from different views. Also, it discusses and analyzes the main achievement results, comparing them with other prior works. Finally, Section 6 includes a conclusion for the main points of the work. Additionally, offers some future directions for improving the use of the proposed method.

2. LITERATURE REVIEW

This section reviews the extended capabilities of deep learning in addressing the prevalence of the COVID-19 virus. Different ideas and methods have been investigated to handle the segmentation task of the lung and infections in CT images. The U-Net is one of the most prominent networks in the field of image segmentation, with successful predictive results.

U-net and its extensions introduced accurate results via different views. Müller et al. [6] developed an automation pipeline approach that avoids overfitting, which can occur when a limited-sized dataset. The approach involves generating a random image on the fly by cooperating a collection of image processing methods, data augmentation, and 3D U-Net. The validation disc score based on 5-fold results a 0.956 when segmenting lungs, and 0.761 when detecting infection. Saood and Hatem [7] used U-Net and SegNet models to distinguish between healthy lung and infected areas. The experiment's findings demonstrate that SegNet is suitable for classifying whether the regions is uninfected or infected using a mean accuracy of 0.95. On the other hand, the UNet was better for a multi-class segmentation, using a mean accuracy of 0.91. Paluru et al. [9] segmented anomalies found in CT of COVID-19 by proposing a lightweight model named Anam-Net. The reported results after running Anam-Net applied on test samples resulted in good scores of dice coefficient, along with the ability to run on mobile and embedded devices with 17.2MB. Zhang et al. [20] developed a collaborative learning approach, which uses a shared knowledge of unaffected lung images to label lung CT affected by infection of COVID-19. The proposed approach is implemented by creating a model with two symmetric encoders sharing one decoder. The main encoder is dedicated to obtaining the general features of affected COVID lesions. Depending on several unrelated COVID lesions, the second encoder(target) is responsible for the infections of COVID-19. The suggested concept succeeds in dealing with the training of limited-sized data, resulting in scoring of up to 3.0% for Dice similarity coefficient (DSC) and 4.2% for normalized surface dice. Chi et al. [21] segmented the COVID-19 infection parts through the proposed MID-UNet, which fuses the reference input image with their transformations. Also, they proposed a directional convolution block (DCB) that concatenates four direction types of kernels. Additionally, A loss-based contour is proposed for providing a curvature histogram. Aswathy and SS [22] developed dual stages as Cascaded 3D UNet for segmenting the places of infection. The suggested method resulted in scores of (98.07%) accuracy, (98.64%) specificity, (93.47%) sensitivity, and (92.46%) dice score for segmenting the lung. Also, other values (99.20%) accuracy, (99.84%) specificity, (83.33%) sensitivity, and (82%) dice score when segmenting the lung infection. Yin et al. [23] proposed SD-UNet that fuses global context with multi-scale details. The SD-UNet produced segmentation values of (0.9906) Accuracy, (0.8988) Sensitivity, (0.9932) Specificity, (0.8696) DSC, and (0.7702) Jaccard metric, when carried out on binary classes. Additionally, other scores of (0.9821) Accuracy, (0.6169) Sensitivity, (0.9907) Specificity, (0.5936) DSC, and (0.4788) Jaccard metric, when applied to multiclass. Asnawi et al. [24] studied the segmentation for both lung infection through the models of 3D UNet, 3D VGGUNet, 3D ResUNet, and 3D DenseUNet. The models are applied to perform the segmentation for binary and multiclass. The 3D UNet yielded the majority of satisfactory scores (99.37%) accuracy, (97.05%) Dice scores, and (94.32%) IoU when performing the binary task. Besides, 3D UNet produced (81.58%) accuracy, (88.61%) Dice scores, and (98.78%) IoU scores when doing

segmentation for multi-class. Antar et al. [12] proposed a paradigm that integrates image processing and data visualization, with U-Net infrastructure. The experimental outcomes demonstrated a performance of 99.71% for accuracy, 0.87 for precision, 0.83 for sensitivity, and 0.85 for dice coefficient. Newson et al. [25] introduced a simple ED-CNN structure for predicting contours and masks for infected parts within a CT lung image. The results of the model give scores (0.994 ± 0.002) accuracy, (0.996 ± 0.001) specificity, and (0.0075 ± 0.0005) mean absolute error. Additionally, the suggested model employs about 49k parameters over a short time. Lenin Marksia and Yesubai Rubavathi [26] presented a combination approach of extracting features on variance scales, bidirectional pyramid units, and global-edge paths. The presented fusion approach is able to identify different infected part structures within the lung. The performance of validation resulted in scores of 98.6%, 96.54%, 97.52%, 98.55%, and 98.59% for accuracy, precision, sensitivity, specificity, and kappa score, respectively. Their suggested model takes a time of 3.2 s to execute. Zhong and Zhang [27] introduced EF-Net to produce an accurate segmentation for the blurred border of COVID-19 disease. Their model enhances the semantic and texture details through the integration of EDA with FFM blocks. EDA while FFM. EF-Net achieves clear results with Dice rate of 97.3 for the CT scan (CS) dataset. Sarsembayeva et al. [28] enhance the performance quality of U-Net via an approach of a pipeline sequence of several preprocessing algorithms. The proposed approach gives results of IoU and Dice coefficient above 0.95 when conducted on the training dataset. Carannante et al. [29] employed the U-Net as basis, then they trained a deterministic U-Net with three other models based on Bayesian and uncertainly perspective, which are MC-Dropout, SUPER-Net, and ensemble five models of U-Nets, offering a stable performance.

Table 1. The main differences between the previous works based U-Net model and the suggested models

Modifications	Literature Works	Suggested Models	
	U-Net	Our Model C-Net	Our Model O-Net
Task per one time	Perform one segmenting task either lung or infection	Perform one segmenting task either lung or infection	Perform dual segmenting tasks lung and infection and use addition operation
Number of levels	5	5	5
Number of convolution layers per level	2	1	Duplicated C-Net
Number of Convolution with kernel size 1 × 1 used as reduction channel	1	1	1 for each C-Net and 1 to construct O-Net topology
Connecting encoder-decoder	Use concatenation leading to increasing the number of parameters	Use the absolute difference, leading to decreasing the number of parameters	
Selection of number of filters	Fixed 32,64,128,256,512,1024	Selected dynamical using either PSO or RS or merging PSO and RS algorithms	
Kernel size	3 × 3	Selected dynamical using either PSO or RS or merging PSO and RS algorithms which are prove that 3 × 3 is the suitable size	
BatchNormalization	Used	Not used	
Features	Resulting high parameters counts due to trail approach, leading consuming more time.	Reducing both of the parameters counts and time, also maintaining the accuracy because it use dynamic approach and optimizing the model size using PSO and RS.	

Note: O-Net = O-shaped Network; PSO = Particle Swarm Optimization; RS = Random Search.

Mi et al. [30] proposed framework merging enhanced U-Net and Transformer based units to segment lung specified for cancer application. The modifications involve adding Squeeze-and-Excitation networks in skip connections for mininizing the noise, in addition to improve fine-grained lesion by presenting auxiliary losses.

The attention strategies boosts the accuracy by highlighting on the relevant information regions for input but it is raising the complexity. Bougourzi et al. [31] proposed PDAtt-Unet model for segmenting the infection of COVID-19 using the CT images. Their model is based on building the encoder in the same as U-Net, including pyramid routes with attention gates. For the decoder, they use two Att-Unet for segmenting the lung and infection at one time. Besides, an edge loss is used for processing the restriction of the BCE function. The performance of PDAtt-Unet was assessed by handling various datasets either as an intra-way or cross-way, realizing superior results. Buongiorno et al. [32] compared and analyzed the performance of UNet, Attention-UNet, R2-UNet, and R2-Attention UNet to do binary segmentation for the infections of COVID-19. The analysis results show that the Attention-UNet offered the best values of 2D Dice metric of 81.93%.

Mustapha et al. [1] introduced an optimized U-Net model based on merging residual links with new hybrid attention. The model can collect the local and global details and segment both the lung and infection parts. The model was assessed with the COVID-19 CT and COVID-QU-Ext public databases. The mean values of the results for testing COVID-QU-Ex are IoU (0.9740), Dice Coefficient (0.9784), and accuracy (0.990) when segmenting the lung regions. Besides IoU (0.8272), and Dice Coefficient (0.7837), accuracy (0.990) was realized when segmenting the infection regions.

The research gap involves overlapped lung tissues and infections, image quality, limited annotations for patient cases, moderate balancing that allows for merging accuracy, and processing time; the search for new and more efficient methods continues. In the current paper, the optimized O-Net based C-Net design adds new and improved contributions to previous approaches.

The main differences between suggested models and fundamental U-Net and its modifications are summarized in Table 1.

3. METHODOLOGIES

This section describes the optimal framework designed to solve the problem of finding the maximum performance of the

suggested segmentation architecture in the lung application. Moreover, it explains the major components of the framework, which are a combination of O-Net, PSO, and RS. Figure 1 illustrates the schematic representation of the framework.

3.1 Problem definition

The objective is to select the optimal search value for Hyperparameters (HPs) that maximizes the evaluation metrics to consider the model successful in performing CT segmentation. Eq. (1) represents the mathematical objective function with the best solution (Bs).

$$Bs = argmax_{HPs \epsilon \wedge} f(HPs) \quad (1)$$

The factor ∧ refers to the search space for the defined HPs, and f(HP) is the evaluation performance metric. The f is not differentiable and has computational complexity. Thus, both RS and PSO will be considered.

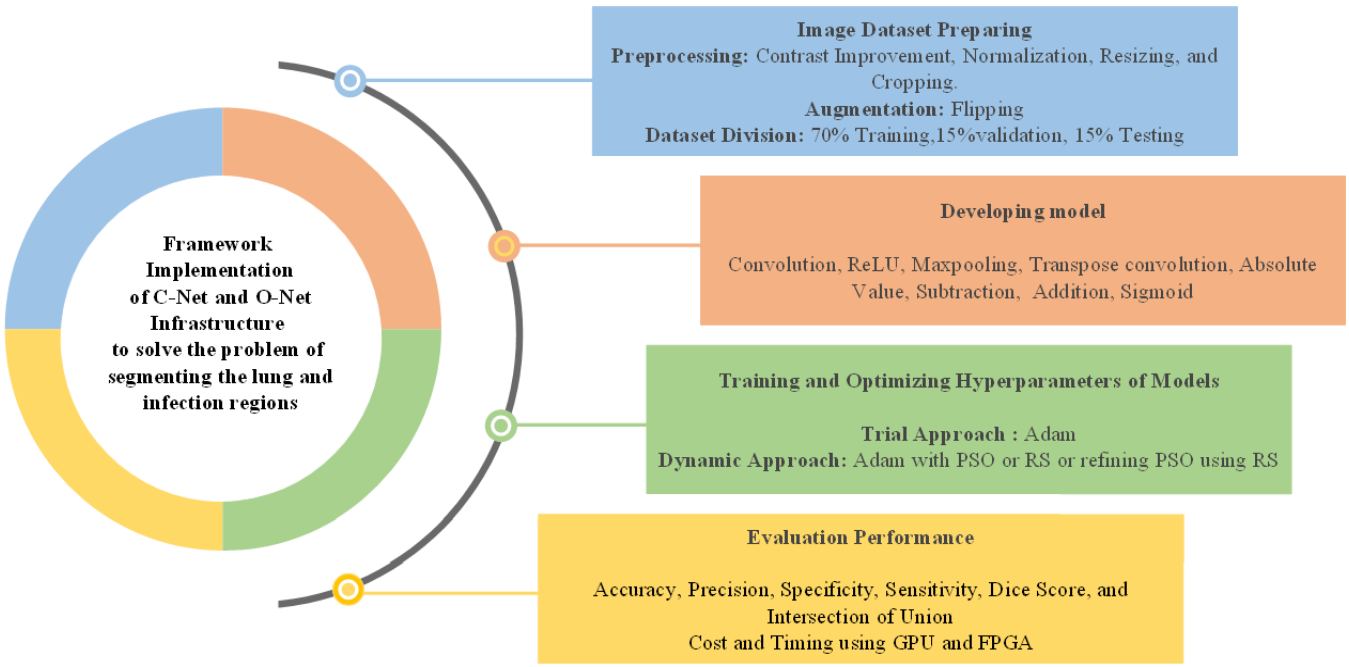


Figure 1. A schematic representation of the overall framework

3.2 Optimized O-shaped Network

Optimized O-Net is an extended and optimized variant model that will be applied to boost the success of producing accurate and fast delineation. It is utilized as a solution for segmenting the lung CT imaging into four regions: infections, right lung, left lung, and background. It takes a small target dataset for CT, then learning through pixel-to-pixel and end-to-end approaches. O-Net is extended to the previous works, which combine the important properties of the basis 2D U-Net with the Siamese or symmetric concept, as illustrated in Figure 2(a). The O-Net segments lung and infection areas simultaneously using two identical low-cost new C-Net architectures, dictating one C-Net for each. Their structure is constructed with the help of diverse units shown in Figure 2(b) and (c), then the information outputs are passed to a channel reduction to allow the O-Net to learn and improve feature correlations among variant channels. Finally, the segmented features for both lung and infections are overlapped using an addition function, followed by a channel reduction of linear 1x1 padded convolution embedded with a Sigmoid activation function, assigning the data vector to the specified classes as depicted in Figure 2(d).

The symmetry concept supports the capturing of more details across various multi-scale levels. Mathematically, all data flow over the O-Net infrastructure can be formulated as follows.

$$LF_{Encoder1} = DS(A_{ReLU}(SC(LF_{input}))) \quad (2)$$

$$IF_{Encoder1} = DS(A_{ReLU}(SC(IF_{input}))) \quad (3)$$

$$LF_{Encoderi} = DS(A_{ReLU}(SC(LF_{Encoderi-1}))) \; 1 \le i \le N \quad (4)$$

$$IF_{Encoderi} = DS(A_{ReLU}(SC(IF_{Encoderi-1}))) \; 1 \le i \le N \quad (5)$$

$$LF_{Decoderi} = A_{ReLU}\left(SC(|LF_{Encoderi} - US(LF_{Encoderi})|)\right) \; 1 \le i \le N \quad (6)$$

$$IF_{Decoderi} = A_{ReLU}\left(SC(|IF_{Encoderi} - US(IF_{Encoderi})|)\right) \; 1 \le i \le N \quad (7)$$

$$LF_{CR} = A_{Sigmoid}(conv_{1\times1}(LF_{Decoder5})) \quad (8)$$

$$IF_{CR} = A_{Sigmoid}(conv_{1\times1}(IF_{Decoder5})) \quad (9)$$

$$F_{CO}^{Classes\times H\times W} = A_{Sigmoid}(conv_{1\times1}(LF_{Decoder5} + IF_{Decoder5})) \quad (10)$$

where, SC represent the standard convolution, A_{ReLU} and $A_{Sigmoid}$ referrer to ReLU and sigmoid functions, CR means channel reduction, DS is down sampling based maxpooling, F_{CO} is fully classification layer, LF means lung feature, IF means infection feature, i level number, and N is count of filters.

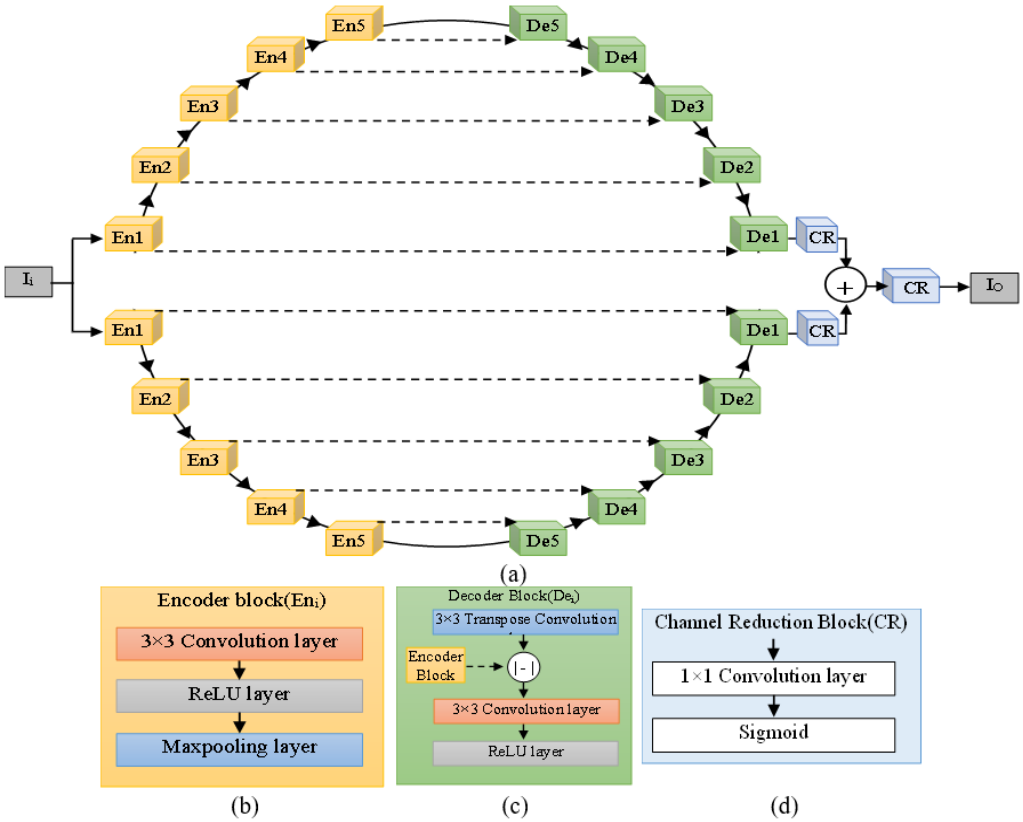


Figure 2. O-shaped Network (O-Net) architecture. (a) encoder and decoder block diagram; (b) components of each encoder block; (c) components of each decoder block; (d) represents the contents of the channel reduction block

3.3 C-Net

The C-Net is a C-shaped topology takes a convolutional neural network (CNN) as a fundamental basis. It is designed with light infrastructure, considering the encoding and decoding approach for learning the input samples in an end-to-end way. The major properties of this architecture are embedding one convolution per level and replacing the concatenation operation with an absolute difference. Also, both the encoder and decoder are designed with five levels where their number of feature filters is determined two times, either using a fixed method or a dynamic method. The fixed method follows the same approach as U-Net, the number of feature filters is times by 2 at the encoding side, while the number of feature filters is divided by a factor of 2 at the decoding side. On the other hand, in the dynamic method, the filter numbers are irregular and specified according to PSO and RS algorithms.

The encoding section consists of down-resolution blocks shown in Figure 2, where the image size is shrunk by a stride of 2 for each block. Figure 2(b) shows a building encoding block, which is a chain of one convolution with the same padding, Rectified Linear Unit (ReLU) activation function, and Maxpooling layer.

In correspondence, the decoding section consists of up-resolution blocks (see Figure 2), where the image size is enlarged by 2 per block.

Figure 2(c) illustrates the building decoding block, which uses transpose convolution to enlarge the incoming encoded data by 2. After that, the corresponding data of the encoding-decoding level undergoes an absolute difference layer, followed by a stack of one convolution layer that applies same-padding, and ReLU.

3.4 Loss function

In various segmentation applications, the Dice loss (L_{DSC}) and Binary Cross-Entropy (L_{BCE}) are combined as a weighted sum to utilize the best possible properties for improving accuracy. The merged loss function is denoted by Eq. (11). This combination helps to speedup the network convergence, but will lead to a need for more computations.

$$L = L_{BCE} + L_{DSC} \tag{11}$$

The L_{BCE} loss is selected to perform binary classification functions and work at the pixel level. It measures the divergence between the true (y) and predicted (p) labels for every pixel i according to Eq. (12). Although it introduces stable, probabilistic learning and handles the issue of gradient hiding, it is ineffective for imbalanced classes and does not immediately measure spatial overlap. Moreover, it is inappropriate for tiny object structures because of the dependence on the quality of image pixels.

$$L_{BCE} = -\frac{1}{N}\sum_{i=1}^{N}\left(y_i \log(p_i) + (1-y_i) \times \log(1-p_i)\right) \tag{12}$$

On the contrary, the L_{DSC} loss mitigates the effect of the unbalanced category. It finds the overlaps between true and predicted areas using Eq. (13). Nonetheless, the L_{DSC} has less stability during the training; the unsuitable weights will create inaccurate results. Additionally, it is more complicated and may need soothing methods.

$$L_{DSC}=1-2\frac{\sum_{i=1}^{N} y_i p_i}{\sum_{i=1}^{N} y_i^{\,2}+\sum_{i=1}^{N} p_i^{\,2}} \quad (13)$$

The value of N denotes the overall number of image pixels, yi represents the segmented value labeled by the expert, while pi represents the segmented value predicted by the suggested network. Both of y_i and p_i are computed for every pixel and belong to the set {0,1}.

3.5 Particle Swarm Optimization

PSO is an efficient meta-heuristic algorithm that simulates the behavior of a swarm in nature, such as birds flocking, and was developed by Vanneschi and Silva [33]. The PSO is utilized to optimize complex problems concerning a defined criterion, known as the objective function. It is simple and does not differentiate the specified objective function, also containing a few parameters. PSO includes a group of candidate solutions referred to as swarm particles(m), where every particle(i) is expressed in the Eq. (14) as a position vector of a collection of different parameters (X_i^d) that needs to be optimized to realize a maximum or minimum objective function f(HPs) given in Eq. (1).

$$X_i = [x_i^1, x_i^2, x_i^3, \dots x_i^k] \quad (14)$$

All the particles move through the search space given in Eq. (15) and change their location according to the variables of velocity, personal best, and the global best. Besides, their velocity and position are updated using Eqs. (16) and (17). The particle continues searching the dedicated space until it finds the optimum and best solution (d) for the problem.

$$x^k \in [x^k_{min}, x^k_{max}] \quad (15)$$

where, x^k_{min} and x^k_{max} are minimum and maximum limitations for K-th parameter(particle).

$$V_i^{k+1} = w.V_i^k + c_1.r_1.\left(pbest_i - X_i^k\right) + c_2.r_2.\left(gbest_i - X_i^k\right),\ i \in [1,2,3,\dots,m] \quad (16)$$

$$X_i^{k+1}=X_i^k+V_i^{k+1} \quad (17)$$

where, d denote the number of parameters, i represent particles index, X_i^k and V_i^k referred to the vector position and velocity for the particle i at some iteration k, $pbest_i$ denotes to a personal best position that by a particle i detected, gbest represent the best position that all swarm detected, w denotes to the inertia weigh, c1 and c2 represent the acceleration coefficients, and finally both r1 and r2 are some random values within a range of [0,1].

3.6 Random Search

RS is a simple numerical approach that solves the complicated problem efficiently when searching several hyperparameters[34]. It boosts fast implementations, and can use one of the techniques to stop at an appropriate time to preserve available computational hardware. In RS, the maximum or minimum objective function f(xi) can be obtained by picking out a random group of configurations(xi) uniformly from a dedicated range of hyperparameters R over N iterations. Mathematically, this is formulated as in Eq. (18).

$$X_i = [x_i^1, x_i^2, x_i^3, \dots x_i^k], X \subseteq R \quad (18)$$

For each iteration(i), the objective function is calculated by Eq. (19) and compares if the new value (Bi) is more reliable than the current best function factor (Bs), then both of will be updated as in Eqs. (20) and (21).

$$Bs=f(X_i) \quad (19)$$

$$Xbest=X_i \quad (20)$$

$$Bs=B_i \quad (21)$$

The random search for the best solution is repeated until the specified trial number (N) is reached or there is no improvement, and the nearest factor for the desired objective is generated.

4. DATASET, ENVIRONMENT, AND IMPLEMENTATION SPECIFICATIONS

This section clarifies the requirements necessary to execute the suggested solutions, covering the dataset, image processing paradigms, software, and hardware, ending with an explanation of both training and testing implementations.

4.1 Assessment dataset

In this article, the public dataset of COVID-19 CT scans is utilized [35]. It contains 20 volume CT samples with infection, left and right lungs, annotated by two radiologists and validated by a specialist radiologist. The data cases contains multiple slice, size of each CT slice in the dataset is 512 × 512 and in Nifti format.

4.2 Dataset preparation

The CT scan images undergo several preprocessing stages to improve performance.

Resizing: The dataset is rescaled to a size of 256 × 256 to avoid a mismatch in size and to match the input dimensions of the suggested architecture.

Improve contrast: CLAHE is applied to the dataset to alleviate the noise effect and get clearer feature extraction.

Normalizing: The dataset is normalized to a consistent range of [0,1] to support the generality and enhance the stability and convergence speed.

Data split: The overall lung dataset is divided using two approaches to confirm fair performance comparison and support the generality. In the first approach, the data divided randomly to construct portions of 70% for training, 15% for validation, and the remaining 15% is dedicated to testing. The second approach is splitting the dataset slices using 4-fold cross validation protocol but the first three folds are specified for training and validation stages then mean from three folds is computed, while the fold 4 is specified for testing.

Cropping: This step helps in extracting the target object based on the boundary, leading to a reduction in image noise and focusing on the necessary features.

Augmentation: To increase amount of data, avoid overfitting and deal with scanning variations, the training

dataset is artificially augmented, likely the reference images, using two types of geometry transformations of vertical and horizontal flips.

4.3 Experimental setup

The training stage for the suggested architecture is executed on an advanced PC with an Intel(R) i7 CPU that includes 8 cores and 16 GB of RAM working at a frequency of 2.9 GHz. The machine is equipped with an advanced RTX 4060 Ti, which has 4352 cores and 16 GB of memory. The architecture is built with Windows 11 (64-bit), Python 3.8 language interfaced with Tensorflow 2.9.0, Keras 2.9.0 frameworks, and VHDL. On the other hand, the test stage is implemented using the same RTX card, as well as it is realized using a high-performance computing FPGA board named Zynq (XC7Z020).

4.4 Training strategies and implementations

The suggested C-Net and O-net structures are trained according to assembling of the Binary Cross Entropy (BCE) and Dice Loss (DL) shown in Eqs. (1) and (2). The ADAM, PSO, and RS optimizers have been leveraged as objective functions to make the disparity between reference and predicted labels small. The preprocessed CT images are segmented in four experiments with 8 batch samples at the step-training, to be compatible with the memory bounds. At each experiment, several epochs with variant HPs refinements are applied until the training process reaches the cloeset best DSC results, depending on HPs shown in Table 2. Therefore, the specification analysis of O-Net for each experiment is shown in Table 3.

The first experiment develops the C-Net topology concerning the fixed method, using the same filter numbers as the U-Net model. The HPs are adjusted according to the classical approach, try-and-error.

The second, third, and fourth experiments are refining the model parameters (MPs) and the training parameters (TPs) dynamically to create the C-Net structure.

The second experiment uses a PSO algorithm shown in Figure 3(a). The PSO is initiated with parameters of a maximum number of particles (N = 8), a maximum number of iterations (8 and 20), w equal to 0.5, c1 and c2 are equal to 1. In addition, the search space is defined as in Table 2, while Xi is selected randomly within the specified space, Vi is set to zero, and both r1 and r2 take random values.

The third applies the RS method as illustrated in Figure 3(b). The RS is configured with a maximum number of iterations (8 and 20).

Finally, the fourth experiment, which considers the main objective and Leveraged consecutive of PSO and RS approaches. The best outcome parameters from the PSO at experiment 2 are refined and optimized by RS.

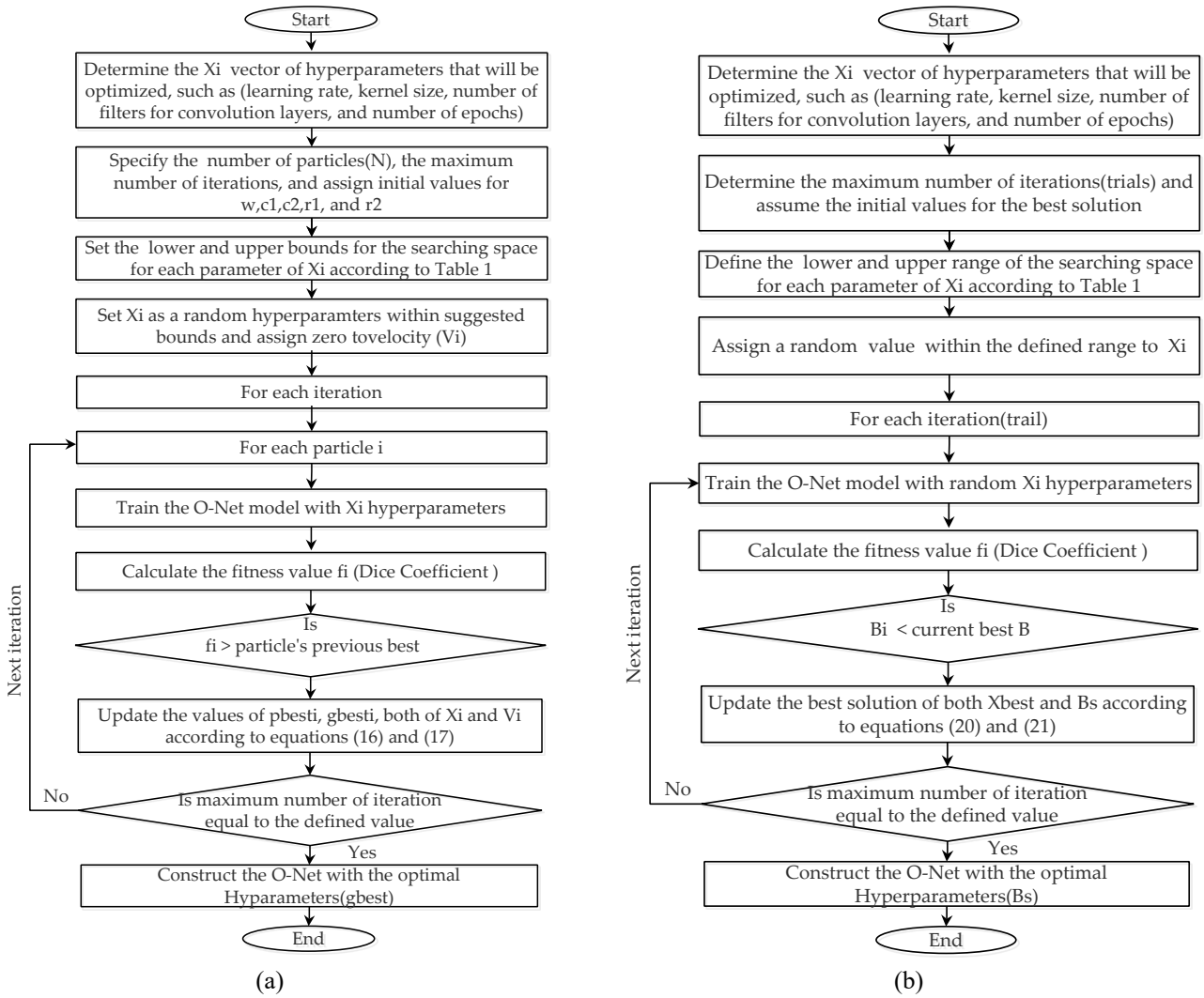


Figure 3. Workflow of optimization algorithms: (a) Particle Swarm Optimization (PSO) and (b) Random Search (RS)

Table 2. Hyperparameter setups of the trial and optimization of the O-shaped Network (O-Net) architecture

Parameter	Trial O-Net	Range	Optimal Values					
			PSO		RS		PSO and RS	
			A		B		C	
Max. iteration	-	8 or 20	20	8	20	8	20	8
Mini Batch size	8	8	8	8	8	8	8	8
Learning rate	0.0001	[0.0001-0.005]	0.0031405	0.00165605	0.00091537	0.00163479	0.00092024	0.00202068
Epochs	100	[5-100]	97	20	95	24	79	47
Optimizer	Only Adam	Adam, PSO, RS	Adam, PSO		Adam, RS		Adam, PSO, RS	
EN1(conv)&DN1(conv)	32	[16-31]	28	18	22	27	25	19
EN2(conv)&DN2(conv)	64	[32-63]	36	42	59	44	47	61
EN3(conv)&DN3(conv)	128	[64-127]	113	64	87	77	111	109
EN4(conv)&DN4(conv)	256	[128-255]	243	140	207	164	217	137
EN5(conv)&DN5(conv)	512	[256-512]	434	393	276	355	256	285
Kernel size	3 × 3	[3-5]	3	3	3	3	3	3

Note: PSO = Particle Swarm Optimization; RS = Random Search.

Table 3. Specification analysis of the O-shaped Network (O-Net) infrastructure at different experiments

Level	Layers	n	Feature Maps			
			Trail	PSO	RS	PSO and RS
				($H \times W \times A_{20}$ or A_8)	($H \times W \times B_{20}$ or B_8)	($H \times W \times C_{20}$ or C_8)
Input	-	1	256 × 256 × 1	256 × 256 × 1	256 × 256 × 1	256 × 256 × 1
EN1	Conv3 × 3,ReLU	2	256 × 256 × 32	256 × 256 × 28 or 18	256 × 256 × 22 or 27	256 × 256 × 25 or 19
	Maxpool		128 × 128 × 32	128 × 128 × 28 or 18	128 × 128 × 22 or 27	128 × 128 × 25 or 19
EN2	Conv3 × 3,ReLU	2	128 × 128 × 64	128 × 128 × 36 or 42	128 × 128 × 59 or 44	128 × 128 × 47 or 61
	Maxpool		64 × 64 × 64	64 × 64 × 36 or 42	64 × 64 × 59 or 44	64 × 64 × 47 or 61
EN3	Conv3 × 3,ReLU	2	64 × 64 × 128	64 × 64 × 113 or 64	64 × 64 × 87 or 77	64 × 64 × 111 or 109
	Maxpool		32 × 32 × 128	32 × 32 × 113 or 64	32 × 32 × 87 or 77	32 × 32 × 111 or 109
EN4	Conv3 × 3,ReLU	2	32 × 32 × 256	32 × 32 × 243 or 140	32 × 32 × 207 or 164	32 × 32 × 217 or 137
	Maxpool		16 × 16 × 256	16 × 16 × 243 or 140	16 × 16 × 207 or 164	16 × 16 × 217 or 137
EN5	Conv3 × 3,ReLU	2	16 × 16 × 512	16 × 16 × 434 or 393	16 × 16 × 276 or 355	16 × 16 × 256 or 285
	Maxpool		8 × 8 × 512	8 × 8 × 434 or 393	8 × 8 × 276 or 355	8 × 8 × 256 or 285
DN5	Tconv2 × 2, sub, abs, Conv3 × 3, ReLU	2	16 × 16 × 512	16 × 16 × 434 or 393	16 × 16 × 276 or 355	16 × 16 × 256 or 285
DN4	Tconv2 × 2, sub, abs, Conv3 × 3, ReLU	2	32 × 32 × 256	32 × 32 × 243 or 140	32 × 32 × 207 or 164	32 × 32 × 217 or 137
DN3	Tconv2 × 2, sub, abs, Conv3 × 3, ReLU	2	64 × 64 × 128	64 × 64 × 113 or 64	64 × 64 × 87 or 77	64 × 64 × 111 or 109
DN2	Tconv2 × 2, sub, abs, Conv3 × 3, ReLU	2	128 × 128 × 64	128 × 128 × 36 or 42	128 × 128 × 59 or 44	128 × 128 × 47 or 61
DN1	Tconv2 × 2, sub, abs, Conv3 × 3, ReLU	2	256 × 256 × 32	256 × 256 × 28 or 18	256 × 256 × 22 or 27	256 × 256 × 25 or 19
Output C-Net	Conv1 × 1, Sigmoid	2	256 × 256 × 1	256 × 256 × 1	256 × 256 × 1	256 × 256 × 1
Output branch	Add	1	256 × 256 × 1	256 × 256 × 1	256 × 256 × 1	256 × 256 × 1
Output	Conv1 × 1, Sigmoid	1	256 × 256 × 1	256 × 256 × 1	256 × 256 × 1	256 × 256 × 1

* **n** represents a similar duplicated layer and A,B,C represent the optimal values and variables based on the Maximum Iteration 20 or 8

Note: PSO = Particle Swarm Optimization; RS = Random Search.

4.5 Hardware implementation

The developed architecture, described in Section 2, takes time to execute because it receives three images (reference, lung, and infection) from the input directions. To raise the speedup of execution, the architecture is converted to a hardware-based FPGA, but it will be restricted by the available specifications of the FPGA. Thus, four solutions are applied as shown in the designed Figure 4. The first solution is to store each column image in software as a row image in the FPGA's memory.

The second solution will read three pixels of the image samples stored in DDR memory for the targeted device from three ports in one clock in parallel using a time-sharing approach. The data width of each pixel is read as 8 bits, then these bits are distributed to the lung and infection feature registers. Each of the three pixels corresponds to one row of a 3x3 register (kernel's size), and the address will be calculated based on image size and kernel size. This approach will eliminate the need for memory storage, which is used for the convolution operation based on a sliding window. All units are designed with 16-bit for the incoming and outgoing feature values. However, some delay will appear due to the multi-level nature of the O-Net. Therefore, the fourth solution will be running the symmetric networks(C-Nets) in parallel, and each convolution unit will use 9 DSPs.

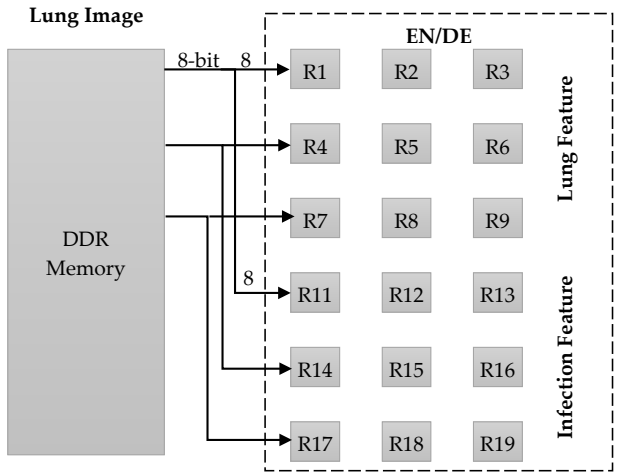


Figure 4. Hardware design for one filter of one encoding and decoding at some level

5. EXPERIMENTAL RESULTS

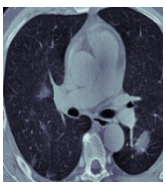
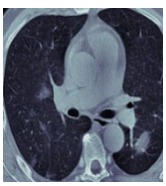
This section explores all the results obtained from the practical implementations. It contains visual interpretations and the overall mathematical scores for the entire test dataset. The key factors that are contribute in success of the suggested models are aggregating complementary features using the duplicated C-Net designs, and integrating PSO and RS natural algorithm to obtain the optimal training and model parameters

5.1 Visualization results of segmentation

Table 4 gives an immediate visual comparison of the manual and automated delineations for selecting one CT sample.

As demonstrated, both the O-Net including C-Net has good segmented images in focusing on infection parts within both lungs at different positions, especially O-Net, which presented high differentiation between the background, lungs, and infections. Although there is diversity among optimizing model approaches, the O-Net model provided accurate segmentation results with minimum mean squared error (MSE). According to the visual images and MSE factor, the first experiment with trial O-Net has the least error and yields accurate results. On the other hand, the approaches of PSO and RS is affected by reducing the number of parameters and the maximum iteration value, but still reliable and approximately matched the reference images with the other experiments, the fourth experiment is considered the moderate approach. This indicates the suggested models have been able to introduce satisfactory segmentation results.

Table 4. Results of random sample using expert and automated models of O-Net for together infection and lungs when the maximum iteration for PSO and RS is equal to 20 and 8

Original CT	Expert			Predicted			
	Lung	Infection	Lung + Infection	Trial O-Net	O-Net with PSO	O-Net with RS	O-Net with PSO and RS
							
MSE				0.0104	0.0213	0.0178	0.0199
Maximum iteration equal to 20							
							
MSE				0.0104	0.0195	0.0179	0.0200
Maximum iteration equals to 8							

Note: CT = Computed tomography; O-Net = O-shaped Network; MSE = mean squared error; PSO = Particle Swarm Optimization; RS = Random Search.

5.2 Statistical analysis

To quantify the effective performance of the adopted approach, six metrics are reported as shown in Table 5. Accuracy (Acc), DSC, precision (Prec), specificity (SPE), sensitivity (SEN), and Intersection of Union (IOU). Table 6 presents the practical values of measurements using random dataset spliting. As noted, the C-Net and O-Net based on PSO and RS optimizations realizes remarkable performance in terms of DSC, and IOU in segmenting infection areas, while the model tuned by trail approach has the highest results when segmenting both lung regions and introduced a good performance when segmenting infection parts. The diversity is a reflection of the configuration of MPs and TPs. Alongside, applying the absolute value instead of the concatenation approach, which decreases the number of features at each level, and the segmentation combination of lung and infection.

To validated generality of the presented models C-Net and O-Net, the experiment table 7 gives a report for the obtained results using 4-fold cross validation. As seen good scores are achieved and the fold 3 results the top scores and its weights are used to assess models using test set(fold4). However, the dice score decreased compared to random split approach because the number of trained pattern is also reduced.

5.3 Cost analysis

As shown in Table 8, the software cost for the Trial O-Net is the highest, due to the different nature of the structure parameters. On the other hand, the combination of PSO and RS represents the balanced solution. To speed up the testing phase and work within the available range of hardware resources, all experimental models are structured units based on fixed symmetric C-Nets designs with a semi-parallelism component per unit level and leveraging 9 DSP units to design parallel convolution, but this has led to some delay, although each one contains a semi-parallelism component per unit level. However, the proposed design is flexible and can be accommodated for diverse tasks by changing the distribution of workload with respect to the capabilities of the available FPGA platform.

5.4 Time analysis

Table 9 compares the execution time of different experiments conducted on variant speeds of platforms. As illustrated, the tuning time for the second experiment based on the PSO method is less than the tuning time of both RS and hybrid PSO and RS at the same number of iterations. This is

because the PSO's searching mechanism is guided and performed according to the details of prior iterations obtained from a sharing group of particles, while the RS does not depend on a specified direction point. In addition, the approximated HPs produced from PSO have created a model with a smaller number of parameters compared to the RS strategy.

Table 5. Performance metrics are used to assess the quality of segmentation

Metric	Definition	Formula
Acc	Express the overall areas which is segmented correctly relative to all tested samples	(TP+TN)/(TP+TN+FP+FN)
DSC	Compute the similarity between two samples	(2 × TP)/(2 × TP+FP+FN)
Prec	Express the positive instances that are correctly predicted relative to all predicted instances as positive	(TP)/(TP+FP)
SEN	Express the percentage of normal areas which is segmented correctly	(TP)/(TP+FN)
SPE	Express the percentage of abnormal areas which is segmented correctly	(TN)/(TN+FP)
IOU	Represent the overlap between the true and predicted annotations	(TP)/(TP+FP+FN)

Note: TP = True Positive; TN = True Negative; FP = False Positive; FN = False Negative; Acc = Accuracy; DSC = dice similarity coefficient; Prec = precision; SEN = sensitivity; IOU = Intersection of Union.

Table 6. Practical scores based on C-Net and O-Net architectures with random data spling and different implementations for lung segmentation, infection segmentation, lung and infection segmentations

Segmented Area	Metric	Trial	PSO (A)		RS (B)		PSO and RS (C)	
			MI = 20	MI = 8	MI = 20	MI = 8	MI = 20	MI = 8
Lung	Acc	**0.9782**	0.9163	0.9701	0.9565	0.9724	0.975	0.9668
	Prec	**0.9445**	0.7983	0.921	0.8923	0.9289	0.9359	0.9156
	DSC	**0.9667**	0.8833	0.9551	0.9353	0.9583	0.9621	0.95
	IOU	**0.9803**	0.9778	**0.9837**	0.9659	0.9793	0.9798	0.9753
	SEN	**0.9726**	0.8822	0.9599	0.9441	0.9643	0.968	0.9571
	SPE	**0.99**	0.9888	**0.9918**	0.9827	0.9895	0.9898	0.9874
Infection	Acc	0.988	0.9809	0.9871	0.9902	0.9889	**0.992**	0.9864
	Prec	**0.9726**	0.8346	0.8738	0.9307	0.8565	0.929	0.8379
	DSC	0.8303	0.7362	0.8335	0.8719	0.8649	**0.8979**	0.8304
	IOU	0.5677	0.7098	0.6621	0.695	0.7756	**0.7679**	0.6993
	SEN	**0.9991**	0.9945	0.9951	**0.9974**	0.9938	0.9972	0.9932
	SPE	0.7243	0.6586	0.7967	0.8201	**0.8736**	0.8687	0.8231
Lung and infection	Acc	0.9879	0.9405	0.9855	0.9625	0.9895	**0.99**	0.9732
	Prec	0.9933	0.8601	**0.9782**	0.9162	0.9844	0.9834	0.9353
	DSC	0.9821	0.9193	0.9788	0.9472	0.9847	**0.9856**	0.9621
	IOU	0.944	0.9751	0.9599	0.9617	0.9706	0.976	**0.9814**
	SEN	**0.9966**	0.9160	0.9886	0.9531	0.9918	0.9913	0.9641
	SPE	0.971	0.9874	0.9796	0.9805	0.9851	0.9879	**0.9906**

Note: Acc = Accuracy; DSC = dice similarity coefficient; Prec = precision; SEN = sensitivity; IOU = Intersection of Union; O-Net = O-shaped Network; PSO = Particle Swarm Optimization; RS = Random Search; MI represent the maximum number of iterations

Table 7. Dice rates resulted from validation C-Net and O-Net model using four fold protocol, Particle Swarm Optimization (PSO), and Random Search (RS) at maximum iteration of 20

Fold	Lung Using C-Net	Infection Using C-Net	Lung and Infection Using O-Net
Fold 1	0.8250	0.9364	0.9668
Fold 2	0.8255	0.9334	0.9676
Fold 3	0.83	0.9256	0.9686
Mean	0.8268	0.9318	0.9677
Fold 4	0.8352	0.9240	0.9740

Table 8. Software and Hardware costs for implementing the optimised O-Net design

Implementation	Cost	Trial O-Net	O-Net with PSO		O-Net with RS		O-Net with PSO and RS	
			Maximum Iteration = 20	Maximum Iteration = 8	Maximum Iteration = 20	Maximum Iteration = 8	Maximum Iteration = 20	Maximum Iteration = 8
Software	Number of parameters	12,915,140	9,821,030	6,237,116	5,084,692	5,881,040	5,065,574	4,347,512
	Memory(software) in MB	49.27	37.46	23.79	19.4	22.43	19.32	16.58
	GOPs	21.8	13.2	7.5	11.9	10.4	12.5	11.3
Hardware	Slices				2621			
	LUTs				8123			
	FFs				5620			
	DSPs				193			

Note: O-Net = O-shaped Network; PSO = Particle Swarm Optimization; RS = Random Search.

Table 9. Running time for variant stages and using different processors

Phase	Method	Time in Sec	
		Maximum Iteration = 20	Maximum Iteration = 8
Training/epoch	Trial O-Net	34.39	
	O-Net with PSO	28.26	22.57
	O-Net with RS	25.23	27.88
	O-Net with PSO and RS	25.77	24.44
HPO	PSO	48.31	22.7
HPO	RS	26661.23	7013,35
HPO	Refining PSO using RS	24332.69	8934.68
Test/image(GPU)	Trial O-Net	1.32	
	O-Net with PSO	1.22	0.96
	O-Net with RS	1.04	0.97
	O-Net with PSO and RS	1.05	0.935
Test/image(FPGA)	Trial O-Net	1.050464	
	O-Net with PSO	0.90441266	0.69592
	O-Net with RS	0.689569	0.706502918
	O-Net with PSO and RS	0.694861	0.64723583

Note: O-Net = O-shaped Network; PSO = Particle Swarm Optimization; RS = Random Search.

Table 10. Comparison of DSC and IOU scores between C-Net design and baseline works for lung segmentation

Ref.	Architecture	Image size	Performance(%)	Optimizer
[1]	ECA-Attention with DANet	256 × 256	Acc: 97.57 DSC: **97.14** IOU: 97.75	Bayesian optimization (TPE)
Our	C-Net with Trail	256 × 256	Acc: **97.82** DSC: 96.67 IOU: **98.03**	Only ADAM
	C-Net with PSO and RS	256 × 256	Acc: 97.5 DSC: 96.21 IOU: 97.98	PSO and RS

Note: Acc = Accuracy; DSC = dice similarity coefficient; IOU = Intersection of Union; The bold values are considered the highest results.

Table 11. Comparison of evaluation time and cost between the optimised O-Net design and baseline works

Ref.	Architecture	Image Size	Testing Time/Step in ms	Training Time in min	Model size in MB
[1]	ECA-Attention with DANet	256 × 256	**22** 8 images/step	-	**54.27**
Our	Trail	256 × 256	One image GPU:1320, FPGA: 1050	47.1/100	49.27
Our	Optimized O-Net Using PSO and RS	256 × 256	One image For maximum iteration = 20 GPU: 1050, FPGA: 694 For maximum iteration = 8 GPU: 935, FPGA: 647	33.93/79	19.32 for Maximum iteration = 20 16.58 for Maximum iteration = 8
[9]	Anam-Net	512 × 512	**362**	**27/100 epochs**	**17.21**
[21]	MID-UNet	512 × 512	1360	1012.8/100 epochs	-
Our	O-Net with PSO and RS	512 × 512	For maximum iteration = 20 GPU: 3130, FPGA: 2779	-	19.32 for Maximum iteration = 20

Note: O-Net = O-shaped Network; PSO = Particle Swarm Optimization; RS = Random Search; The bold values are the best reported results.

Another observation, the training time for experiment 1 using the trial approach consumes the largest implementation time compared to both PSO and RS versions, which are approximately equal, and takes the least time. The cause of equaling PSO and RS is due to having an approximate total count of operations that is decreased by the GPU optimization matrix and run-to-run variations implemented. Besides the size of memory, restrict the epoch period.

Finally, the table demonstrates that the testing time of implementing all experiments on an FPGA is more suitable than a GPU for real-time applications when using 16-bit precision. The reason is that the two C-Nets are designed to work in parallel.

5.5 Discussion and comparison with benchmark architectures

Although the related researchers' work produces remarkable success in terms of accuracy, it suffers from a long time, computation overheads and memory threshold. On the other hand, the suggested O-Net aims to yield a balance point between the cost and the accuracy criteria. This section discusses the most significant improvement implications for the presented model (O-Net) to work within resource-limited environments with good performance results. Tables 10 and 11 abstract all the advancements of the suggested concepts.

In general, the reported results stand out the optimising the model design's effect on the accuracy and time, where deep models give promising scores.

Table 9, the current work-based trail method achieved the best acc and IOU scores when segmenting lung areas, and the work [1] has the highest DSC score. However, the results of the C-Net with PSO and RS are considered a good enhancement. This variation is due to the used machine and HPs. On the other hand, Table 10 reveals the diversity between the time and mode size across variant models. The ECA-

Attention with DANet developed with a large model size and consumes the largest amount of memory when compared with current models at the same image resolution. In contrast, the Anam-Net needs the smallest memory size when compared with the combination of PSO and RS at a maximum iteration of 20 when applied on the same image size of 512 × 512. The increase in model parameters, together with image resolution, is reflected in the training and testing time as seen in Table 10. The infrastructure of ECA-Attention with DANet takes the smallest testing time, and the Anam-Net consumes the smallest training time when compared to the suggested models. The suggestion gives moderated measures compared to the related works and can be developed for the future by applying and implementing more advanced optimizing for the hardware design.

6. CONCLUSION AND FUTURE WORK

This paper presents an optimized O-Net architecture for segmenting either the lungs or infectious parts, or both tasks at the same time. The suggested approach is based on inserting two new similar topologies named C-Net, which alleviates the overhead cost by including the absolute difference as an alternative to concatenation and utilizing two convolution layers per level, one for lung segmentation and the other for segmenting the infections. To complete the work and avoid overfitting cases, augmentation, RS, and PSO algorithms are applied to balance between accuracy, dataset availability, number of parameters, and training time. Moreover, the high capability of the FPGA platform is utilized to improve the testing time. The integration both PSO with RS provides the tradeoff experimental results, with comparable accuracy metrics at FPGA testing time of 694 msec. Although the augmentation and fold validation are used, the work is tested on a specific task dataset of COVID-19 and the tunning-based RS algorithm consumes time, so the accuracy and time can be further improved. Future directions will include exploring other optimization algorithms to improve the model and expanding the work to apply it to different large dataset applications with performing parallel optimization and measuring power consumption.

REFERENCES

[1] Mustapha, B., Zhou, Y., Nawel, B., Chunyan, S., Zhitao, X. (2025). Optimized attention U-Net for enhanced lung and area of infection segmentation in chest X-rays and CT scans. Journal of Radiation Research and Applied Sciences, 18(3): 101650. https://doi.org/10.1016/j.jrras.2025.101650

[2] Keen, C., Grenier, J., Šereš, P., Stobbe, R., et al. (2025). MRI assessment of lung water density in individuals previously infected with COVID-19: A cross-sectional study. Journal of Magnetic Resonance Imaging, 62(3): 767-778. https://doi.org/10.1002/jmri.29814

[3] Kim, J., Chae, G., Kim, W.Y., Chung, C.R., et al. (2024). Pulmonary fibrosis followed by severe pneumonia in patients with COVID-19 infection requiring mechanical ventilation: a prospective multicentre study. BMJ Open Respiratory Research, 11(1): e002538. https://doi.org/10.1136/bmjresp-2024-002538

[4] Zheng, J., Wang, L., Gui, J., Yussuf, A.H. (2024). Study on lung CT image segmentation algorithm based on threshold-gradient combination and improved convex hull method. Scientific Reports, 14(1): 17731. https://doi.org/10.1038/s41598-024-68409-4

[5] Ronneberger, O., Fischer, P., Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical Image Computing and Computer-Assisted Intervention, Germany, pp. 234-241. https://doi.org/10.1007/978-3-319-24574-4_28

[6] Müller, D., Soto-Rey, I., Kramer, F. (2021). Robust chest CT image segmentation of COVID-19 lung infection based on limited data. Informatics in Medicine Unlocked, 25: 100681. https://doi.org/10.1016/j.imu.2021.100681

[7] Saood, A., Hatem, I. (2021). COVID-19 lung CT image segmentation using deep learning methods: U-Net versus SegNet. BMC Medical Imaging, 21(1): 19. https://doi.org/10.1186/s12880-020-00529-5

[8] Zhao, C., Xu, Y., He, Z., Tang, J., et al. (2021). Lung segmentation and automatic detection of COVID-19 using radiomic features from chest CT images. Pattern Recognition, 119: 108071. https://doi.org/10.1016/j.patcog.2021.108071

[9] Paluru, N., Dayal, A., Jenssen, H.B., Sakinis, T., Cenkeramaddi, L.R., Prakash, J., Yalavarthy, P.K. (2021). Anam-net: Anamorphic depth embedding-based lightweight CNN for segmentation of anomalies in COVID-19 chest CT images. IEEE Transactions on Neural Networks and Learning Systems, 32(3): 932-946. https://doi.org/10.1109/TNNLS.2021.3054746

[10] Fan, C., Zeng, Z., Xiao, L., Qu, X. (2022). GFNet: Automatic segmentation of COVID-19 lung infection regions using CT images based on boundary features. Pattern Recognition, 132: 108963. https://doi.org/10.1016/j.patcog.2022.108963

[11] Chen, C., Zhou, J., Zhou, K., Wang, Z., Xiao, R. (2021). DW-UNet: Loss balance under local-patch for 3D infection segmentation from COVID-19 CT images. Diagnostics, 11(11): 1942. https://doi.org/10.3390/diagnostics11111942

[12] Antar, S., Abd El-Sattar, H.K.H., Abdel-Rahman, M.H., FM Ghaleb, F. (2023). COVID-19 infection segmentation using hybrid deep learning and image processing techniques. Scientific Reports, 13(1): 22737. https://doi.org/10.1038/s41598-023-49337-1

[13] Subramaniam, U., Subashini, M.M., Almakhles, D., Karthick, A., Manoharan, S. (2021). Research article an expert system for COVID-19 infection tracking in lungs using image processing and deep learning techniques. Lung, 110: 30. https://doi.org/10.1155/2021/1896762

[14] Jannat, M., Birahim, S.A., Hasan, M.A., Roy, T., et al. (2025). Lung segmentation with lightweight convolutional attention residual u-net. Diagnostics, 15(7): 854. https://doi.org/10.3390/diagnostics15070854

[15] Wang, W., Pei, Y., Wang, S.H., manuel Gorrz, J., Zhang, Y.D. (2022). PSTCNN: Explainable COVID-19 diagnosis using PSO-guided self-tuning CNN. Biocell, 47(2): 373-384. https://doi.org/10.32604/biocell.2023.025905

[16] Le Blevec, H., Léonardon, M., Tessier, H., Arzel, M. (2023). Pipelined architecture for a semantic segmentation neural network on fpga. In 2023 30th IEEE International Conference on Electronics, Circuits and Systems (ICECS), Istanbul, Turkiye, pp. 1-4.

https://doi.org/10.1109/ICECS58634.2023.10382715
[17] Posso, J., Kieffer, H., Menga, N., Hlimi, O., et al. (2025). Real-time semantic segmentation of aerial images using an embedded u-net: A comparison of cpu, gpu, and fpga workflows. arXiv preprint arXiv:2503.08700. http://arxiv.org/abs/2503.08700
[18] Dawwd, S.A. (2013). The multi 2D systolic design and implementation of Convolutional Neural Networks. In 2013 IEEE 20th International Conference on Electronics, Circuits, and Systems (ICECS), Abu Dhabi, United Arab Emirates, pp. 221-224. https://doi.org/10.1109/ICECS.2013.6815394
[19] Goel, G., Gondhalekar, A., Qi, J., Zhang, Z., Cao, G., Feng, W. (2021). ComputeCOVID19+: Accelerating COVID-19 diagnosis and monitoring via high-performance deep Learning on CT images. In Proceedings of the 50th International Conference on Parallel Processing, Lemont IL, USA, pp. 1-11. https://doi.org/10.1145/3472456.3473523
[20] Zhang, Y., Liao, Q., Yuan, L., Zhu, H., Xing, J., Zhang, J. (2021). Exploiting shared knowledge from non-COVID lesions for annotation-efficient COVID-19 CT lung infection segmentation. IEEE Journal of Biomedical and Health Informatics, 25(11): 4152-4162. https://doi.org/10.1109/JBHI.2021.3106341
[21] Chi, J., Zhang, S., Han, X., Wang, H., Wu, C., Yu, X. (2022). MID-UNet: Multi-input directional UNet for COVID-19 lung infection segmentation from CT images. Signal Processing: Image Communication, 108: 116835. https://doi.org/10.1016/j.image.2022.116835
[22] Aswathy, A.L., SS, V.C. (2022). Cascaded 3D UNet architecture for segmenting the COVID-19 infection from lung CT volume. Scientific Reports, 12: 3090. https://doi.org/10.1038/s41598-022-06931-z
[23] Yin, S., Deng, H., Xu, Z., Zhu, Q., Cheng, J. (2022). SD-UNet: A novel segmentation framework for CT images of lung infections. Electronics, 11(1): 130. https://doi.org/10.3390/electronics11010130
[24] Asnawi, M.H., Pravitasari, A.A., Darmawan, G., Hendrawati, T., Yulita, I.N., Suprijadi, J., Nugraha, F.A.L. (2023). Lung and infection CT-scan-based segmentation with 3D UNet architecture and its modification. Healthcare, 11(2): 213. https://doi.org/10.3390/healthcare11020213
[25] Newson, K.S., Benoit, D.M., Beavis, A.W. (2024). Encoder-decoder convolutional neural network for simple CT segmentation of COVID-19 infected lungs. PeerJ Computer Science, 10: e2178. https://doi.org/10.7717/PEERJ-CS.2178
[26] Lenin Marksia, U., Yesubai Rubavathi, C. (2024). Accurate segmentation of COVID-19 infected regions in lung CT scans with deep learning. Neural Computing and Applications, 36(35): 22511-22531. https://doi.org/10.1007/s00521-024-10336-6
[27] Zhong, W., Zhang, H. (2024). EF-net: Accurate edge segmentation for segmenting COVID-19 lung infections from CT images. Heliyon, 10(23): e40580. https://doi.org/10.1016/j.heliyon.2024.e40580
[28] Sarsembayeva, T., Mansurova, M., Abdildayeva, A., Serebryakov, S. (2025). Enhancing u-net segmentation accuracy through comprehensive data preprocessing. Journal of Imaging, 11(2): 50. https://doi.org/10.3390/jimaging11020050
[29] Carannante, G., Bouaynaya, N., Dera, D., Fathallah-Shaykh, H., Rasool, G. (2026). Testing the trust: verification and validation of bayesian segmentation under uncertainty. In Medical Imaging with Deep Learning-Validation Papers.
[30] Mi, F., Xu, Y., Mi, S. (2026). LungNet: Leveraging state-space models with SE-enhanced skip connections for precise CT-based lung lesion segmentation. PloS One, 21(4): e0346561. https://doi.org/10.1371/journal.pone.0346561
[31] Bougourzi, F., Distante, C., Dornaika, F., Taleb-Ahmed, A. (2023). Pdatt-unet: Pyramid dual-decoder attention unet for covid-19 infection segmentation from ct-scans. Medical Image Analysis, 86: 102797. https://doi.org/10.1016/j.media.2023.102797
[32] Buongiorno, R., Del Corso, G., Germanese, D., Colligiani, L., Python, L., Romei, C., Colantonio, S. (2023). Enhancing covid-19 CT image segmentation: A comparative study of attention and recurrence in unet models. Journal of Imaging, 9(12): 283. https://doi.org/10.3390/jimaging9120283
[33] Vanneschi, L., Silva, S. (2023). Particle Swarm Optimization. In Lectures on Intelligent Systems, Cham: Springer International Publishing, pp. 105-111. https://doi.org/10.1007/978-3-031-17922-8_4
[34] Bergstra, J., Bengio, Y. (2012). Random search for hyper-parameter optimization. Journal of Machine Learning Research, 13(2): 281-305. https://doi.org/10.5555/2188385.2188395
[35] Jun, M., Cheng, G., Yixin, W., Xingle, A., et al. (2020). COVID-19 CT lung and infection segmentation dataset. CiNii Research. https://doi.org/10.5281/zenodo.3757476