A METHOD FOR IMPROVING THE TRAINING OF SEMANTIC SEGMENTATION MODELS USING ERROR FINGERPRINT MAPS

Vitalii Vlasenko[0009-0008-6951-8496] *and Andrii Dashkevych*[0000-0002-9963-0998]
National Technical University "Kharkiv Polytechnic Institute"
Kharkiv, Ukraine
Email: vitalii.vlasenko@infiz.khpi.edu.ua

Abstract - One of the key tasks of computer vision is semantic segmentation, which involves classifying each pixel of an image into predefined categories. With the development of machine learning and deep learning, convolutional neural networks (CNNs) and other artificial neural networks have become the foundation of modern segmentation methods, providing significantly higher accuracy and stability compared to classical image processing algorithms. These networks enable end-to-end feature extraction, learning of spatial hierarchies and contextual connections, which is particularly beneficial for analyzing complex and large-scale visual data such as aerial images. However, current architectures, such as U-Net, DeepLabV3+, FPN, and PSPNet, consistently face problems including class imbalance, boundary uncertainty, and overfitting to frequent patterns, and are inefficient in rare or complex domains. These challenges highlight the need for adaptive training strategies that enable models to focus more effectively on regions of high uncertainty and structural complexity. In this paper, we introduce a new method for improving the training process through dynamic Error Fingerprint Maps (EFMs) – spatially adaptive maps that capture patterns of uncertainty and model misclassifications during learning. EFM maps, which are based on a combination of error distributions and entropy, are integrated into the loss function as adaptive weights, enabling the network to focus its attention on complex or ambiguous regions. The proposed method is architecture-agnostic and can be applied to different segmentation models without structural modifications. Experimental results demonstrate that EFM-guided training provides more stable convergence, improves boundary accuracy, and enhances the generalization ability of segmentation models.

Keywords: Semantic segmentation, Deep learning, Convolutional neural networks, Machine Learning, Artificial Neural Network, Feature Extraction, Image Processing, Adaptive training, Error Fingerprint Map, Aerial images.

1. Introduction and Related Work

Today, with the rapid development of technologies related to unmanned aerial vehicles (UAVs), there is a constantly growing demand for the creation of automated methods that can quickly and qualitatively analyze large volumes of visual information. When working with aerial images obtained from UAVs, it is essential to achieve accurate and context-sensitive classification at the pixel level to identify various objects (buildings, roads, vegetation, water, etc.). Semantic segmentation, as a part of machine learning and image processing, is one of the key tasks in computer vision, as it enables the division of an image into distinct semantic areas, each distinguished by individual objects [1, 2]. This approach is actively used for environmental monitoring, urban planning, agriculture, and for the elimination of the consequences of natural disasters [3, 4].

The field of semantic segmentation was revolutionized significantly when the concept of deep learning emerged. Convolutional Neural Networks (CNNs), as a form of artificial neural network, became the basis of modern segmentation methods [5, 6]. The models offer significant improvements in accuracy and robustness compared to classical image processing algorithms [7]. The use of U-Net [12], DeepLabV3+ [3], Feature Pyramid Network (FPN) [8], and Pyramid Scene Parsing Network (PSPNet) [20] architectures is very popular, which can implement unique mechanisms to improve multi-scale feature extraction, contextual

© 2025 The Author(s). Published by Cefin Publishing House – Bucharest, https://ijomam.com/

https://doi.org/10.17683/ijomam/issue22.v2.10

understanding, and accurate object boundary detection.

When the U-Net architecture was developed [12], it was proposed for biomedical image segmentation and quickly became one of the most widely used models due to its encoder-decoder structure and bandwidth connections. The successful implementation of the architecture has inspired researchers to create various extensions that aim to improve performance on complex datasets through multi-scale pooling, dense connectivity, and advanced loss functions such as Lovasz-Softmax [2] and Boundary Loss [6].

Another very popular architecture for segmentation is DeepLabV3+ [3], which utilizes the Atrous Spatial Pyramid Pooling (ASPP) module to aggregate features across multiple receptive fields, thereby improving the understanding of multi-scale context. Current researchers are focused on optimizing DeepLabV3+ to adapt more effectively to remote sensing data. They use attention mechanisms, Bayesian uncertainty modeling, and transfer learning [9, 14]. Such improvements allow DeepLabV3+ to be effectively utilized on UAVs and in real-time applications [19].

The FPN [8] introduced a hierarchical top-down structure that allows combining semantic and spatial information from multiple levels of features. Such a design will enable FPN to work effectively with objects of different scales, which is very useful for remote sensing [13]. Researchers are also improving the model, for example, by combining it with attention modules such as CBAM [16] or dual-channel attention [5]. Additions to the model further enhance the segmentation accuracy for small objects, such as roads, the roofs of houses, and small vegetation areas.

The PSPNet architecture [20] is also popular among researchers, allowing for the effective use of the principle of global context aggregation using pyramidal pooling. This approach allows for the network to more accurately recognize the global spatial arrangement of a scene while preserving local details. Several applications of PSPNet have been proposed by researchers, including building extraction from high-resolution remote sensing images [18] and real-time semantic segmentation of UAV images [19]. The improved Shift-Pooling model of PSPNet [18] has demonstrated significant improvements in structural recognition tasks by improving the pooling of local features.

Despite the progress and development of deep learning-based segmentation methods, modern comprehensive studies on this topic [10, 11, 17, 21] show that common problems related to class imbalance, boundary uncertainty, and structural distortions in complex aerial scenes remain unresolved. The existing limitations motivate the development of adaptive learning strategies that allow models to focus on areas of high uncertainty and structural complexity.

To address such issues, we propose a novel approach in our work that integrates error fingerprint maps (EFMs) into the training process of segmentation models. EFM can dynamically identify spatial patterns of errors and uncertain models, forming adaptive weight maps that direct optimization to hard-to-recognize regions. Unlike many current model-specific improvements, our approach is architecture-independent and can be seamlessly applied to existing frameworks such as FPN or PSPNet.

Building on the results of our previous research, which conducted a comparative analysis of popular deep learning architectures for semantic segmentation of aerial images and developed a standardized approach to visualize errors and model entropy [15], it was found that FPN consistently provides the best balance between segmentation quality and computational efficiency. We extend previous results by implementing the EFM mechanism in the FPN and PSPNet architectures to demonstrate the versatility and effectiveness of the method in different network structures. The results obtained are expected to contribute to a more stable learning process, improve the accuracy of object boundary detection, and enhance the overall efficiency of aerial image processing and feature-based segmentation.

2. Method Overview

2.1. Concept of Error Fingerprint Maps

The proposed method improves the training process of semantic segmentation models by using spatial representations of prediction uncertainty and model errors, in the form of EFMs. At each training stage, the EFM highlights areas where the model makes inconsistent or ambiguous predictions. The maps are generated dynamically and used to adaptively tune the loss function adaptively, directing the model to focus on hard-to-learn regions and object boundaries.

Unlike traditional uniform loss weighting, EFM provides localized feedback that reflects changes in the weak sides of the model during training. This approach offers adaptive optimization, which improves both convergence stability and bounds accuracy.

2.2 EFM Generation and Mathematical Formulation

For an input image $I \in R^{H\times W\times 3}$ with ground-truth mask $Y \in \{1,\dots,C\}^{HxW}$, the segmentation network outputs class probabilities:

$$P(x,y) = \text{Softmax } (f_0(I))(x,y) \tag{1}$$

where $P_c(x,y)$ denotes the probability of class c at pixel (x,y).

To quantify the spatial distribution of model uncertainly and misclassification, two complementary maps are introduced: the Error Map $E(x,y)$ and the Entropy Map $H(x,y)$.

The error map highlights pixels where the predicted class label differs from the ground truth, allowing the capture of systematic spatial patterns of incorrect predictions:

$$E(x,y) = \begin{cases} 1, \text{if } \operatorname{argmax}_c \ P_c(x,y) \neq Y(x,y) \\ 0, \text{otherwise} \end{cases} \tag{2}$$

where $Y(x,y)$ denotes the ground-truth class label for pixel (x,y).

The entropy map measures the uncertainty of the model's probabilistic output for each pixel, thus reflecting the confidence of the network in its decision:

$$H(x,y) = -\sum_{c=1}^{C} P_c(x,y)\log(P_c(x,y)+\varepsilon), \tag{3}$$

where ε is a small constant ensuring numerical stability.

To jointly capture both spatial uncertainty and prediction inconsistency, the EFM is constructed as a weighted combination of the normalized error and entropy maps. This allows the model to focus on regions that are not only frequently misclassified but also have high prediction entropy, representing ambiguous or structurally complex areas. The formulation of the combined EFM is presented in (4).

$$EFM(x,y) = \alpha\overline{E}(x,y) + (1-\alpha)\overline{H}(x,y), \tag{4}$$

where $\overline{E}(x,y)$ denote the min-max normalized versions of the error, $\overline{H}(x,y)$ – the min-max normalized versions of the entropy maps, respectively, and $\alpha \in [0,1]$ controls the contribution of error versus uncertainty (empirically, $\alpha = 0.6$).

To improve spatial coherence and reduce isolated noisy regions, the error components and entropy components must be spatially smoothed before merging. This operation will ensure that the focus map shows more coherent structural regions rather than pixel-level noise. The smoothing is performed by convolving each normalized component with a Gaussian kernel G_δ, as presented in (5).

$$\overline{E}(x,y) = G_\delta * E(x,y)\,,\ \overline{H}(x,y) = G_\delta * H(x,y)\,, \tag{5}$$

where G_δ denotes a Gaussian function with standard deviation δ, which controls the level of spatial blurring.

The final focus map is recalculated every few epochs and re-normalized to the range [0,1] to ensure consistent scaling across training iterations, allowing the weighting to adapt dynamically as the model improves.

2.2. Integration into the Loss Function

The resulting normalized and smoothed focus map needs to be effectively incorporated into the model optimization process. The goal is to ensure that the network does not treat all pixels equally during training, but instead pays more attention to uncertain and structurally ambiguous areas. To achieve this goal, an EFM is introduced. The map represents an adaptive weighting component in the loss function, dynamically modulating the gradient contribution of each pixel based on its estimated complexity and uncertainty.

Let L_{CE} denote the pixel-wise cross-entropy loss and L_{Dice} the multi-class Dice loss. The baseline training objective can be formulated as:

$$L_{base} = \lambda_1 L_{CE} + \lambda_2 L_{Dice}\,, \tag{6}$$

where λ_1 and λ_2 control the contribution of each component.

To emphasize uncertain and complex regions, EFM is integrated as an adaptive weighting mask that scales the per-pixel loss according to the local difficulty of prediction:

$$L_{EFM} = \frac{1}{N}\sum_{x,y}(1+\alpha_f EFM(x,y)^{\beta_f} \cdot L_{base}(x,y)\,, \tag{7}$$

where α_f controls the strength of the focusing effect, β_f regulates its nonlinearity, and $N = H \times W$ is the number of pixels in the image.

This formulation adaptively increases the gradient contribution from spatial regions that are both uncertain and frequently misclassified, allowing the network to refine its attention over time.

To further enhance boundary localization, a boundary-sensitive weighting term is computed via morphological pooling:

$$W_b(x,y) = 1 + \gamma \,\|\, \text{maxpool } (Y) - \text{avgpool } (Y) \,\|\,, \tag{8}$$

where γ defines the intensity of boundary emphasis and Y is the ground truth mask.

The final pixel-wise weight that combines spatial uncertainty and edge importance is expressed as:

$$W(x,y) = W_b(x,y) \cdot (1 + \alpha_f \mathrm{EFM}(\mathrm{x},\ \mathrm{y})^{\beta_\mathrm{f}}), \qquad (9)$$

Finally, the total loss function for training becomes in (10), which effectively integrates both adaptive spatial focus and boundary precision into the model's learning objective.

$$L_{total} = \lambda_1 \frac{1}{N} \sum W(x,y) L_{CE}(x,y) + \lambda_2 L_{Dice}. \qquad (10)$$

2.3. Adaptive Training Algorithm

To incorporate the proposed EFM mechanism into the optimization process, an adaptive training procedure is designed.

The workflow consists of the following sequential steps:

(1) At the initialization stage, the training and validation datasets are loaded, and the segmentation model f_0, optimizer, learning rate scheduler, and evaluation metrics are set up.

(2) For each training epoch, a forward pass is performed to compute the pixel-wise cross-entropy loss L_{CE} and the Dice loss L_{Dice} as defined in (5).

(3) When the epoch number satisfies $epoch \geq E_{start}$ and $epoch \bmod k = 0$, the EFM is recalculated according to (3)-(4), and the corresponding EFM and entropy visualizations are saved for monitoring.

(4) A boundary-sensitive weight map $W_b(x,y)$ is then computed using (7) to highlight edge regions.

(5) The boundary and EFM-based weights are combined to form the final adaptive weighting map using (9).

(6) The training process focuses on the most uncertain samples by selecting the top $k\%$ of pixel-wise losses:

$$L_{CE}^{topk} = \mathrm{mean}\left(\mathrm{TopK}(L_{CE}(x,y),k)\right) \qquad (11)$$

which improves robustness to noise and label ambiguity.

(7) The model parameters are updated using mixed-precision training (AMP) with gradient clipping to stabilize convergence.

(8) After each optimization step, the model's weights are smoothed using the Exponential Moving Average (EMA):

$$\theta_{EMA} = \mu \theta_{EMA} + (1-\mu)\theta, \qquad (12)$$

where $\mu = 0.999$ defines the update rate.

(9) At the end of each epoch, the EMA weights are applied, and validation metrics such as IoU, F1-score, Accuracy, Precision, and Recall are evaluated. The checkpoint corresponding to the highest IoU is saved for testing.

(10) During inference, Test-Time Augmentation (TTA) is used to enhance prediction stability by averaging model outputs over four flipped variants of each input:

$$P_{TTA} = \frac{1}{4} \sum_{i=1}^{4} \mathrm{Flip}_i^{-1}\left(\mathrm{Softmax}\left(f_0(\mathrm{Flip}_i(I))\right)\right). \qquad (13)$$

3. Experimental Setup

3.1. Dataset and Preprocessing

The experiments were conducted on a custom dataset of aerial RGB images and their corresponding semantic segmentation masks, representing land-cover categories such as vegetation, buildings, roads, and water bodies.

All images were resized and normalized to a spatial resolution of 512x512 pixels, and segmentation masks were encoded as integer label maps within the range [0, 6] (see Figure 1).

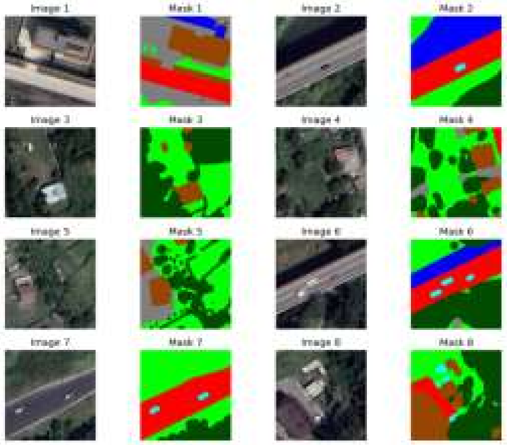

Figure 1: Prepared image pairs (aerial image and corresponding segmentation)

For model development, the dataset was initially divided into 80% for training and 20% for validation to ensure balanced class representation.

A separate test subset was then derived from the validation portion and used exclusively for final performance evaluation after model convergence.

This approach enables control over the model's generalization while maintaining unbiased testing.

To increase robustness and prevent overfitting, standard data augmentation was applied. We used random horizontal and vertical rotations, brightness and contrast adjustments, and Gaussian noise. The transformations allowed us to model the variability of the airspace as in real-world tasks (e.g., differences in illumination, changes in camera perspective, or texture diversity).

3.2. Model Architectures

To evaluate the effectiveness of the proposed EFM-based training strategy, two state-of-the-art deep learning architectures were employed.

The first model FPN is built on a ResNet-34 encoder and leverages multi-level feature aggregation to construct a hierarchical spatial representation, allowing for the effective segmentation of objects with varying scales.

The second model PSPNet enhances global contextual understanding through pyramid pooling, which captures multi-scale spatial relationships and improves the network's awareness of large-scale scene structures.

Both models were implemented using the PyTorch Segmentation Models framework.

Each architecture outputs seven segmentation classes and applies the Softmax activation function for pixel-wise classification.

The encoder backbones were initialized with ImageNet-pretrained weights, which significantly accelerated convergence and improved the stability of early training stages.

3.3. Training Configuration

All experiments were performed in PyTorch 2.0 with CUDA acceleration on a GPU environment provided by Google Colab.

The overall training procedure consisted of three sequential stages: basic training without error maps, complete training integrated with EFM, and adaptive fine-tuning to improve robustness.

The training of the models at the baseline stage took place over 50 epochs. The AdamW optimizer was used with a learning rate of 5×10^{-4} and a weight decay of 1×10^{-4}. A cosine annealing scheduler ($T_{\max}=50$) was used to adjust the learning rate gradually.

The loss function combined cross-entropy (with label smoothing of 0.05) and Dice loss, providing a balanced optimization between class-level accuracy and spatial consistency. Gradient cutoff (1.0) and mixed-precision learning (AMP) were used to stabilize the learning process throughout all experiments.

The second stage used the same 50-epoch configuration, but the EFM mechanism was integrated into the training pipeline. EFMs and entropy maps were computed on the validation data every two epochs, starting from the fifth epoch. These maps dynamically modulated the per-pixel weighting in the loss function, directing the model's attention to areas with higher uncertainty and structural complexity. The weighting factors were empirically set to $\alpha=2.0$ and $\beta=1.5$, which determines the strength and nonlinearity of the focusing effect, while the edge weighting factor $\gamma=1.2$ emphasized the edges of the object.

Finally, an adaptive fine-tuning step was performed for an additional 15 epochs using the same EFM-driven configuration. An exponential moving average (EMA) of the model parameters, with a decay factor of $\mu=0.999$, was applied to stabilize convergence, which improved the temporal consistency of the learned weights and reduced gradient noise.

3.4. Evaluation Metrics

Model performance was evaluated using a set of standard quantitative metrics, including Accuracy, Precision, Recall, Intersection over Union (IoU), and the F1-score.

Accuracy reflects the overall correctness of classification across all pixels, while Precision measures the reliability of optimistic predictions. Recall quantifies the model's sensitivity to accurate positive detections, and IoU represents the spatial overlap between the predicted segmentation and the ground truth mask. Finally, the F1-score, defined as the harmonic mean of Precision and Recall, provides a balanced indicator of model performance, especially in cases of class imbalance.

All metrics were computed on the test subset after each training phase using the torchmetrics and scikit-learn libraries. The best-performing model checkpoint was determined based on the highest IoU value achieved on the validation data, ensuring consistent and unbiased evaluation.

To further enhance robustness during inference, TTA was applied. This procedure involved averaging predictions across four transformed variants of each input image – including horizontal, vertical, and diagonal flips – as defined by (13).

4. Results and Discussion

The experimental results clearly demonstrate the effectiveness of the proposed error fingerprint map (EFM) mechanism in enhancing both the stability and performance of semantic segmentation models.

The combination of quantitative analysis, learning dynamics, visual heat maps, and qualitative segmentation examples confirms that optimization using EFM enables the network to focus on the most uncertain and structurally complex regions during training, leading to improved generalization and spatial accuracy.

Table 1 presents the quantitative results for the FPN architecture, and Table 2 presents the quantitative results for the PSPNet. Each model was tested under three training configurations: basic training, training with EFM integration, and fine-tuning using EFM. In both cases, the implementation of the proposed EFM consistently improved all performance metrics, confirming its architecture-independent effectiveness.

In the case of FPN, IoU increased from 0.7586 to 0.7817 after EFM integration and further improved to 0.8275 after fine-tuning. Similarly, the F1-score improved from 0.8603 to 0.9045, and overall Accuracy rose from 0.8621 to 0.9054.

In contrast, PSPNet showed a more gradual improvement trend. Direct inclusion of EFM yielded only modest performance improvements – IoU increased from 0.7163 to 0.7172 and F1 from 0.8306 to 0.8312.

However, the fine-tuning stage demonstrated a significant performance improvement, with an IoU of 0.7612 and an F1-score of 0.8622. Accuracy similarly increased from 0.8322 to 0.8643.

Table 1. Quantitative comparison of FPN performance across training stages

Training Stage	Baseline	+EFM	+EFM Fine-Tuning
Accuracy	0.8621	0.8771	0.9054
Precision	0.8612	0.8769	0.9052
Recall	0.8621	0.8771	0.9054
IoU	0.7586	0.7817	0.8275
F1	0.8603	0.8758	0.9045

Table 2. Quantitative comparison of PSPNet performance across training stages

Training Stage	Baseline	+EFM	+EFM Fine-Tuning
Accuracy	0.8322	0.8337	0.8643
Precision	0.8308	0.8312	0.8640
Recall	0.8322	0.8337	0.8643
IoU	0.7163	0.7172	0.7612
F1	0.8306	0.8312	0.8622

PSPNet needs smoother adaptation to EFM weighting because it has a pyramidal pooling structure, taking into account the global context. Once properly tuned, the model fully utilizes EFM recommendations, achieving stable convergence and improved semantic consistency.

These results underscore that spatially adaptive weighting enables the network to capture fine-grained boundaries more effectively and handle heterogeneous textures.

The evolution of the training and validation metrics for the FPN model is shown in Figure 2.

During baseline training, the model exhibited oscillating validation losses and slower convergence, indicating a tendency to retune specific spatial patterns and local textures. The convergence process became noticeably smoother after integrating the error trace map. A steady decrease in validation loss was observed, accompanied by a consistent increase in IoU, F1-score, and overall accuracy.

The EFM-based optimization allowed the network to focus more effectively on complex areas (object edges and fine structural details) rather than processing all pixels uniformly. This adaptability improved gradient stability and reduced noise in the optimization. Overall, the EFM mechanism acted as a dynamic regularizer, accelerating learning and improving generalization.

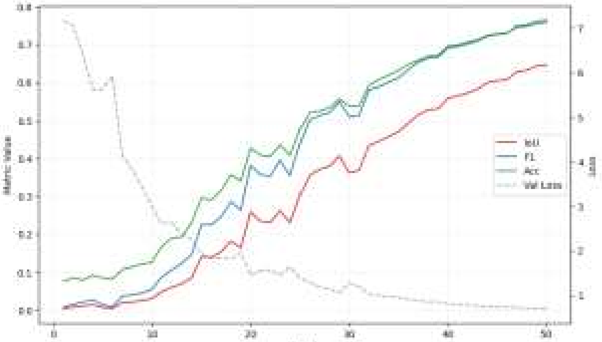

Figure 2: Training and validation curves for loss, IoU, F1, and Accuracy for the FPN model during EFM-guided optimization

The learning curves for the PSPNet model are presented in Figure 3.

Unlike FPN, the PSPNet baseline configuration exhibited unstable convergence, with noticeable oscillations in the validation loss, which is typical for architectures with large receptive fields and strong global context aggregation.

After the initial integration of EFM, the training process became moderately more stable, although the improvements in validation metrics remained limited.

However, during the fine-tuning stage, PSPNet demonstrated a clear performance boost — the validation loss decreased smoothly, and both IoU and F1-score steadily increased, indicating a more confident and consistent optimization process.

This suggests that the PSPNet architecture benefits most from a gradual adaptation to spatial weighting rather than direct early integration.

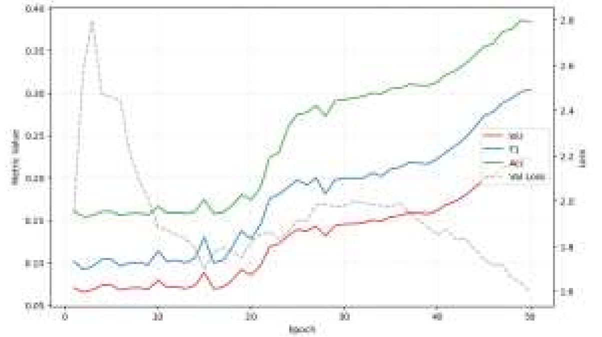

Figure 3: Training and validation curves for loss, IoU, F1, and Accuracy for the PSPNet model during EFM-guided optimization

Overall, the proposed EFM mechanism complements the pyramidal pooling strategy by improving spatial selectivity and improving contextual coherence as the network stabilizes its internal feature representations.

To further investigate the spatial behavior of the proposed approach, the evolution of EFM and entropy distributions across several training epochs for both FPN and PSPNet architectures is shown.

For the FPN model (see Figure 4), the early EFM maps reveal strong activation along object boundaries, small structural elements, and regions with irregular textures – areas most susceptible to misclassification.

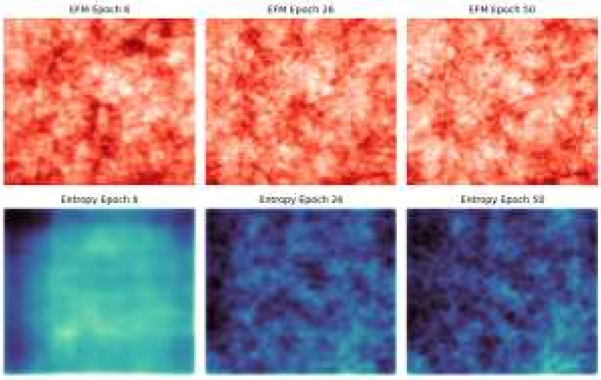

Figure 4: Evolution of error footprint maps (top) and entropy maps (bottom) for FPN over training epochs

During training, the high-error areas gradually shrink and smooth out, demonstrating that the model is effectively learning to correct its marginal errors and stabilize segmentation in complex spatial regions. The corresponding entropy maps also show a noticeable decrease in uncertainty, confirming that the model confidence increases as optimization progresses.

The PSPNet model (see Figure 5) exhibits a slightly different evolution compared to FPN, primarily due to its pyramidal pooling mechanism, which emphasizes large-scale contextual features.

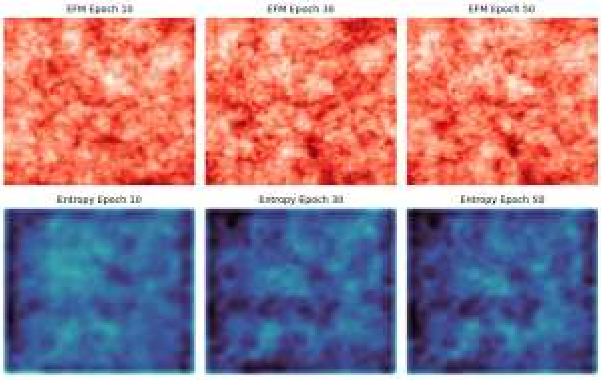

Figure 5: Evolution of error footprint maps (top) and entropy maps (bottom) for PSPNet over training epochs

In the early stages of training, the EFM highlights broad and ambiguous regions with unclear class boundaries and overlapping textures – regions where the PSPNet global-context aggregation is less sensitive to acceptable structural variations.

After fine-tuning, the diffuse error zones gradually decrease and become more localized, indicating improved spatial consistency and class separation. The overall entropy decrease is more gradual than that of FPN, indicating that PSPNet needs additional training adaptation to use EFM weighting and fully stabilize multi-scale feature integration.

Finally, Figure 6 shows a visualization of the difference error maps (start-end) for the FPN model, and Figure 7 shows the difference error maps for the PSPNet. The visualization is obtained by subtracting the final EFM map from the initial one.

The red-blue heatmap highlights the areas where prediction errors were most effectively reduced during training, providing a clear spatial representation of the network's gradual improvement.

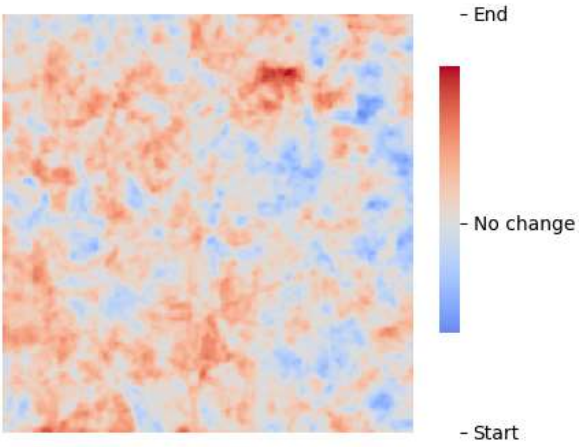

Figure 6: Difference Error Map (Start – End) visualization for the FPN model

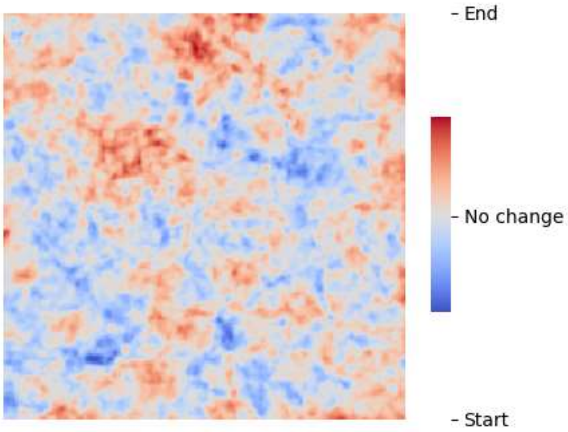

Figure 7: Difference Error Map (Start – End) visualization for the PSPNet model

For FPN, the most significant improvements are concentrated along object boundaries. At the same time, PSPNet shows a broader reduction in texture-dense or contextually ambiguous areas, highlighting the complementary strengths of both architectures.

To visually demonstrate the effectiveness of the proposed approach, Figures 8 and 9 present the qualitative segmentation results obtained after fine-tuning using EFM for the FPN and PSPNet architectures. Each figure shows the input image, the ground truth mask, and the corresponding model prediction.

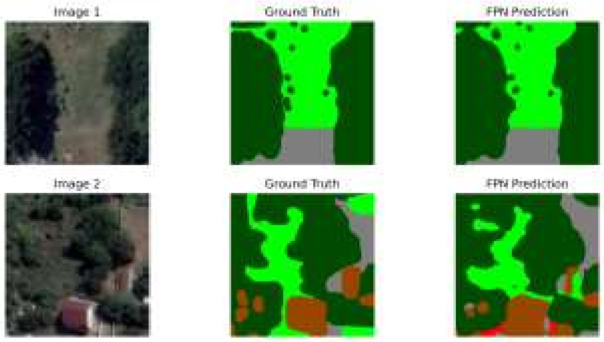

Figure 8: Final segmentation results for FPN: Input, Ground Truth, and EFM-guided prediction

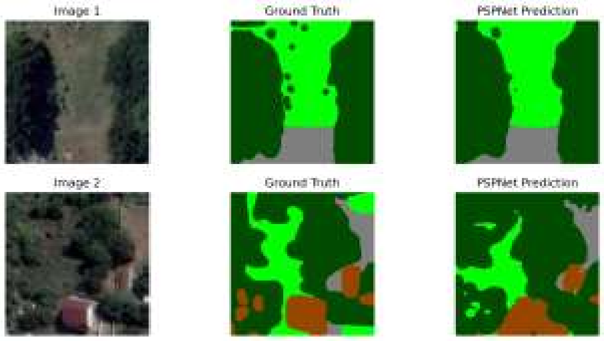

Figure 9: Final segmentation results for PSPNet: Input, Ground Truth, and EFM-guided prediction

Experimental data confirm that integrating error fingerprint maps into the training process provides consistent benefits for both architectures.

This stabilizes convergence, reduces overtraining, improves local boundary accuracy, and increases overall segmentation consistency.

Importantly, these improvements were achieved without changing the network architecture or significantly increasing computational costs.

Since EFM functions as an external adaptive weighting mechanism, it can be seamlessly integrated into any existing segmentation system. The parallel performance improvements observed for both FPN and PSPNet demonstrate that EFM is an architecture-agnostic, lightweight, and efficient enhancement for deep semantic segmentation.

5. Conclusions

This study introduced an EFM, a spatially adaptive mechanism that dynamically directs training attention to uncertain and misclassified regions.

Integrating EFM into the loss function improved convergence stability and segmentation accuracy across architectures, although the degree of improvement varied depending on the model design.

For the FPN architecture, the inclusion of EFM resulted in a stable and consistent performance improvement. At the same time, PSPNet benefited more significantly after the adaptive fine-tuning stage, indicating that architectures with strong global context aggregation require smoother adaptation to spatial weighting.

On average, EFM-based optimization increased the IoU and F1-score values by 4-5%, and improved the overall accuracy by approximately 4% compared to baseline training.

The improvements confirm that EFM effectively enhances the model's ability to capture complex textures, object boundaries, and spatial inconsistencies, resulting in more stable and reliable predictions.

Significantly, the proposed mechanism operates as an external component, requiring no architectural modifications or additional computational overhead, making it compatible with a wide range of modern segmentation networks.

Future work will focus on extending EFM to multimodal and temporal segmentation tasks, as well as exploring its role in dynamic scene understanding and self-adaptation.

References

[1] Adegun, A.A., Viriri, S., & Tapamo, J.-R. (2023). Review of Deep Learning Methods for Remote Sensing Satellite Images Classification: Experimental Survey and Comparative Analysis. Journal of Big Data, Volume 10, Article 93. https://doi.org/10.1186/s40537-023-00772-x

[2] Berman, M., Triki, A.R., & Blaschko, M.B. (2018). The Lovasz-Softmax Loss: A Tractable Surrogate for the Optimization of the Intersection-Over-Union Measure in Neural Networks. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Pp: 4413–4421. https://doi.org/10.1109/CVPR.2018.00464

[3] Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., & Adam, H. (2018). Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. Computer Vision – ECCV 2018, Lecture Notes in Computer Science, Volume 11211, Pp: 833–851. Springer. https://doi.org/10.1007/978-3-030-01234-2_49

[4] Chen, Z., Xie, Y., & Wei, Y. (2025). Toward High-Resolution UAV Imagery Open-Vocabulary Semantic Segmentation. Drones, Volume 9(7), Article 470. https://doi.org/10.3390/drones9070470

[5] Jiang, J., Feng, X., & Huang, H. (2024). Semantic Segmentation of Remote Sensing Images Based on Dual-Channel Attention Mechanism. IET Image Processing, Volume 18, Pp: 2346–2356. https://doi.org/10.1049/ipr2.1310

[6] Kervadec, H., Bouchtiba, J., Desrosiers, C., Granger, E., Dolz, J., & Ben Ayed, I. (2021). Boundary Loss for Highly Unbalanced Segmentation. Medical Image Analysis, Volume 67, 101851. https://doi.org/10.1016/j.media.2020.101851

[7] Li, J., Cai, Y., Li, Q., Kou, M., & Zhang, T. (2024). A Review of Remote Sensing Image Segmentation by Deep Learning Methods. International Journal of Digital Earth, Volume 17(1). https://doi.org/10.1080/17538947.2024.2328827

[8] Lin, T.-Y., Dollar, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature Pyramid Networks for Object Detection. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Pp: 936–944. https://doi.org/10.1109/CVPR.2017.106

[9] Lin, T.-Y., Goyal, P., Girshick, R., He, K., Dollár, P., 2020. Focal Loss for Dense Object Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 42(2), Pp: 318–327. https://doi.org/10.1109/TPAMI.2018.2858826

[10] Meng, X., Zhu, L., Han, Y., & Zhang, H. (2023). We Need to Communicate: Communicating Attention Network for Semantic Segmentation of High-Resolution Remote Sensing Images. Remote Sensing, Volume 15(14), Article 3619. https://doi.org/10.3390/rs15143619

[11] Minaee, S., Boykov, Y.Y., Porikli, F., Plaza, A.J., Kehtarnavaz, N., & Terzopoulos, D. (2021). Image Segmentation Using Deep Learning: A Survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 44(7), Pp: 3523–3542. https://doi.org/10.1109/TPAMI.2021.3059968

[12] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, Lecture Notes in Computer Science, Volume 9351, Pp: 234-241. Springer. https://doi.org/10.1007/978-3-319-24574-4_28

[13] Thapa, A., Horanont, T., Neupane, B., & Aryal, J. (2023). Deep Learning for Remote Sensing Image Scene Classification: A Review and Meta-Analysis. Remote Sensing, Volume 15(19), Article 4804. https://doi.org/10.3390/rs15194804

[14] Upadhyay, U., Karthik, S., Chen, Y., Mancini, M., & Akata, Z. (2022). BayesCap: Bayesian Identity Cap for Calibrated Uncertainty in Frozen Neural Networks. Computer Vision – ECCV 2022, Lecture Notes in Computer Science, Volume 13680, Pp: 299–317. Springer. https://doi.org/10.1007/978-3-031-19775-8_18

[15] Vlasenko, V.O. (2025). Standardized Error and Entropy Maps as Tool for Visualizing Model Quality Assessment in Deep Learning for Aerial Image Processing [in Ukrainian]. Modern Modeling Problems, Volume 30, Pp: 32–43. https://doi.org/10.33842/2313-125X-2025-30-32-43

[16] Woo, S., Park, J., Lee, J.-Y., & Kweon, I.S. (2018). CBAM: Convolutional Block Attention Module. Computer Vision – ECCV 2018, Lecture Notes in Computer Science, Volume 11211, Pp: 3–19. Springer. https://doi.org/10.1007/978-3-030-01234-2_1

[17] Yu, A., Quan, Y., Yu, R., Guo, W., Wang, X., Hong, D., Zhang, H., Chen, J., Hu, Q., & He, P. (2023). Deep Learning Methods for Semantic Segmentation in Remote Sensing with Small Data: A Survey. Remote Sensing, Volume 15(20), Article 4987. https://doi.org/10.3390/rs15204987

[18] Yuan, W., Wang, J., & Xu, W. (2022). Shift Pooling PSPNet: Rethinking PSPNet for Building Extraction in Remote Sensing Images from Entire Local Feature Pooling. Remote Sensing, Volume 14(19), Article 4889. https://doi.org/10.3390/rs14194889

[19] Zhang, Z., & Li, G. (2025). UAV Imagery Real-Time Semantic Segmentation with Global-Local Information Attention. Sensors, Volume 25(6), Article 1786. https://doi.org/10.3390/s25061786

[20] Zhao, H., Shi, J., Qi, X., Wang, X., & Jia, J. (2017). Pyramid Scene Parsing Network. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Pp. 6230–6239. https://doi.org/10.1109/CVPR.2017.660

[21] Zhu, X.X., Tuia, D., Mou, L., Xia, G.-S., Zhang, L., Xu, F., & Fraundorfer, F. (2017). Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources. IEEE Geoscience and Remote Sensing Magazine, Volume 5(4), Pp: 8–36. https://doi.org/10.1109/MGRS.2017.2762307