Received: October 16, 2023. Revised: December 28, 2023.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

R2U-Net3+: Recurrent Residual Convolutional Network U-Net3+ for Segmentation of Radio Frequency Interference (RFI) in Weather Radar Images

Alfan Alfarisy[1] **Handayani Tjandrasa[1]***

[1]*Department of Informatics, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia*
* Corresponding author's Email: handatj@its.ac.id

Abstract: In recent years, radio frequency interference (RFI) has emerged as a significant challenge in weather radar systems due to the escalating demand on the RF spectrum. This interference results in distortions in radar imagery, diminishing the accuracy of weather forecasts. While current mitigation methods rely on threshold algorithms and linear detection, there's been a notable enhancement with the incorporation of deep learning techniques, especially through convolutional neural networks (CNNs). In this context, we introduce the R2U-Net3+ architecture, a fusion of U-Net3+ and recurrent residual convolutional neural networks (RRCNN). Focusing on detecting RFI-affected areas in radar images, this architecture has exhibited commendable results in tests. Based on evaluation metrics, R2U-Net3+ achieved a dice coefficient of 94.980% with a single recurrence iteration and improved to 95.352% with two iterations. For the IoU, R2U-Net3+ scored 90.745% with one iteration and 91.409% with two iterations. These results affirm that R2U-Net3+ offers a significant improvement in detecting RFI, positioning it as a leading solution to challenges faced by weather radar systems.

Keywords: Weather radar, Radio frequency interference, Recurrent residual network, U-Net3+.

1. Introduction

Weather radar systems are fundamental in today's meteorology, offering crucial data for real-time weather monitoring and forecasting [1]. However, these systems often face radio frequency interference (RFI), leading to distortions and inaccuracies in radar imagery, which affects the reliability of the resulting weather forecasts [2].

Currently, the increasing demand for bandwidth in the radio frequency (RF) spectrum has led to shared use, raising significant concerns about the potential impact of unwanted RF signals on weather radars [3]. A key example of this is the allocation of the 5470–5720 MHz band to mobile services, including radio local area networks (RLANs), approved by the world radio-communication conference in 2003 [4]. Although weather radars have a dedicated bandwidth in the 5600-5650 MHz (C-Band) frequency range, there is a partial overlap with bands used by existing C-band radars, creating potential RFI issues [5].

This overlapping, especially with C-Band WIFI networks, causes recurrent RFI in weather radar data, showing as dots, spokes, or stripes in radar images [6]. Therefore, it is essential to develop methods to effectively identify and mitigate RFI to preserve the quality of radar data.

Previous research in this domain primarily centered on the use of U-Net architectures integrated with convolutional neural networks (CNN) for image-based segmentation to address RFI challenges [7–9]. While these preliminary investigations have provided insights into potential mitigation techniques and methodologies, there remains significant scope for refinement and optimization in model performance. Acknowledging the prevalent use of the base U-Net architecture in previous research and addressing the identified gaps, this study embarks on a significant advancement by introducing the R2U-Net3+ architecture. This innovative model is a carefully crafted combination of U-Net3+ [10] and recurrent residual convolutional networks (RRCNN) [11]. U-Net3+ an evolution of the widely used U-Net [12] and its enhanced version, U-Net++ [13] has a

proven track record in segmentation tasks. The integration of RRCNN, inspired by other U-Net variants like R2U-net [14] and R2U++ [15], brings a dynamic advantage to RFI identification in weather radar images through the R2U-Net3+ architecture. By leveraging full-scale skip connections for enriched feature representation, R2U-Net3+ sets the stage for improved performance with less parameter complexity than its predecessors. The goal of this research is to broaden the scope of RFI identification.

This research endeavor aims to address the persistent challenges of RFI in weather radar images with a robust and innovative approach. Our project not only introduces a solution to these challenges but also encourages advancements in interconnected fields, particularly in the domain of image segmentation. Key contributions include: (a) Precise localization and segmentation of RFI areas in weather radar images, offering a more nuanced grasp of RFI's implications. (b) The introduction of the pioneering R2U-Net3+ architecture, tailored for adeptly segmenting RFI areas. (c) A proven superior performance of the R2U-Net3+ architecture compared to its contemporaries, as evidenced by benchmarks like the dice coefficient, IoU score, and pixel-level accuracy.

The paper is organized as follows: after the introduction, section 2 presents a literature review, section 3 outlines the methodology, section 4 describes the experimental setup and results, and section 5 concludes the study.

2. Related work

Methods for detecting radio frequency interference (RFI) are essential in various fields, including radio communications, weather radar systems, and satellite technologies. Historically, RFI detection has mainly used threshold algorithms focusing on the physical characteristics of RFI in the time-frequency domain, with linear detection being commonly used [16-18].

However, this approach revealed certain limitations. While effective to some extent, the SumThreshold mask exhibited inaccuracies, including erroneously masked pixels and undetected RFI pixels. These limitations underscored the challenges in setting parameters for mask dilation and smoothing, as the characteristics of RFI vary significantly during different times, making it challenging to identify a singular parameter set that effectively addresses all scenarios. This underscores the complexity of accurately mitigating RFI contamination in observational data, especially in scenarios with varying RFI characteristics over time [7].

Semantic segmentation technologies have advanced significantly in recent years, becoming vital in applications such as medical image analysis, robotic perception, and satellite image segmentation [19-21]. The evolution of convolutional neural networks (CNN) introduced groundbreaking architectures like fully convolutional networks (FCNs) [22], U-Net [12], PSPNet [23], and various versions of DeepLab [24-26], boosting the capabilities of semantic segmentation technologies.

Expanding on this groundwork, researchers achieved notable milestones in the realm of RFI detection using CNN technology, particularly in image segmentation. A significant advancement was made [7], employed a vanilla U-Net architecture for RFI mitigation, setting a new standard in RFI area identification in 2D data, surpassing traditional RFI marking algorithms such as SEEK's SumThreshold in receiver operating characteristic (ROC) and precision-recall assessments. This innovation inspired further developments, with Yang et al. [8] enhancing the vanilla U-Net to develop the RFI-Net, showing promising results in RFI detection. Additionally, studies by Lepetit et al. [9] explored using the same segmentation frameworks for different applications, expanding the field's possibilities.

However, the U-Net architectures used in earlier research are foundational and have seen substantial enhancements, one of which is U-Net3+. This enhanced U-Net variant has proven to be particularly effective in segmentation tasks, delivering superior results. This version improves the traditional U-Net by introducing full-scale skip connections, enabling the architecture to capture details at various semantic levels and achieving this with fewer parameters and improved performance. This innovation allows the architecture to understand details of different sizes from the feature maps created by each convolutional block. Notably, U-Net3+ does this with fewer parameters than its predecessors while enhancing performance, offering a well-balanced compromise between complexity and capability.

The advancements in U-Net architectures do not stop at skip connections; the backbone of the U-Net has also seen significant improvements. A key development has been the introduction of recurrent convolutional layers (RCL) [11], allowing the network to utilize sequential information in the data. This led to the creation of frameworks like R2U-Net (Fig. 1a) and R2U++ (Fig. 1b) infusing them with enhanced performance. Integrating RCL into U-Net architecture represents a shift towards efficiency and deeper analysis [14, 15].

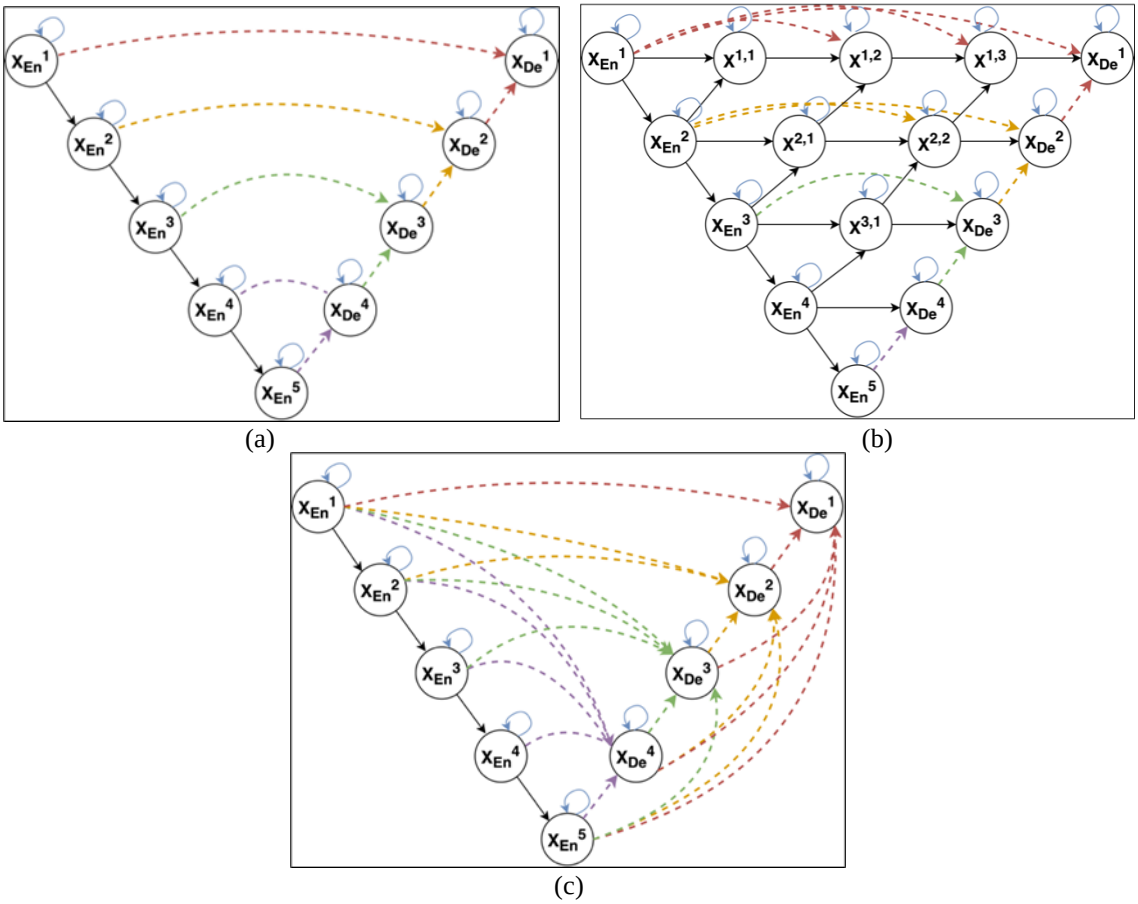


Figure. 1 Integration of the RRCNN with different U-Net architectures: (a) R2U-Net, (b) R2U++, and (c) proposed method: R2U-Net3+

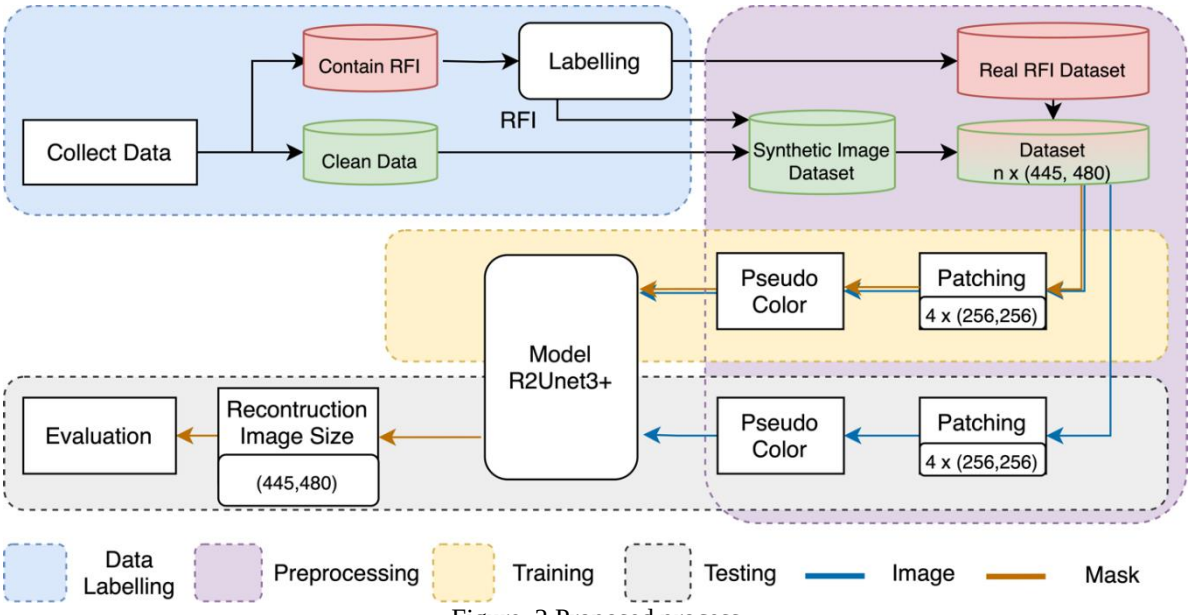
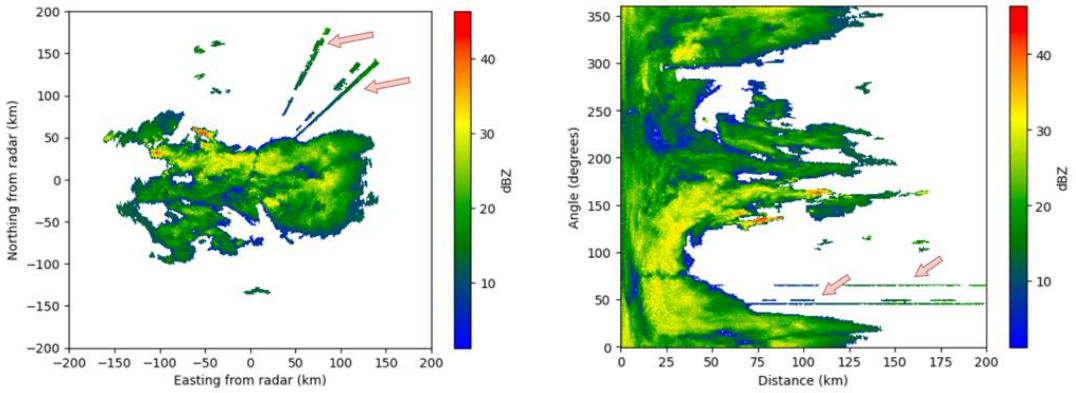


Figure. 2 Proposed process

The use of RCL enables a deeper understanding of patterns in RFI-affected zones while maintaining a streamlined parameter count due to weight sharing. This balance between depth and efficiency in the framework opens up possibilities for architectures with unprecedented accuracy in semantic segmentation tasks while optimizing computational resource utilization. In this context, the role of RCL in enhancing the network layers without increasing parameter count is highlighted, emphasizing its



(a) (b)

Figure. 3 Visualization reflectivity in: (a) Plan position indicator (PPI) and (b) Cartesian plot

contribution to improving semantic segmentation tasks [11]. This reflects a commitment to innovation and resource efficiency, paving the way for growth and continuous improvement in the sector.

In pursuit of advancing previous research in RFI detection through image segmentation models based on the U-Net architecture, we propose a novel architecture. Illustrated in Fig. 1c, our innovation combines RRCNN with U-Net3+, resulting in our newly proposed model, R2U-Net3+. This design aims to harness the strengths of U-Net3+ while integrating the advantages of recurrent convolutional layers for superior performance. With R2U-Net3+, we seek to push the boundaries of RFI detection capabilities, leveraging the latest advancements to provide even more accurate and efficient solutions in the field of semantic segmentation.

3. Research method

Our research method begins with extracting weather data, which forms the base for the subsequent steps. Next, we carefully assemble our dataset to ensure it is relevant and sturdy for the upcoming tasks. During the pre-processing phase, we apply three crucial steps: augmentation to diversify the data; enhancement to boost image quality using methods like pseudo-coloring; and patching to maintain image proportions for efficient data management. Once the data is pre-processed, we implement the proposed advanced R2U-Net3+ model, aiming to maximize the dataset's potential. The model is thoroughly trained to align with the data's intricacies. After training, we test the model to assess its effectiveness in real-world situations. Fig. 2 provides a clear overview of this research process.

3.1 Weather radar data

In this study, the data used is collected from a Gematronik radar system. This system saves raw data in *.vol* format or volumetrically, which we then convert into JSON format with the help of wradlib [27]. This conversion enables the extraction of both the data and its associated metadata. The reflectivity (dBZ) data is organized as a matrix, with columns representing all possible measurable angles, and rows representing the range bins.

We will represent this dBZ data as pixels to serve as model input. During a full cycle, the data undergoes 445 sweeps, resulting in an angular resolution that can be calculated as:

$$a = \frac{360^\circ}{sweeps} = \frac{360^\circ}{445} \approx 0.8^\circ \quad (1)$$

In this case, the number of sweeps is 445, and each sweep contains 480 data points, covering distances up to 240 km. The radial resolution, or distance resolution, can be calculated with the following formula:

$$\delta r = \frac{\text{max distance}}{\text{n of data points}} = \frac{240\text{ km}}{480} = 0.5\text{ km} \quad (2)$$

To provide a more in-depth understanding, the data is represented in two formats: polar (Fig. 3a) and Cartesian (Fig. 3b). While the polar format provides immediate geospatial insights, we find the Cartesian format to be more advantageous, particularly for input model semantic segmentation. This means that each data point covers a half-kilometer distance. This data comes from the first layer, where radio frequency interference (RFI) is commonly observed [28].

3.2 Build dataset

Upon preparing the images in Cartesian format, they are imported into Label Studio, an open-source platform specializing in data annotation [29]. In this stage, our team, consisting of three experts in radar

imagery, identifies and marks the areas affected by radio frequency interference (RFI) by drawing bounding boxes around these areas.

Following the annotation, we start the mask creation process. In this step, any pixel within the bounding boxes that is different from the background is assigned a value of 1. This method leverages the fact that RFI areas are predominantly isolated from the primary reflectivity. Consequently, every pixel inside the bounding box and different from the background value is designated a value of 1, representing the mask. This results in a clear visual representation of the regions affected by RFI, facilitating further data processing steps.

3.3 Pre-processing

The pre-processing stage is essential in refining the dataset for subsequent analyses and consists of three key steps: data augmentation, image enhancement, and patching. Each of these steps plays a critical role in ensuring the reliability and quality of the final output.

3.3.1. Data augmentation

To foster a more robust semantic segmentation model, we employ data augmentation to generate synthetic radar visuals. This technique involves utilizing two primary types of data: "RFI-Clean Data" visuals, which are devoid of radio frequency interference (RFI), and "RFI-Affected Data," which feature real-world interference patterns.

From the RFI-affected visuals, we derive an "RFI Blueprint" that captures the essence of interference elements. This blueprint is subsequently superimposed onto the clean data, resulting in a synthesized set of radar visuals that balance the interference-free backdrop with controlled RFI elements. This method enables the exploration of diverse interference scenarios using actual radar signatures.

3.3.2. Image enhancement

In this phase, we focus on enriching the visual quality of the images. The original images, with values ranging between 0 and 65, necessitate enhancement techniques to extract detailed insights. We employ a pseudo-coloring technique, a widely recognized method in image processing, to augment the contrast and details, thereby facilitating easier data interpretation and analysis[30–33].

The applied colormap features a sequential spectrum of blue, green, yellow, and red, designed to distinctly differentiate between various intensity levels, thus offering a richer and clearer perspective.

3.3.3. Patching

During the development phase of the dataset, maintaining the images' realistic proportions was critical, while also ensuring that they could be processed efficiently. To achieve this, we implemented a patching strategy that was finely tuned for our dataset, which comprised images no larger than 512 x 512 pixels. This approach involved segmenting each image into four equal-sized patches, each one anchored at a different corner of the image: top-left, top-right, bottom-left, and bottom-right. Given an array of dimensions $M \times N$, we divide it into four $\alpha \, x \, \alpha$ patches anchored at four corners, where $\alpha = 256$. Let's denote the patches as follows:

$$\begin{aligned} P_{TL} &= I[0{:}\,\alpha, 0{:}\,\alpha] \\ P_{TR} &= I[0{:}\,\alpha, N-\alpha{:}\,N] \\ P_{BL} &= I[M-\alpha{:}\,M, 0{:}\,\alpha] \\ P_{BR} &= I[M-\alpha{:}\,M, N-\alpha{:}\,N] \end{aligned} \quad (3)$$

Where:

- P_{TL} denotes the top-left patch of the image, starting from the top-left corner and spanning $\alpha \, x \, \alpha$ pixels.
- P_{TR} is the top-right patch, starting $\alpha \, x \, \alpha$ pixels from the right boundary and extending leftwards for $\alpha \, x \, \alpha$ pixels.
- P_{BL} represents the bottom-left patch, beginning $\alpha \, x \, \alpha$ pixels above the bottom boundary and extending upwards.
- P_{BR} captures the bottom-right section of the image, which is the $\alpha \, x \, \alpha$ square of pixels at the bottom-right corner.

In the final evaluation phase, it was imperative to reintegrate these segmented patches to reconstruct the complete image. This reintegration is crucial for the holistic evaluation of the model's outcomes against the original data.

3.4 Architecture R2U-Net3+

To overcome the limitations associated with U-Net and its derivatives, we introduce a new model, termed R2U-Net3+. This model comprises two main elements: the skip pathways and the backbone, which are detailed below.

3.4.1. Skip pathways

Addressing the existing challenges, R2U-Net3+ draws its architectural inspiration from U-Net3

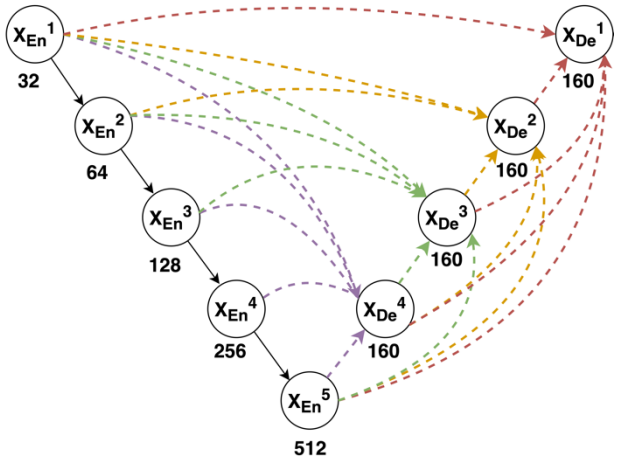


Figure. 4 Skip pathways of U-Net3+

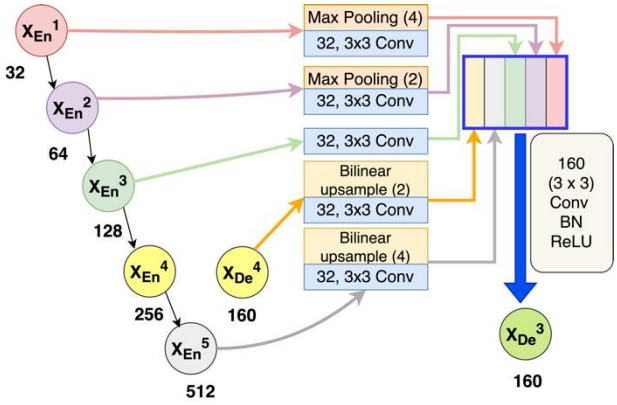


Figure. 5 Illustration of the process employed to derive the feature map X_{De}^3

adapting modifications suitable for our tasks. Traditional models like U-Net and U-Net++ have the ability to capture some information but often fail to gather details comprehensively from all scales. U-Net3+ improves this by combining feature maps from different sizes in each decoder layer. This ensures that both small details and larger patterns are captured across all scales.

Incorporating the full-scale skip connections from U-Net3+ (Fig. 4) optimizes the model in terms of reducing parameter quantity, thereby promising an improvement in computational efficiency while ensuring enhanced performance. The effectiveness of U-Net3+ in enhancing the accuracy of medical image segmentation and reducing computational resources has been well-documented [10]. Formally, the integration of skip connections in the architecture can be described as follows:

The first step in this intricate process is the extraction of the feature map at the k^{th} encoding block. The math underlying this is:

$$Y_{En}^k = \left[C\left(D\left(X_{En}^k\right)\right)\right] \quad (4)$$

Here, C denotes the convolution operation, while D signifies the down sampling mechanism. This produced feature map, Y_{En}^k, is instrumental in deriving the primary feature map in the subsequent encoding block.

As we transition to the decoding side, the generation of the feature map at the k^{th} decoding block is paramount. This is described by:

$$Y_{De}^k = \left[C\left(U\left(X_{De}^k\right)\right)\right] \quad (5)$$

Where U embodies the up sampling operation. Finally, to synthesize the main feature map X_{De}^i in the decoding block, the aggregated features from both encoding and decoding phases are combined. This fusion is mathematically expressed as:

$$X_{De}^i = H\left(Z\left[\left[Y_{En}^k\right]_{k=1}^{i-1}, C\left(X_{En}^k\right), \left[Y_{De}^k\right]_{k=i+1}^{N}\right]\right) \quad (6)$$

Here, Z represents concatenation. The function H encompasses a convolution operation, followed by batch normalization and a ReLU activation function. The index i pinpoints a specific layer in the decoding process, while N denotes the depth of the downsampling.

Fig. 5 illustrates the process of building the feature map from the X_{De}^3 block. The encoder-layer feature map of the same scale (X_{En}^3) directly feeds into the X_{De}^3 decoder. The skip connections between the encoder and decoder serve the purpose of transmitting low-level information from the smaller scale encoder layers (like X_{En}^1 and X_{En}^2) using non-overlapping max pooling operations. Meanwhile, skip connections within the decoders transmit high-level semantic information from larger scale decoder layers (like X_{De}^5 and X_{De}^4) using bilinear interpolation. To generate five feature maps of identical resolution in this architecture, channels are unified and redundant information is pruned. This process employs 32 filters of size 3 × 3. Additionally, to amalgamate shallower semantic information with deeper semantic cues, feature aggregation is performed on the merged feature maps, consisting of 160 filters of size 3 × 3, batch normalization, and the ReLU activation function.

3.4.2. Backbone

Drawing from the inspiration of the R2U-Net and R2U++, we have incorporated recurrent residual convolution layers (RRCL) into the foundational unit convolutional layers of the U-Net3+ architecture.

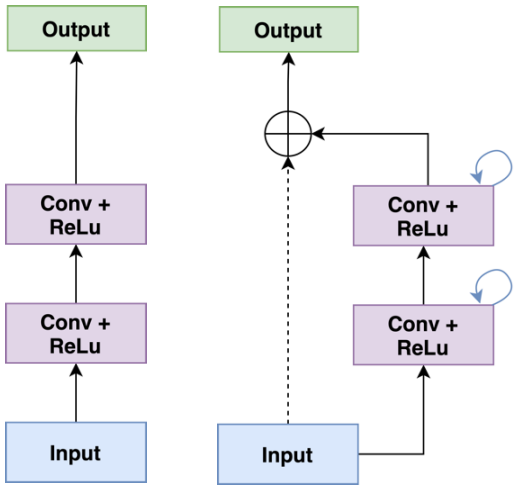


Figure. 6 Convolutional unit showing a comparison between: (a) U-Net and (b) the proposed R2U-Net3+

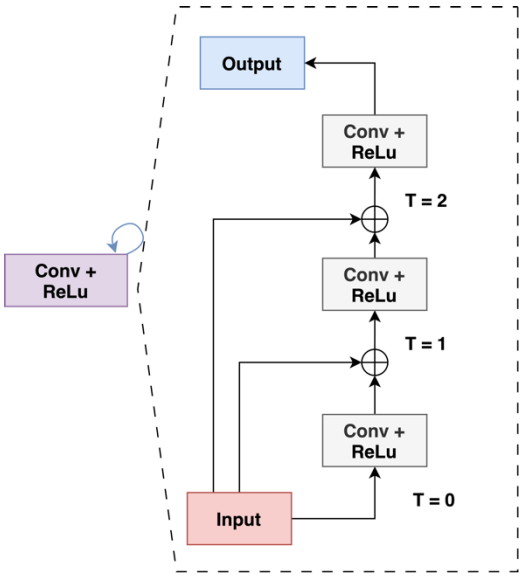


Figure. 7 Recurrent convolutional layer demonstrated across different timestamps (t)

This evolution sees the recurrent convolution layer (RCL) mature over discrete time steps, which we denote as t. These recurrent convolution operations find their mathematical basis in the enhancements seen in the RCNN discussed. The RCL's operations are executed in alignment with these discrete time steps, as articulated by the RCNN. In our proposed RRCNN architecture for backbone use, the output is defined using the equation:

$$O_k^l(t) = w_k^f . x_l^f(t) + w_k^r . x_l^r(t-1) + b_k \quad (7)$$

where $x_l^f(t)$ and $x_l^r(t-1)$ represent the inputs to the standard convolution layers at l^{th} the recurrent convolutional layer (RCL). The terms w^f and w^r are feed-forward weights and recurrent weights and b_k denotes the bias term. Following this, the output is processed using the Rectified Linear Unit (ReLU) activation function, formulated as:

$$f\left(O_k^l(t)\right) = \max\left(0, O_k^l(t)\right) \quad (8)$$

Subsequently, the output from the RRCL block is computed by combining the original input with the processed input, if $F(x_l, w_l)$ is the output from the RCL then the final output from RRCL, as represented by:

$$x_{l+1} = x_l + F(x_l, w_l) \quad (9)$$

In Fig. 6, the left side depicts the convolutional block unit standard to the U-Net, which consists of two feedforward convolutional ReLU layers. On the right (Fig. 6b), we see the block unit for the residual recurrent unit intended for use in the R2U-3Net+. Every recurrent residual block is composed of two unit recurrent convolutional blocks. The residual connection is then utilized to generate the final output by combining the original input with the output derived from the second recurrent unit.

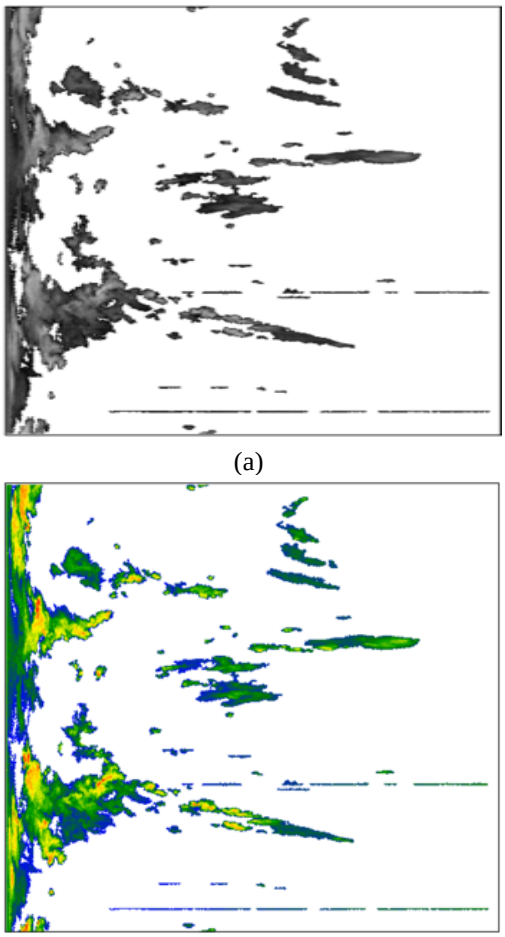
In Fig. 7, the recurrent convolutional layer (RCL) with a time step of t=2 is illustrated. The unfolding of the recurrent convolution block is depicted for the specific time step. When t=0, the output emerges exclusively from the convolution layer, devoid of any recurrent contributions. At t=1, the output from t=0 is combined with the original input at t=1 and is subsequently fed into the network. This methodology is echoed for t=2. The term "Conv" symbolizes the convolution process.

4. Result and discussion

The experimentation process comprises two primary steps: training and testing, as depicted in Fig. 2. For training, pre-processed images are fed to R2U-Net3+ to train the model using a random split method. Upon completion of the training process, unseen testing data is introduced to the trained model to generate predictions.

4.1 Dataset

The datasets under study comprise two distinct categories: data containing radio frequency interference (RFI) and data devoid of RFI. The former category contains a total of 368 data entries, each with a resolution of 445 x 480. This dataset primarily focuses on the first elevation scanning,

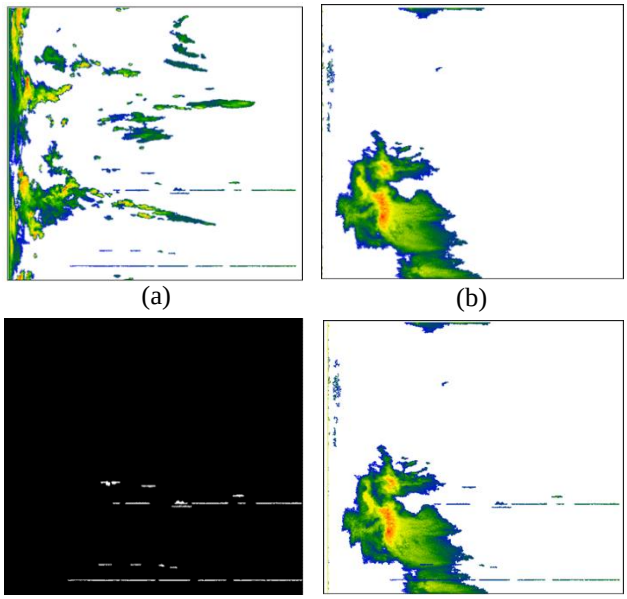


(a) (b)

Figure. 8 Pseudo-coloring comparison: (a) Original grayscale dataset and (b) Data after pseudo-coloring

where RFI occurrences are frequently observed. In contrast, the latter category comprises 219 data entries with the same resolution, sourced from the third layer. At this elevation, RFI instances are uncommon but not entirely absent. As a result, not all entries from this layer were classified as RFI-free.

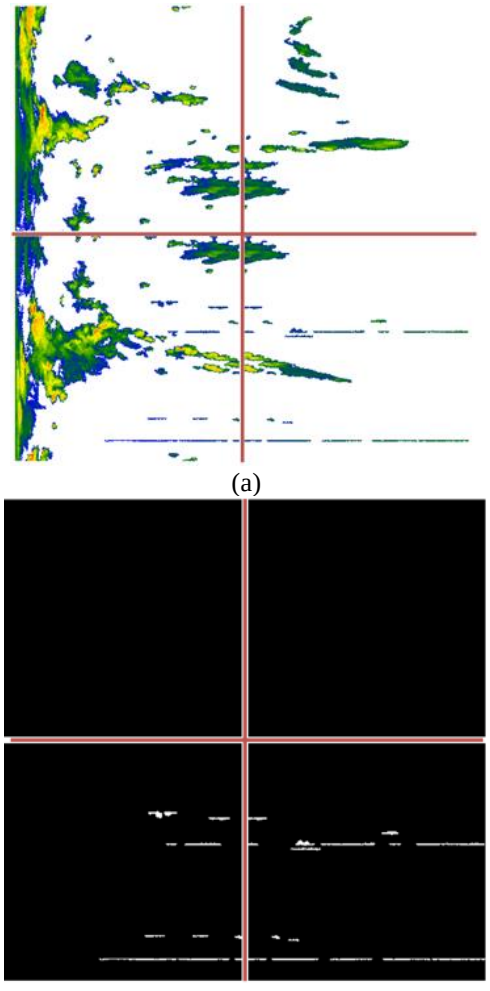
Before delving deeper into the dataset discussion, Fig. 8 illustrates the pseudo-coloring process. Although this technique is typically applied after the patching process, it is introduced at this stage to enhance clarity and facilitate a more straightforward understanding of the forthcoming dataset discussion. Fig. 8a displays the original data which, due to its limited grayscale range of 0-65, tends to blur the distinctions between different elements. This lack of contrast can make it challenging for models to learn effectively. However, by introducing pseudo-coloring as shown in In Fig. 8b, we introduce a wider range of colors, emphasizing the minor variations in the data. This enhanced feature palette offers greater information, facilitating more effective and comprehensive learning during training.



(a) (b)

(c) (d)

Figure. 9 Data augmentation process: (a) Data with inherent RFI, (b) Data free from interference, (c) Mask isolating RFI, and (d) Synthetic image



(a) (b)

Figure. 10 Patching process in image segmentation according to Eq. (3): (a) input image and (b) ground truth

Fig. 9 illustrate of the datasets utilized in our augmentation process. The objective behind this methodology is to enrich our data, enhancing the diversity and robustness required for advanced analytics and model training. Specifically: (a) showcases the original dataset, which contains radio frequency interference (RFI); (b) presents the clean RFI dataset, purified and devoid of RFI; (c) details the Mask specifically designed to isolate RFI; and (d) illustrates the outcome when RFI from the original dataset (a) is transplanted onto the Clean RFI dataset (b). This augmentation strategy, particularly the incorporation of synthetic data like in (d), ensures a broader spectrum of data, catering to a more comprehensive learning landscape and, ultimately, a more resilient and accurate model.

Fig. 10 illustrates the pre-processing, adhering to the methodology detailed in Eq. 3. In Fig. 10a, the input image has been processed and segmented into four images, each sized 256x256 pixels. This step ensures the quality and integrity of the image information while also accommodating an optimal size for further segmentation tasks. Conversely, Fig. 10b represents the ground truth, which has also been segmented to the same dimensions. It is crucial to note that, during the evaluation phase, these segmented portions of the ground truth will be reassembled to provide a comprehensive evaluation of segmentation quality.

4.2 Metrics

Several evaluation metrics to assess the model's performance. These metrics include sensitivity (Se), specificity (Sp), accuracy (Ac), F1 score, dice coefficient (Dice), and Intersection over union (IoU) sensitivity (Se) measures how well a model can correctly identify positive cases, calculated by dividing true positives (TP) by the sum of true positives and false negatives (FN), represented as:

$$\mathrm{Se} = \frac{\mathrm{TP}}{\mathrm{TP+FN}} \tag{10}$$

Specificity (Sp), on the other hand, measures how well a model can correctly identify negative cases, calculated by dividing true negatives (TN) by the sum of true negatives and false positives (FP), represented as:

$$\mathrm{Sp} = \frac{\mathrm{TN}}{\mathrm{TN+FP}} \tag{11}$$

Accuracy (Ac) is another pivotal metric, quantifying the overall correctness of the model across all cases, represented as:

$$\mathrm{Ac} = \frac{\mathrm{TP+TN}}{\mathrm{TP+TN+FP+FN}} \tag{12}$$

The F1 Score (dice coefficient) is the harmonic mean of precision and recall, offering a balance between the two, and is calculated as:

$$\mathrm{Dice} = 2 \cdot \frac{\mathrm{TP}}{2 \cdot \mathrm{TP+FP+FN}} \tag{13}$$

Intersection over union (IoU) measure the overlap between two samples or object boundaries. Their respective formulas are:

$$\mathrm{IoU} = \frac{\mathrm{TP}}{\mathrm{TP+FP+FN}} \tag{14}$$

By utilizing these comprehensive metrics, the study facilitates an in-depth assessment of the model's performance, highlighting areas that may necessitate further refinement.

4.3 Result

Upon examining Table 1, which presents a quantitative comparison of various architectural models including U-Net, U-Net++, U-Net3+, R2U-Net, R2U++, and our innovatively developed R2U-Net3+, it becomes apparent that there is a noticeable variability in the number of parameters across the different models. Our proposed model, R2U-Net3+, was evaluated under two different conditions, represented by timestamps t=1 and t=2, and exhibited parameter counts of 9.5 million and 17.6 million, respectively. These timestamps indicate the recursion iterations, reflecting the diverse conditions under which the models were assessed. When comparing the R2U++ models, our developed architecture, R2U-Net3+, demonstrates a reduction in the number of parameters compared to R2U++, even though the reduction is marginal. This optimization in model parameters is vital for enhancing efficiency without compromising the integrity and accuracy of the results.

In addition to the parameters, the hyperparameter settings have been set for optimal performance. The number of filters are, ranging from 32 to 512; number of filters consists of 32, 64, 128, 256, 512 for each layer depth. A Batch Size of 16, a Learning Rate of 2e-4, and 35 epochs, with the Adam optimizer, have been employed. Furthermore, distinct loss functions have been allocated to each model; binary cross entropy for U-Net and R2U-Net, BCE dice loss [13] for U-Net++ and R2U++, and a multi hybrid loss function [10] for U-Net3+ and R2U-Net3+. The selection of different loss functions aligns with previous research on each respective architecture.

Table 1. Result of experiments

Model	t	Parameter	Dice Coef	IoU	Ac	Se	Sp
U-Net	-	7.7 M	93.527	88.467	99.837	92.236	99.936
U-Net++	-	9 M	94.089	89.306	99.846	92.524	99.949
U-Net3+	-	6.7 M	94.767	90.469	99.866	**95.212**	99.926
R2U-Net	2	16.7 M	94.417	89.814	99.856	93.770	99.936
R2U++	2	18 M	94.581	90.160	99.863	92.854	**99.955**
R2U-Net3+ (Ours)	1	9.5 M	94.980	90.745	99.869	95.061	99.934
R2U-Net3+ (Ours)	2	17.6 M	**95.352**	**91.409**	**99.881**	94.952	99.945

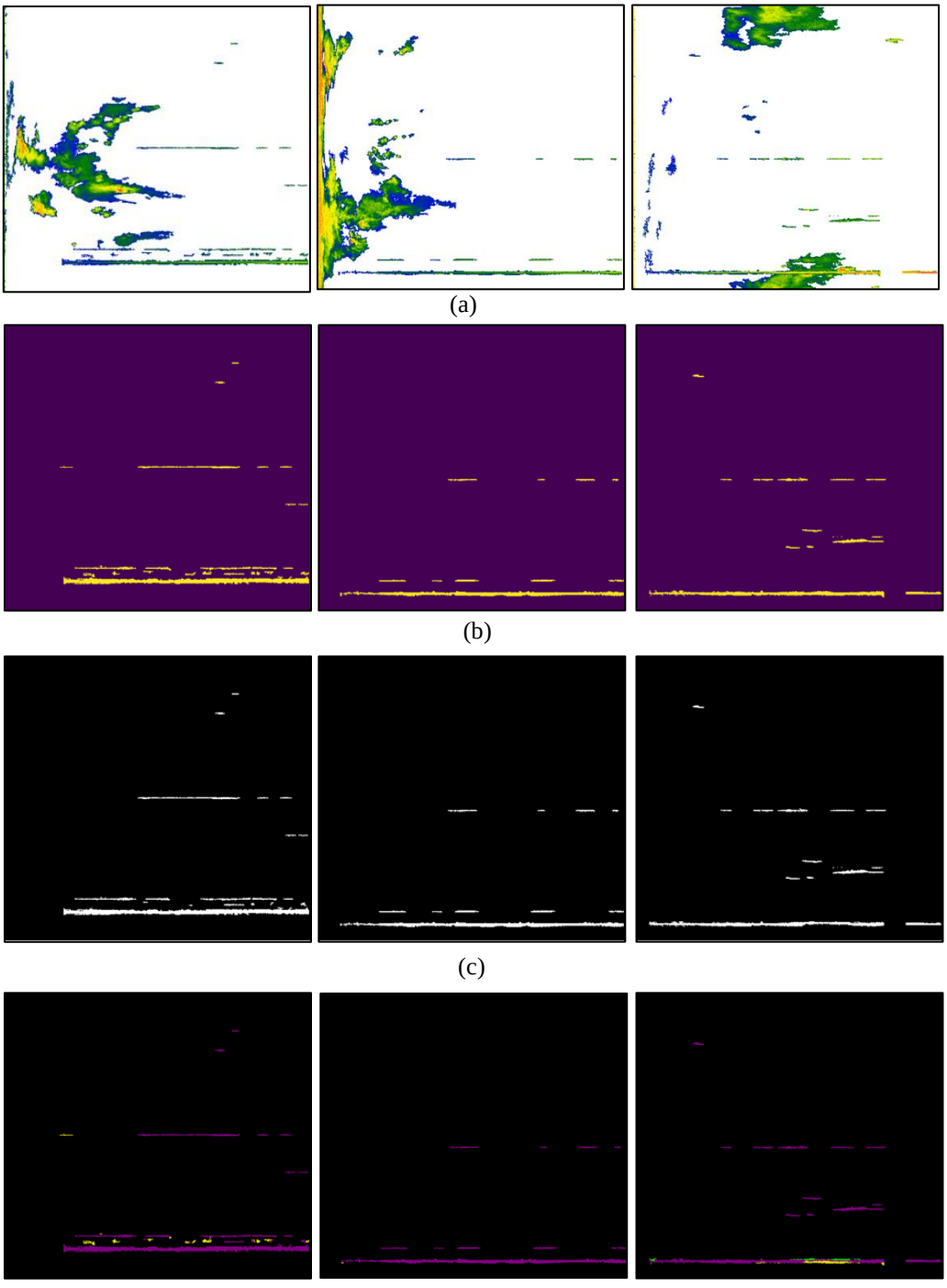


Figure. 11 Comparison results of three samples (R2U-Net3+): (a) input, (b) ground truth, (c) masks predict, and (d) predict, true positives (purple), false negatives (yellow), and false positives (green)

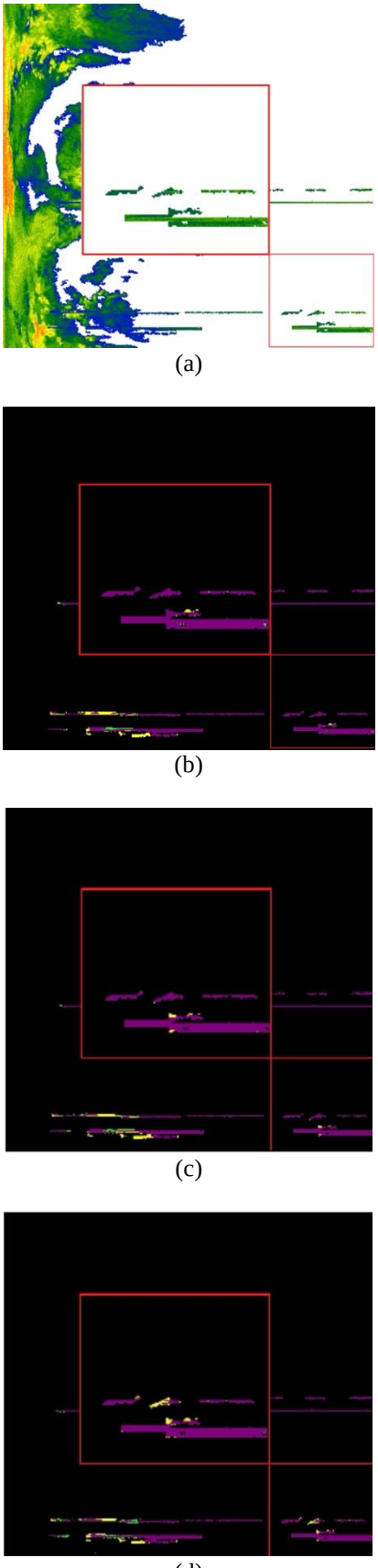
Specifically, binary cross entropy is utilized for U-Net and R2U-Net, adhering to the prevalent methodologies found in related studies. Similarly, BCE dice loss is applied to U-Net++ and R2U++, while a multi hybrid loss function is implemented for U-Net3+ and R2U-Net3+. Fine-tuning these hyperparameters is crucial for ensuring the strength and consistency of the experimental results.

Based on Table 1, it is evident that our proposed R2U-Net3+ model excels in terms of performance across various metrics. When tested at two different timestamps, t=1 and t=2, this model consistently delivers high values for IoU, accuracy (Ac), sensitivity (Se), specificity (SP), and dice coef. This suggests it is both effective at correctly identifying objects in images and has a good balance of precision and recall. In terms of IoU, R2U-Net3+ scored 90.745% at t=1 and 91.409% at t=2, showing a strong match between its predictions and the actual results. The model also achieved an accuracy of 99.869% at t=1 and 99.881% at t=2, further highlighting its reliability.

Furthermore, the model's sensitivity values were 95.061% at t=1 and 94.952% at t=2, and its Specificity values were 99.934% and 99.945% respectively. These results show its strength in correctly identifying true positives and its ability to keep false positives low. This means it is good at spotting objects in images and avoiding mistakes.

The dice coefficient for R2U-Net3+ at t=2 is 95.352%, which is among the highest in the table. This score represents a balance between precision (getting things right) and recall (catching everything that's relevant), suggesting the model performs well overall. In conclusion, R2U-Net3+ performs well across various measures, making it a strong model for tasks like medical image segmentation. Its efficient use of parameters and balanced performance metrics make it a good choice for both further study and practical use.

Fig. 11 provides a comprehensive analysis of masking results for three distinct samples using the proposed R2U-Net3+ model. Fig. 11a displays the model's input data, representing the initial radar reflections captured, whereas Fig. 11b presents the ground truth. Fig. 11c illustrates the masking results of the R2U-Net3+ model. Meanwhile, Fig. 11d highlights the masking outcomes with color-coded pixels indicating true positives (purple), false negatives (yellow), and false positives (green). Here, true positives are highlighted in purple, indicating areas where the model's predictions are in close harmony with the ground truth. Notably, when radio frequency interference (RFI) is absent within the reflectivity region, the model demonstrates



(a)

(b)

(c)

(d)

Figure. 12 Qualitive analysis: (a) Input, (b) R2U-Net3+, (c) R2U++, and (d) R2U-Net

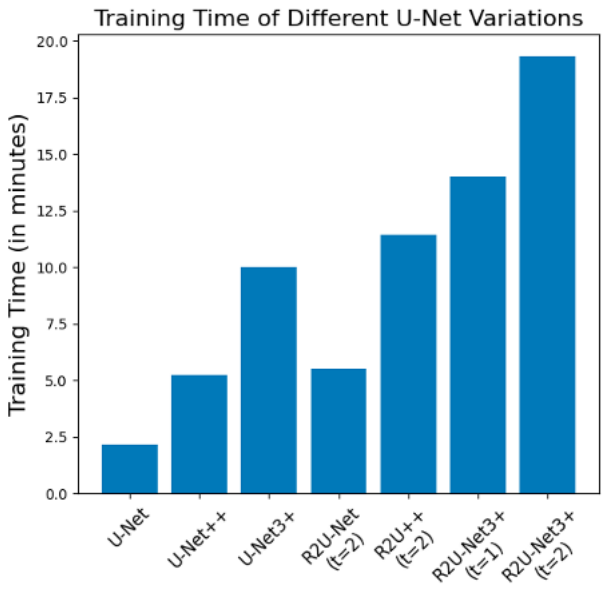
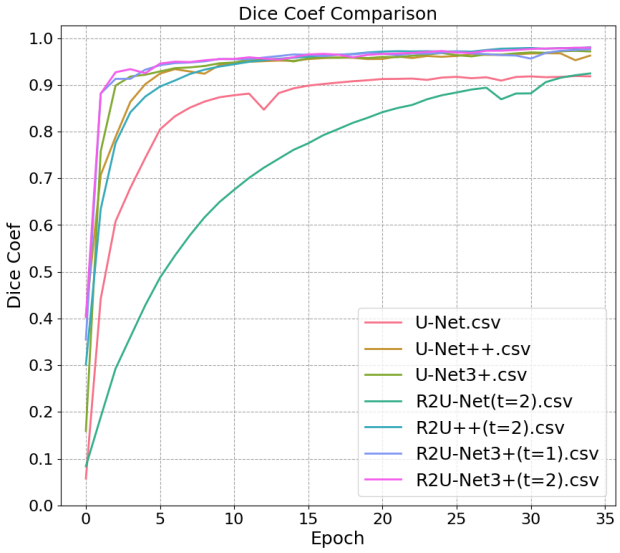


Figure. 13 Comparative analysis of training times for various models, displayed in seconds. The t denotes the timestamp for recursive iterations in RRCNN

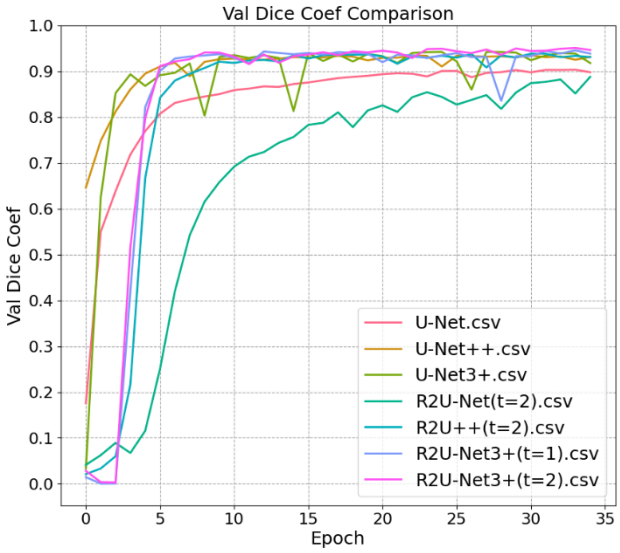
remarkable segmentation precision. Conversely, in the presence of reflectivity, the model encounters challenges, as shown by the presence of false negatives (yellow) and false positives (green). These color markings respectively signify areas the model missed and locations inaccurately identified. The R2U-Net3+ model's capability in detecting RFI underscores its potential and areas that could benefit from further optimization for enhanced real-world deployment.

Fig. 12 offers a side-by-side comparison of three distinct models: R2U-Net3+, R2U++, and R2U-Net. Within the figure, there are two red boxes. The central red box provides an enlarged view of the specific area captured within the smaller red box located in the bottom-right corner. The first image in the sequence displays the input data (Fig. 12 (a)), followed by the outputs of R2U-Net3+ (Fig. 12 (b)), R2U++ (Fig. 12 (c)), and R2U-Net (Fig. 12 (d)). Among the results, the segmentation achieved by R2U++ stands out, as evident from the regions highlighted. This model notably produces superior results, exhibiting fewer false positives and negatives compared to R2U++ and R2U-Net. Additionally, R2U-Net3+ delivers better performance than R2U-Net++. This assessment underscores that our proposed model demonstrates greater accuracy than previous RRCNN models integrated into U-Net and U-Net++.

In Fig. 13, we analyze the computational time of various models. Referring to Table 1, U-Net3+ has the fewest parameters yet still requires a longer computational time compared to U-Net, U-Net++,



(a)



(b)

Figure. 14 Comparison of history: (a) Train dice coefficient and (b) Validation dice coefficient

and even R2U-Net which has almost double the number of parameters. Both versions of our proposed R2U-Net3+ with time step (t=1 and t=2) use fewer parameters than R2U++ with time step 2 but require more processing time. Factors other than parameter count, such as the intricacies of the architecture, influence the processing duration. However, the streamlined parameter structure of R2U-Net3+ with t=1 indicates its efficient use of resources. This efficiency does not compromise the segmentation quality; in fact, it may enhance it, as shown by the superior dice coef and IoU metrics.

Fig. 14 displays the graphs of the train dice coefficient history (a) and the validation dice coefficient (b). In the training data, all models exhibit slight fluctuations. The U-Net and R2U-Net models

tend to perform below other models, while U-Net++, R2U++, U-Net3+, R2U-Net3+ (t=1, t=2) have relatively similar performances. In the validation data, a similar trend is observed, although U-Net3+ and R2U-Net3+(t=1) show fluctuations at certain epochs. Notably, the R2U-Net3+ in the validation dice coefficient stands out with a significantly superior performance compared to the other models.

5. Conclusion

In this research, we introduced the recurrent residual convolution block architecture as the backbone for a full-scale skip connection U-Net in the context of radio frequency interference (RFI) radar weather data segmentation. Several key improvements that we applied to this architecture include: The utilization of Recurrent Residual Units instead of plain convolutions, enabling the network to extract low-level features with precision without encountering degradation issues. This approach allowed us to capture more relevant features from RFI data and enhance the segmentation quality.

We also drew architectural inspiration from U-Net3+, adapting modifications suitable for our task. Traditional models like U-Net and U-Net++ can capture some information but often fail to comprehensively gather details from all scales. U-Net3+ addresses this limitation by combining feature maps of different sizes in each decoder layer. This ensures that both small-scale details and larger patterns are captured across all scales.

Throughout this study, our proposed R2U-Net3+ model was rigorously benchmarked against several esteemed architectures, including U-Net, U-Net++, R2U-Net, R2U++, and U-Net3+, using the dice coefficient (Dice Coef) and intersection over union (IoU) as evaluation metrics. U-Net has been a frequently employed model for RFI segmentation in previous research. When compared with this classic model (U-Net), our R2U-Net3+ (t=2) showed a promising improvement of approximately 1.825% in Dice Coefficient and 2.942% in IoU. Furthermore, when pitted against earlier RRCNN fusions such as R2U-Net and R2U++, our architecture clearly stands out. Specifically, it outperformed R2U-Net by 0.935% in dice coefficient and 1.595% in IoU, and against R2U++, our model marked an advancement with a 0.771% increment in dice coefficient and a 1.249% rise in IoU. Delving into the foundational architecture, U-Net3+, which we enhanced by integrating RRCNN to create our R2U-Net3+, the latter demonstrated a growth of 0.585% in dice coefficient and a 0.94% elevation in IoU. Conclusively, the empirical data attests to the significant scientific contribution of our R2U-Net3+ model.

Furthermore, we also paid attention to the role of the timesteps, which represents the recurrent time in the RRCN within our architecture. The experimental results showed that an increase the time step recurrent value had a positive impact on the model's performance. Particularly, when comparing the R2U-Net3+ model with t=2 to t=1, there is a notable improvement, with an approximate 0.372% increase in the Dice Coefficient and IoU (0.664%), Accuracy (0.012%), Sensitivity (-0.109%), and Specificity (0.011%). This research contributes significantly to the development of segmentation techniques in this context and underscores the importance of recurrent parameters in designing optimal architectures for RFI segmentation tasks in radar weather data. These findings provide a solid foundation for further research and development in this field.

Conflict of interest

The authors declare no conflict of interest.

Author contributions

Conceptualization by Alfan Alfarisy and Handayani Tjandrasa; methodology by Alfan Alfarisy; software by Alfan Alfarisy; validation and formal analysis by Handayani Tjandrasa and Alfan Alfarisy; The original manuscript draft was by Alfan Alfarisy and supervised by Handayani Tjandrasa.

References

[1] N. Besic, J. FiguerasVentura, J. Grazioli, M. Gabella, U. Germann, and A. Berne, "Hydrometeor classification through statistical clustering of polarimetric radar measurements: A semi-supervised approach", *Atmos Meas Tech*, Vol. 9, No. 9, pp. 4425–4445, Sep. 2016, doi: 10.5194/amt-9-4425-2016.

[2] J. Yin, P. Hoogeboom, C. Unal, and H. Russchenberg, "Radio Frequency Interference Characterization and Mitigation for Polarimetric Weather Radar: A Study Case", *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 60, 2022, doi: 10.1109/TGRS.2021.3093565.

[3] J. Y. N. Cho, "A new radio frequency interference filter for weather radars", *J Atmos Ocean Technol*, Vol. 34, No. 7, pp. 1393–1406, Jul. 2017, doi: 10.1175/JTECH-D-17-0028.1.

[4] ITU-R, "ITU-R: Resolution 229 [COM5/16] (WRC-03); Use of the bands 5150-5250 MHz, 5250-5350MHz and 5470-5725MHz by the

mobile service for the implementation of wireless access systems including radio local area networks", *The World Radiocommunication Conference, Geneva.*, 2003, Accessed: Apr. 28, 2023. [Online]. Available: https://www.itu.int/oth/R0A0E00002E/en

[5] ETSI, "En 301 893", 2012. [Online]. Available: http://portal.etsi.org/chaircor/ETSI_support.asp

[6] A. Huuskonen, E. Saltikoff, and I. Holleman, "The operational weather radar network in Europe", *Bull Am Meteorol Soc*, Vol. 95, No. 6, pp. 897–907, 2014, doi: 10.1175/BAMS-D-12-00216.1.

[7] J. Akeret, C. Chang, A. Lucchi, and A. Refregier, "Radio frequency interference mitigation using deep convolutional neural networks", *Astronomy and Computing*, Vol. 18, pp. 35–39, 2017, doi: 10.1016/j.ascom.2017.01.002.

[8] Z. Yang, C. Yu, J. Xiao, and B. Zhang, "Deep residual detection of radio frequency interference for FAST", *Mon Not R Astron Soc*, Vol. 492, No. 1, pp. 1421–1431, 2020, doi: 10.1093/mnras/stz3521.

[9] P. Lepetit, C. Ly, L. Barthes, C. Mallet, N. Viltard, Y. Lemaitre, and L. Rottner, "Using Deep Learning for Restoration of Precipitation Echoes in Radar Data", *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 60, 2022, doi: 10.1109/TGRS.2021.3052582.

[10] H. Huang, L. Lin, R. Tong, H. Hu, Q. Zhang, Y. Iwamoto, X. Han, Y. W. Chen, and J. Wu, "UNet 3+: A Full-Scale Connected UNet for Medical Image Segmentation", *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, Vol. 2020-May, No. ii, pp. 1055–1059, 2020, doi: 10.1109/ICASSP40776.2020.9053405.

[11] M. Liang and X. Hu, "Recurrent convolutional neural network for object recognition", In: *Proc of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE Computer Society*, 2015, pp. 3367–3375. doi: 10.1109/CVPR.2015.7298958.

[12] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation", *Medical Im- Age Computing and Computer-Assisted Intervention*, pp. 234–241, 2015, [Online]. Available: http://arxiv.org/abs/1505.04597

[13] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet++: A Nested U-Net Architecture for Medical Image Segmentation", *Deep Learning in Medical Image Anylysis and Multimodal Learning for Clinical Decision Support*, pp. 3–11, 2018, [Online]. Available: http://arxiv.org/abs/1807.10165

[14] M. Z. Alom, M. Hasan, C. Yakopcic, T. M. Taha, and V. K. Asari, "Recurrent residual convolutional neural network based on u-net (r2u-net) for medical image segmentation", *ArXiv Preprint ArXiv:1802.06955*, 2018.

[15] M. Mubashar, H. Ali, C. Grönlund, and S. Azmat, "R2U++: a multiscale recurrent residual U-Net with dense skip connections for medical image segmentation", *Neural Comput Appl*, Vol. 34, No. 20, pp. 17723–17739, 2022, doi: 10.1007/s00521-022-07419-7.

[16] A. R. Offringa, A. G. de Bruyn, S. Zaroubi, and M. Biehl, "A LOFAR RFI detection pipeline and its first results", *ArXiv Preprint ArXiv:1007.2089*, 2010, [Online]. Available: http://arxiv.org/abs/1007.2089

[17] C. J. Wolfaardt, D. Davidson, and T. Niesler, "Statistical classification of radio frequency interference (RFI) in a radio astronomy environment", in *2016 Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference, PRASA-RobMech 2016, Institute of Electrical and Electronics Engineers Inc.*, 2017. doi: 10.1109/RoboMech.2016.7813164.

[18] R. D. Ekers and J. F. Bell, "Radio Frequency Interference", *arXiv:astro-ph/0002515*, 2000.

[19] F. B. Espillco, E. Gascó, C. I. L. González, M. J. G. Silva, and G. Pajares, "Semantic segmentation based on Deep learning for the detection of Cyanobacterial Harmful Algal Blooms (CyanoHABs) using synthetic images", *Appl Soft Comput*, Vol. 141, 2023, doi: 10.1016/j.asoc.2023.110315.

[20] J. Yu, J. Zhang, A. Shu, Y. Chen, J. Chen, Y. Yang, W. Tang, and Y. Zhang, "Study of convolutional neural network-based semantic segmentation methods on edge intelligence devices for field agricultural robot navigation line extraction", *Comput Electron Agric*, Vol. 209, 2023, doi: 10.1016/j.compag.2023.107811.

[21] X. Wang, S. Jing, H. Dai, and A. Shi, "High-resolution remote sensing images semantic segmentation using improved UNet and SegNet", *Computers and Electrical Engineering*, Vol. 108, p. 108734, 2023, doi: 10.1016/j.compeleceng.2023.108734.

[22] E. Shelhamer, J. Long, and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation", *IEEE Trans Pattern Anal Mach Intell*, Vol. 39, No. 4, pp. 640–651, 2015, doi: 10.1109/TPAMI.2016.2572683.

[23] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia,

"Pyramid scene parsing network", In: *Proc. of 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, Vol. 2017-Janua, pp. 6230–6239, 2017, doi: 10.1109/CVPR.2017.660.

[24] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 40, No. 4, pp. 834-848, 2018, doi: 10.1109/TPAMI.2017.2699184.

[25] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs", *IEEE Trans Pattern Anal Mach Intell*, Vol. 40, No. 4, pp. 834–848, 2018, doi: 10.1109/TPAMI.2017.2699184.

[26] L. C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking Atrous Convolution for Semantic Image Segmentation", *ArXiv, abs/1706.05587*, 2017.

[27] M. Heistermann, S. Jacobi, and T. Pfaff, "Technical note: An open source library for processing weather radar data (wradlib)", *Hydrol. Earth Syst. Sci.*, Vol. 17, pp. 863–871, 2013, doi: https://doi.org/10.5194/hess-17-863-2013.

[28] M. Vaccarono, C. V. Chandrasekar, R. Bechini, and R. Cremonini, "Survey on electromagnetic interference in weather radars in Northwestern Italy", *Environments*, Vol. 6, No. 12, 2019, doi: 10.3390/environments6120126.

[29] M. Tkachenko, M. Malyuk, A. Holmanyuk, and N. Liubimov, "Label Studio: Data labeling software", [Online]. Available: https://github.com/heartexlabs/label-studio

[30] B. Kuang, S. G. Nnabuife, and Z. Rana, "Pseudo-image-feature-based identification benchmark for multi-phase flow regimes", *Chemical Engineering Journal Advances*, Vol. 5, 2021, doi: 10.1016/j.ceja.2020.100060.

[31] Y. Yang, M. Zhu, Y. Wang, H. Yang, Y. Wu, and B. Li, "Super-resolution reconstruction of cell Pseudo-color image based on raman technology", *Sensors*, Vol. 19, No. 19, 2019, doi: 10.3390/s19194076.

[32] C. Rodríguez, A. V. Eeckhout, E. G. Caurel, A. Lizana, and J. Campos, "Automatic pseudo-coloring approaches to improve visual perception and contrast in polarimetric images of biological tissues", *Sci Rep*, Vol. 12, No. 1, 2022, doi: 10.1038/s41598-022-23330-6.

[33] K. Srivastava, S. Gogia, and G. Rohith, "An approach to pseudocoloring of grey scale image using deep learning technique", *Journal of Physics: Conference Series*, Institute of Physics, 2023. doi: 10.1088/1742-6596/2466/1/012030.