

## IMPROVING THE YOLOv11 NETWORK FOR RADIO FREQUENCY INTERFERENCE DETECTION IN SENTINEL-1A LEVEL-1 DATA

Luu Hoang Dat<sup>1</sup>, Nguyen Tien Phat<sup>1\*</sup>, Nguyen Minh Tuan<sup>2</sup>, Ngo Xuan Son<sup>3</sup>, Tran Van Anh<sup>4</sup>

<sup>1</sup>Le Quy Don Technical University, <sup>2</sup>Operations Division - Department of Electronic Warfare

<sup>3</sup>Center 80 - Department of Electronic Warfare, <sup>4</sup>Radar Institute – Military Institute of Science and Technology

ARTICLE INFO	ABSTRACT
Received: 18/02/2025	Radio frequency interference is a significant issue that affects the data quality of Sentinel-1A Level-1 satellite imagery, leading to difficulties in data analysis and application. Therefore, radio frequency interference detection and removal are crucial step in Sentinel-1A data preprocessing. This study focuses on developing an advanced detection method based on the YOLOv11 network. The YOLOv11 model is a state-of-the-art model known for its fast and accurate object detection capabilities. However, to enhance the effectiveness of radio frequency interference detection in Sentinel-1A data, this research presents an improved model by integrating Attention Module into the network architecture, namely: ECA (Efficient Channel Attention), GAM (Global Attention Mechanism), SA (Shuffle Attention), and ResCBAM (ResBlock + Convolutional Block Attention Module). The paper also constructed a high-precision, manually labeled RFI image dataset to facilitate the training and evaluation of the models. Experimental results demonstrate that the improved YOLOv11 + SA model achieves higher accuracy and faster execution speed compared to the original model.
Revised: 05/6/2025	
Published: 08/6/2025	

### KEYWORDS

Radio frequency interference  
Sentinel-1A Level-1  
YOLOv11  
Object detection  
Deep Learning

## CẢI TIẾN MẠNG YOLOv11 ĐỂ PHÁT HIỆN NHIỀU TÀN SỐ VÔ TUYẾN TRONG DỮ LIỆU SENTINEL-1A LEVEL1

Luu Hoàng Đạt<sup>1</sup>, Nguyễn Tiến Phát<sup>1\*</sup>, Nguyễn Minh Tuấn<sup>2</sup>, Ngô Xuân Sơn<sup>3</sup>, Trần Văn Ánh<sup>4</sup>

<sup>1</sup>Đại học Kỹ thuật Lê Quy Đôn, <sup>2</sup>Phòng Tham mưu - Cục Tác chiến điện tử,

<sup>3</sup>Trung tâm 80 - Cục Tác chiến điện tử, <sup>4</sup>Viện Radar – Viện Khoa học và Công nghệ quân sự

THÔNG TIN BÀI BÁO	TÓM TẮT
Ngày nhận bài: 18/02/2025	Nhiều tần số vô tuyến là một vấn đề đáng kể ảnh hưởng đến chất lượng dữ liệu ảnh vệ tinh Sentinel-1A Level-1, gây ra những khó khăn trong việc phân tích và ứng dụng dữ liệu. Do đó, việc phát hiện và loại bỏ nhiễu tần số vô tuyến là một bước quan trọng trong tiền xử lý dữ liệu Sentinel-1A. Nghiên cứu này tập trung vào việc phát triển một phương pháp phát hiện nhiễu tần số vô tuyến tiên tiến dựa trên mạng YOLOv11. Mô hình YOLOv11 là một mô hình mới nhất hiện nay với khả năng phát hiện đối tượng nhanh chóng và chính xác. Tuy nhiên, để nâng cao hiệu quả phát hiện trong dữ liệu Sentinel-1A, nghiên cứu này đã trình bày một mô hình cải tiến bằng cách tích hợp thêm Attention Module vào kiến trúc mạng là: ECA (Efficient Channel Attention), GAM (Global Attention Mechanism), SA (Shuffle Attention) và ResCBAM (ResBlock + Convolutional Block Attention Module). Bài viết cũng đã xây dựng một bộ dữ liệu được gán nhãn thủ công với độ chính xác cao phục vụ cho quá trình huấn luyện và đánh giá các mô hình. Kết quả thực nghiệm cho thấy phương pháp mô hình cải tiến YOLOv11+ SA có độ chính xác cao và tốc độ thực thi nhanh hơn so với mô hình ban đầu.
Ngày hoàn thiện: 05/6/2025	
Ngày đăng: 08/6/2025	

### TỪ KHÓA

Nhiều tần số vô tuyến  
Sentinel-1A Level-1  
YOLOv11  
Phát hiện đối tượng  
Học sâu

DOI: <https://doi.org/10.34238/tnu-jst.12082>

\* Corresponding author. Email: [nguyenphat@lqdtu.edu.vn](mailto:nguyenphat@lqdtu.edu.vn)

## 1. Giới thiệu

Nhiều tần số vô tuyến (RFI) đang trở thành thách thức nghiêm trọng đối với hệ thống radar khẩu độ tổng hợp (SAR), làm suy giảm đáng kể chất lượng dữ liệu thu thập. Hiện tượng này phát sinh từ nhiều nguồn khác nhau, chẳng hạn hệ thống thông tin liên lạc, thiết bị radar lân cận và các thiết bị điện tử hoạt động trùng dải tần [1], [2]. Một số nghiên cứu chỉ ra rằng RFI không chỉ làm giảm độ tương phản và độ phân giải của ảnh SAR mà còn gây sai lệch trong các phép đo định lượng, từ đó ảnh hưởng đến khả năng phân tích địa vật lý [3], [4]. Hậu quả này đặc biệt nghiêm trọng trong các ứng dụng thực tiễn như giám sát môi trường, quản lý tài nguyên và ứng phó thiên tai, nơi dữ liệu chính xác là yếu tố then chốt [5]. Ví dụ, trong công tác đánh giá thiệt hại sau thiên tai, dữ liệu SAR bị nhiễu có thể dẫn đến ước lượng sai phạm vi ảnh hưởng, gây chậm trễ hoặc định hướng sai các hoạt động cứu trợ. Tương tự, việc giám sát biến đổi đất đai hay giám sát tàu thuyền trên biển cũng bị ảnh hưởng do RFI làm sai lệch thông tin [6]. Đáng chú ý, RFI còn làm sai lệch đặc tính tán xạ của mục tiêu – một yếu tố ít được thảo luận nhưng có nguy cơ gây ra nhầm lẫn trong theo dõi tài nguyên hoặc phân tích hậu thiên tai [7]. Những hệ lụy trên cho thấy, việc giảm thiểu RFI không chỉ cải thiện chất lượng dữ liệu mà còn nâng cao độ tin cậy của các ứng dụng SAR trong thực tế. Trong những năm gần đây, nhiều phương pháp đã được nghiên cứu và phát triển để giải quyết vấn đề RFI trong ảnh SAR, có thể được phân loại thành các nhóm chính như sau:

Các phương pháp xử lý tín hiệu truyền thống: Các phương pháp này thường sử dụng các kỹ thuật như lọc, phân tích phổ, phân tách tín hiệu dựa trên ma trận, hoặc các thuật toán thống kê, xử lý đơn giản [8] - [12]. Ưu điểm của chúng là đơn giản và dễ triển khai, tuy nhiên thường gặp khó khăn khi xử lý RFI có cấu trúc phức tạp hoặc không ổn định theo thời gian, hoặc trong việc bảo toàn các tín hiệu hữu ích trong dữ liệu, dẫn đến mất thông tin quan trọng.

Các phương pháp dựa trên phân tách ma trận/tensor: Các phương pháp này thường tận dụng tính chất thừa của RFI trong không gian tín hiệu để tách chúng ra khỏi tín hiệu SAR [13] - [17]. Tuy nhiên, hiệu suất của các phương pháp này phụ thuộc lớn vào giả định về tính chất thừa và có thể không hoạt động tốt trong môi trường có RFI dày đặc.

Các phương pháp dựa trên học sâu: Các phương pháp này sử dụng mạng nơ-ron tích chập (CNN) hoặc các kiến trúc mạng sâu khác để học các đặc trưng của RFI và phân loại chúng [18], [19]. Mặc dù có tiềm năng lớn trong việc xử lý các loại RFI phức tạp, nhưng các phương pháp học sâu thường đòi hỏi lượng lớn dữ liệu huấn luyện và có thể gặp khó khăn trong việc tổng quát hóa cho các loại RFI chưa gặp trước đó.

Nghiên cứu này đề xuất một phương pháp mới để phát hiện RFI trong dữ liệu Sentinel-1A Level-1 bằng cách cải tiến mạng YOLOv11, một kiến trúc mạng nơ-ron tích chập mạnh mẽ trong bài toán phát hiện đối tượng. Đóng góp chính của nghiên cứu bao gồm:

Cải tiến kiến trúc YOLOv11: Bằng cách tích hợp các Attention Module ECA, GAM, SA, và ResCBAM, mạng được tối ưu hóa để tăng cường khả năng học đặc trưng và hiệu suất phát hiện RFI.

Xây dựng bộ dữ liệu RFI: Một bộ dữ liệu RFI đa dạng được xây dựng từ ảnh Sentinel-1A Level-1, bao gồm các loại nhiễu phổ biến và được gán nhãn cẩn thận.

Phương pháp được đánh giá trên bộ dữ liệu đã xây dựng và so sánh với các phương pháp khác, chứng minh hiệu quả vượt trội về độ chính xác, khả năng khái quát hóa và tốc độ xử lý. Các phần tiếp theo của bài báo được sắp xếp như sau: Phần 2 trình bày phương pháp cải tiến mạng YOLOv11; phần 3 phân tích kết quả thử nghiệm; và phần 4 đưa ra kết luận chung.

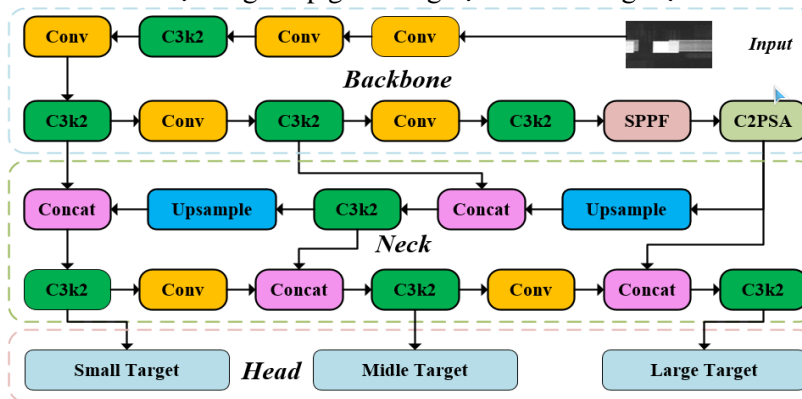
## 2. Phương pháp nghiên cứu

YOLOv11 là một mô hình phát hiện đối tượng bằng cách chia ảnh đầu vào thành một lưới các ô, mỗi ô chịu trách nhiệm dự đoán các hộp bao quanh đối tượng và xác suất xuất hiện của đối tượng trong ô đó. YOLOv11 sử dụng kiến trúc CNN để trích xuất đặc trưng từ ảnh và đưa ra dự đoán. So với các phiên bản trước, YOLOv11 có những cải tiến đáng kể về tốc độ xử lý, độ chính xác và khả năng phát hiện đối tượng nhỏ.

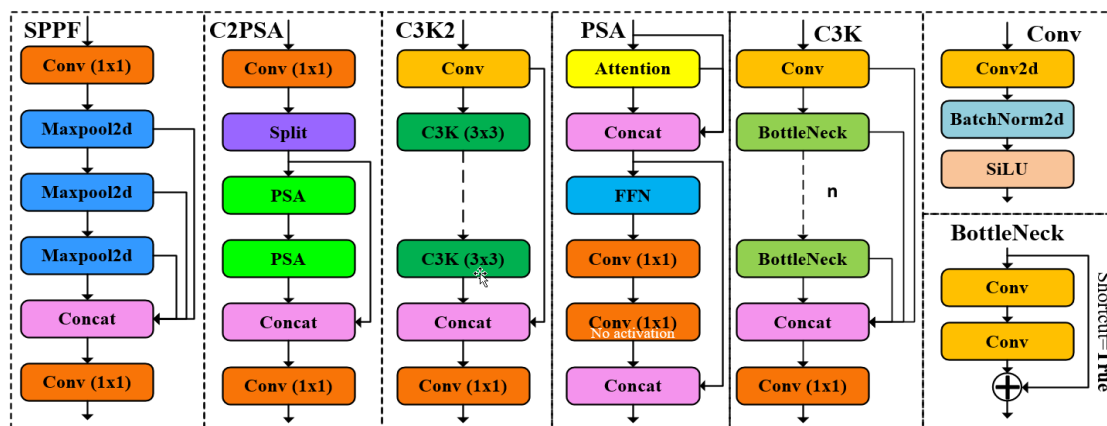
2.1. Mô hình Yolov11

Họ mô hình YOLO (You Only Look Once) đã tạo nên tiếng vang lớn trong lĩnh vực thị giác máy tính với khả năng phát hiện đối tượng theo thời gian thực. Điểm mạnh của YOLO chính là sự kết hợp hoàn hảo giữa tốc độ xử lý nhanh, độ chính xác cao, khả năng khái quát hóa tốt và kiến trúc đơn giản. Kiến trúc của họ mô hình YOLO gồm có ba thành phần chính: Backbone, Neck và Head được trình bày chi tiết ở Hình 1 và Hình 2. Backbone có nhiệm vụ trích xuất các đặc trưng (features) từ hình ảnh đầu vào tạo cơ sở cho Neck tổng hợp từ các lớp khác nhau để tạo ra một biểu diễn đặc trưng phong phú hơn và Head thực hiện dự đoán cuối cùng về bounding box (vị trí, kích thước) và phân loại đối tượng.

Từ phiên bản YOLOv1 đến YOLOv10, các nhà nghiên cứu đã không ngừng cải tiến, mang đến những bước tiến đáng kể về hiệu suất và độ chính xác. Phiên bản YOLOv11, ra mắt tại hội nghị YOLOVision 2024 [20], đánh dấu một cột mốc quan trọng với những đột phá về công nghệ. YOLOv11 sử dụng kiến trúc mạng CSPDarknet53 được cải tiến, kết hợp cùng SPPF và Focus, cho phép tối ưu hóa quá trình trích xuất đặc trưng của đối tượng. Nhờ đó, YOLOv11 đạt được độ chính xác ấn tượng trên các bộ dữ liệu phổ biến như COCO mà vẫn đảm bảo tốc độ xử lý vượt trội. Một điểm đáng chú ý khác là việc sử dụng hàm mất mát CIoU thay thế cho IoU truyền thống. CIoU giúp đánh giá chính xác hơn mức độ trùng khớp giữa vùng dự đoán và vùng thực tế của đối tượng.



Hình 1. Mô hình tổng quan của YOLOv11



Hình 2. Mô hình chi tiết các khối trong mạng YOLOv11

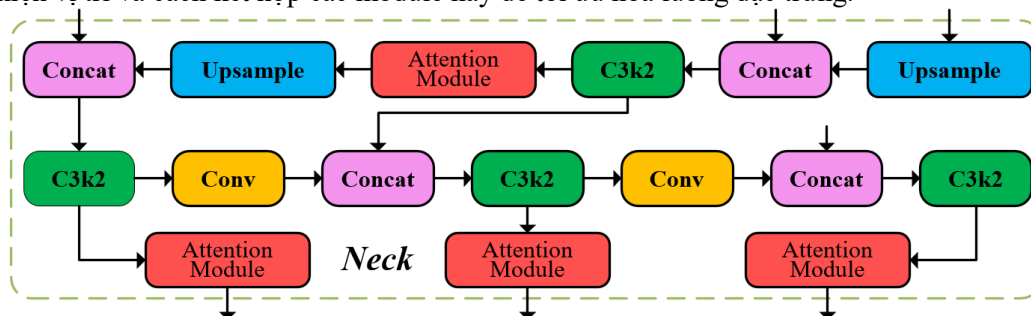
2.2. Cải tiến mô hình Yolov11

2.2.1. Tích hợp Attention Modules vào Neck của YOLOv11

Neck là thành phần quan trọng trong kiến trúc YOLOv11, nằm giữa Backbone (trích xuất đặc trưng) và Head (dự đoán đầu ra). Nhiệm vụ chính của Neck là tổng hợp và xử lý đặc trưng đa

tầng từ Backbone để tạo ra tập đặc trưng mang nhiều thông tin. Tuy nhiên, phương pháp xử lý đồng đều các đặc trưng hiện tại dẫn đến việc bỏ sót thông tin quan trọng. Để khắc phục, nghiên cứu đề xuất tích hợp Attention Module vào Neck. Cơ chế này giúp mô hình tập trung vào vùng đặc trưng quan trọng (như RFI), giảm nhiễu và nâng cao độ chính xác phát hiện.

Cụ thể, bốn loại Attention Module được khảo sát gồm: Shuffle Attention (SA), Efficient Channel Attention (ECA), Global Attention Mechanism (GAM) và ResBlock + Convolutional Block Attention Module (ResCBAM). Việc tích hợp chúng vào Neck cho phép mô hình linh hoạt cân bằng giữa độ phức tạp tính toán và hiệu suất. Kiến trúc cải tiến được minh họa trong Hình 3, thể hiện vị trí và cách kết hợp các module này để tối ưu hóa luồng đặc trưng.



Hình 3. Mô hình phần Neck sau khi cải tiến

### 2.2.2. Chi tiết các Attention Modules

Các Attention Module được nghiên cứu áp dụng trong Neck có cơ chế hoạt động đa dạng, tập trung vào tăng cường khả năng biểu diễn đặc trưng thông qua chú ý theo kênh (channel) hoặc không gian (spatial). Dưới đây là mô tả chi tiết từng module:

#### a) Shuffle Attention

Shuffle Attention (SA) là một Attention Module hiệu quả, được thiết kế để cải thiện khả năng biểu diễn của mạng nơ-ron tích chập (CNN) mà không làm tăng đáng kể độ phức tạp tính toán. SA hoạt động bằng cách chia đặc trưng đầu vào thành các nhóm dọc theo chiều kênh, sau đó áp dụng các cơ chế attention song song (Channel Attention và Spatial Attention) lên từng nhóm để học các mối quan hệ giữa các kênh và không gian. Sau đó, các kênh thông tin được xáo trộn (Channel Shuffle) giữa các nhóm để tăng cường trao đổi thông tin, và các đặc trưng được tinh chỉnh từ mỗi nhóm được ghép lại để tạo ra đầu ra, giúp mạng tập trung vào các đặc trưng quan trọng hơn và nâng cao hiệu suất tổng thể [21].

Xét ví dụ với một đầu vào  $x$  sẽ được chia thành các nhóm nhỏ hơn và nhóm thứ  $i$  được ký hiệu là  $x_i$ . Từng nhóm nhỏ sẽ được phân chia thành hai phần là  $x_{i1}$  và  $x_{i2}$  để xử lý song song nhờ khối “split”. Đối với  $x_{i1}$  sẽ áp dụng Channel Attention bằng cách sử dụng hàm Global Average Pooling (GAP) để trích xuất thông tin toàn cục. Kết hợp với hàm Fully Connected (FC) và Sigmoid được áp dụng để chọn lọc thông tin một cách chính xác và linh hoạt ta có:

$$x_{i1}' = \sigma\{FC[GAP(x_{i1})]\} \otimes x_{i1} \quad (1)$$

Tương tự đối với Spatial Attention, thay GAP bằng hàm Group Normalization (GN) ta được:

$$x_{i2}' = \sigma\{FC[GN(x_{i2})]\} \otimes x_{i2} \quad (2)$$

Cuối cùng đầu ra  $x_i'$  được tạo ra bằng cách ghép nối  $x_{i1}'$  và  $x_{i2}'$

$$x_i' = x_{i1}' \odot x_{i2}' \quad (3)$$

#### b) Efficient Channel Attention

Efficient Channel Attention (ECA) là một module chú ý kênh được đề xuất để cải thiện hiệu suất bằng cách giảm độ phức tạp tính toán của các Channel Attention truyền thống, đồng thời vẫn duy trì hiệu quả trong việc nắm bắt thông tin quan trọng giữa các kênh [21]. Ý tưởng chính của ECA là sử dụng một hàm ánh xạ trực tiếp (thường là một lớp tích chập 1D) để tính toán trọng số cho mỗi kênh,

thay vì sử dụng các lớp fully connected tốn kém về mặt tính toán. Cụ thể, hàm GAP được áp dụng để tính trung bình giá trị của mỗi kênh trên toàn bộ feature map đầu vào  $X_{in}$  và sau đó áp dụng thêm hàm tương tác giữa các kênh (CC - cross-channel interaction) để tạo ra đặc trưng tổng hợp  $X_c$ :

$$X_c = CC[GAP(X_{in})] \quad (4)$$

Sau đó ECA xem xét tương tác giữa mỗi kênh và  $y$  kênh lân cận của nó. ECA tránh giảm chiều bằng tích chập 1D, thay vào đó thực hiện tương tác đa kênh một cách hiệu quả. Trọng số  $\alpha_i$  của các đặc trưng  $X_{ci}$  (thuộc tập  $y$  kênh lân cận) được tính theo công thức:

$$\alpha_i = \sigma \left( \sum_{j=1}^y W^i X_{ci}^j \right) \quad (5)$$

ECA sử dụng một phương pháp thích ứng để xác định kích thước  $y$  của kernel tích chập dựa trên số lượng kênh  $N$ . Mối quan hệ giữa chúng được biểu diễn như sau:

$$N = 2^{\beta * y - k} \quad y = \left\lfloor \frac{\log_2 N}{\beta} + \frac{\beta}{k} \right\rfloor_{odd} \quad (6)$$

Với  $\beta$  được chọn là 2 và  $k$  là 1, hàm *odd* trả về số lẻ gần nhất với giá trị.

c) *ResBlock + Convolutional Block Attention Module*

Convolutional Block Attention Module (CBAM) là một Attention Module kết hợp bao gồm: Channel Attention Module (CAM) và Spatial Attention Module (SAM) [21]. CAM sử dụng hàm GAP và Global Max Pooling (GMP) để tổng hợp thông tin. Hai kết quả này được đưa vào một Multi-Layer Perceptron (MLP) chung với một lớp ẩn. Đầu ra của MLP sau đó được đưa vào hàm Sigmoid như sau:

$$M_c(X) = \sigma\{MLP[GAP(X)] + MLP[GMP(X)]\} \quad (7)$$

Tương tự SAM cũng sử dụng GAP và GMP để trích xuất đặc trưng nhưng kết quả của chúng được nối với nhau và đưa vào nhân tích chập với kích thước  $7 \times 7$ .

$$M_s(X) = \sigma\{f^{7 \times 7}[GAP(X) \otimes GMP(X)]\} \quad (8)$$

Áp dụng vào thuật toán của CBAM đối với feature map đầu vào  $X_{in}$  ta có:

$$X_{out} = M_s(X_c) \otimes X_c \quad \text{với} \quad X_c = M_c(X_{in}) \otimes X_{in} \quad (9)$$

Khi triển khai CBAM trong ResBlock để trở thành ResCBAM thì đầu ra sẽ được triển khai là:

$$X_{out} = X_{in} + M_s(X_c) \otimes X_c \quad (10)$$

d) *Global Attention Mechanism*

Global Attention Mechanism (GAM) kế thừa cấu trúc cơ bản của CBAM, bao gồm cả CAM và SAM, được thiết kế để cải thiện khả năng biểu diễn đặc trưng so với CBAM [21]. Điểm nổi bật của GAM là việc tích hợp Shortcut Connection, một kỹ thuật được mượn từ kiến trúc ResNet. Kết nối này cho phép thông tin từ lớp đầu vào truyền trực tiếp đến lớp đầu ra, bỏ qua một số lớp trung gian theo phương trình tổng quát:

$$X_{out} = X_{in} + \{M_s[M_c(X_{in}) \otimes X_{in}] \otimes [M_c(X_{in}) \otimes X_{in}]\} \quad (11)$$

Trong thành phần CAM được thay đổi sử dụng hoán vị 3D trước, sau đó là một MLP hai lớp để khuếch đại sự phụ thuộc giữa các kênh và không gian.

$$M_c(X) = \sigma\{ReversePermutation[MLP(ReversePermutation(X))]\} \quad (12)$$

Còn đối với SAM sẽ sử dụng hai lớp tích chập  $7 \times 7$  để tích hợp thông tin không gian.

$$M_s(X) = \sigma\{BN[f^{7 \times 7}(BN + ReLU(f^{7 \times 7}(X)))]\} \quad (13)$$

Tóm lại, GAM cải thiện CBAM bằng cách tích hợp kết nối tắt, sử dụng hoán vị 3D trong chú ý theo kênh, và loại bỏ gộp cục bộ. Những cải tiến này giúp GAM nắm bắt các mối quan hệ phức tạp hơn trong dữ liệu và bảo tồn nhiều thông tin hơn từ bản đồ đặc trưng, dẫn đến hiệu suất tốt hơn trong các tác vụ.

Bốn module trên được đánh giá để cân bằng giữa độ chính xác và tốc độ xử lý, trong đó SA và ECA phù hợp cho ứng dụng thời gian thực, trong khi ResCBAM và GAM mang lại hiệu suất cao hơn ở chi phí tính toán lớn.

### 3. Kết quả và thảo luận

#### 3.1. Dữ liệu

Tập dữ liệu bao gồm 1000 mẫu ảnh RFI với kích thước 640x640, được gán nhãn theo khu vực xuất hiện nhiễu với cấu trúc phù hợp theo chuẩn dữ liệu đầu vào của YOLO. Quá trình tiền xử lý được thực hiện theo tài liệu hướng dẫn [22] gồm các bước sau:

Bước 1: Tạo project trên Google Earth Engine và chạy script theo đường dẫn: <https://code.earthengine.google.com/2035f40ee8fae0ca741d0cf895c35a9b>

Bước 2: Trong đồ thị “Radio Frequency Interference” lựa chọn thời điểm mà tại đó mức Vertical-Horizontal (VH) cao nhất để hình ảnh nhiễu RFI được cập nhật.

Bước 3: Di chuyển trên đồ thị để tìm tọa độ các vùng bị ảnh hưởng bởi RFI (xuất hiện dưới dạng các vệt màu xanh hoặc vàng). Mỗi một vùng hình chữ nhật sẽ lưu lại 2 điểm tọa độ trên cùng bên trái và dưới cùng bên phải và dữ liệu đồ thị RFI dưới dạng tệp .csv của vị trí bị ảnh hưởng nhiễu nhất bên trong vùng. Tại bước này, chúng tôi đã thực hiện tải dữ liệu từ các vùng sau: (25.78, 48.92; 27.25, 52.17), (23.06, 52.07; 25.91, 57.05), (23.49, 52.87), (25.42, 57.93), (21.43, 38.64) (các vùng chỉ có một tọa độ vì diện tích tương đối nhỏ).

Bước 4: Tạo tài khoản và truy cập vào website : <https://search.earthdata.nasa.gov/search/>. Chọn mục “SENTINEL-1A\_SLC” ở bảng kết quả tìm kiếm.

Bước 5: Dựa vào dữ liệu các vùng đã lưu, nhập các thông tin về tọa độ vùng (nếu có hai tọa độ thì chọn “Rectangle”, nếu một tọa độ thì chọn “Point”), thời gian được xác định tại các vị trí có mức VH  $\geq -15$  trong đồ thị RFI.

Bước 6: Tải hết toàn bộ các mảnh xuất hiện trong khung kết quả tìm kiếm và thực hiện với toàn bộ các vùng. Mỗi một mảnh sẽ có dung lượng khoảng 3-4 GB, toàn bộ 5 vùng có 171 mảnh bị RFI ảnh hưởng. Từng tệp dữ liệu được xử lý ở bước 7 đến 12 để về định dạng ảnh thông thường.

Bước 7: Sử dụng phần mềm SNAP để thực hiện mở các tệp dữ liệu đã tải về.

Bước 8: Tiến hành hiệu chỉnh dữ liệu: Radar => Radiometric => Calibrate.

Bước 9: Kết hợp các mảnh: Radar => Sentinel-1 TOPS => S-1 TOPS Deburst.

Bước 10: Lọc đốm và làm mịn ảnh: Radar => Speckle Filtering => Single Product Speckle Filter với tham số bộ lọc sử dụng Gamma Map.

Bước 11: Xử lý đa hướng: Radar => Multilooking với tham số Number of Range Looks là 8.

Bước 12: Chọn hướng (VH/VV) và xuất hình ảnh tương ứng với độ phân giải cao nhất sẽ thu được kết quả là ảnh đen trắng với kích thước khoảng 60 MB và phần tên sẽ có thêm cụm “\_Cal\_deb\_Spk\_ML”.

Những bức ảnh này sẽ được đưa vào một công cụ tự xây dựng cho phép người dùng khoanh vùng thủ công các khu vực bị nhiễu ảnh hưởng để cắt thành các bức ảnh với kích thước cố định và gán nhãn theo tọa độ tương đối của nhiễu. Sau khi thực hiện từ 171 tệp dữ liệu SLC thu được hơn 1000 mẫu ảnh và chúng sẽ được chia thành hai phần huấn luyện (80%) và kiểm tra (20%).

#### 3.2. Huấn luyện mô hình

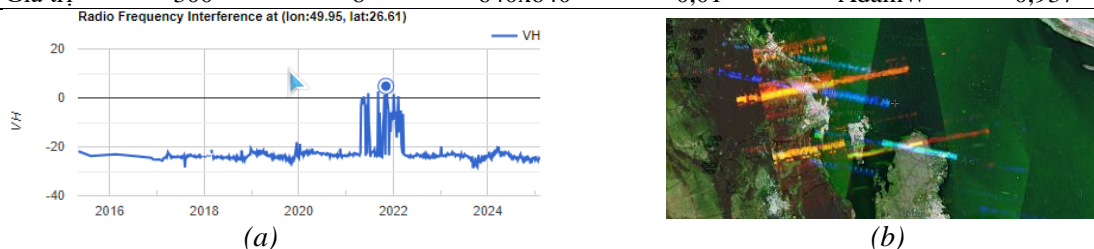
Cấu hình phần cứng được sử dụng cho các thí nghiệm là GPU (GTX1660 6GB), CPU (Intel Xeon(R) W-1270E CPU 3.40GHz  $\times$  16), RAM 16GB. Quá trình đào tạo được thực hiện bằng khung học sâu PyTorch 2.5.1 và Python phiên bản 3.12.8. Bảng 1 tóm tắt các thiết lập tham số cần thiết cho quá trình đào tạo.

#### 3.3. Kết quả

Để đánh giá rõ ràng tác động của các Attention Module đến hiệu quả hoạt động của mô hình YOLOv11, chúng tôi đã tiến hành so sánh hiệu suất của nó với các phiên bản YOLO trước đó. Kết quả so sánh cụ thể được tổng hợp và trình bày chi tiết trong Bảng 2.

**Bảng 1.** Tham số cài đặt huấn luyện mô hình

Tham số	Epochs	Batch Size	Imgsize	Learning rate	Optimizer	Momentum
Giá trị	300	8	640x640	0,01	AdamW	0,937

**Hình 4.** Kết quả chạy script trên Earth Engine: (a) Đồ thị RFI theo thời gian và (b) ảnh khu vực bị RFI tác động**Bảng 2.** Tổng hợp kết quả các mô hình YOLO

Mô hình	mAP <sub>0,5</sub>	mAP <sub>0,5-0,95</sub>	Precision	Recall	Parameters	Inference (ms)	Model Size (MB)	Training time (h)
YOLOv9t	0,901	0,798	0,961	0,860	1.970.979	5,5	4,7	1,874
YOLOv10n	0,908	0,797	0,970	0,852	2.694.806	5,2	5,8	1,628
YOLOv11n	0,903	0,795	0,971	0,868	2.582.347	4,9	5,5	1,383
YOLOv11n+ECA	0,899	0,788	0,988	0,855	2.582.359	4,1	5,5	1,428
YOLOv11n+GAM	0,907	0,794	0,963	0,865	3.263.627	8,0	6,9	2,330
YOLOv11n+SA	0,911	0,803	0,974	0,878	2.582.563	4,1	5,5	1,478
YOLOv11n+ResCBAM	0,907	0,800	0,970	0,865	3.815.571	5,2	8,0	1,637

Dựa trên số liệu kết quả của từng phiên bản, chúng tôi nhận thấy các mô hình cơ sở như YOLOv9t, YOLOv10n và YOLOv11n cho thấy sự tương đồng về hiệu suất phát hiện đối tượng, với các chỉ số mAP@0,5 và mAP@0,5-0,95 khá gần nhau. Trong số đó, YOLOv9t có số lượng tham số ít nhất (1,97 triệu), cho thấy sự cân bằng tốt giữa hiệu suất và độ phức tạp. Tuy nhiên, việc bổ sung thêm Attention Module vào phiên bản YOLOv11n cho thấy những kết quả cải thiện đáng kể ở một vài tham số nhất định. Cụ thể là:

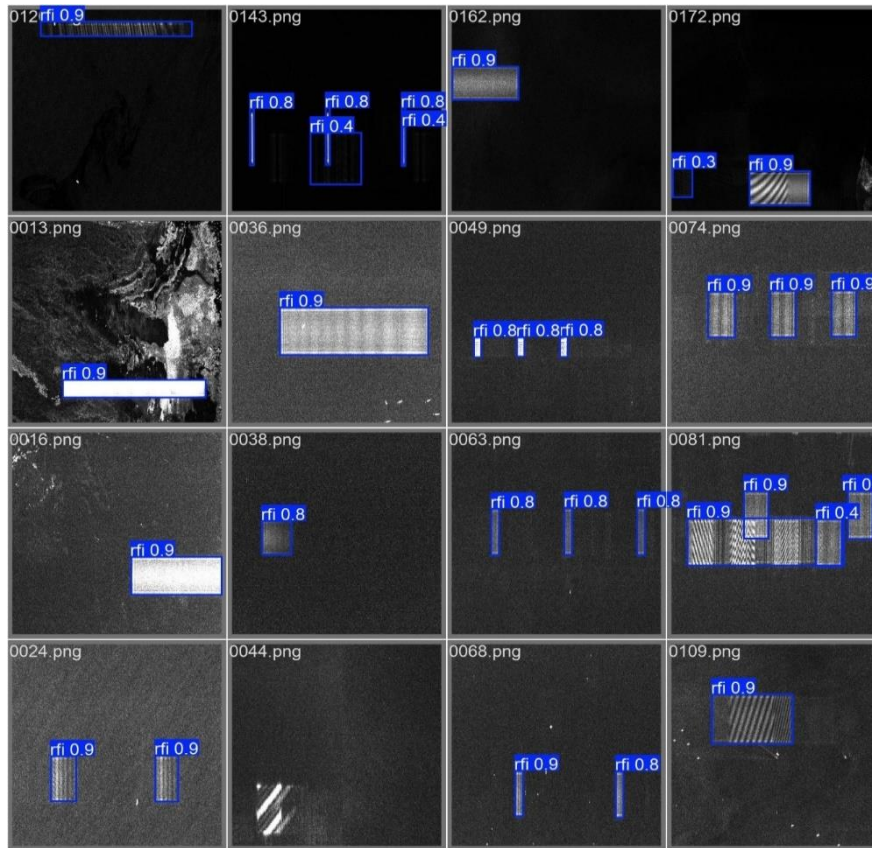
Với module ECA, mặc dù mAP@0,5 giảm nhẹ từ 0,903 xuống 0,899 và mAP@0,5-0,95 giảm từ 0,795 xuống 0,788, Precision lại tăng đáng kể từ 0,971 lên 0,988, cho thấy mô hình có khả năng dự đoán chính xác hơn nhưng lại bỏ lỡ một số đối tượng, thể hiện qua việc Recall giảm từ 0,868 xuống 0,855. Một điểm đáng chú ý là việc thêm ECA giúp giảm thời gian suy luận xuống còn 4,1 ms so với 4,9 ms của YOLOv11n nguyên gốc, trong khi số tham số chỉ tăng không đáng kể. Điều này cho thấy ECA có thể giúp mô hình tập trung vào các kênh đặc trưng quan trọng mà không làm tăng đáng kể độ phức tạp tính toán.

Ngược lại, module GAM giúp tăng nhẹ mAP@0,5 từ 0,903 lên 0,907 và mAP@0,5-0,95 từ 0,795 lên 0,794. Tuy nhiên, sự cải thiện này đi kèm với việc giảm Precision từ 0,971 xuống 0,963 và Recall từ 0,868 xuống 0,865. Điều đáng chú ý nhất là số tham số của mô hình tăng lên đáng kể từ 2,58 triệu lên 3,26 triệu, dẫn đến thời gian suy luận tăng lên 8,0 ms và thời gian huấn luyện mô hình kéo dài hơn (2,330 giờ). Điều này cho thấy GAM có thể mang lại sự cải thiện về hiệu suất phát hiện, nhưng đánh đổi bằng chi phí tính toán và thời gian huấn luyện cao hơn.

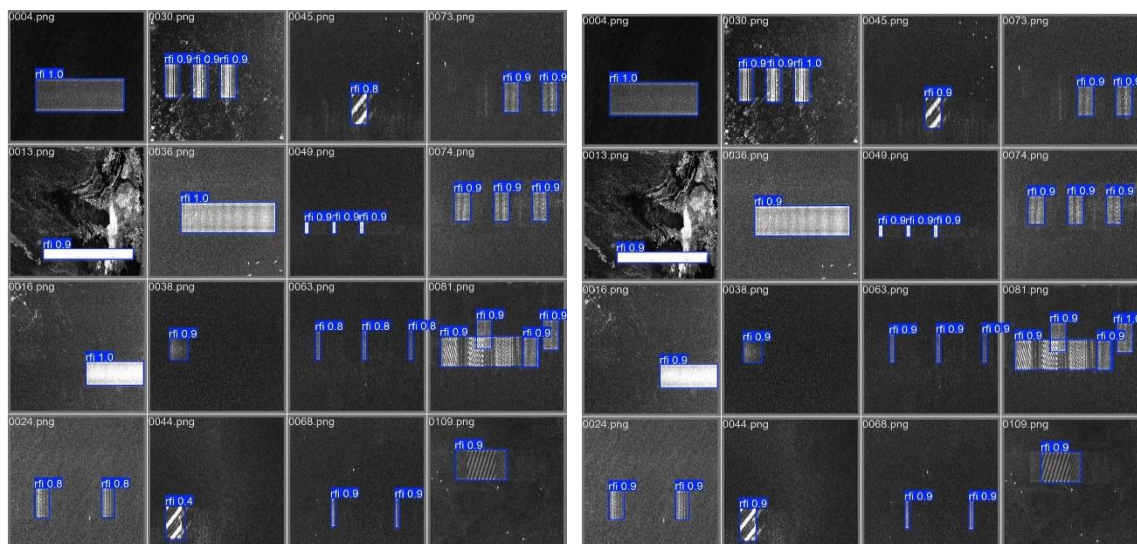
Với kết quả tương tự GAM, mô hình ResCBAM đạt được mAP@0,5 (0,907) và mAP@0,5-0,95 (0,800), đồng thời có Precision (0,970). Tuy nhiên, việc sử dụng ResCBAM khiến số tham số của mô hình tăng lên cao nhất (3,81 triệu) và thời gian suy luận cũng tăng lên 5,2 ms. Điều này cho thấy ResCBAM có thể mang lại một chút cải thiện về hiệu suất phát hiện, nhưng lại đi kèm với sự tăng lên đáng kể về độ phức tạp và chi phí tính toán.

Cuối cùng, mô hình YOLOv11n+SA có thể xem là một sự lựa chọn tối ưu trong số các Attention Module được thử nghiệm. Việc thêm SA vào YOLOv11n không chỉ cải thiện cả mAP@0,5 (từ 0,903 lên 0,911) và mAP@0,5-0,95 (từ 0,795 lên 0,803) mà còn tăng cả Precision (từ 0,971 lên 0,974) và Recall (từ 0,868 lên 0,878). Quan trọng hơn, thời gian suy luận của mô hình

cũng giảm xuống còn 4,1 ms, tương tự như YOLOv11n+ECA. Điều này cho thấy rằng SA có khả năng giúp mô hình tập trung tốt hơn vào các vị trí không gian quan trọng trong ảnh, từ đó cải thiện cả độ chính xác và khả năng phát hiện đối tượng mà không làm tăng đáng kể chi phí tính toán.

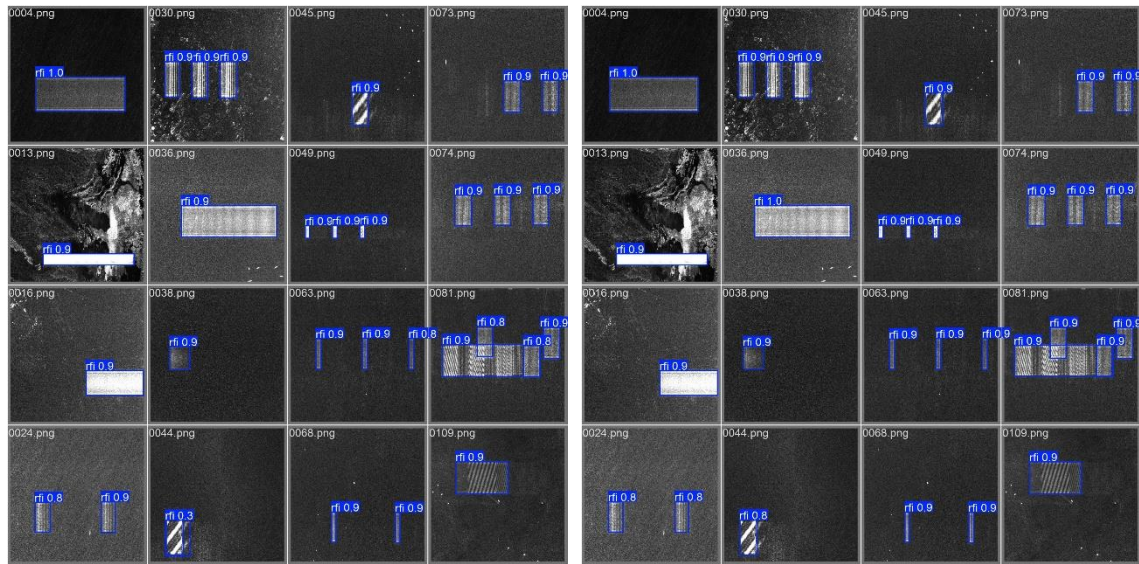


**Hình 5.** Kết quả phát hiện trên 16 ảnh với mô hình YOLOv11n



**Hình 6.** Kết quả phát hiện trên 16 ảnh với các mô hình: (a) YOLOv11n+ECA; (b) YOLOv11n+GAM

Để minh họa một cách trực quan hiệu quả của các cải tiến được đề xuất, Hình 5-7 trình bày một loạt các ví dụ phát hiện được thực hiện trên bộ dữ liệu chuẩn đối với các mô hình. Cụ thể, mô hình nguyên gốc Hình 5 đã nhận định nhầm ở mẫu thứ 2 và bỏ sót một mục tiêu ở mẫu số 14, trong khi đối với các mô hình cải tiến ở Hình 6-7 những sai sót này đã được khắc phục, ngoài ra xác suất khẳng định mục tiêu ở một số mẫu cho thấy sự cải thiện đáng kể chứng minh cho hiệu quả của phương pháp đưa ra.



(a)

(b)

Hình 7. Kết quả phát hiện trên 16 ảnh với các mô hình: (a) YOLOv11n+SA; (b) YOLOv11n+ResCBAM

#### 4. Kết luận

Bài báo đã đề xuất mô hình cải tiến YOLOv11 để phát hiện RFI trong dữ liệu ảnh Sentinel-1A. Kết quả thực nghiệm cho thấy phương pháp có thể mang lại những cải thiện tùy thuộc vào loại Attention Module và cách chúng được sử dụng. Trong khi ECA có thể cải thiện tốc độ suy luận và Precision, các module như GAM và ResCBAM có xu hướng tăng hiệu suất nhưng cũng tăng đáng kể chi phí tính toán và nổi bật với module SA với sự cân bằng giữa hiệu suất và chi phí, mang lại hiệu quả cao về mAP và Recall mà không làm tăng đáng kể thời gian suy luận. Điều này cho thấy rằng việc lựa chọn và tích hợp các Attention Module cần thực hiện một cách cẩn thận, dựa trên các yêu cầu cụ thể của ứng dụng và sự cân bằng giữa hiệu suất, độ phức tạp và chi phí tính toán.

Định hướng phát triển nghiên cứu để cải thiện hiệu quả của phương pháp đề xuất bằng cách tập trung vào việc kết hợp các module khác nhau, để tận dụng ưu điểm của từng loại và đạt được hiệu quả tổng thể cao hơn. Ngoài ra, một hướng đi khác là phát triển các Attention Module hoàn toàn mới, có khả năng tập trung vào các đặc trưng quan trọng của ảnh một cách hiệu quả hơn, đồng thời giảm thiểu độ phức tạp tính toán. Các nghiên cứu có thể hướng đến việc phát triển các Attention Module tự học (self-attention) hoặc các biến thể của attention dựa trên đồ thị (graph attention), hứa hẹn mang lại những cải tiến đáng kể về hiệu suất và độ tin cậy của mô hình.

Từ kết quả thực nghiệm cho thấy phương pháp đã đề xuất có tiềm năng ứng dụng rộng rãi trong việc xử lý ảnh vệ tinh Sentinel-1A Level-1 và đóng góp vào việc nâng cao chất lượng dữ liệu cho các ứng dụng khoa học và thực tiễn.

#### Lời cảm ơn

Nghiên cứu này được tài trợ bởi Trung tâm 80 - Cục Tác chiến điện tử trong đề tài mã số KC-KT.33/23.

## TÀI LIỆU THAM KHẢO/ REFERENCES

- [1] P. P. Sitompul, T. Manik, M. Batubara, and B. Suhandi, "Radio Frequency Interference Measurements for a Radio Astronomy Observatory Site in Indonesia," *Aerospace*, vol. 8, 2021, Art. no. 51.
- [2] M. T. Gamba, B. D. Polidori, A. Minetto, F. Dovis, E. Banfi, and F. Dominici, "GNSS Radio Frequency Interference Monitoring from LEO Satellites: An In-Laboratory Prototype," *Sensors*, vol. 24, 2024, Art. no. 508.
- [3] X. Dong, Y. Sui, Y. Li, Z. Chen, and C. Hu, "Modeling and Analysis of RFI Impacts on Imaging between Geosynchronous SAR and Low Earth Orbit SAR," *Remote Sensing*, vol. 14, 2022, Art. no. 3048.
- [4] W. Shen, Z. Qin, and Z. Lin, "A New Restoration Method for Radio Frequency Interference Effects on AMSR-2 over North America," *Remote Sensing*, vol. 11, 2019, Art. no. 2917.
- [5] M.-G. Andrea, G. Davide, and R. Andrea, "Identification of C-Band Radio Frequency Interferences from Sentinel-1 Data," *Remote Sensing*, vol. 9, 2017, Art. no. 1183.
- [6] D. Yi, F. Weiwei, Z. Zijing, Z. Feng, and L. Bingbing, "Radio Frequency Interference Mitigation for Synthetic Aperture Radar Based on the Time-Frequency Constraint Joint Low-Rank and Sparsity Properties," *Remote Sensing*, vol. 14, 2022, Art. no. 775.
- [7] A. Recchia, N. Franceschi, A. Cotrufo, R. Piantanida, D. Giudici, and A. M. Guarnieri, "Exploiting Sentinel-1 data to assess Radio Frequency Interference impact on geosynchronous SAR," *EUSAR 2021; 13th European Conference on Synthetic Aperture Radar*, 2021, pp. 1-4.
- [8] M. H. Same, G. Gleeton, G. Gandubert, P. Ivanov, and R. J. Landry, "Multiple Narrowband Interferences Characterization, Detection and Mitigation Using Simplified Welch Algorithm and Notch Filtering," *Applied Sciences*, vol. 11, 2021, Art. no. 1331.
- [9] G. Li, W. Ye, G. Lao, S. Kong, and D. Yan, "Narrowband Interference Separation for Synthetic Aperture Radar via Sensing Matrix Optimization-Based Block Sparse Bayesian Learning," *Electronics*, vol. 8, 2019, Art. no. 458.
- [10] A. Chojka, P. Artiemjew, and J. Rapiński, "RFI Artefacts Detection in Sentinel-1 Level-1 SLC Data Based On Image Processing Techniques," *Sensors*, vol. 20, 2020, Art. no. 2919.
- [11] Z. Fu, H. Zhang, J. Zhao, N. Li, and F. Zheng, "A Modified 2-D Notch Filter Based on Image Segmentation for RFI Mitigation in Synthetic Aperture Radar," *Remote Sensing*, vol. 15, 2023, Art. no. 846.
- [12] W. Xing, W. Xu, P. Huang, W. Tan, J. Hu, and M. Zhang, "A Novel Radio Frequency Interference Suppression Method for SAR Based on Spectrum Extrapolation," *Photonics & Electromagnetics Research Symposium - Fall (PIERS - Fall)*, 2019, pp. 1300-1305.
- [13] Y. Huang, L. Zhang, J. Li, Z. Chen, and X. Yang, "Reweighted Tensor Factorization Method for SAR Narrowband and Wideband Interference Mitigation Using Smoothing Multiview Tensor Model," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, pp. 3298 - 3313, 2020.
- [14] H. Zhang, L. Min, J. Lu, J. Chang, Z. Guo, and N. Li, "An Improved RFI Mitigation Approach for SAR Based on Low-Rank Sparse Decomposition: From the Perspective of Useful Signal Protection," *Remote Sensing*, vol. 14, 2022, Art. no. 3278.
- [15] D. W. Shin, C. S. Yang, and S. J. K. Chowdhury, "Enhancement of Small Ship Detection Using Polarimetric Combination from Sentinel-1 Imagery," *Remote Sensing*, vol. 16, 2024, Art. no. 1198.
- [16] J. Yu, J. Li, B. Sun, Y. Jiang, and L. Xu, "Multiple RFI Sources Location Method Combining Two-Dimensional ESPRIT DOA Estimation and Particle Swarm Optimization for Spaceborne SAR," *Remote Sensing*, vol. 13, 2021, Art. no. 1207.
- [17] J. Yu, J. Li, B. Sun, J. Chen, C. Li, W. Li, and L. Xu, "Single RFI Localization Based On Conjugate Cross-Correlation Of Dual-Channel SAR Signals," *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, 2019, pp. 385-388.
- [18] W. Fan, F. Zhou, M. Tao, X. Bai, P. Rong, S. Yang, and T. Tian, "Interference Mitigation for Synthetic Aperture Radar Based on Deep Residual Network," *Remote Sensing*, vol. 11, 2019, Art. no. 1654.
- [19] J. Wang, T. Zheng, P. Lei, and X. Bai, "A Hierarchical Convolution Neural Network (CNN)-Based Ship Target Detection Method in Spaceborne SAR Imagery," *Remote Sensing*, vol. 11, 2019, Art. no. 620.
- [20] A. Sharma, V. Kumar, and L. Longchamps, "Comparative performance of YOLOv8, YOLOv9, YOLOv10, YOLOv11 and Faster R-CNN models for detection of multiple weed species," *Smart Agricultural Technology*, vol. 9, 2024, Art. no. 100648.
- [21] C.-T. Chien, R.-Y. Ju, K.-Yi Chou, E. Xieerke, and J.-S. Chiang, "YOLOv8-AM: YOLOv8 Based on Effective Attention Mechanisms for Pediatric Wrist Fracture Detection," *International Conference on Neural Information Processing*, 2024, pp. 52461-52477.
- [22] NASA EarthData, "Create an RGB Composite from Multi-Temporal Sentinel-1 Data," 2024. [Online]. Available: <https://www.earthdata.nasa.gov/learn/data-recipes/create-rgb-composite-from-multi-temporal-sentinel-1-data>. [Accessed February 15, 2025].