CAAI Transactions on Intelligence Technology

WILEY

The Institution of Engineering and Technology

ORIGINAL RESEARCH OPEN ACCESS

TMSA-Net: Transformer-Based Multi-Scale Attention U-Net for Flood Image Segmentation

Parham Imanzadeh Charandabi | Shivakumara Palaiahnakote | Apostolos Antonacopoulos | Christian Clausner

PRImA Research Lab, School of Science, Engineering and Environment, University of Salford, Salford, UK

Correspondence: Shivakumara Palaiahnakote (S.Palaiahnakote@salford.ac.uk)

Received: 21 January 2026 | **Revised:** 14 May 2026 | **Accepted:** 15 July 2026

Keywords: attention mechanism | deep learning | disaster response | flood detection | multi-scale processing | semantic segmentation | vision transformer

ABSTRACT

Flood detection is essential for real-time applications, including disaster management, emergency response, and alerting people in flood zones. For successful flood detection, accurate flood region segmentation is essential. However, the flood region segmentation is challenging due to the complex background and occlusions with debris and the effect of external adverse factors. In contrast to existing models that focus on satellite and remote sensing images, for real-time applications, this study proposes a new Transformer-based Multi-Scale-Attention U-Net (TMSA-Net) model for flood region segmentation in images with cluttered backgrounds. The proposed model extracts features hierarchically using a Convolutional Neural Network (CNN) encoder. The deepest features are fed to an adapted vision transformer for extracting global context. In the decoder, Multi-Scale Attention Gates are used in skip connections to selectively filter encoder features before fusion. The proposed model is compared with state-of-the-art methods, including Swin-UNet, DeepLabV3+, TransUNet, UNet++, and Attention U-Net. Experimental results show that TMSA-Net achieves a test IoU of 92.01%, with improvements of 2.82% and 5.95% IoU over DeepLabV3+ and TransUNet, respectively, demonstrating the effectiveness of the proposed approach. The newly created experimental dataset, with ground truth at the pixel level, will be publicly available (released upon acceptance).

1 | Introduction

Due to unpredictable weather and climate change, flood incidents have become one of the most destructive natural hazards [1, 2]. Real-time and accurate flood region segmentation is essential for disaster management, emergency response, and alerting affected populations. However, segmenting flood regions presents significant challenges: complex backgrounds with trees and debris, unstructured water region shapes, large variations in water surface appearance, and unpredictable scene conditions, as illustrated in Figure 1 [3–5].

Most existing models are developed for satellite and remote sensing images, which suffer from temporal delays, cloud occlusion during storm events, and coarse spatial resolution inadequate for street-level decision-making [3, 5]. Ground-level imagery from Unmanned Aerial Vehicles (UAVs), Closed-Circuit Television (CCTV) networks, and smartphone cameras offers immediate, high-resolution data critical for real-time disaster response [6]. However, this heterogeneous imagery presents unique challenges: (1) extreme viewpoint variation from street-level photos to drone angles, (2) dramatic scale variation where distant floods occupy 5–10 pixels while foreground water spans 100+ pixels, (3) variable image quality from 4K drone footage to compressed CCTV streams, and (4) frequent occlusion by buildings, vegetation, and infrastructure [7].

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2026 The Author(s). *CAAI Transactions on Intelligence Technology* published by John Wiley & Sons Ltd on behalf of The Institution of Engineering and Technology and Chongqing University of Technology.

Some models use sophisticated networks for flood image detection [8, 9]. However, these models require a large number of training samples and are computationally expensive. In addition, these models fail to perform well when the features of the flood region resemble the features of background objects. Therefore, the conventional pipeline shown in Figure 2, including U-Net and different attention-based models, may not cope with the challenges of flood images with cluttered backgrounds [10]. These observations motivated us to introduce multi-scale attention gates to integrate with the baseline U-Net and dual attention refinement for addressing the challenges of flood image detection. Inspired by the multi-scale attention gates and dual attention refinement, which are special for extracting focused information in images [10, 11] We explore the same models in a new way in this work for flood region segmentation. Therefore, the key contributions of the proposed work are listed below.

(i) We create a new ground-level flood segmentation dataset with 320 images and pixel-level binary segmentation, addressing the lack of ground-level flood segmentation benchmarks in the literature [we will release it upon acceptance].

(ii) We introduce a transformer bottleneck that replaces the standard CNN bottleneck and captures global context across the entire scene, enabling the model to connect distant flood regions separated by occlusion.

(iii) We propose Multi-Scale Attention Gates in the decoder skip connection that selectively filter encoder features at multiple receptive fields (1×1, 3×3, and 5×5 kernels), allowing the model to handle flood regions ranging from small puddles to large inundation zones.

(iv) We fuse sequential channel attention and spatial attention in a new way, inspired by CBAM [12], for fine-tuning the boundary of the detected water region.

The remainder of this paper is structured as follows. A comprehensive literature review on flood detection and image segmentation is presented in Section 2. Section 3 describes the steps of the proposed architecture and concepts. Experimental results and analysis to validate the performance of the methods are presented in Section 4. Section 5 summarises the findings and inferences of the proposed work.



FIGURE 1 | Examples of complex flooded scenes.

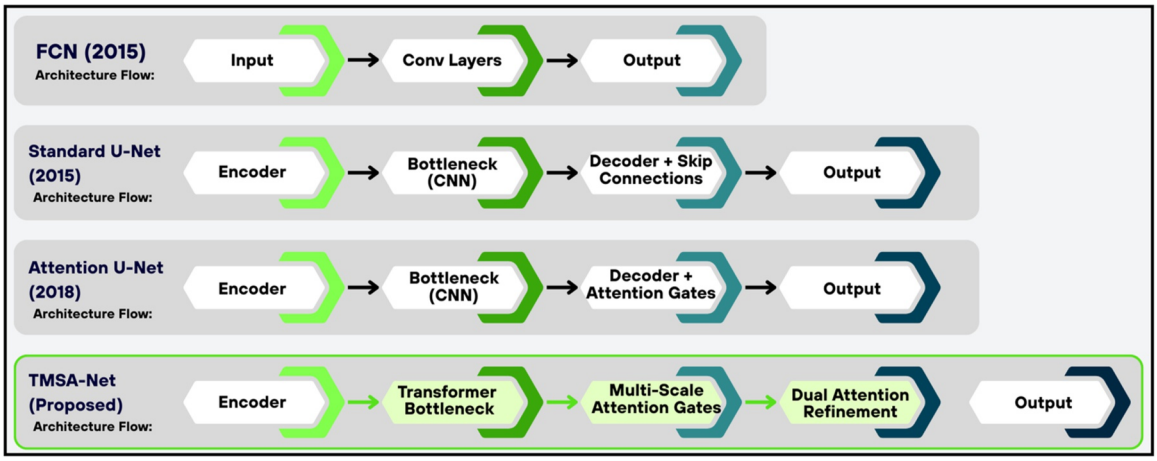


FIGURE 2 | Architectural evolution comparison.

2 | Literature Review

In this section, we review flood detection methodologies, which include conventional approaches and state-of-the-art deep learning architectures.

2.1 | Conventional Flood Detection Methods

Conventional flood detection has evolved through two primary approaches: ground-based monitoring systems and satellite remote sensing, each offering distinct capabilities while exhibiting fundamental limitations for real-time emergency response.

Ground-Based Sensor Networks: Traditional monitoring uses sensor networks, river gauges, and weather stations measuring rainfall and water levels at discrete locations [13]. These measurements feed into hydrological models that simulate water flow [1]. However, three critical limitations emerge: spatial coverage is restricted to sensor locations, and installation costs are prohibitively expensive [4], and models may not capture rapidly evolving flash floods.

Satellite Remote Sensing: To address spatial coverage gaps, researchers adopted satellite imagery with regular revisit cycles [14]. Synthetic Aperture Radar (SAR) satellites like Sentinel-1 overcome optical sensors' cloud occlusion limitations through radio wave transmission [15]. supported by benchmark datasets like Sen1Floods11 [3]. Recent advances include specialised deep learning models for SAR monitoring [16].

Despite these capabilities, satellite approaches face constraints for emergency response: temporal delays from revisit intervals [5], coarse spatial resolution (10–30m per pixel; inadequate for street-level decisions), and the inability to detect flooding obscured by canopies or buildings. These limitations create a critical operational gap, necessitating ground-level imagery approaches providing immediate, high-resolution information that conventional methods cannot deliver.

In summary, since the methods follow a conventional approach for flood image detection, they work well for simple images but not for complex images with large variations in the background and quality of the images. Therefore, the methods cannot be robust and effective for complex images.

2.2 | Deep Learning Models for Flood Detection

Deep learning, particularly CNNs, revolutionised automated flood detection through hierarchical feature learning. Early CNNs like AlexNet [17] and VGG [18] excelled at image-level classification but could not localise water. Long et al. [19] introduced Fully Convolutional Networks (FCN), replacing classification layers with convolutional layers to produce spatial prediction maps [20]. SegNet [21] introduced encoder-decoder structures to reconstruct full-resolution segmentation maps, while PSPNet [22] added pyramid pooling for multi-scale context. These architectures establish dense predictions but share a fundamental constraint: convolutional operations are inherently local. Each neuron's limited receptive field prevents understanding that distant water patches belong to the same system.

The U-Net architecture [9] became the dominant approach through its skip connections, transferring high-resolution encoder features to decoder levels, particularly effective for boundary delineation [23]. U-Net's success inspired numerous variants. ResUNet [24] added residual connections for better gradient flow. U-Net++ [25] introduced nested skip connections, and U-Net 3+ [26] connected every encoder layer to every decoder layer. However, standard skip connections transfer features indiscriminately, propagating noise. Attention U-Net [11] introduced attention gates using decoder context to filter encoder features and suppress irrelevant background. With focal loss, it achieved state-of-the-art results for Sentinel-1 SAR detection [13]. Spatial attention mechanisms integrated with bilateral U-Net structures have shown promise for handling imbalanced datasets with small-scale samples [27]. However, its gates use only single-scale (1×1) convolutions, limiting multi-scale pattern capture.

Beyond flood detection applications, the attention-enhanced U-Net architecture has recently shown strong results in water body segmentation from satellite imagery. Jonnala et al. [28] proposed DSIA U-Net, which utilises a deep-shallow interaction mechanism paired with a Nested Attention Module to process local and global features jointly, achieving 0.954 IoU on Sentinel-2 water body extraction—similarly, Jonnala et al. [29] introduced AER U-Net, which strengthens skip connections through residual blocks and self-attention refinement, reaching 0.945 IoU on the same data source. These methods demonstrate clear gains from integrating attention into U-Net variants. Still, both rely on satellite imagery where water bodies can be identified through spectral indices like NDWI, making segmentation comparatively straightforward. Ground-level flood scenes are considerably more difficult since water blends with roads and mud, buildings block the view, and lighting shifts between frames. This makes it essential for models to grasp the broader scene while still capturing precise boundaries. To handle objects of varying sizes, DeepLabV3+ [30] introduced Atrous Spatial Pyramid Pooling (ASPP), applying dilated convolutions at multiple rates (6, 12, 18) to capture context at different scales, demonstrating effectiveness for post-disaster aerial imagery [31]. However, ASPP operates uniformly without context-aware adaptation and cannot prioritise flood-relevant regions. While deep learning flood research primarily targets satellite imagery with advanced architectures (Graph Attention networks [32], Vision Transformers [33] ground-level research remains sparse, largely restricted to classification [34–36] or specific platforms like Unmanned Aerial Vehicle (UAV) search-and-rescue [6]. Recent segmentation studies for drone [7] or CCTV imagery [37] Do not resolve standard CNN limitations regarding local receptive fields, noise transfer, and scale variation.

In summary, it is noted from the above review that although the CNN-based segmentation (including U-Net-based models) has been developed for flood image detection, the methods do not perform well because of the missing global context information.

2.3 | Transformer-Based Models for Segmentation

Transformers, originally developed for natural language processing [38], have been adapted to computer vision to address CNNs' limited receptive fields through global self-attention.

The Vision Transformer (ViT) [39] divides images into patches and processes them as tokens using self-attention, enabling every patch to attend to all others. This captures long-range dependencies essential for connecting distant flood regions. However, ViT loses spatial structure, sacrificing the inductive biases that make CNNs effective for precise localisation. To combine local precision with global context, researchers proposed hybrids. TransUNet [40] pairs a CNN encoder with a Transformer middle section before a CNN decoder, achieving strong medical imaging results. However, placing Transformers in the encoder can disrupt hierarchical feature extraction. Swin-UNet [41] uses a pure Transformer design with shifted windows for efficiency but sacrifices convolutional inductive biases. Other hybrids like UTNet [42], TransFuse [43], and CoTr [44] Integrate Transformers at various stages. Recent work has demonstrated that serial and parallel integration of CNN and Transformer can promote feature interactions across various semantic levels through cross-window self-attention mechanisms [45]. Systematic reviews identify three integration approaches: sequential, parallel, and encoder-decoder hybrids, concluding that placement critically impacts performance and cost, with strategic bottleneck placement remaining underexplored.

Parallel to Transformer development, attention mechanisms emerged for feature selection. Channel attention (SENet) [46] emphasises important feature channels, while spatial attention highlights key locations. CBAM [12] combines both sequentially, validated across autonomous driving, segmentation, and medical imaging. Variational attention mechanisms that model uncertainty in latent spaces have been proposed to enhance boundary perception through multi-scale pooling operations [47]. However, these mechanisms are typically applied independently rather than integrated with decoder-guided multi-scale gating.

Despite these advances, no existing architecture strategically places Transformers at the bottleneck while simultaneously integrating decoder-guided attention gates with multi-scale processing (1×1, 3×3, 5×5) and dual refinement. This holistic integration is essential for ground-level flood imagery presenting extreme viewpoint variation, dramatic scale differences, and frequent occlusion.

Overall, it is observed from the above review that conventional methods were developed for flood monitoring, but the methods cannot extract spatial and temporal information to obtain fine local information. CNN-based deep learning, particularly the U-Net, revolutionised automated detection but struggles with limited global context, indiscriminate feature fusion, and inadequate multi-scale processing. The transformer-based hybrids introduced global reasoning but often disrupt spatial hierarchies or add prohibitive computational cost. To alleviate the above limitations, the proposed work aims to develop a new model for flood image detection in complex and cluttered backgrounds.

3 | The Proposed Method

As discussed previously, flood region segmentation is a crucial step for disaster management. Due to background variations and unstructured water regions, dynamic surface turbulence, and the illumination effect, accurate flood region detection remains challenging. Motivated by the ability of CNNs, transformers, and U-Net for robust feature extraction and segmentation, we explore the same concepts in a new way to adapt the models for flood region segmentation in this work.

Therefore, CNN has been used for extracting hierarchical features. This extracts rich local information. The transformer is adapted for extracting global contextual features from the hierarchical features. It finds relationships between the flood region and other non-flood regions to define contextual features. Furthermore, multi-scale attention gates of U-Net are proposed for actual flood region segmentation through multi-heads. This step helps us to fine-tune the extracted features such that the method does not miss vital details in the images. The steps of the proposed flood region segmentation can be seen in Figure 3, and sample flood images and the ground truth are shown in Figure 4, showing the labelled water region for experimentation. The complete architecture can be seen in Figure 5, including detailed steps of each component of the proposed work.

3.1 | Hierarchical Feature Extraction

The encoder splits raw input images into hierarchical feature representations through progressive downsampling and channel expansion. This step uses CNNs with convolutional operations, capturing patterns at multiple semantic levels. We use pretrained CNN encoders as backbones for feature extraction, specifically EfficientNet-B4 [48]. These encoders are initialised with ImageNet pretrained weights, providing robust low-level and mid-level feature representations that transfer well to flood detection. The encoder has a depth of 5, progressively downsampling the input while increasing channel depth as defined in Equation (1).

$$\mathbf{F}_{\text{enc}} = (F_0, F_1, F_2, F_3, F_4, F_5), \tag{1}$$

where F_i represents the feature map at the stage i with spatial dimensions $H_i \times W_i$ and channel count C_i. For an input image of size 256×256, the spatial dimensions are as defined in Equation (2).

$$CapH_i = W_i \frac{256}{2^i}, \tag{2}$$

The channel configurations of the encoders of different backbone networks are as follows. EfficientNet-B4 employs [3, 48, 32, 56, 160, 448] channels across levels 0–5. Table 1 summarises the encoder output characteristics.

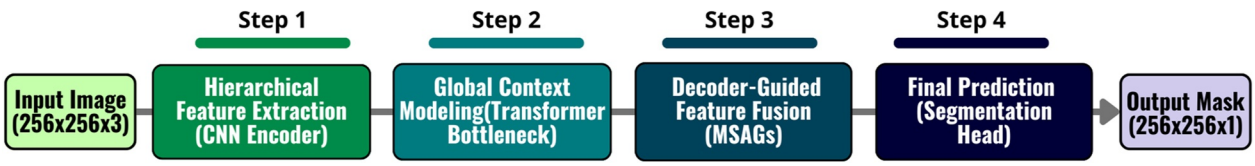


FIGURE 3 | The TMSA-Net 4-step processing workflow.

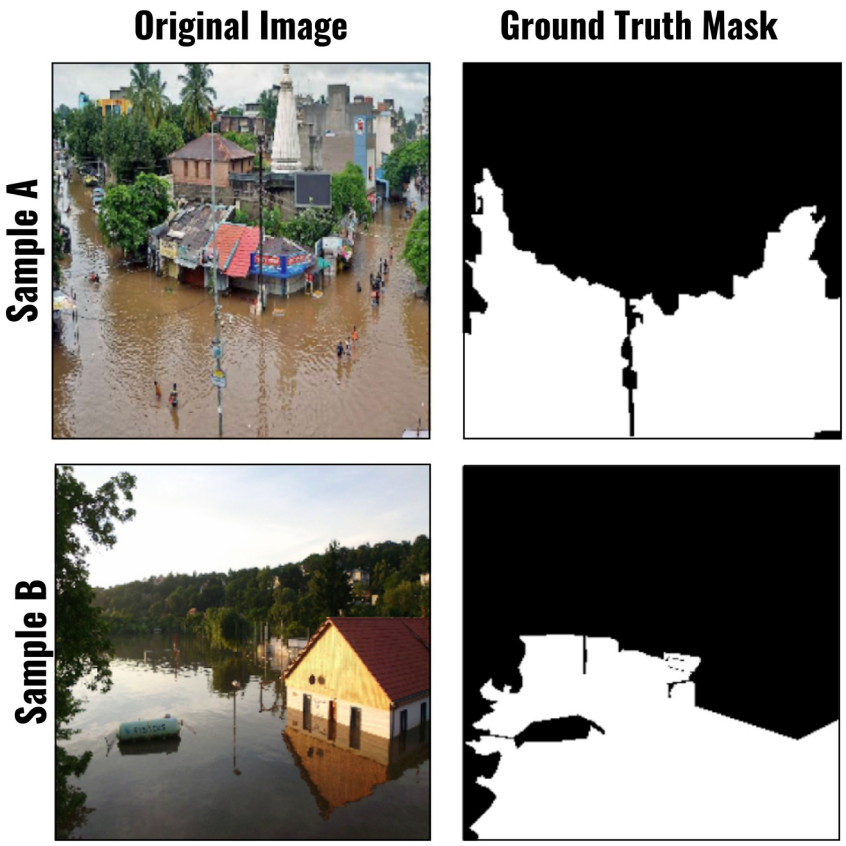


FIGURE 4 | Representative urban flood scenarios (water regions shown in white).

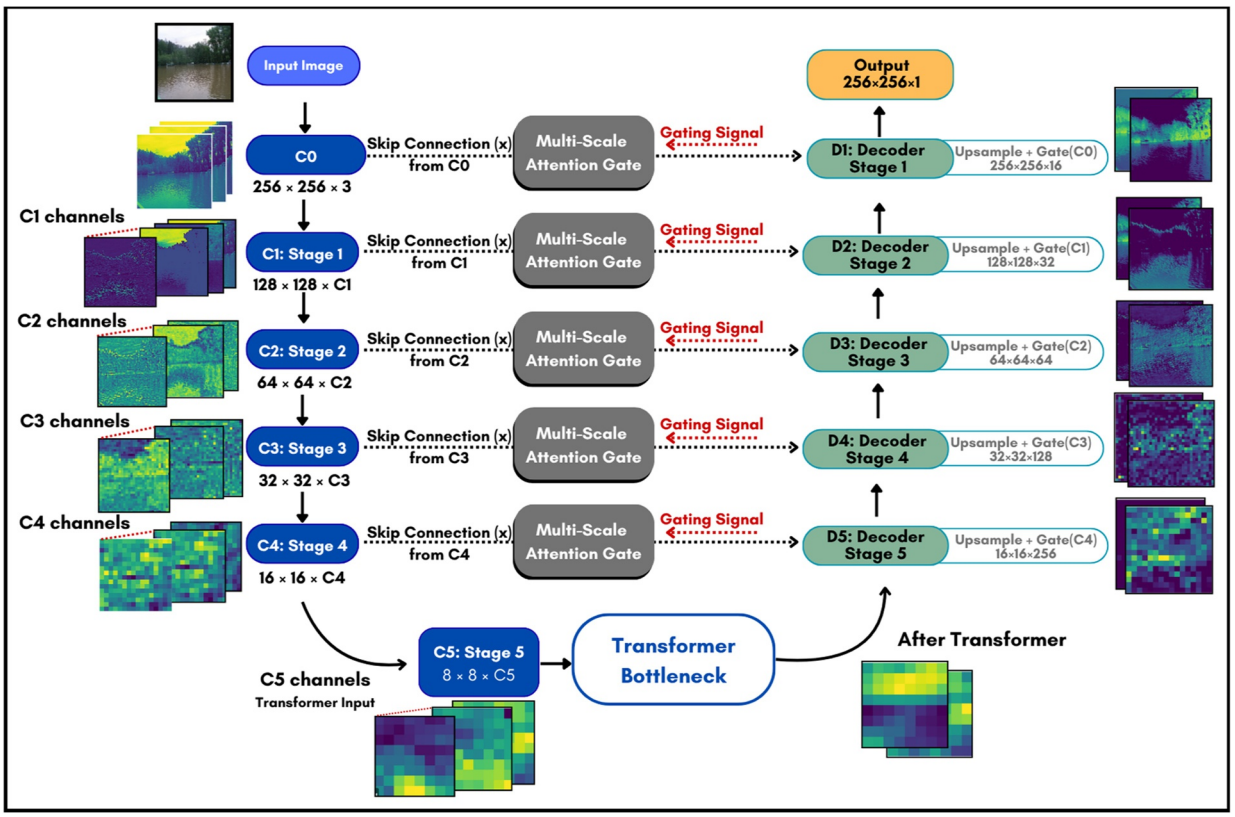


FIGURE 5 | TMSA-Net architecture overview. The 4-stage pipeline integrates: (i) CNN encoder for hierarchical spatial features, (ii) Transformer bottleneck for global context at minimal computational cost, (iii) decoder with Multi-Scale Attention Gates for intelligent feature fusion, (iv) segmentation head for final prediction.

The effectiveness of the hierarchical feature extraction is illustrated in Figure 6, which shows encoder activations across all levels for two samples (A and B). Level 1 (128 × 128) captures low-level features like edges and boundaries. Progressive abstraction through Levels 2–4 yields increasingly semantic representations. The final Level 5 (8 × 8) produces highly compressed semantic features but lacks global reasoning about spatial relationships between distant regions, a limitation we address through Transformer processing. Therefore, this step contributes to extracting features that represent the flood region in the images.

TABLE 1 | Encoder feature hierarchy.

Level	Feature map	Spatial size	Semantic content
F_0	256 × 256 × 3	Full resolution	Raw RGB input
F_1	128 × 128 × C_1	1/2 resolution	Edges, textures, colour gradients
F_2	64 × 64 × C_2	1/4 resolution	Surface patterns, local structures
F_3	32 × 32 × C_3	1/8 resolution	Object-level features (buildings, vegetation)
F_4	16 × 16 × C_4	1/16 resolution	Complex spatial layouts
F_5	8 × 8 × C_5	1/32 resolution	High-level semantic abstractions

Note: Channel dimensions are backbone-dependent. EfficientNet-B4: [48, 32, 56, 160, 448].

3.2 | Transformer Bottleneck for Extracting Global Contextual Features

Standard CNN bottlenecks employ 3 × 3 convolutions with inherently limited receptive fields. Unlike TransUNet, which places the Transformer in the encoding path and processes larger feature maps at a higher computational cost, we confine the Transformer to the bottleneck where it operates over only 64 tokens (8 × 8 spatial resolution). This keeps our computational cost low (5.52 GFLOPs), as detailed later in Table 2 (Section 4), while still achieving global context across the entire scene. Even with deep stacking, effective receptive fields grow sublinearly [10], preventing models from connecting distant flooded regions separated by obstructions, distinguishing water from visually similar surfaces, and recognising that foreground puddles and background inundation belong to the same flood event. We

TABLE 2 | Performance of the proposed and existing models in terms of computations (all models use 256 × 256 input dimension except Swin-UNet, which uses 224 × 224 to enable implementation).

Model	Encoder	Params (M)	GFLOPs
TMSA-Net (ours)	**EfficientNet-B4**	**30.58**	**5.52**
TransUNet	ResNet-50	32.52	10.67
Swin-UNet	Swin-T	31.63	6.89
DeepLabV3+	ResNet-101	45.67	14.04
UNet++	VGG-16	39.35	93.35
Attention U-Net	ResNet-34	24.55	7.85

Note: The bold indicates the best results.

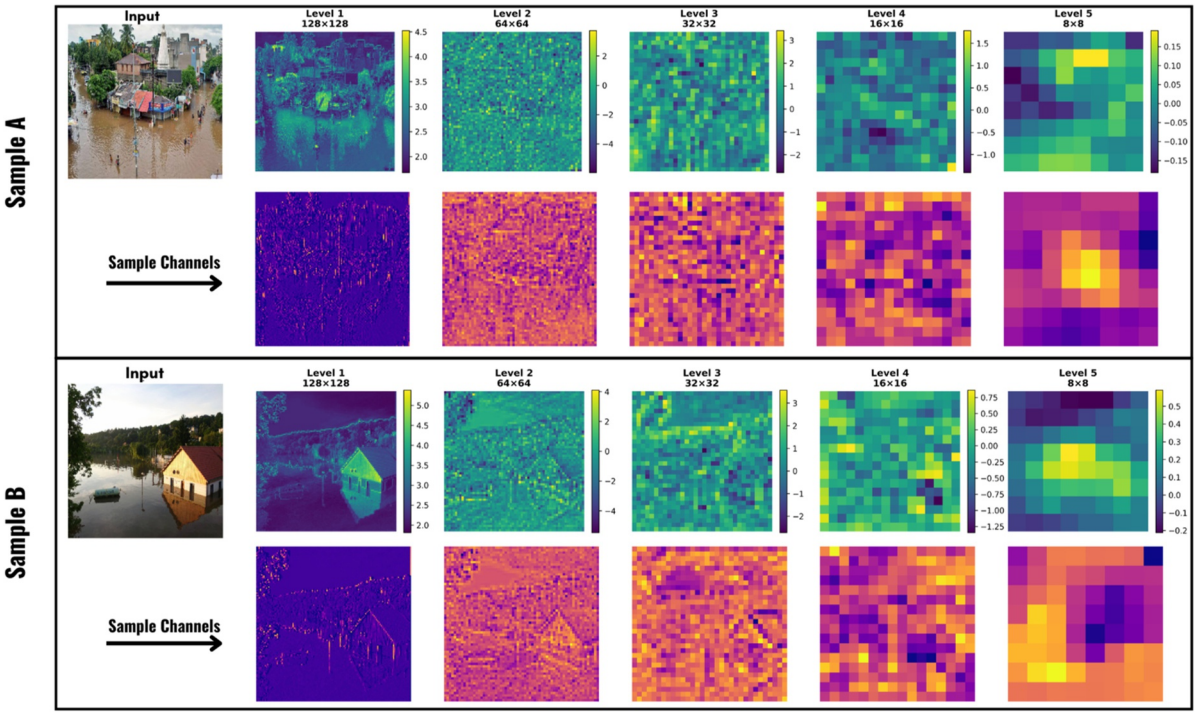


FIGURE 6 | Encoder feature visualisation across all levels for samples (A) & (B).

address this limitation by replacing the CNN bottleneck with a Transformer module that processes the deepest feature map (F_5, $8 \times 8 \times C_5$) as a sequence of spatial tokens, enabling global self-attention across all locations. The Transformer Bottleneck F_5 Consists of four steps. (i) Spatial tokenisation, (ii) Positional encoding, (iii) Multi-head self-attention, and (iv) Spatial de-tokenisation.

3.2.1 | *Spatial Tokenisation*

We flatten the 2D feature map into a sequence of spatial patches as defined in Equation (3).

$$\mathbf{X}_{\text{flat}} = \text{Flatten}(F_5) \in \mathbb{R}^{B \times N \times C5}, \tag{3}$$

where B is batch size, $N = H \times W = 64$ is the number of patches (tokens), and C_5 Is the channel dimension. This converts the spatial structure ($B \times C_5 \times 8 \times 8$) into a token sequence ($B \times 64 \times C_5$) where each of the 64 tokens represents one spatial location.

3.2.2 | *Positional Encoding*

Since self-attention is permutation-invariant, we add learnable positional embeddings to encode spatial relationships as defined in Equation (4).

$$\mathbf{X}_{\text{pos}} = \mathbf{X}_{\text{flat}} + \mathbf{P} \in \mathbb{R}^{B \times N \times C_5}, \tag{4}$$

where $\mathbf{P} \in \mathbb{R}^{1 \times 64 \times C_5}$ is a learnable parameter initialised with zeros and broadcast across the batch.

3.2.3 | *Multi-Head Self-Attention Processing*

The sequence is processed through $L = 4$ Transformer blocks, each applying layer normalisation (LN), multi-head self-attention (MHSA), and feed-forward MLP with residual connections as defined in Equations (5) and (6).

$$\mathbf{Z}' = \text{MHSA}(\text{LN}(\mathbf{Z})) + \mathbf{Z}. \tag{5}$$

$$\mathbf{Z}_{\text{out}} = \text{MLP}\big(\text{LN}\big(\mathbf{Z}'\big)\big) + \mathbf{Z}'. \tag{6}$$

Multi-head self-attention with $h = 8$ heads compute as defined in Equations (7) and (8)

$$\text{MHSA}(\mathbf{Z}) = \text{Concat}(\text{head}_1, ..., \text{head}_h)\mathbf{W}^O, \tag{7}$$

$$\text{head}_i = \text{Softmax}\left(\frac{\mathbf{Q}_i \mathbf{K}_i^T}{\sqrt{d_k}}\right)\mathbf{V}_i, \tag{8}$$

where $\mathbf{Q}_i = \mathbf{Z}\mathbf{W}_i^Q$, $\mathbf{K}_i = \mathbf{Z}\mathbf{W}_i^K$, $\mathbf{V}_i = \mathbf{Z}\mathbf{W}_i^V$ Are the query, key, and value projections for the head i, and $d_k = C_5/h$ Is the dimension per head?

The MLP consists of two linear layers with GELU activation and dropout, as defined in Equation (9)

$$\text{MLP}(\mathbf{Z}) = \text{Dropout}(\mathbf{W}_2 \cdot \text{GELU}(\mathbf{W}_1 . \mathbf{Z})), \tag{9}$$

where $\mathbf{W}_1 \in \mathbb{R}^{C_5 \times D_{\text{mlp}}}$ and $\mathbf{W}_2 \in \mathbb{R}^{D_{\text{mlp}} \times C_5}$ with $D_{\text{mlp}} = 2048$.

3.2.4 | *Spatial De-Tokenisation*

After processing through all Transformer blocks and final layer normalisation, we reshape the sequence back to 2D spatial format as shown in Equation (10)

$$F_5^{\text{Transformer}} = \text{Reshape}(\text{LN}(\text{Z}_L)) \in \mathbb{R}^{B \times C_5 \times 8 \times 8}, \tag{10}$$

where Z_L is the output of the L-th Transformer block.

The complete architecture is illustrated in Figure 7, where it can be seen how the step architecture can help us to extract the relevant features for flood region segmentation. The effectiveness of the transformer bottleneck for flood region detection is shown in Figure 8, where it is noted that the water regions are separated from other regions. The "Before" column on the left shows the CNN encoder's compressed representation with dominant central activations reflecting local processing limitations. After Transformer processing, the "After" column exhibits more distributed activation patterns.

It can be observed from "Difference" maps that extreme peaks are moderate while peripheral regions gain representation. This

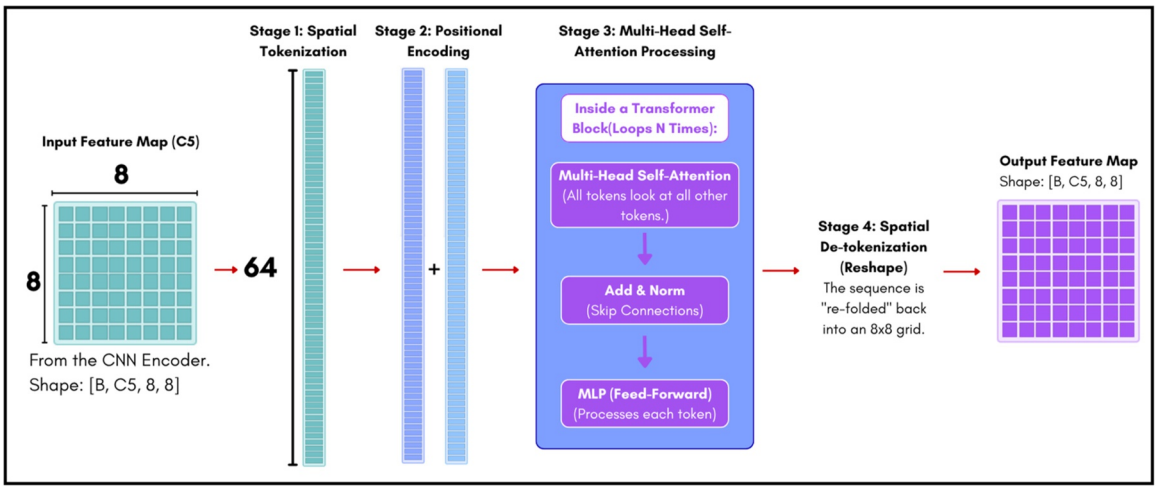


FIGURE 7 | Transformer bottleneck architecture.

redistribution encodes global contextual reasoning; each of the 64 patches now contains information about all other patches through self-attention, enabling scene-level understanding (e.g., recognising that distant water patches belong to the same river system).

The effectiveness of the combination of the CNN encoder and the transformer bottleneck for complex flood scene images and its ground truth shown in Figure 9a–b, respectively, is illustrated in Figure 9, where one can see that the CNN alone does not segment the flood region completely compared to the ground truth, as shown in Figure 9c. It misses the details, especially where the flood region is occluded by a person and a fence. However, when we look at the CNN + Transformer output for the same complex input image, it restores significant missing details—in contrast to the output of CNN only, as shown in Figure 9d. This kind of combination of the CNN and the Transformer is a useful synergy. However, it is noted from Figure 9d that the combination of the CNN and the Transformer bottleneck does not segment the flood region completely compared to the ground truth. This shows that the combination alone is not sufficient for segmenting flood regions in complex scene images.

3.3 | Multi-Scale Attention Gates for Classification

The decoder reconstructs the segmentation map through progressive upsampling while intelligently integrating encoder features via Multi-Scale Attention Gates (MSAGs). Standard attention gates [11] use single-scale 1×1 convolutions that lack multi-scale context, and Swin-UNet provides no skip connection filtering at all. In contrast, MSAGs simultaneously implement three innovations: (i) multi-scale processing at three receptive fields (1×1, 3×3, 5×5), (ii) decoder-guided gating using high-level context to generate attention coefficients, and (iii) dual refinement through sequential channel and spatial attention.

For each input, the decoder gating signal $\mathbf{g} \in \mathbb{R}^{B \times C_g \times H_g \times W_g}$ (upsampled from the previous decoder level) and the encoder

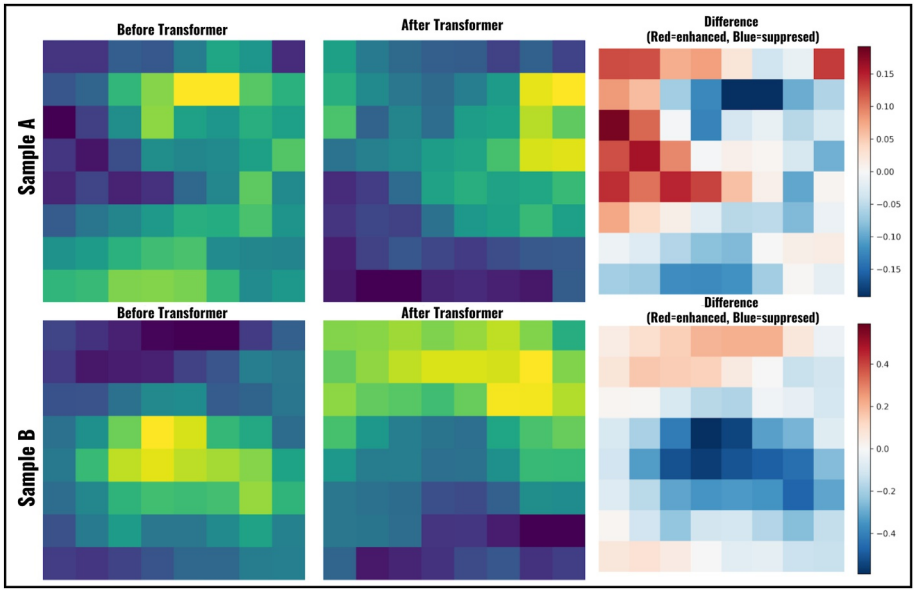


FIGURE 8 | Effect of transformer bottleneck on spatial feature representations.

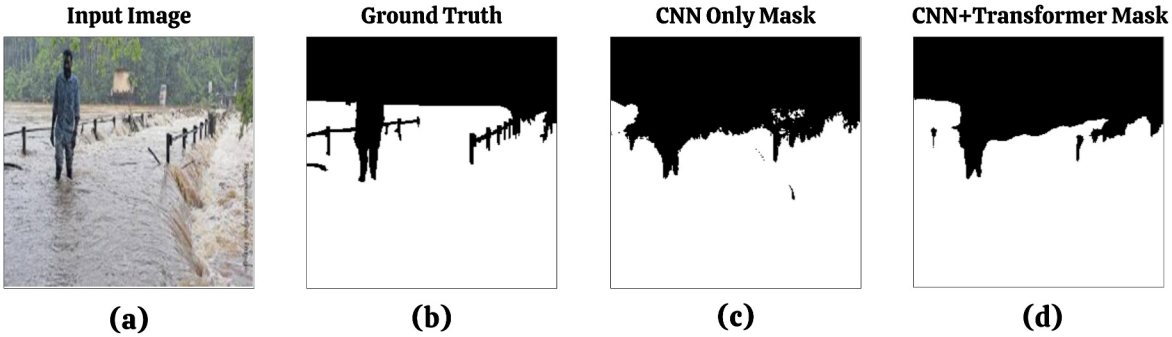


FIGURE 9 | Visual analysis of CNN-Transformer synergy on a complex flood scene containing a person, fence posts, and turbulent water. (a) Complex input image, (b) ground truth for flood region segmentation. (c) CNN-only prediction: the model segments the main flood body but misses water regions between and around thin occluding structures such as fence posts, where local receptive fields cannot connect visually separated water patches. (d) CNN + Transformer prediction: the Transformer's global self-attention enables the model to recognise that water patches separated by fence posts belong to the same flood event, recovering regions missed in (c).

skip connection $\mathbf{x} \in \mathbb{R}^{B\times C_l\times H_x\times W_x}$. The module produces refined skip features through five sequential operations:

We process the gating signal at three different kernel sizes to capture multi-scale context as defined in Equation (11).

$$\mathbf{g}_k = \mathrm{BN}(\mathrm{Conv}_{k\times k}(\mathbf{g}))\text{for } k \in \{1, 3, 5\}, \tag{11}$$

where each convolution maps C_g channels to intermediate dimension $C_{\text{int}} = C_l/2$. If spatial dimensions don't match the skip connection, we apply bilinear upsampling as defined in Equation (12).

$$\mathbf{g}_k^{\text{resized}} = \mathrm{Interpolate}\big(\mathbf{g}_k, \text{size} = (H_x, W_x)\big). \tag{12}$$

The three scales are fused through concatenation followed by a 1×1 convolution as shown in Equation (13).

$$\mathbf{g}_{\text{fused}} = \mathrm{ReLU}\big(\mathrm{BN}\big(\mathrm{Conv}_{1\times1}\big(\mathrm{Concat}\big([\mathbf{g}_1, \mathbf{g}_3, \mathbf{g}_5]\big)\big)\big)\big) \in \mathbb{R}^{B\times C_{\text{int}}\times H_x\times W_x}. \tag{13}$$

This multi-scale fusion enables simultaneous capture of point-wise features (1×1), local context (3×3), and broader patterns (5×5) which are essential for handling floods ranging from small puddles to large inundation zones.

The encoder skip connection is projected to the intermediate dimension as defined in Equation (14).

$$\mathbf{x}' = \mathrm{BN}(\mathrm{Conv}_{1\times1}(\mathbf{x})) \in \mathbb{R}^{B\times C_{\text{int}}\times H_x\times W_x}. \tag{14}$$

We combine the multi-scale decoder context with the transformed skip features through element-wise addition and generate attention coefficients as defined in Equations (15) and (16).

$$\boldsymbol{\psi} = \mathrm{ReLU}\Big(\mathbf{g}_{\text{fused}} + \mathbf{x}'\Big). \tag{15}$$

$$\boldsymbol{\alpha} = \mathrm{Sigmoid}(\mathrm{BN}(\mathrm{Conv}_{1\times1}(\boldsymbol{\psi}))) \in \mathbb{R}^{B\times1\times H_x\times W_x}. \tag{16}$$

The attention coefficients $\boldsymbol{\alpha} \in [0, 1]$ represent the decoder's assessment of feature relevance: values near 1 indicate flood-relevant regions, values near 0 indicate background or noise. This implements decoder-guided attention, the high-level semantic understanding from the decoder directs which low-level encoder features should be emphasised.

We apply the attention coefficients through element-wise multiplication as defined in Equation (17).

$$\mathbf{x}_{\text{gated}} = \mathbf{x} \odot \boldsymbol{\alpha} \in \mathbb{R}^{B\times C_l\times H_x\times W_x}, \tag{17}$$

where $\odot$ denotes element-wise multiplication with broadcasting. This soft gating allows gradients to flow during training while suppressing irrelevant features. In summary, each kernel operation gains different information in parallel. For instance, the 1×1 kernel operation extracts point-wise feature relevance at a specific location, the 3×3 kernel operation captures local structural patterns such as water-road boundaries and edges of occluding objects, while the 5×5 kernel operation captures broader spatial context, enabling the gate to recognise continuous water regions. These three observations refer to different, diverse contextual information. Further, the proposed work fuses these three responses through concatenation and a learnt 1×1 projection, producing attention coefficients that simultaneously encode fine, medium, and coarse spatial context. The gated features then pass through sequential channel attention and spatial attention, inspired by CBAM [12], for further refinement.

We apply two sequential attention mechanisms to further refine the gated features:

The channel Attention emphasises important feature types as defined in Equation (18).

$$\mathbf{x}_{\text{ca}} = \mathbf{x}_{\text{gated}} \odot \mathrm{Sigmoid}\big(\mathrm{MLP}\big(\mathrm{AvgPool}\big(\mathbf{x}_{\text{gated}}\big)\big) + \mathrm{MLP}\big(\mathrm{MaxPool}\big(\mathbf{x}_{\text{gated}}\big)\big)\big), \tag{18}$$

where AvgPool and MaxPool perform global average and max pooling across spatial dimensions, and MLP is a two-layer network with a reduction ratio of 8 as defined in Equation (19).

$$\mathrm{MLP}(\mathbf{z}) = \mathbf{W}_2(\mathrm{ReLU}(\mathbf{W}_1(\mathbf{z}))), \tag{19}$$

with $\mathbf{W}_1 \in \mathbb{R}^{(C_l/8)\times C_l}$ and $\mathbf{W}_2 \in \mathbb{R}^{C_l\times(C_l/8)}$.

Equation (20) highlights important spatial locations.

$$\mathbf{x}_{\text{out}} = \mathbf{x}_{\text{ca}} \odot \mathrm{Sigmoid}\big(\mathrm{Conv}_{7\times7}\big(\mathrm{Concat}\big(\big[\mathrm{AvgPool}_{\text{channel}}(\mathbf{x}_{\text{ca}}), \mathrm{MaxPool}_{\text{channel}}(\mathbf{x}_{\text{ca}})\big]\big)\big)\big), \tag{20}$$

where $\mathrm{AvgPool}_{\text{channel}}$ and $\mathrm{MaxPool}_{\text{channel}}$ compute average and max pooling along the channel dimension, producing two spatial maps that are concatenated and processed by a 7×7 convolution.

The final output $\mathbf{x}_{\text{out}}$ represents the refined skip connection features that will be concatenated with the upsampled decoder features. Figure 10 illustrates the complete MSAG module integrating all three innovations.

3.4 | Decoder and Segmentation Head

The decoder consists of five stages corresponding to encoder levels $4 \rightarrow 0$. At each stage i, three operations occur:

Step (i): Upsample decoder features by $2 \times$ using bilinear interpolation as defined in Equation (21)

$$\mathbf{d}_i^{\text{up}} = \mathrm{Interpolate}(\mathbf{d}_{i-1}, \text{scale} = 2). \tag{21}$$

followed by 1×1 convolution with batch normalisation and ReLU as defined in Equation (22).

$$\mathbf{d}_i^{\text{conv}} = \mathrm{ReLU}\big(\mathrm{BN}\big(\mathrm{Conv}_{1\times1}\big(\mathbf{d}_i^{\text{up}}\big)\big)\big). \tag{22}$$

Step (ii): Apply MSAG to corresponding encoder skip connection as defined in Equation (23).

$$\mathbf{s}_i = \mathrm{MSAG}\big(\mathbf{g} = \mathbf{d}_i^{\text{conv}}, \mathbf{x} = F_{5-i}\big), \tag{23}$$

where F_{5-i} is the encoder feature at the corresponding resolution.

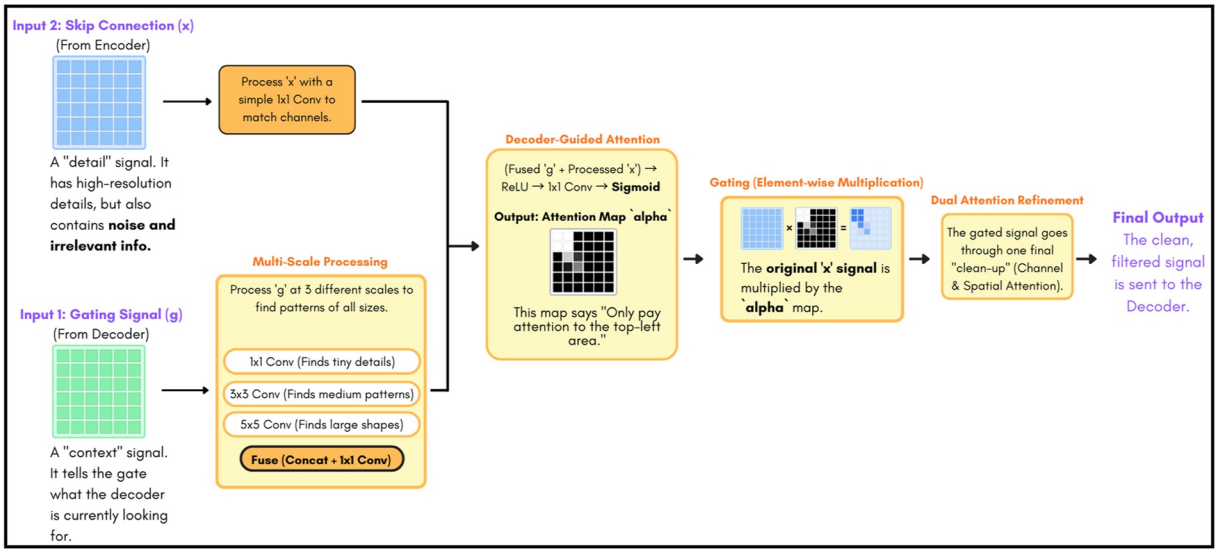


FIGURE 10 | The decoder reconstructs the segmentation map through progressive upsampling while intelligently integrating encoder features via Multi-Scale Attention Gates (MSAGs). Unlike standard U-Net skip connections that indiscriminately concatenate all encoder features, MSAGs simultaneously implement three innovations: (i) multi-scale processing at three receptive fields (1×1, 3×3, 5×5), (ii) decoder-guided gating using high-level context to generate attention coefficients, and (iii) dual refinement through sequential channel and spatial attention.

Step (iii): Concatenate and process through two 3×3 convolutional blocks as shown in Equations (24 and 25).

$$\mathbf{d}_i^{\text{cat}} = \text{Concat}\left(\left[\mathbf{d}_i^{\text{conv}}, \mathbf{s}_i\right]\right) \in \mathbb{R}^{B\times\left(C_g+C_l\right)\times H_i\times W_i}, \tag{24}$$

$$\mathbf{d}_i = \text{ReLU}\left(\text{BN}\left(\text{Conv}_{3\times3}\left(\text{ReLU}\left(\text{BN}\left(\text{Conv}_{3\times3}\left(\mathbf{d}_i^{\text{cat}}\right)\right)\right)\right)\right)\right), \tag{25}$$

where each Conv3 $\times$ 3 Block includes convolution, batch normalisation, and ReLU activation. This reduces the concatenated features to the target decoder channel count for level i. The decoder channel dimensions are [256, 128, 64, 32, 16] for levels 5 through 1, progressively reducing capacity as spatial resolution increases.

The MSAG operation across decoder levels is demonstrated in Figure 11 for both sample images. The attention coefficients (α) directly indicate which spatial regions the decoder considers relevant for flood segmentation. At early levels (i)–(ii), the model performs coarse semantic separation between foreground and background. At middle levels (iii), a clear water/non-water distinction emerges. At late levels (iv)–(v), attention maps show refined boundary delineation, increasingly resembling soft segmentation masks. This progressive refinement is precisely what standard U-Net skip connections lack.

The segmentation head consists of a single 1×1 convolution that maps the final decoder features to a single-channel output as defined in Equation (26).

$$\mathbf{M}_{\text{logits}} = \text{Conv}_{1\times1}(\mathbf{d}_5) \in \mathbb{R}^{B\times1\times256\times256}. \tag{26}$$

During inference, we apply sigmoid activation to obtain probability maps as defined in Equation (27).

$$\mathbf{M}_{\text{prob}} = \sigma\left(\mathbf{M}_{\text{logits}}\right), \tag{27}$$

and threshold at 0.5 to produce binary masks as shown in Equation (28).

$$\mathbf{M}_{\text{binary}} = 1\left[\mathbf{M}_{\text{prob}} > 0.5\right]. \tag{28}$$

The effectiveness of the MSAG for segmenting the flood region is illustrated in Figure 12 for the same sample shown in Figure 9a. It is observed from Figure 12a–c that the CNN-only model segments the main flood body but produces noisy boundaries and misses water regions around thin occluding structures, as shown in Figure 12a. When MSAG is added without the Transformer as shown in Figure 12b, the multi-scale gates substantially improve segmentation by selectively filtering encoder features at each decoder level, producing cleaner boundaries and recovering previously missed regions. The impact of the CNN + Transformer bottleneck + MSAG can be seen in Figure 12c, where the correct segmentation, which is very close to the ground truth, is obtained.

3.5 | Loss Functions

In this work, the following loss functions are defined. We employ a combined loss balancing pixel-wise classification and spatial overlap as defined in Equation (29).

$$\mathcal{L}_{\text{total}} = 0.5 \cdot \mathcal{L}_{\text{focal}} + 0.5 \cdot \mathcal{L}_{\text{dice}}. \tag{29}$$

Focal Loss addresses class imbalance by down-weighting easy examples as shown in Equation (30).

$$\mathcal{L}_{\text{focal}} = -\frac{1}{N}\sum_{i=1}^{N}\alpha\left(1-p_i\right)^{\gamma}\left[y_i\log\left(p_i\right) + \left(1-y_i\right)\log\left(1-p_i\right)\right], \tag{30}$$

where $p_i = \sigma(m_i)$ is the predicted probability for pixel i, $y_i \in \{0, 1\}$ is the ground truth label, $\alpha = 0.25$ is the balancing factor, and $\gamma = 2.0$ Is the focussing parameter.

Dice Loss directly optimises the Dice coefficient, measuring spatial overlap as Equation (31).

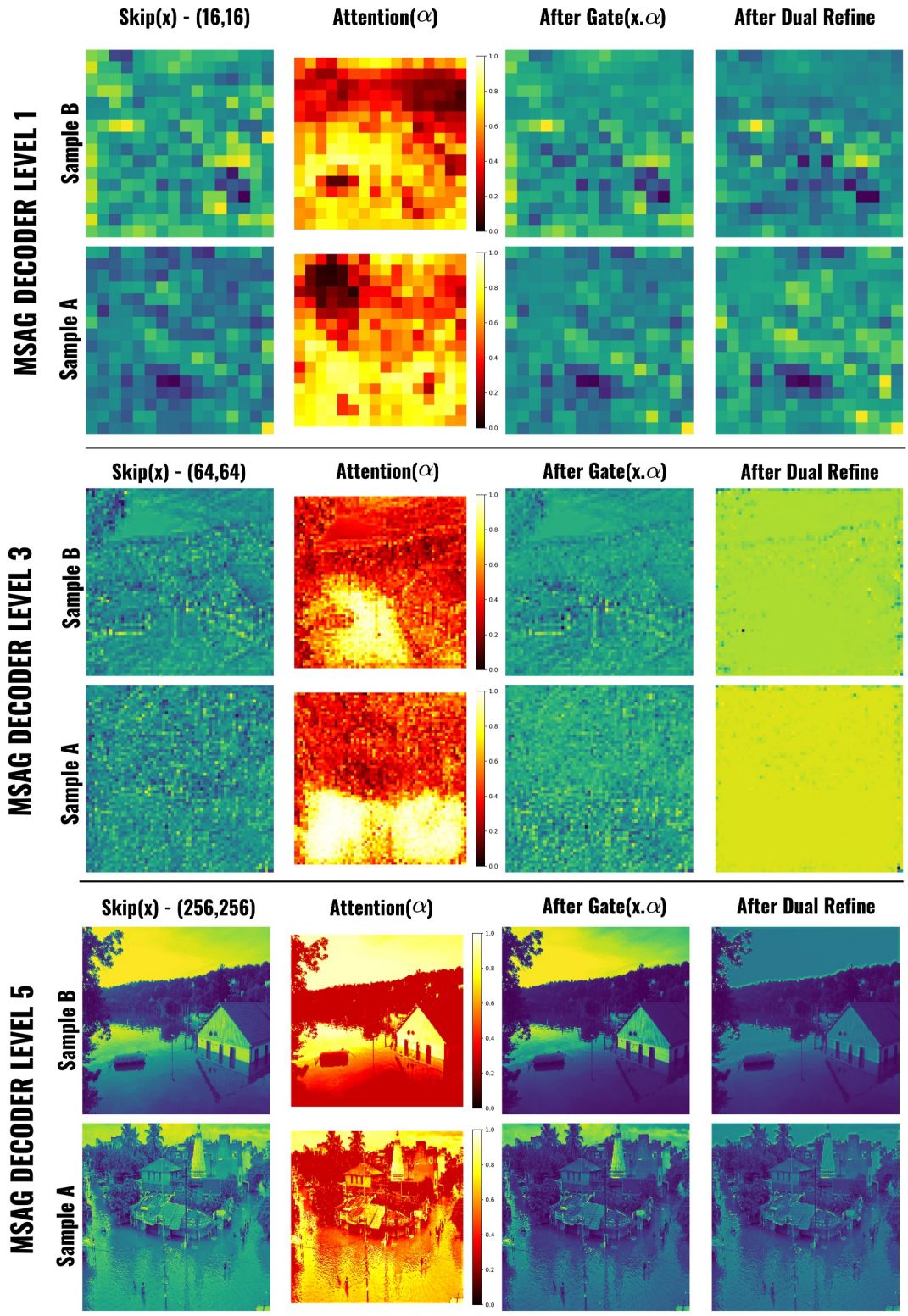


FIGURE 11 | Different levels of decoder for flood segmentation.

$$\mathcal{L}_{\text{dice}} = 1 - \frac{2\sum_{i=1}^{N} p_i y_i + \epsilon}{\sum_{i=1}^{N} p_i + \sum_{i=1}^{N} y_i + \epsilon}, \tag{31}$$

where $\epsilon = 1$ Is a smoothing term to avoid division by zero.

The focal loss encourages the model to focus on hard-to-classify pixels (e.g., ambiguous water boundaries), while the Dice loss ensures good spatial overlap between predictions and ground truth.

The algorithmic steps of the proposed model are presented in Algorithm 1, which summarises all the steps of the proposed method in order.

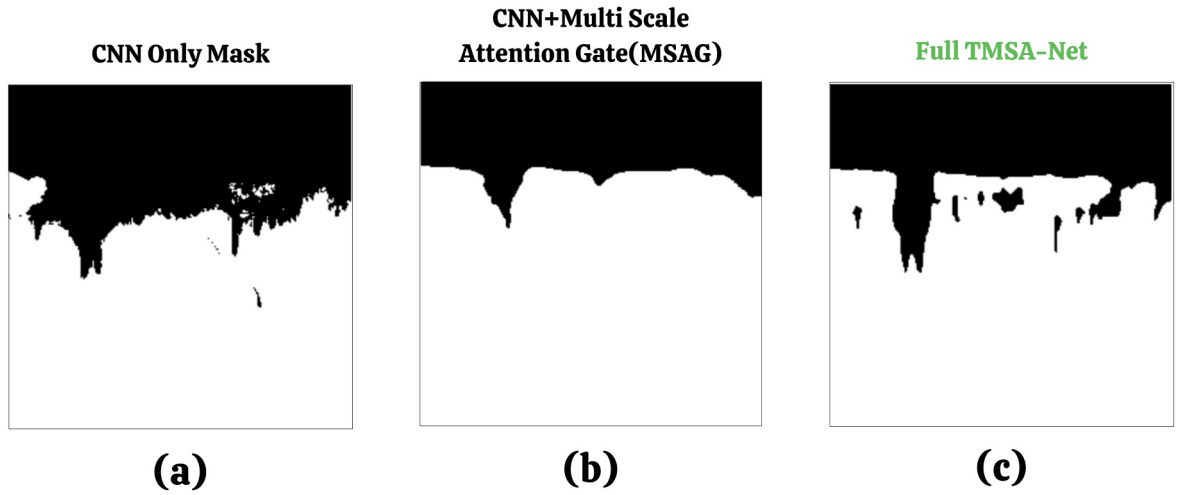


FIGURE 12 | Visual analysis of the MSAG contribution on the same complex flood scene shown in Figure 9a. (a) CNN-only prediction, which is the same as the results in Figure 9c. (b) CNN + MSAG prediction (without Transformer): the multi-scale gates with dual channel-spatial attention substantially improve segmentation by filtering skip connections at three receptive fields (1×1, 3×3, 5×5), producing cleaner boundaries and recovering water regions missed in (a). (c) Full TMSA-Net prediction: adding the Transformer bottleneck to MSAG further refines the segmentation, correctly connecting distant water patches and achieving the most complete flood mask with precise boundary delineation around occluding structures.

ALGORITHM 1 | TMSA-Net Flood Segmentation Pipeline.

Input: RGB image I ($256 \times 256 \times 3$)
Output: Binary flood mask M ($256 \times 256 \times 1$)

Stage 1: Hierarchical Feature Extraction (Section 3.1)
Extract multi-scale features using EfficientNet-B4 encoder:
$F_{\text{enc}} = (F_0, F_1, F_2, F_3, F_4, F_5) \leftarrow \text{EfficientNet-B4}(I)$

Stage 2: Transformer Bottleneck (Section 3.2)
Flatten deepest features into a token sequence:
$Z \leftarrow \text{Flatten}(F_5) \in \mathbb{R}^{B \times 64 \times C_5}$
Add learnable positional encoding:
$Z \leftarrow Z + P_{\text{pos}}$
Apply L layers of multi-head self-attention:
for $l = 1$ **to** L **do**
$Z' \leftarrow \text{MHSA}(\text{LN}(Z)) + Z$
$Z \leftarrow \text{MLP}(\text{LN}(Z')) + Z'$
end for
Reshape back to spatial feature map:
$F_5^{\text{T}} \leftarrow \text{Reshape}(\text{LN}(Z))\ \mathbb{R}^{B \times C_5 \times 8 \times 8}$

Stage 3: Decoder with Multi-Scale Attention Gates (Section 3.3)
for $i = 5$ **to** 1 **do**
// Upsample and project decoder features
$g \leftarrow \text{ReLU}(\text{BN}(\text{Conv}_{1\times1}\ (\text{Upsample}(d_{\{i-1\}}))))$
// Multi-scale gating with three parallel kernels
$g_k \leftarrow \text{BN}(\text{Conv}_{\{k\times k\}}(g))$ for $k \in \{1, 3, 5\}$
// Fuse multi-scale responses
$g_{\text{fused}} \leftarrow \text{ReLU}(\text{BN}(\text{Conv}_{1\times1}\ (\text{Concat}(g_1, g_3, g_5))))$
// Generate attention coefficients
$\alpha \leftarrow \text{Sigmoid}(\text{BN}(\text{Conv}_{1\times1}(\text{ReLU}(g_{\text{fused}} + W_x(x)))))$
// Apply gating and dual attention refinement
(Continues)

(Continued)
$x_{\text{gated}} \leftarrow \alpha \odot x$
$x_{\text{out}} \leftarrow \text{SpatialAtt}(\text{ChannelAtt}(x_{\text{gated}}))$
// Decode
$d_i \leftarrow \text{Conv}_{3\times3}(\text{Conv}_{3\times3}\ (\text{Concat}(g, x_{\text{out}})))$
end for

Stage 4: Segmentation Head (Section 3.4)
$\boldsymbol{M}_{logits} \leftarrow \text{Conv}_{1\times1}(d_5)$
$M \leftarrow \mathbb{1}[\sigma(\boldsymbol{M}_{logits}) > 0.5]$

return M

To demonstrate each component's contribution, Figure 13 presents t-SNE plots for visualisations of feature representations at four stages of TMSA-Net. In (a), using only the EfficientNet-B4 backbone, flood pixels (red) and non-flood pixels (blue) are scattered and mixed everywhere. This means the backbone alone is inadequate for detecting the flood region in the images. In (b), after adding the Transformer bottleneck, the points start grouping into small clusters because the model now understands relationships between different parts of the image. However, red and blue are still mixed. In (c), the Multi-Scale Attention Gates help the model focus on important features, and we can see more organised patterns forming. The biggest improvement is in (d), where the full TMSA-Net creates two clearly separate groups: blue points cluster on one side and red points on the other. This clear separation shows that combining all three components, Transformer, multi-scale processing, and attention, allows the model to effectively distinguish flood from non-flood regions.

4 | Experimental Results

We present experimental results and analysis for evaluating the proposed method. Since there is a limited number of standard

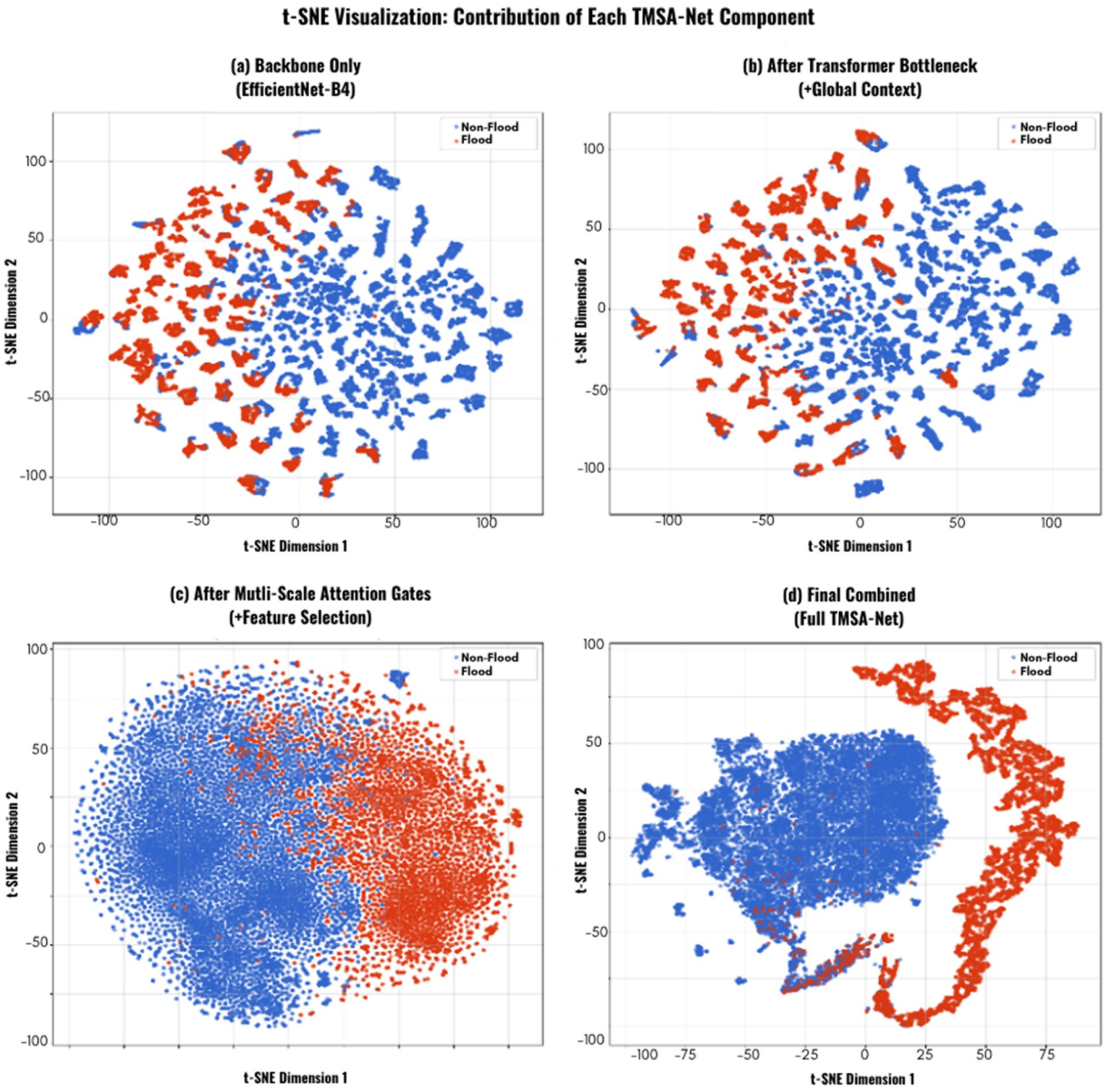


FIGURE 13 | t-SNE visualisation of feature representations at each TMSA-Net stage: (a) EfficientNet-B4 backbone only shows mixed flood/non-flood distributions, (b) Transformer bottleneck begins clustering similar features, (c) Multi-Scale Attention Gates add structure, and (d) complete TMSA-Net achieves clear class separation. The progressive improvement validates each component's contribution to discriminative feature learning.

datasets suitable for ground-level flood image segmentation, we created our own dataset for experimentation.

4.1 | Dataset and Evaluation

Existing flood segmentation datasets primarily target satellite and SAR imagery, such as Sen1Floods11 [3] and WorldFloods, or UAV-based aerial imagery, such as FloodNet [49] and RescueNet [50]. Although these datasets provide valuable pixel-level annotations, they do not include samples captured by CCTV or smartphones, resulting in mixed viewpoints. These challenges are common for real-time street-level disaster response. Therefore, we collected relevant samples from the Flooding Image Dataset available on Kaggle [51], which provides 320 samples with the possible variations as shown in Figure 14. Furthermore, we manually created binary segmentation masks for annotating the flood regions at pixel level. The selected subset exhibits challenging characteristics representative of real-world ground-level flood imagery: extreme viewpoint variation from street-level to drone altitudes, dramatic scale variation, variable image quality, and frequent occlusion by buildings, vegetation, and infrastructure.

Since it is a new challenge and the focus of work is to address the challenge, this study aims at problem-solving, not at a new complete benchmark dataset creation. Thus, the number of samples collected is limited. However, the collected samples include diversified images with large variations in the background and image quality, as shown in Figure 14. But, for training, we perform different augmentation techniques, such as

geometric transforms, colour jittering, noise injection, advanced mixing-based techniques like MixUp ($\alpha = 0.2$) and CutMix ($\alpha = 1.0$) to increase the number of samples. This gives a total of 13,000 samples for training. For all the experiments, the work uses a five-fold cross-validation on 85% of the data, with a 15% held-out test set. The same setup has been used for all the experiments conducted by the proposed and existing methods.

To further validate generalisability, a second benchmark dataset of 660 aerial RGB flood images with binary segmentation masks, publicly available on Kaggle [52]. Unlike our dataset, which captures ground-level perspectives with complex visual ambiguities such as shadows, wet asphalt, and reflections, the benchmark dataset mostly provides top-view imagery. Aerial flood images typically exhibit clearer water-land boundaries but introduce different challenges, including varying water colour due to sediment, partially submerged vegetation, and scale variations across flight altitudes. We apply the same experimental protocol to this dataset, using identical augmentation strategies, loss functions, and hyperparameters. For measuring the performance of the proposed method, we use the following standard measures as defined in Equations (32–36).

$$\text{Intersection over Union (IoU)} = \frac{TP}{TP + FP + FN}, \tag{32}$$

$$\text{Dice} = \frac{2 \cdot TP}{2 \cdot TP + FP + FN}, \tag{33}$$

$$\text{Precision} = \frac{TP}{TP + FP}, \tag{34}$$

$$\text{Recall (Sensitivity)} = \frac{TP}{TP + FN}, \tag{35}$$

$$\text{F-1 Score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}, \tag{36}$$

where TP (True Positives) are correctly predicted flood pixels, FP (False Positives) are non-flood pixels incorrectly predicted as flood, FN (False Negatives) are flood pixels incorrectly predicted as non-flood, and TN (True Negatives) are correctly predicted non-flood pixels. IoU and Dice coefficient measure spatial overlap between predictions and ground truth. Precision measures the proportion of predicted flood pixels that are correct, while recall measures the proportion of actual flood pixels detected. We report all metrics as percentages averaged across the test set. In addition, we generate a confusion matrix with TP, TN, FP, and FN counts to test the performance of the proposed model on each class.

Implementation details: Comprehensive augmentation pipeline including geometric (flips, rotations, elastic transformations), photometric (brightness, contrast, hue shifts), and mixing-based techniques (MixUp with $\alpha = 0.2$, CutMix with $0\ \alpha = 1.$) applied with probability 0.33 each. Optimiser: Adam ($\beta_1 = 0.9$, $\beta_2 = 0.999$), Learning rate: 10^{-4} with ReduceLROnPlateau (patience = 5, factor = 0.5), Batch size: 8, Maximum epochs: 50, Early stopping: patience = 10 based on validation IoU.

4.2 | Pixel-Level Classification Performance

The pixel-level classification performance is presented in Figure 15. It illustrates a normalised confusion matrix showing per-class classification rates. The proposed TMSA-Net achieves **94.07%** accuracy for flood pixels and **97.31%** accuracy for non-flood pixels, yielding a mean diagonal accuracy of **95.69%**. The low false positive rate (**2.69%**) indicates the model rarely misclassifies non-flood regions as flooded, which is critical for avoiding false alarms in disaster response systems. Similarly, the false negative rate of **5.93%** demonstrates effective flood detection with minimal missed water regions.



FIGURE 15 | Confusion matrix for TMSA-Net.



FIGURE 14 | Representative samples from flood dataset [51] demonstrating diversity across acquisition platforms, scales, and environmental conditions.

To validate the performance of the proposed method in terms of efficiency and accuracy for flood region segmentation, Figure 16 presents the training and validation loss convergence. The key observations emerge from the training dynamics. First, validation IoU reaches **70%** within 10 epochs, indicating effective knowledge transfer from ImageNet pretrained encoders. Second, validation metrics stabilise after epoch **40** without degradation, suggesting the combined **BCE + Dice** loss and early stopping effectively prevent overfitting. Third, validation curves consistently outperform training curves (**88%** vs. **70%** IoU at plateau), a counterintuitive but desirable pattern indicating strong regularisation from data augmentation, dropout, and batch normalisation.

4.3 | Comparative Study With the State-of-the-Art Methods

To demonstrate the effectiveness of the proposed model, we compare it with state-of-the-art segmentation methods: DeepLabV3+ (multi-scale ASPP with ResNet-101), TransUNet (hybrid CNN-Transformer with ResNet-50), UNet++ (nested skip connections with VGG16), and Attention U-Net (spatial and channel attention with ResNet-34).

For experimentation, the work uses five-fold cross-validation and considers the best results as the results of the methods out of five. The performance of the proposed and existing methods on each fold is reported in Table 3, where it is noted that the best results are obtained for the 4th and 5th folds of the experiments compared to the existing models. When we consider the mean of all five-fold experiments, the SwinUNet is the best compared to the proposed and other models. Similarly, the standard deviation is slightly higher for the proposed method compared to the existing methods. This is because of heterogeneity in our dataset. Since the work considers the best fold results as the actual performance of the method out of fivefold results, the mean and standard deviation do not affect the final comparative accuracy. The key reason for the low mean and high standard deviation of the proposed method is that the method focuses on flood regions, while the existing method focuses on non-flood regions. The contextual features between flood and non-flood regions are not stable in the case of the proposed method. The existing methods are good in terms of

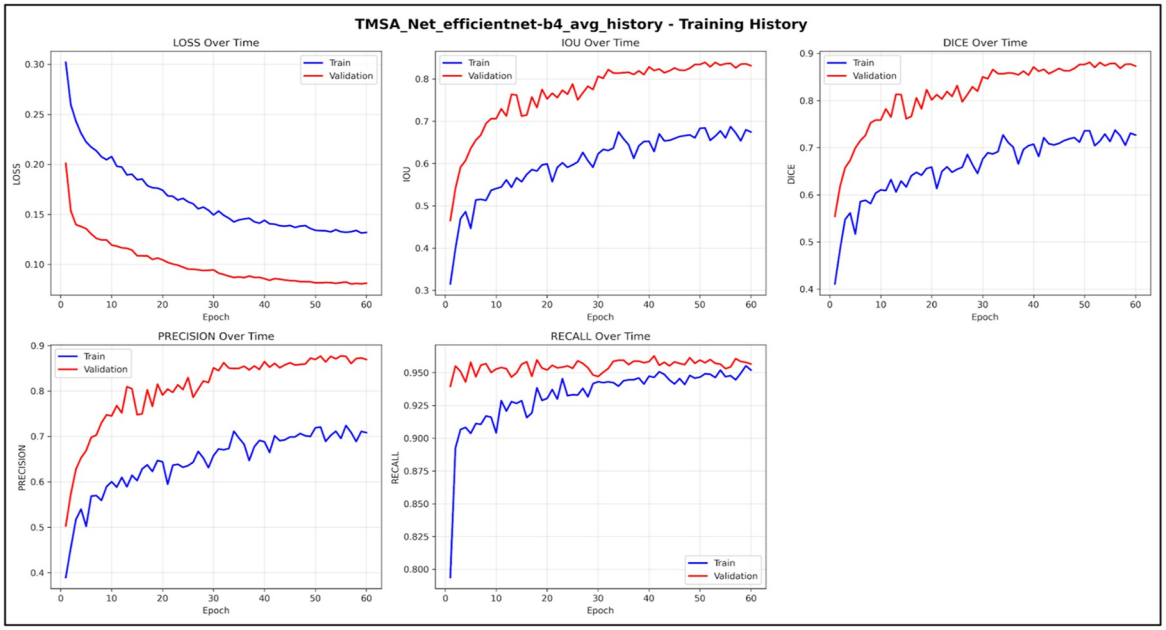


FIGURE 16 | Average training dynamics across all 5 folds for TMSA-Net.

TABLE 3 | Details of five-fold cross-validation results (IoU%) for the proposed and existing models.

	Our dataset							Benchmark dataset [52]						
Model	**Fold 1**	**Fold 2**	**Fold 3**	**Fold 4**	**Fold 5**	**Mean**	**Std**	**Fold 1**	**Fold 2**	**Fold 3**	**Fold 4**	**Fold 5**	**Mean**	**Std**
TMSA-Net (ours)	83.41	89.73	87.86	**91.85**	**93.66**	89.30	3.53	**93.70**	**93.50**	**93.18**	**92.20**	**92.98**	**93.13**	0.54
Swin-UNet [41]	**88.68**	**90.73**	**92.14**	91.50	93.18	**91.25**	1.51	92.11	91.94	92.65	92.08	92.21	92.20	**0.24**
DeepLabV3+ [30]	87.89	89.34	87.35	88.61	90.53	88.74	**1.12**	90.48	90.59	90.65	89.20	89.67	90.12	0.58
TransUNet [40]	84.08	90.10	90.24	87.04	89.60	88.21	2.37	91.97	90.57	92.57	91.05	91.47	91.53	0.70
UNet++ [25]	83.04	87.09	88.49	86.30	89.34	86.85	2.18	92.65	92.07	92.69	91.54	91.56	92.10	0.50
Attention U-Net [11]	86.68	84.30	86.80	87.02	90.09	86.98	1.84	91.94	90.62	91.46	90.48	89.53	90.81	0.83

Note: The bold indicates the best results.

mean and standard deviation compared to the proposed model on our dataset, but at the cost of accuracy. However, for the benchmark dataset, the proposed method reports the best results for all the fold experiments compared to the existing models. But the standard deviation of the proposed method for the benchmark dataset is higher than that of the SwinUNet. Therefore, the proposed method is second best in terms of standard deviation, but it is the best in terms of accuracy and mean compared to the existing models. The reason for the best performance of the proposed method on the benchmark dataset is that images are almost similar except for the flood regions.

The test results of the proposed and existing models on our dataset and the benchmark datasets are recorded in Table 4, where, for all the measures, the proposed method is the best compared to the existing models on both datasets. This shows that the proposed method accurately segments flood regions while maintaining high precision, indicating minimal false positives for diversified datasets. The consistent improvement across all metrics demonstrates the effectiveness of combining the Transformer and Multi-Scale Attention U-Net for flood region segmentation. However, it is noted that the difference is marginal. When we consider overall performance, the proposed model is better than existing models. The reason for achieving the best results is the combination of the Transformer and Mult-Scale Attention U-Net for flood region segmentation. On the other hand, the existing methods are not robust for complex images because of ineffective feature extraction. In addition, most of the existing models do not use multi-scale attention for extracting the local information to address the challenges of an unstructured flood region with the presence of arbitrarily shaped debris.

The same conclusions can be drawn from the qualitative results shown in Figure 17, where one can see successful flood segmentation for all the samples with different backgrounds. This shows that the proposed method is capable of handling complex situations. Figure 17 shows that the result of the proposed model is close to the ground truth, while the results of existing models are not so close to the ground truth. Therefore, the above inferences make sense. The reason for the poor results of the existing models is the same as the quantitative results.

To further demonstrate the improved performance of the proposed model over the existing models, qualitative results for the proposed and existing models on complex flood scene images are shown in Figure 18. It can be seen that the proposed model segments the flood regions in all the samples accurately, which is very close to the ground truth. But the existing methods do not perform well for flood region segmentation. This is because none of the methods focused on flood region segmentation but focused on classification and detection using image-level features. Therefore, the existing methods are unlikely to be effective for flood region segmentation, especially for complex background images and images with debris. Figure 18a shows a daytime urban flood with heavy occlusion and reflective camouflage, Figure 18b shows nighttime urban flood with intense artificial lighting and Figure 18c shows rural flood scene with water surface colour varying from light (sky reflections in foreground) to dark (vegetation reflections in background), and two people on a bamboo raft splitting the water region.

For image (a) in Figure 18, the models, DeepLabV3+, UNet++, Attention UNet, and Swin-UNet miss water patches where reflections alter the surface appearance. For image (b), all the existing models miss large water regions between reflective highlights. For image (c), TransUNet, Attention UNet and Swin-UNet fail on the darker flooded area. The key reason for the poor segmentation of the existing models is that the features are not robust enough for separating the flood region from non-flood regions. In addition, the models focus on visual features but not on contextual features between flood and non-flood regions. However, the proposed model extracts features based on context information and hence, the proposed method works well for complex images.

4.4 | Cross-Dataset Validation

To demonstrate the robustness and the generalisation ability of the proposed TMSA-Net method, we conducted cross-dataset validation for all the models and the results are reported in Table 5. It is observed that the proposed method achieves the best results for both experiments in terms of almost all the measures. However, the TransUNet achieves the best precision compared to all the methods for the benchmark dataset. But the TransUNet fails to achieve the best results in terms of all other measures compared to the proposed method. It is also noted from the Table 5 that performance in terms of F1-score is high when the methods are trained on our dataset and tested on the benchmark dataset. This makes sense because the samples of

TABLE 4 | Performance of the proposed and existing methods on our dataset and the benchmark dataset.

	Our dataset					Benchmark dataset [52]				
Model	**IoU (%)**	**Dice (%)**	**Precision (%)**	**Recall (%)**	**F1 (%)**	**IoU (%)**	**Dice (%)**	**Precision (%)**	**Recall (%)**	**F1 (%)**
TMSA-Net (ours)	**92.01**	**95.84**	95.67	**96.01**	**95.84**	**94.75**	**97.30**	**97.63**	**96.98**	**96.96**
Swin-UNet [41]	90.72	95.13	**96.53**	93.78	95.13	93.87	96.84	97.32	96.35	96.84
DeepLabV3+ [30]	89.19	94.29	94.26	94.31	94.29	92.92	96.33	96.15	96.51	96.33
UNet++ [25]	87.35	93.25	94.52	92.01	93.25	93.74	96.77	97.16	96.38	96.77
Attention U-Net [11]	86.30	92.64	93.21	92.09	92.64	93.60	96.70	96.94	96.45	96.70
TransUNet [40]	86.06	92.51	90.81	94.26	92.51	93.66	96.72	96.94	96.51	96.72

Note: The bold indicates the best results.



FIGURE 17 | Qualitative comparison of TMSA-Net against state-of-the-art models on three challenging flood scenarios. Each scenario shows the original image, predicted binary masks from all models, and ground truth. Overlay visualizations display predictions (red) superimposed on original images, with ground truth shown in green.

our dataset are more complex compared to the samples of the benchmark dataset. Therefore, the experiments show that the proposed model is robust and generic.

It is noted from Table 5 that although the TransNet and Swin-Net use transformer layers, these models are widely used for medical imaging, but not for flood region segmentation. Therefore, the models may not work well for flood region segmentation without modifications. It is evident from the results reported in Table 5, where the performance of the baseline models is inferior to the proposed model on our and benchmark datasets. The Swin-UNet achieves the highest precision compared to the proposed model and other models on our dataset. But it is worse for the benchmark dataset compared to the proposed model. This is because the Swin-UNet is good for extracting visual features that represent non-flood regions in the flood images. Therefore, the difference between flood and non-flood regions increases, resulting in low false positives. However, the Swin—UNet does not achieve the best results for other parameters compared to the proposed method. This indicates that visual features alone are not adequate to segment the flood region, and hence, it misses flood regions. The reason for the best performance of the proposed model is that use of Transformer bottleneck with Multi-Scale Attention Gates for extracting features. This extracts the necessary global context to distinguish water from visually similar terrain (like shadows) with minimum expense.

To validate the efficiency of the proposed model, the number of parameters and GFLOPs are calculated on our dataset and the results are reported in Table 2, where the proposed model is efficient in terms of the number of parameters and GFLPs compared to other models, including existing models. The existing models are heavy and use an expensive encoder compared to the proposed EfficientNet-B4. Therefore, the methods are computationally expensive.

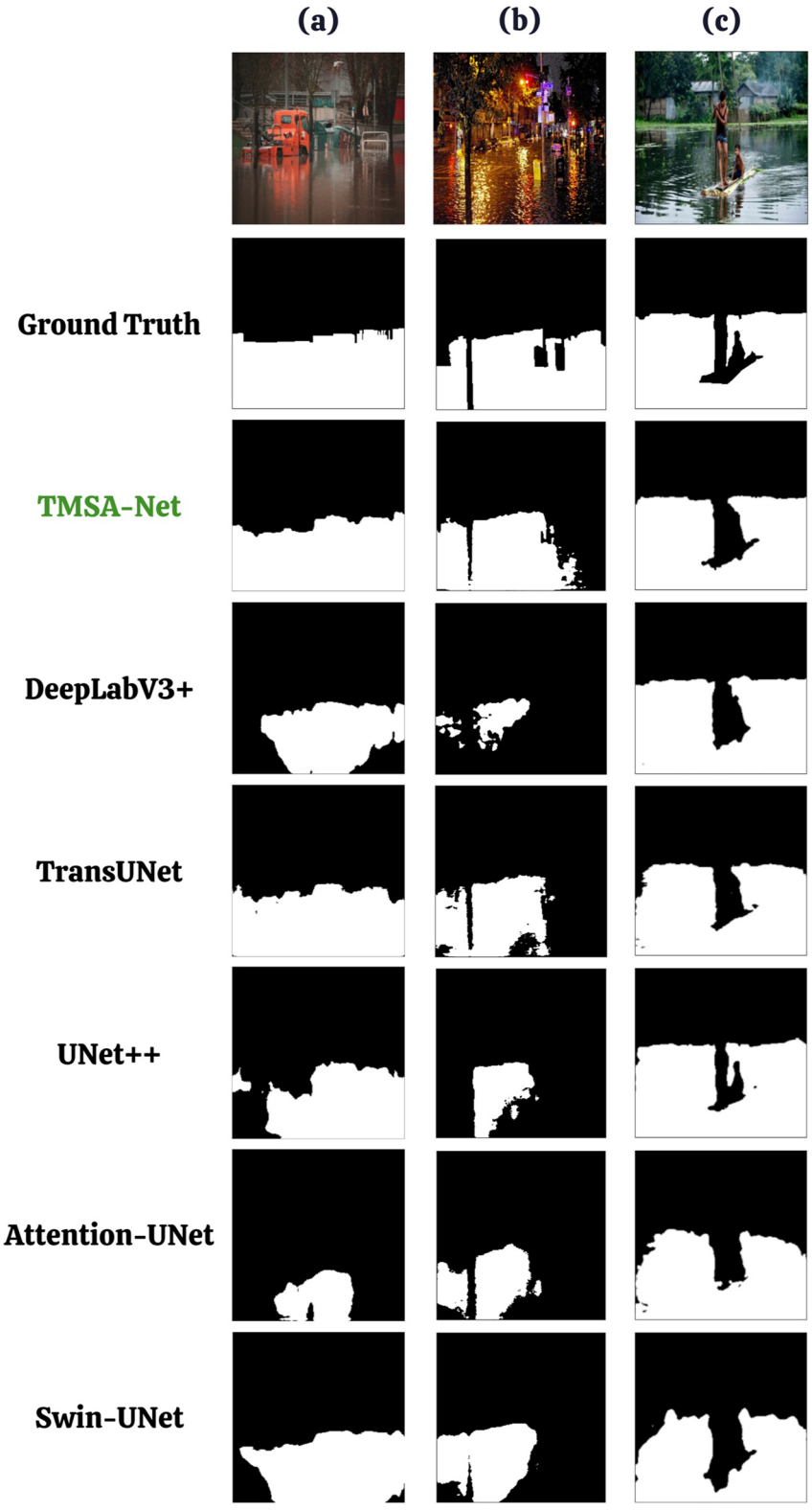


FIGURE 18 | Qualitative comparison of TMSA-Net against state-of-the-art methods on challenging flood scenes. (a) Daytime urban flood with heavy occlusion and reflective camouflage. (b) Nighttime urban flood with intense artificial lighting. Specular reflections from traffic lights and streetlamps on the water surface closely resemble the light sources themselves, creating ambiguity between illuminated non-flood regions and reflective floodwater. (c) Rural flood scene with water surface colour varying from light (sky reflections in foreground) to dark (vegetation reflections in background), and two people on a bamboo raft splitting the water region.

4.5 | Ablation Studies

In the proposed work, we use EfficientNet-B4 as the encoder for feature extraction. To show that this is better than other baseline networks, the experiments are conducted on our and benchmark datasets for a comparative study with the RestNet34. The results are reported in Table 6, where the EfficientNet-B4 is better than the RestNet34 in terms of IoU for both datasets. Therefore, the proposed work prefers to use the EfficientNet-B4 as a backbone for flood image segmentation. This is because EfficientNet is computationally inexpensive and extracts robust features that represent boundaries in the low contrast images, whilst the ResNet is expensive, and the features depend on a large number of parameters.

Similarly, the proposed model is integrated with the transformer bottleneck network, multi-scale attention gates of U-Nets for flood region segmentation. To show the effectiveness of each key step of the proposed model, we calculated measures for baseline U-Net, Baseline Attention, Baseline Transformer and the proposed model on both our and standard datasets. The results of the experiments are recorded in Table 7, where the baseline models report lower results compared to the proposed model. At the same time, the results of all the measures indicate that the baseline models are capable of flood region segmentation because the baseline models contribute equally to achieving the best results by the proposed model. Table 7 shows that all the baseline models and the proposed model are the best at recall compared to other measures. This implies that the models are good for flood region segmentation but not for detecting non-flood regions accurately. This indicates that the models output more false positives and hence poor precision. Therefore, one can infer that the samples are complex due to cluttered backgrounds and diversity in the samples.

Interestingly, when we add standard attention to the baseline model, the IoU, Dice, Precision and F1-score drop significantly compared to the baseline model performance. This is because the standard attention, which uses a 1×1 single convolution to generate a gate signal, does not extract sufficient contextual information [11]. Therefore, non-flood regions are misclassified

TABLE 5 | Cross-dataset validation—train on our dataset (320) and test it on the benchmark dataset (600) (in %).

	Train on our dataset and test it on the benchmark dataset					Train on the benchmark dataset and test it on our dataset				
Model	**IoU**	**Dice**	**Precision**	**Recall**	**F1**	**IoU**	**Dice**	**Precision**	**Recall**	**F1**
TMSA-Net	**81.45**	**89.78**	93.60	**86.25**	**89.78**	**77.58**	**87.37**	82.46	**92.91**	**87.37**
UNet++	80.66	89.30	94.22	84.87	89.30	73.30	84.60	81.82	87.56	84.60
TransUNet	79.83	88.79	93.32	84.67	88.79	75.14	85.80	**83.52**	88.21	85.80
Swin-UNet	78.94	88.23	**97.17**	80.79	88.23	76.38	86.61	83.10	90.42	86.61
DeepLabV3+	76.17	86.47	93.72	80.27	86.47	74.28	85.24	83.49	87.07	85.24

Note: The bold indicates the best results.

TABLE 6 | Assessing the contribution of different encoders for flood region segmentation (in %).

		Our dataset					Benchmark dataset				
Model	**Encoder**	**IoU (test)**	**Dice**	**Precision**	**Recall**	**F1**	**IoU (test)**	**Dice**	**Precision**	**Recall**	**F1**
TMSA-Net	EfficientNet-B4	**92.01**	**95.84**	**95.67**	**96.01**	**95.84**	**94.75**	**97.30**	**97.63**	**96.98**	**96.96**
TMSA-Net	ResNet34	87.00	89.98	92.50	94.29	93.34	89.33	94.09	93.75	95.20	94.47

Note: The bold indicates the best results.

TABLE 7 | Ablation study experiments for the proposed and existing methods on our and the benchmark datasets.

	Our dataset					Benchmark dataset				
Model variant	**IoU (%)**	**Dice (%)**	**Precision (%)**	**Recall (%)**	**F1 (%)**	**IoU (%)**	**Dice (%)**	**Precision (%)**	**Recall (%)**	**F1 (%)**
Baseline (standard U-Net)	81.29	84.08	84.33	96.61	84.08	90.70	94.68	94.97	94.85	94.68
Baseline + Std. Attention	73.04	75.80	75.16	**97.55**	75.80	91.17	95.05	94.78	95.83	95.05
Baseline + Transformer bottleneck	81.49	84.91	85.66	95.46	84.91	91.40	95.28	95.35	95.61	95.28
Baseline + multi-scale gates	85.03	91.91	93.70	90.18	91.91	92.35	96.02	95.91	96.14	96.02
Baseline + multi-scale + channel Att.	86.35	92.67	90.86	94.56	92.67	92.30	95.99	96.36	95.63	95.99
Baseline + full MSAG (MS + channel + spatial)	84.89	91.83	93.40	90.31	91.83	92.05	95.86	95.34	96.38	95.86
TMSA-Net (full architecture)	**92.01**	**95.84**	**95.67**	96.01	**95.84**	**94.75**	**97.30**	**97.63**	**96.98**	**96.96**

Note: The bold indicates the best results.

as flood regions. Hence, IoU, Dice and precision are poor. However, the recall is higher than the baseline model's performance instead of dropping (97.55% vs. 96.61%). This indicates that the attention gate is not suppressing genuine flood pixels but is instead failing to reject visually similar non-flood surfaces such as wet asphalt and shadows, resulting in a sharp increase in false positives. The precision-recall gap of 22.39% (75.16% precision vs. 97.55% recall) confirms this over-prediction behaviour. It is also evident from the results on the benchmark dataset, which do not provide images with a complex background. In this case, a single convolution extracts useful information, rather than confusing the background information with the flood regions and the standard attention variant improves slightly over the baseline (91.17% vs. 90.70%).

Further, it is observed that adding multi-scale gates recovers the performance substantially (85.03% IoU), confirming that processing features at multiple receptive fields resolves the textural ambiguity that single-scale gating cannot handle. Adding channel attention improves precision further (90.86%), while adding spatial attention alongside channel attention improves recall (94.56% vs. 90.18% for multi-scale gates alone). This indicates that channel and spatial attentions complement each other: channel attention helps the model prioritise flood-relevant feature channels, while spatial attention refines the spatial extent of detected regions. The proposed method (TMSA-Net), which integrates the Transformer bottleneck with the full MSAG, achieves the highest IoU (92.01%) with both high precision (95.67%) and high recall (96.01%). Although the recall is not the highest among all variants, TMSA-Net achieves the smallest precision-recall gap of just 0.34% points, compared to 12.28 for the baseline, 3.70 for multi-scale with channel attention, and 22.39 for standard attention. This balance between precision and recall is desirable for practical deployment, where both over-prediction and under-prediction carry risks. Overall, one can conclude that each component contributes to achieving the best results for both our and standard datasets.

To assess the contribution of the transformer bottleneck, the measures are calculated for our and the benchmark datasets and the results are reported in Table 7. The baseline + Transformer bottleneck achieves 81.49% and 91.40% of IoU for our and benchmark datasets, respectively. Compared to the proposed model, these results are lower, but the improvement over the baseline (81.29% and 90.70%) confirms that global context modelling contributes to the overall architecture. This shows that the transformer bottleneck step is effective and contributes well to achieving the best results.

In this work, the spatial and channel attention modules contribute equally to achieving the best results for flood region segmentation, as can be seen from Table 7. In addition, we also present qualitative results to show the effectiveness of both spatial and channel attention. The refinement model suppresses the non-flood regions but not the flood regions, as illustrated in the complex image of Figure 19, where the suppressed pixels are marked for different weight parameters, α of the attention gates at every decoder level. The levels are determined experimentally in this work.

4.6 | Discussion

When we compare the proposed model with conventional CNNs for flood region segmentation in scene images, standard CNNs struggle in complex flood scenes due to fixed, localised convolution kernels (limited receptive fields) and rigidity when objects drastically change size due to camera perspective (scale variation). Standard CNNs lose global context because their kernels only "see" a small local neighbourhood of pixels. To fix this, the proposed TMSA-Net integrates a Transformer bottleneck at the deepest layer of the encoder. Unlike convolutions, the Transformer utilises global self-attention. This allows the network to mathematically link distant pixels across the entire image. For example, connecting a patch of water in the bottom-left corner directly to a patch of water in the top-right corner,

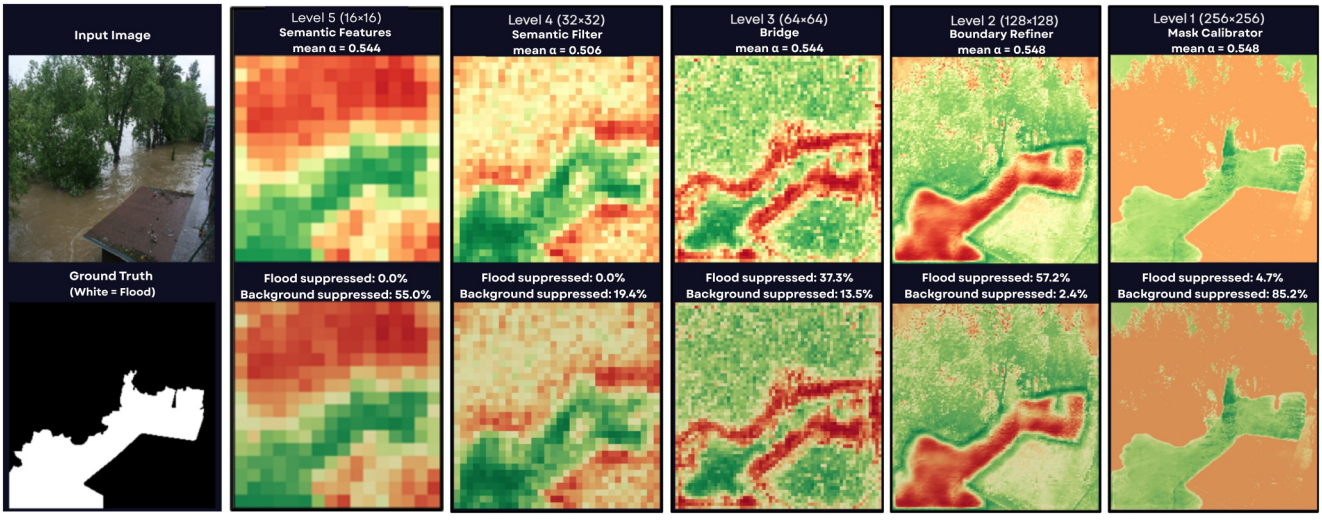


FIGURE 19 | Attention gate weights α at each decoder level for the image. Top row: spatial gate weight maps using the RdYlGn colourmap, where red indicates gate closed (feature suppressed) and green indicates gate open (feature preserved), with mean α value shown above each map. Bottom row: same gate maps with quantitative labels showing the percentage of flood pixels and background pixels with gate weight below 0.3. Ground truth mask (white = flood) is shown bottom-left for spatial reference.

bypassing the rigid spatial constraints of a CNN and ensuring the model understands the global topology of the flood. In ground-level images, flood scales vary wildly; water close to the lens appears massive, while distant flood zones appear tiny. Standard U-Nets use fixed-size skip connections that fail to capture this variance. TMSA-Net addresses this by replacing standard skip connections with MSAG modules. By running parallel convolutional kernels of varying sizes (1×1, 3×3, 5×5) inside the attention gate, the network dynamically extracts multi-scale features simultaneously. The smaller kernels capture the fine boundaries of distant puddles and thin debris, while the larger kernels capture the broad, continuous textures of massive water bodies nearby. By fusing global Transformer attention with multi-scale local gating, TMSA-Net effectively breaks the constraints of traditional CNNs. The proposed model does not limit receptive fields, unlike the conventional CNN, which has limited receptive fields. It is evident from the ablation experiments because the proposed method achieves the best results compared to the baseline model and even existing models.

4.7 | Limitations

Even though TMSA-Net exhibits superior performance over the state-of-the-art, there is a scope for improvement. First, while the base dataset is adequate for validation and comparative evaluation, it should be further augmented and additional fine-tuning experiments carried out before the method can be deployed in the real world. Cross-validation variance was low ($\pm$ 5.76%); however, generalisation to highly novel scenarios may be constrained. Second, testing on different geographic domains or conditions (nighttime imagery, snow melt, different architectural styles) requires future validation. Third, parallel 1, 3×3, and 5×5 pathways increase memory usage compared to the standard U-Net, presenting challenges for limited GPU memory. Fourth, the margin over DeepLabV3+ (+2.82% IoU) represents a meaningful improvement, though DeepLabV3+ may still be considered where simplicity is prioritised. Finally, real-time video processing characteristics require additional investigation for real-world streaming applications.

In addition, the proposed method is trained and tested on binary segmentation (flood vs. non-flood), and it does not distinguish between different types of water-related regions, such as standing water, flowing water, or water bodies like rivers and lakes. This may limit the usefulness of the method for applications that require detailed flood classification. The number of samples in different weather conditions, such as snow, fog, and heavy rain, is limited, and the model needs to be validated under such conditions. Further, to improve the performance of the flood region segmentation, alternative gate designs, such as deformable attention gates with learnable kernel offsets, or non-local attention mechanisms operating across the full spatial extent, could be used. However, this is beyond the scope of the work.

To overcome the above limitations, we plan to bring multi-modal language models and diffusion models to differentiate the flood regions from the non-flood regions. The language model can help us to extract the high-level features of the whole image to extract global information, while the diffusion model can help us to extract the rich local information. If the model combines the global and local features properly, one can extract robust and effective features irrespective of background variations and the quality of the images.

5 | Conclusion and Future Work

In this work, we proposed a new model, the Transformer-based Multi-Scale Attention U-Net (TMSA-Net), for flood region segmentation. In contrast to existing models that primarily target satellite and remote sensing data, TMSA-Net addresses the unique challenges of flood region segmentation in ground-level images. The key challenges are limited receptive fields of CNNs, indiscriminate feature transfer in skip connections, and dramatic scale variation. To overcome these, we combined a Transformer bottleneck for global context, Multi-Scale Attention Gates for decoder-guided feature selection, and Dual Attention Refinement to suppress background noise. This is the first work to integrate a modified CNN, a Transformer, and a U-Net for flood region segmentation. The proposed model achieves superior performance compared with the state of the art across all standard evaluation metrics. Future directions include exploring the use of multi-modal language models and diffusion models for feature extraction. Furthermore, the proposed idea can be extended to be deployed in a real-time environment. Finally, we plan to extend the dataset with more diverse samples to improve the generalisation ability of the proposed model.

Funding

The authors have nothing to report.

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

The dataset and ground truth will be released to the public after acceptance to support reproducibility.

References

1. A. Domeneghetti, "On the Use of SRTM and Altimetry Data for Flood Modeling in Data-Sparse Regions," *Water Resources Research* 52, no. 4 (2016): 2901–2918, https://doi.org/10.1002/2015WR017967.

2. Y. Hirabayashi, R. Mahendran, S. Koirala, et al., "Global Flood Risk Under Climate Change," *Nature Climate Change* 3, no. 9 (2013): 816–821, https://doi.org/10.1038/nclimate1911.

3. D. Bonafilia, B. Tellman, T. Anderson, and E. Issenberg, "Sen1-Floods11: A Georeferenced Dataset to Train and Test Deep Learning Flood Algorithms for Sentinel-1," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (IEEE Xplore, 2020), 835–845, https://doi.org/10.1109/CVPRW50498.2020.00113.

4. M. S. Rahman and L. Di, "The State of the Art of Spaceborne Remote Sensing in Flood Management," *Natural Hazards* 85, no. 2 (2017): 1223–1248, https://doi.org/10.1007/s11069-016-2601-9.

5. N. Tsutsumida, T. Tanaka, and N. Sultana, "Automated Flood Detection From Sentinel-1 GRD Time Series Using Bayesian Analysis for Change Point Problems," *IEEE Transactions on Geoscience and*

Remote Sensing 63 (2025): 1–13, https://doi.org/10.1109/TGRS.2025.3620907.

6. M. Erdelj, E. Natalizio, K. R. Chowdhury, and I. F. Akyildiz, "Help From the Sky: Leveraging UAVs for Disaster Management," *IEEE Pervasive Computing* 16, no. 1 (2017): 24–32, https://doi.org/10.1109/MPRV.2017.11.

7. K. Amini, Y. Liu, J. E. Padgett, G. Balakrishnan, and A. Veeraraghavan, "Debris Segmentation Using Post-Hurricane Aerial Imagery," *Computer-Aided Civil and Infrastructure Engineering* 40, no. 25 (2025): 4116–4131, https://doi.org/10.1111/mice.70033.

8. Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," *Nature* 521, no. 7553 (2015): 436–444, https://doi.org/10.1038/nature14539.

9. O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," *arXiv* (2015), https://doi.org/10.48550/arXiv.1505.04597.

10. L. Zhou, T. Chen, F. Yang, et al., "MSFlood-Net: A Physically Informed Deep Learning Model Integrating Multi-Source Data for Flood Inundation Mapping," *Environmental Modelling & Software* 196 (2026): 106779, https://doi.org/10.1016/j.envsoft.2025.106779.

11. O. Oktay, J. Schlemper, L. L. Folgoc, et al., "Attention U-Net: Learning Where to Look for the Pancreas," *arXiv* (2018), https://doi.org/10.48550/arXiv.1804.03999.

12. S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional Block Attention Module," *arXiv* (2018), https://doi.org/10.48550/arXiv.1807.06521.

13. F. Fakhri and I. Gkanatsios, "Quantitative Evaluation of Flood Extent Detection Using Attention U-Net Case Studies From Eastern South Wales Australia in March 2021 and July 2022," *Scientific Reports* 15, no. 1 (2025): 12377, https://doi.org/10.1038/s41598-025-92734-x.

14. A. Refice, D. Capolongo, M. Chini, and A. D'Addabbo, "Improving Flood Detection and Monitoring Through Remote Sensing," *Water* 14, no. 3 (2022): 364, https://doi.org/10.3390/w14030364.

15. E. Nemni, J. Bullock, S. Belabbes, and L. Bromley, "Fully Convolutional Neural Network for Rapid Flood Segmentation in Synthetic Aperture Radar Imagery," *Remote Sensing* 12, no. 16 (2020): 2532, https://doi.org/10.3390/rs12162532.

16. M. Siddique and T. Ahmed, "CCD-Conv1D: A Deep Learning Based Coherent Change Detection Technique to Monitor and Forecast Floods Using Sentinel-1 Images," *Remote Sensing Applications: Society and Environment* 37 (2025): 101440, https://doi.org/10.1016/j.rsase.2024.101440.

17. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification With Deep Convolutional Neural Networks," *Communications of the ACM* 60, no. 6 (2017): 84–90, https://doi.org/10.1145/3065386.

18. K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *arXiv* (2015), https://doi.org/10.48550/arXiv.1409.1556.

19. J. Long, E. Shelhamer, and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," *arXiv* (2015), https://doi.org/10.48550/arXiv.1411.4038.

20. E. Shelhamer, J. Long, and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39, no. 4 (2017): 640–651, https://doi.org/10.1109/TPAMI.2016.2572683.

21. V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation," *arXiv* (2016), https://doi.org/10.48550/arXiv.1511.00561.

22. H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid Scene Parsing Network," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE Xplore, 2017), 6230–6239, https://doi.org/10.1109/CVPR.2017.660.

23. B. Bahrami and H. Arbabkhah, "Enhanced Flood Detection Through Precise Water Segmentation Using Advanced Deep Learning Models," *Journal of Civil Engineering Research* 6, no. 1 (2024): 1–8, https://doi.org/10.61186/JCER.6.1.1.

24. F. I. Diakogiannis, F. Waldner, P. Caccetta, and C. Wu, "ResUNet-a: A Deep Learning Framework for Semantic Segmentation of Remotely Sensed Data," *ISPRS Journal of Photogrammetry and Remote Sensing* 162 (2020): 94–114, https://doi.org/10.1016/j.isprsjprs.2020.01.013.

25. Zhou, Z., M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet++: A Nested U-Net Architecture for Medical Image Segmentation," *arXiv* (2018), https://doi.org/10.48550/arXiv.1807.10165

26. H. Huang, L. Lin, R. Tong, et al., "UNet 3+: A Full-Scale Connected UNet for Medical Image Segmentation," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, (2020), 1055–1059, https://doi.org/10.1109/ICASSP40776.2020.9053405.

27. G. Zhao, Y. Zhang, M. Ge, and M. Yu, "Bilateral U-Net Semantic Segmentation With Spatial Attention Mechanism," *CAAI Transactions on Intelligence Technology* 8, no. 2 (2023): 297–307, https://doi.org/10.1049/cit2.12118.

28. N. S. Jonnala, R. C. Bheemana, K. Prakash, et al., "DSIA U-Net: Deep Shallow Interaction With Attention Mechanism UNet for Remote Sensing Satellite Images," *Scientific Reports* 15, no. 1 (2025): 549, https://doi.org/10.1038/s41598-024-84134-4.

29. N. S. Jonnala, S. Siraaj, Y. Prastuti, et al., "AER U-Net: Attention-Enhanced Multi-Scale Residual U-Net Structure for Water Body Segmentation Using Sentinel-2 Satellite Images," *Scientific Reports* 15, no. 1 (2025): 16099, https://doi.org/10.1038/s41598-025-99322-z.

30. L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking Atrous Convolution for Semantic Image Segmentation," *arXiv* (2017), https://doi.org/10.48550/arXiv.1706.05587.

31. A. A. Sundaresan and A. A. Solomon, "Post-Disaster Flooded Region Segmentation Using DeepLabv3+ and Unmanned Aerial System Imagery," *Natural Hazards Research* 5, no. 2 (2025): 363–371, https://doi.org/10.1016/j.nhres.2024.12.003.

32. M. U. Danish, M. Buwaneswaran, T. Fonseka, and K. Grolinger, "Graph Attention Convolutional U-NET: A Semantic Segmentation Model for Identifying Flooded Areas," *arXiv* (2025), https://doi.org/10.48550/arXiv.2502.15907.

33. N. K. Sharma and M. Saharia, "DeepSARFlood: Rapid and Automated SAR-Based Flood Inundation Mapping Using Vision Transformer-Based Deep Ensembles With Uncertainty Estimates," *Science of Remote Sensing* 11 (2025): 100203, https://doi.org/10.1016/j.srs.2025.100203.

34. F. Alam, F. Ofli, and M. Imran, "CrisisMMD: Multimodal Twitter Datasets From Natural Disasters," in *Proceedings of the International AAAI Conference on Web and Social Media* 12, no. 1 (2018), https://doi.org/10.1609/icwsm.v12i1.14983.

35. A. Hussain, G. Latif, J. Alghazo, and E. Kim, "Flood Detection Using Deep Learning Methods From Visual Images," in *AIP Conference Proceedings* 3063 (2024): 030004, https://doi.org/10.1063/5.0194669.

36. M. Imran, C. Castillo, F. Diaz, and S. Vieweg, "Processing Social Media Messages in Mass Emergency: Survey Summary," in *Companion of the the Web Conference 2018 on the Web Conference 2018-WWW'18*, (2018), 507–511, https://doi.org/10.1145/3184558.3186242.

37. Y. Wang, Y. Shen, B. Salahshour, et al., "Urban Flood Extent Segmentation and Evaluation From Real-World Surveillance Camera Images Using Deep Convolutional Neural Network," *Environmental*

Modelling & Software 173 (2024): 105939, https://doi.org/10.1016/j.envsoft.2023.105939.

38. A. Vaswani, N. Shazeer, N. Parmar, et al., "Attention Is All You Need," *arXiv* (2023), https://doi.org/10.48550/arXiv.1706.03762.

39. A. Dosovitskiy, L. Beyer, A. Kolesnikov, et al., "An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale," *arXiv* (2021), https://doi.org/10.48550/arXiv.2010.11929.

40. J. Chen, Y. Lu, Q. Yu, et al., "TransUNet: Transformers Make Strong Encoders for Medical Image Segmentation," *arXiv* (2021), https://doi.org/10.48550/arXiv.2102.04306.

41. H. Cao, Y. Wang, J. Chen, et al., "Swin-Unet: Unet-Like Pure Transformer for Medical Image Segmentation," *arXiv* (2021), https://doi.org/10.48550/arXiv.2105.05537.

42. Y. Gao, M. Zhou, and D. Metaxas, "UTNet: A Hybrid Transformer Architecture for Medical Image Segmentation," *arXiv* (2021), https://doi.org/10.48550/arXiv.2107.00781.

43. Y. Zhang, H. Liu, and Q. Hu, "TransFuse: Fusing Transformers and CNNs for Medical Image Segmentation," *arXiv* (2021), https://doi.org/10.48550/arXiv.2102.08005.

44. Y. Xie, J. Zhang, C. Shen, and Y. Xia, "CoTr: Efficiently Bridging CNN and Transformer for 3D Medical Image Segmentation," *arXiv* (2021), https://doi.org/10.48550/arXiv.2103.03024.

45. B. Yu, Q. Zhou, L. Yuan, H. Liang, P. Shcherbakov, and X. Zhang, "3D Medical Image Segmentation Using the Serial–Parallel Convolutional Neural Network and Transformer Based on Cross-Window Self-Attention," *CAAI Transactions on Intelligence Technology* 10, no. 2 (2025): 337–348, https://doi.org/10.1049/cit2.12411.

46. J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-And-Excitation Networks," *arXiv* (2019), https://doi.org/10.48550/arXiv.1709.01507.

47. Y. Zhang, Q. Yang, T. Li, et al., "V-UNet: Medical Image Segmentation Based on Variational Attention Mechanism," *CAAI Transactions on Intelligence Technology* 10, no. 5 (2025): 1350–1362, https://doi.org/10.1049/cit2.70053.

48. M. Tan, and Q. V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," in *Proceedings of the 36 th International Conference on Machine Learning*, Vol. 97 (PMLR, 2019).

49. M. Rahnemoonfar, T. Chowdhury, A. Sarkar, D. Varshney, M. Yari, and R. R. Murphy, "FloodNet: A High Resolution Aerial Imagery Dataset for Post Flood Scene Understanding," *IEEE Access* 9 (2021): 89644–89654, https://doi.org/10.1109/ACCESS.2021.3090981.

50. M. Rahnemoonfar, T. Chowdhury, and R. Murphy, "RescueNet: A High Resolution UAV Semantic Segmentation Dataset for Natural Disaster Damage Assessment," *Scientific Data* 10, no. 1 (2023): 913, https://doi.org/10.1038/s41597-023-02799-4.

51. R. Karanjit, R. Pally, and V. Samadi, "Flooding Images Dataset," (2023) [Dataset], https://doi.org/10.34740/kaggle/ds/1864149.

52. L. Yang, "Flood Semantic Segmentation Dataset," Kaggle, (2024) [Dataset], https://www.kaggle.com/datasets/lihuayang111265/flood-semantic-segmentation-dataset.