



Article

A Novel Attention-Based Generalized Efficient Layer Aggregation Network for Landslide Detection from Satellite Data in the Higher Himalayas, Nepal

Naveen Chandra ¹, Himadri Vaidya ^{2,3}, Suraj Sawant ⁴ and Sansar Raj Meena ^{5,6,*}

- ¹ Geomorphology, Environmental, and Engineering Geology, Wadia Institute of Himalayan Geology, Dehradun 248001, Uttarakhand, India; naveenchandra@wihg.res.in
- ² Department of Computer Science and Engineering, Graphic Era Hill University, Dehradun 248002, Uttarakhand, India; himadrivaidya@gehu.ac.in
- ³ Graphic Era Deemed to be University, Dehradun 248002, Uttarakhand, India
- ⁴ Department of Computer Science, COEP Technological University, Pune 411005, Maharashtra, India; sts.comp@coep.ac.in
- ⁵ Machine Intelligence and Slope Stability Laboratory, Department of Geosciences, University of Padova, 35139 Padova, Italy
- ⁶ Center for Remote Sensing, Department of Earth and Environment, Boston University, Boston, MA 02215, USA
- * Correspondence: sansarraj.meena@unipd.it

Abstract: Landslide event detection poses a significant challenge in the remote sensing community, especially with the advancements in computer vision technology. As computational capabilities continue to grow, the traditional manual and partially automated methods of landslide recognition from remote sensing data are transitioning towards automatic approaches using deep learning algorithms. Moreover, attention models, encouraged by the human visual system, have emerged as crucial modules in diverse applications including natural hazard assessment. Therefore, we suggest a novel and intelligent generalized efficient layer aggregation network (GELAN) based on two prevalent attention modules, efficient channel attention (ECA) and convolutional block attention module (CBAM), to enrich landslide detection techniques from satellite images. CBAM and ECA are separately integrated into GELAN at different locations. The experiments are conducted using satellite images of the Nepal Himalayan region. Standard metrics such as precision, recall, F-score, and mAP (mean average precision) are considered for quantitative evaluation. GELANc+CBAM (F-score = 81.5%) demonstrates the best performance. This study underscores the suitability of the proposed approach in up-to-date inventory creation and accurate landslide mapping for disaster recovery and response efforts. Moreover, it contributes to developing early prediction models for landslide hazards.

Keywords: attention; CBAM; ECA; GELAN; landslides; YOLOv9



Citation: Chandra, N.; Vaidya, H.; Sawant, S.; Meena, S.R. A Novel Attention-Based Generalized Efficient Layer Aggregation Network for Landslide Detection from Satellite Data in the Higher Himalayas, Nepal. *Remote Sens.* **2024**, *16*, 2598. <https://doi.org/10.3390/rs16142598>

Academic Editors: Federico Raspini, Chao Zhou and Kunlong Yin

Received: 22 May 2024

Revised: 3 July 2024

Accepted: 8 July 2024

Published: 16 July 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Landslides represent critical geohazards influenced by climatic, seismic, and human-induced factors, resulting in significant loss of life, infrastructure, and economic resources [1–3]. With current weather patterns, urban extension, and population growth, landslides are expected to rise, particularly in mountainous regions worldwide [4,5]. Hence, effective mitigation strategies and risk reduction measures are essential for humankind. Extensive monitoring, forecasting, and detection form the foundation of landslide risk management [1,6]. Monitoring involves tracking the displacement of landslides over time, while forecasting, or prediction, involves timely estimation of potential events. Detection provides detailed and precise information about landslide occurrences. Continuous monitoring is crucial for predicting landslides, while detection facilitates the identification of

key parameters, such as location and scale, essential for reducing their cascading effects. Furthermore, accurate detection of landslide events is vital for post-hazard recovery efforts and preparing up-to-date inventories. Additionally, detecting landslide events aids in mapping vulnerable areas, which is essential for sustainable planning. So, there is a growing trend towards rapidly and precisely detecting landslide events.

The traditional approach to landslide extraction relied upon ground-based measurements [7,8], which were time-consuming, labor-intensive, and less effective in emergency response situations [7]. However, with advancements in remote sensing technology, high-resolution images have become vital in landslide hazard assessment studies [1].

Four primary techniques [8] leverage remote sensing data for landslide detection: visual interpretation, pixel-based, object-oriented, and artificial intelligence (AI) methods. Although relying on expert knowledge, visual interpretation methods are time-consuming and may not meet the requirements for rapid response actions [7]. Pixel-based approaches overcome the restrictions of visual analysis by employing binary algorithms [8] to classify pixels in an image into specific categories (e.g., landslide and background). However, accurately distinguishing objects with spectral appearances similar to landslides can be challenging. Object-oriented procedures rely on multiscale segmentations that consider image primitives such as texture, shape, and spectrum [9]. These techniques require empirical settings (threshold criteria) and may struggle with the rapid segmentation of large geographical areas using high-resolution satellite imagery [10]. Keyport et al. (2018) [11] conducted a comparative study to infer the capabilities and limitations of both methods for landslide detection. Furthermore, AI-based techniques have witnessed remarkable advancements across multiple fields, including hazard exploration. Sequentially, remote sensing-centered big data [12,13] have provided auxiliary support to AI initiatives.

Additionally, machine learning (ML) algorithms have shown significant progress in landslide prevention and assessment studies [4]. Specifically, logistic regression [14], support vector machines [15], random forest [16], and decision trees [17] have been prominent. With advancements in computational tools, particularly graphical processing units (GPUs), deep learning (a subdiscipline of ML) algorithms have demonstrated exceptional performance across various fields, including object detection [18], image segmentation [19], and scene classification [20]. Moreover, deep learning has been successfully applied in geohazard analysis, including earthquakes, avalanches, and landslides [6].

Previously, various deep learning algorithms, particularly convolutional neural networks (CNNs), with diverse architectures have been employed for landslide information recognition [21], specifically Mask-RCNN [9,21] and U-Net [22–24]. Currently, YOLO (you only look once) models have gained popularity in the areas of computer vision [25]. For landslide detection, a series of YOLO models, namely, YOLOv3 [8], YOLOv4 [26], YOLOv5 [27], YOLOv7 [28], and YOLOv8 [29], have been augmented and applied. In addition, attention modules [30], inspired by human visual mechanisms, have been integrated into CNN architectures to enhance detection accuracy, specifically 3D-SCAM [31], SW-MSA [32], YOLO-SA [2], and LA-YOLO-LLL [33]. Despite the notable contributions by researchers, identifying a suitable deep learning-based model for landslide detection remains a challenge. In addition, evaluating the performance of attention models fused within a one-stage algorithm is in the preliminary stage and is, hence, crucial. We suggest an attention-driven generalized efficient layer aggregation network (GELAN) to leverage its strength and efficiency concerning landslide detection using satellite images with complex environments to address these concerns. The major contributions of this research work are threefold:

1. Developing an intelligent and optimized GELAN for automated landslide mapping in the Himalayan terrain.
2. Incorporating attention-based cognitive models, mainly CBAM and ECA, within GELAN to augment landslide hazard analysis.
3. Providing a comprehensive model evaluation and comparison considering earth observational images of different geological and geomorphological settings.

This paper is structured as follows: Section 2 provides the details of the study area and dataset, while Section 3 outlines the overall methodology. Section 4 presents the results and Section 5 contains the discussion. Lastly, Section 6 includes the conclusion.

2. Study Area and Dataset

The study area is in Nepal, the Greater Himalayas (Figure 1), comprising 275 GeoTIFF-8 bit Landsat images of 416×416 pixels. This region is prone to landslides due to various factors such as seismic activity, topographical conditions, seasonal rainfall, and human activities such as deforestation [34,35]. The Nepal region has experienced significant economic losses and numerous fatalities from natural disasters such as earthquakes and landslides. For instance, the Gorkha earthquake, with a magnitude of 7.8, occurred on 25 April 2015, followed by aftershocks on 12 May 2015, resulting in extensive damage and approximately 9000 fatalities [34,36]. The land cover pattern of the study site includes bare soil, vegetation, water bodies, rocks, and urban areas, making the extraction of landslide information from this complex environment particularly challenging. The dataset is available at <https://zenodo.org/record/3675410#.ZBv3UZBxD8> (accessed on 15 March 2024). Figure 2 represents sample images of the Nepal landslide detection database.

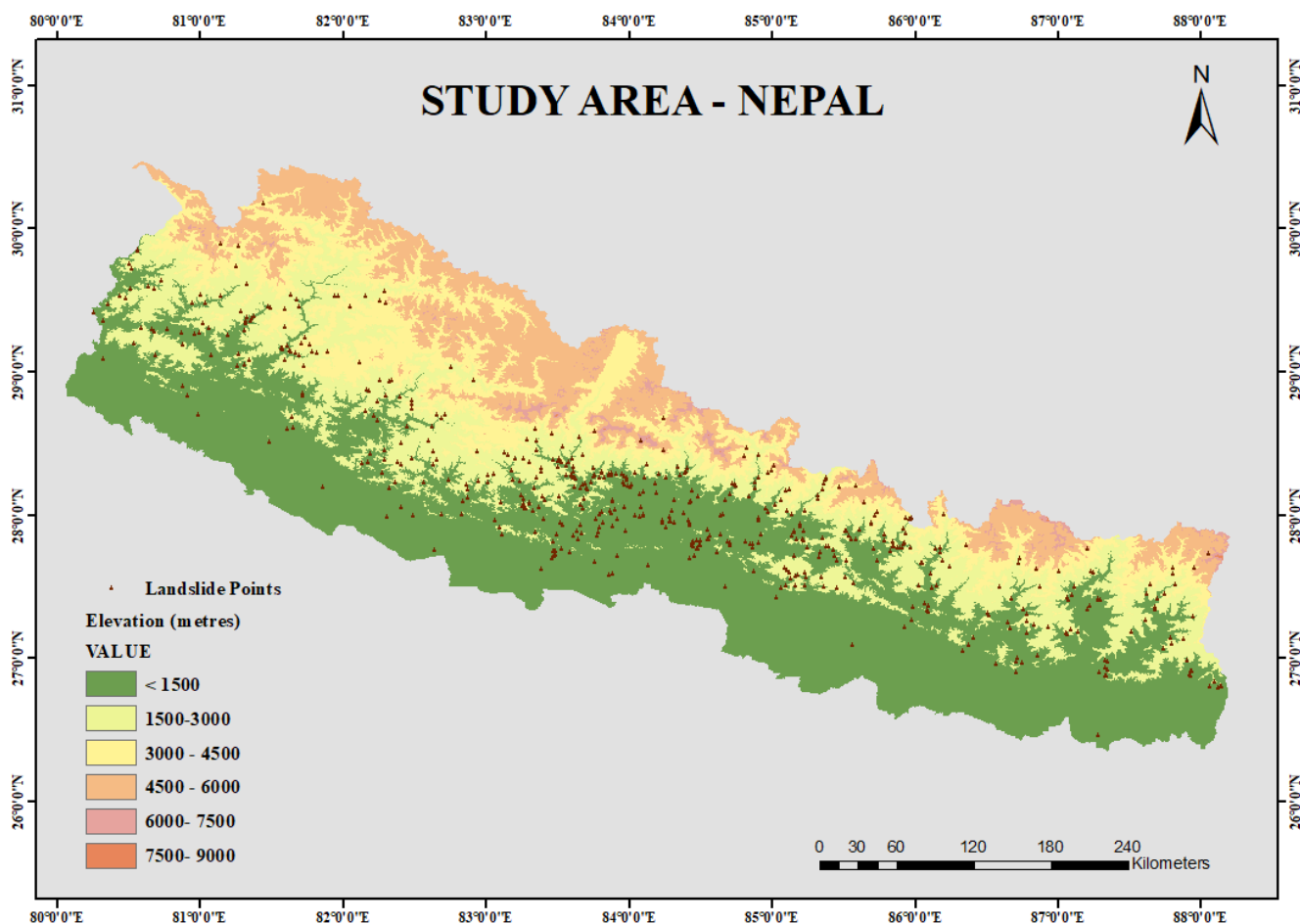


Figure 1. Location map: Nepal (study area) and the landslide event.

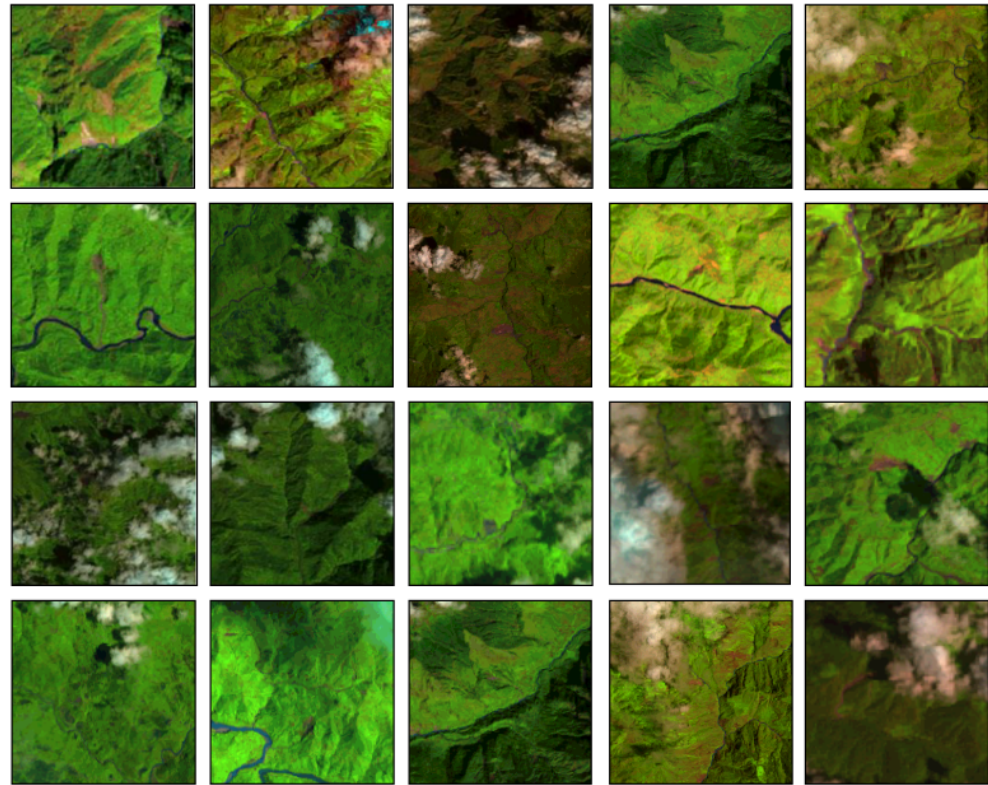


Figure 2. Examples from Nepal landslide detection database.

3. Methodology

This section outlines the technique for automated landslide recognition, which encompasses four key stages: (1) data preparation, (2) model development, (3) experimental settings, and (4) evaluation measures. Algorithm 1 provides the pseudocode for the proposed method.

Algorithm 1 Pseudocode of the novel GELAN based on an attention mechanism.

- 1: **Input:** Labeled landslide-based satellite imagery
 - 2: **Output:** Predicted bounding box coordinates
 - 3: **Annotation transformation:** Convert annotations into YOLO format.
 - 4: **Data formation:** Organize data into training, validation, and test sets.
 - 5: **Model training:**
 - 6: GELANc, and GELANc+attention
 - 7: GELANe, and GELANe+attention
 - 8: YOLOv9c, YOLOv9e, and YOLOv9c+attention
 - 9: **for each epoch do**
 - 10: **Hyperparameter tuning:** Fine-tune image size, batch size, etc.
 - 11: Calculate the loss function (classification, bounding box, and confidence), and update the network's parameters.
 - 12: **Model validation:** Evaluate detection performance on the validation dataset.
 - 13: **end for**
 - 14: Save the model weights with the best performance (best_weight.pt).
 - 15: **Model testing:** Apply the trained network to detect landslides in test images.
 - 16: Predicted bounding box coordinates for detected landslides.
-

3.1. Data Preparation

The dataset initially contained annotations represented as masks in .png format. To align with our requirements for the YOLO format, we transformed the annotations into five

elements: object class ID, X_center, Y_center, height, and width. Algorithm 2 represents the procedure of annotation conversion.

Algorithm 2 Procedure for annotation conversion.

- 1: **INPUT** Satellite images with landslide events, and ground truth images in binary format.
- 2: **OUTPUT** Annotations in YOLO format.
- 3: Read ground truth images (mask).
- 4: Convert images to grayscale.
- 5: Calculate the height (h) and width (w) of the image.
- 6: Contour estimation method: Identify the edges of the ground truth image.
- 7: **for** each identified contour **do**
- 8: Get the minimum and maximum coordinates of the bounding box:
- 9: Minimum x-coordinate (x_{min}), Minimum y-coordinate (y_{min}), Maximum x-coordinate (x_{max}), Maximum y-coordinate (y_{max}).
- 10: **end for**
- 11: Non-maximum suppression to remove overlapping bounding boxes.
- 12: **for** each remaining bounding box **do**
- 13: Calculate normalized bounding box coordinates:

$$x_{center} = \frac{x_{min} + x_{max}}{2w}$$

$$y_{center} = \frac{y_{min} + y_{max}}{2h}$$

$$\text{normalized-height} = \frac{y_{max} - y_{min}}{h}$$

$$\text{normalized-width} = \frac{x_{max} - x_{min}}{w}$$

- 14: Create a line in the YOLO annotation file in the format:

$$\langle \text{class} \rangle \langle x_{center} \rangle \langle y_{center} \rangle \langle \text{normalized-width} \rangle \langle \text{normalized-height} \rangle$$

- 15: **end for**
 - 16: Repeat step 7 for all remaining bounding boxes.
 - 17: Save YOLO annotations.
-

Moreover, we created a YAML file comprising essential information, specifically the file paths for the training, validation, and testing sets and the number and names of classes. For experimentation, we subdivided the annotated data into training (70%), validation (20%), and testing (10%) sets.

3.2. Model Development

This section illustrates the background of the original YOLOv9 algorithm and describes the applied attention modules, particularly CBAM and ECA. Subsequently, a detailed explanation of the proposed GELAN+attention network including training details and evaluation measures will be given.

3.2.1. YOLOv9 Algorithm

Neural networks usually come across the challenge of information loss as the input data undergo numerous layers of feature extraction and spatial transformation, leading to the loss or degradation of the original information. Hence, to report this concern, the YOLOv9 model [37] uses PGI (programmable gradient information) and GELAN (generalized efficient layer aggregation network) for effective key feature extraction.

PGI is a supporting framework aimed at managing the propagation of gradient information across several semantic levels to enrich the detection performance of the network. PGI consists of three primary modules: (1) the main branch, (2) the auxiliary reversible branch, and (3) multilevel auxiliary information. During the inference stage, only the main branch is accountable for both forward- and backpropagation. As the network becomes deeper, an information bottleneck may occur, leading to loss functions that fail to produce valuable gradients. In such cases, reversible functions, employed by the auxiliary reversible branch, become active, preserving information integrity and minimizing information degradation in the main branch. Further, multilevel auxiliary information addresses the problem of errors commencing from the deep supervision process, improving the learning capacity of the model through the introduction of supplementary information at multiple levels.

During model training, to boost information integration and propagation efficiency, the YOLOv9 model presents an innovative lightweight network architecture called GELAN. It assimilates ELAN (efficient layer aggregation network) and CSPNet (cross-stage partial network) to effectively combine network information, thereby minimizing information loss during propagation and improving interlayer information communication. This architecture is well suited for hazard assessment with restricted computing resources because of its lesser parameter count and computational complexity. Additionally, based on the parameter count, four variants of YOLOv9 are released, namely, v9-S, v9-M, v9-C, and v9-E, pretrained on the MS-COCO dataset, providing a balance between speed and accuracy. However, considering the size of the database and the available computational resources, this study concentrates on GELANc, GELANe, YOLOv9c, and YOLOv9e only.

3.2.2. CBAM

CBAM [38], an effective attention-based module, can be integrated into neural networks with a remarkably low parameter count. As illustrated in Figure 3, CBAM consists of a channel attention module (CAM) and a spatial attention module (SAM). To generate feature maps by multiplying weights for adaptive refinement, CBAM computes attention weights across two dimensions. CAM executes two vital operations: global average pooling (GAP) and maximum global pooling, while SAM executes maximum and average pooling. This dual-module framework aids the estimation of attention weights for the precise refinement of feature maps. In GAP, each feature map of the convolutional layer is spatially averaged to produce a single value. Unlike traditional pooling operations like max pooling, which select the maximum value within a region, GAP computes the average value across the entire feature map. This process encompasses three stages: (1) Spatial average calculation, where the values of all the neurons are averaged together for each feature map in the convolutional layer. (2) The spatial average operation, which reduces the spatial dimensions of the feature maps to a single value per channel. (3) Channel-wise aggregation, where the averaged values from each channel are concatenated to form the output of the pooling operation. GAP is generally applied at the end of the convolutional layers in a CNN, before the fully connected layers. GAP also introduces translation invariance, making the network less sensitive to small spatial variations in the input and enhancing the network's performance in challenging tasks like landslide detection.

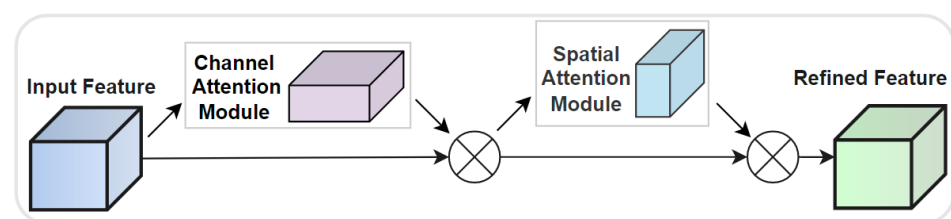


Figure 3. Architecture of CBAM.

Through convolution and pooling operations, CBAM derives the 1D channel attention $X_{Ch_A} \in \mathbb{R}^{C \times 1 \times 1}$ and 2D spatial attention $X_{Sp_A} \in \mathbb{R}^{1 \times H \times W}$ maps, given an input feature map $Z \in \mathbb{R}^{C \times H \times W}$. The inclusive attention is depicted by Equations (1) and (2).

$$Z' = X_{Ch_A}(Z) \otimes Z \quad (1)$$

$$Z'' = M_{Sp_A}(Z') \otimes Z' \quad (2)$$

where \otimes signifies element-wise multiplication and Z'' denotes the refined output.

Moreover, Equations (3) and (4) define the weights of CAM and SAM, respectively.

$$CAM_{Weights} = \sigma\left(\text{Multilayer}_{\text{perceptron}}\left(\text{Max}_{\text{pooling}}(Z)\right) + \text{Multilayer}_{\text{perceptron}}\left(\text{Average}_{\text{pooling}}(Z)\right)\right) \quad (3)$$

$$SAM_{Weights} = \sigma\left(\text{Convolution}\left(\text{concatenation}\left(\text{Max}_{\text{pooling}}(Z), \left(\text{Average}_{\text{pooling}}(Z)\right)\right)\right)\right) \quad (4)$$

3.2.3. ECA Network

Efficient channel attention (ECA) [39] is a mechanism designed to enhance the representation power of CNNs by focusing on important channels within feature maps. It aims to capture long-range dependencies between channels while maintaining computational efficiency, making it suitable for real-time applications such as landslide event extraction employing large-scale datasets. ECA, a lightweight framework related to SENet [39], readily integrates into CNNs to streamline the model's complexity while preserving the original dimensionality.

Initially, the input feature map is transformed using a 1D convolutional layer. This transformation enables the network to learn channel-wise relationships. Further, the transformed feature map is then passed through a GAP layer, capturing the global context of each channel to recalibrate the feature map. This is achieved by applying a sigmoid activation function followed by element-wise multiplication with the original feature map. The sigmoid function scales the channel-wise importance scores between 0 and 1, effectively emphasizing informative channels while suppressing less relevant ones.

To ensure computational efficiency, ECA employs a kernel-wise operation, where each channel is processed independently. This reduces the computational overhead compared to traditional attention mechanisms that consider pairwise interactions between channels. The convolutional layers' output forms a 4D tensor, which serves as input to the ECA network, encompassing four features, namely, the number of channels (Ch), height (He), width (Wi), and batch size (Bs). Similarly, the output of ECA-Net remains a 4D tensor. Its architecture consists of three modules: (1) the global feature descriptor, (2) adaptive neighborhood interaction, and (3) broadcast scaling. The GAP (global average pooling) method involves processing the input tensor by calculating the average of all pixels within an explicit feature map, followed by a reduction to a single pixel. Afterward, a tensor represented as $C \times 1 \times 1$ endures a 1D striding convolution. Further, the adaptive estimation of kernel size K_{Size} is estimated by Equations (5) and (6), where γ and 'X' indicate the predefined hyperparameters. The architecture of the ECA model is given in Figure 4.

$$Z'' = M_{Sp_A}(Z') \otimes Z' \quad (5)$$

$$Ch = \mathcal{O}(K_{Size}) = 2^{(\gamma \cdot K_{Size} - X)} \quad (6)$$

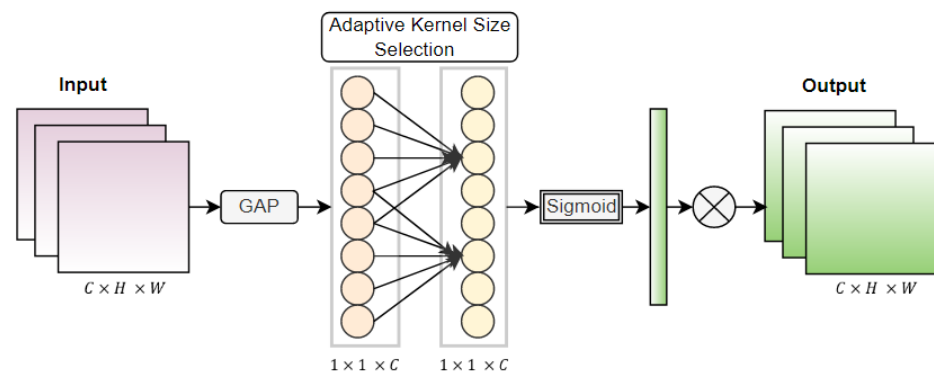


Figure 4. Architecture of ECA model.

3.2.4. Proposed Architecture

The proposed architecture encompasses the backbone and head. The backbone includes a series of convolutional layers that progressively extract hierarchical features from the input data. These convolutional layers allow the model to capture complex patterns and relationships within the data. These layers apply convolutional filters to the input, learning spatial patterns and extracting features. The first layer has a -1 input channel and outputs 64 channels (indicating the layer will generate 64 feature maps), with a kernel size of 3×3 and a stride of 2 for downsampling. The output feature maps from the first convolutional layer (with 64 channels) serve as an input to the second convolutional layer (with 128 channels). This process continues through the entire backbone. The architecture of the backbone is aimed to efficiently aggregate network information while minimizing information loss during propagation. This is achieved through the integration of specialized modules such as CSPNet and ELAN, resulting in RepNCSPPELAN, which enhances inter-layer information interaction and preserves information integrity. The output feature maps from the previous convolutional layer (with 128 channels) are fed into the RepNCSPPELAN block. Average pooling (a down layer) downsamples feature maps spatially, reducing their resolution while preserving some spatial information to create additional feature maps at different scales. Later, the pooled features are processed by convolution, resulting in further downsampled feature maps (with 256 channels). Hence, the backbone progressively reduces the spatial resolution of the feature maps while extracting increasingly complex features. The backbone follows a similar pattern for the remaining layers, and the number of output channels typically increases as the network progresses (256, 512, 512). This allows the model to capture both fine-grained and high-level information from the input image, which is crucial for accurate landslide detection.

Further, the neck receives the feature maps extracted by the backbone and refines them for landslide detection. The SPPELAN block performs spatial pyramid pooling (SPP), which partitions the input feature maps into grids of different sizes and extracts features from each region. This allows the network to capture features at different scales within the input. The upsampled features are concatenated with the features from the backbone, creating a richer representation at a higher resolution. Concatenation combines information from different scales, allowing the network to leverage both high-level semantic information and low-level spatial details for landslide detection.

The concatenated features are processed by the RepNCSPPELAN blocks (similar to those in the backbone), potentially involving group normalization, channel splitting, and residual connections. Further, these features are enhanced by the attention module. Attention, a fundamental cognitive ability of human beings, enables them to prioritize relevant information while filtering out irrelevant details. Inspired by this aspect of human vision, attention modules show significant potential when integrated within a CNN to enhance its performance. Therefore, we deliberately integrated two substantial attention modules, CBAM and ECA, within GELAN. Particularly, an attention module is added in the neck after each RepNCSPPELAN layer. The architecture of the head follows repeated upsampling,

concatenation, RepNCSPeLAN block, attention to the remaining layers, and processing features at different scales. Figure 5 represents the algorithmic framework of the proposed GELAN+attention model. The addition of the attention module enhances feature representation as it allows the model to dynamically focus on the most relevant features within the input data. By assigning higher weights to important features and suppressing less relevant ones, the attention mechanism boosts the representation of key information, leading to more discriminative feature maps. Further, the attention layer facilitates efficient integration of information across different layers of the network. By selectively attending to relevant features from multiple layers, the model can effectively combine information from various levels of abstraction, leading to richer and more informative representations. The attention mechanism enables the model to adaptively learn which features to prioritize based on the input data. This characteristic allows the model to selectively consider different parts of the input, depending on their importance to the task, leading to more flexible and adaptive feature learning. The attention mechanism highlights the regions of the input that are most influential for the predictions, enabling increased interpretability and supporting model debugging and refinement. Integrating attention models within GELAN empowers the network to become more robust towards input data, consequently improving its ability to generalize across different conditions and environments.

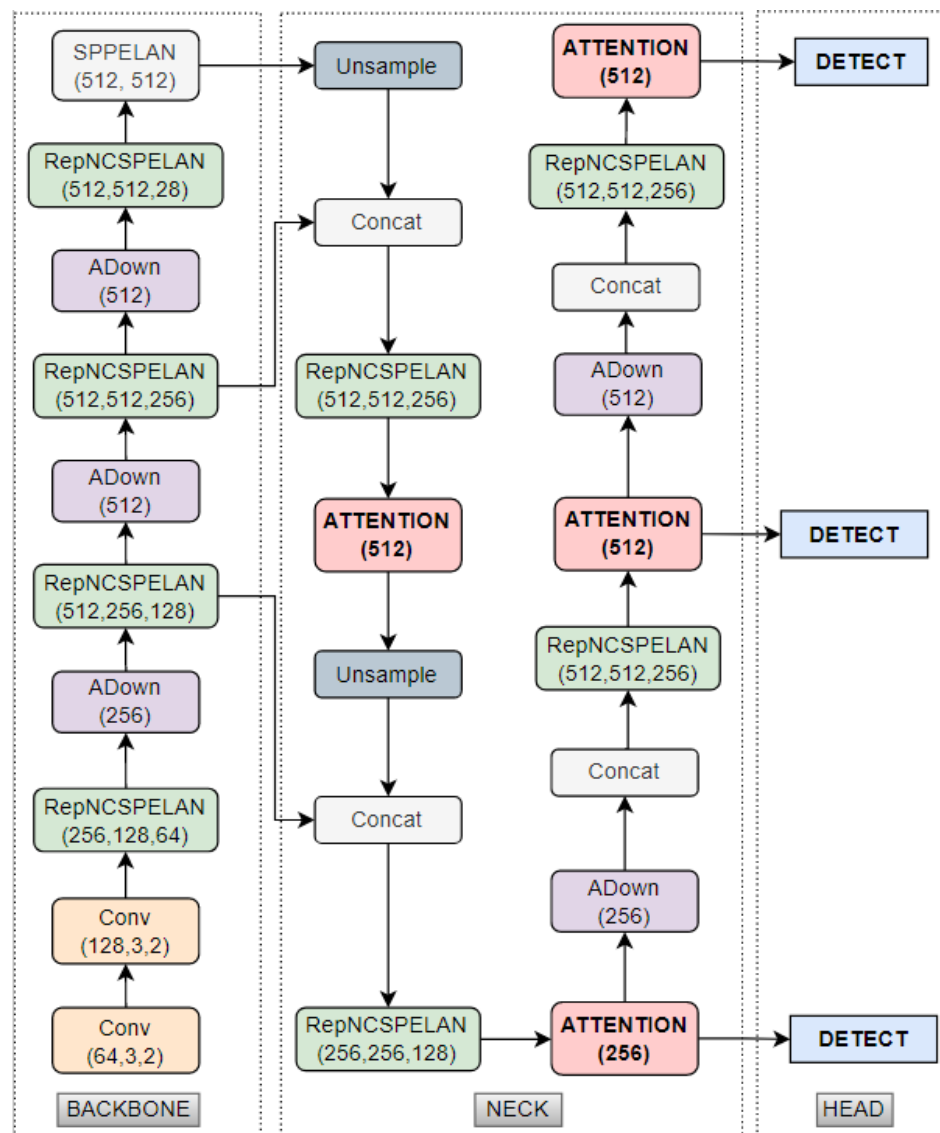


Figure 5. Schematic representation of the proposed architecture.

Lastly, the detection layer performs the complex task of landslide detection. It takes the processed features and predicts the bounding boxes for potential objects in each grid cell of the feature maps and the class probabilities for each bounding box. The proposed architecture improves accuracy and facilitates real-time landslide information extraction from remotely sensed data.

3.3. Experimental Settings and Evaluation Measures

For experimentation, an image size of 416×416 pixels is chosen as the input to the network. The architecture is optimized to effectively process images of the selected size, capturing intricate details in the satellite image and ensuring efficient processing by the network. The batch size is set to 4 due to memory constraints, allowing for efficient training. Although the aim is to train the model for a maximum of 500 epochs, early stopping criteria are adopted to avoid inefficient utilization of computing resources and time. For optimization, the training units utilize a stochastic gradient descent (SGD) optimizer that uniformly scales gradients in all directions and facilitates faster convergence by collecting gradients from previous data points. Furthermore, GELAN uses an SGD optimizer by default, which is therefore employed in this research work. The initial learning rate is set at 0.01, with an initial momentum factor of 0.937 and an initial weight decay of 0.0005. These parameters are chosen to ensure effective training and optimization of the model. The training environment comprises an Intel® Xeon-CPU E3-1231 v3 @ 3.40 GHz CPU and NVIDIA GeForce RTX 3080Ti GPU running on the Ubuntu 18.04 operating system. The deep learning frameworks used are PyTorch 1.7 and CUDA 11.4. The model's performance evaluation encompasses both quantitative and qualitative assessments. For quantitative evaluation, four metrics, precision, recall, f-score, and mean average precision (mAP), are computed using Equations (7)–(10). Simultaneously, qualitative evaluation involves visual analysis of the outcomes to assess the model's capability.

$$\text{Precision} = \frac{\text{True positive}}{\text{True positive} + \text{False positive}} \quad (7)$$

$$\text{Recall} = \frac{\text{True positive}}{\text{True positive} + \text{False negative}} \quad (8)$$

$$\text{F-Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (9)$$

$$\text{mAP} = \sum_{i=0}^n \left(\frac{1}{n} \int_0^1 \text{Precision}(\text{Recall}) d(\text{Recall}) \right) \quad (10)$$

4. Results

This section presents the experimental findings of this study. Initially, quantitative outcomes based on the metrics outlined in Section 3.3 are provided. Subsequently, the qualitative or visual representations of the detected landslide events by the proposed model are described.

4.1. Quantitative Evaluation

The outcomes of quantitative measures, specifically f-score, precision, recall, and mAP, of each attention-induced GELAN are given in Table 1. The computed f-score of GELANe and GELANc is 78.0% and 79.4%, respectively. Integration of the ECA module within GELANe showed an improvement of +2.6%. Further, the f-score significantly increased by +2.8% considering CBAM. In addition, an enhancement of +1.8% and +2.1% is noted when the ECA model and CBAM are added within the architecture of GELANc, respectively. The GELANc+CBAM outperformed other models and demonstrated the highest accuracy (81.5%).

Table 1. Experimental results of the proposed model.

Model	GFLOP	Params	P	R	F-Score	mAP@50	Time (h)	GPU (GBs)
GELANe	190.8	58.00	88.3	69.9	78.0	75.2	1.814	3.74
GELANe + ECA	190.8	58.00	86.4	75.5	80.6	76.2	1.973	3.76
GELANe + CBAM	191.5	58.85	84.2	77.6	80.8	75.0	1.869	4.06
GELANc	102.5	25.41	86.3	73.6	79.4	78.3	1.037	1.95
GELANc + ECA	102.5	25.41	90.2	73.8	81.2	77.8	1.027	1.95
GELANc + CBAM	120.6	31.06	85.7	77.7	81.5	75.2	1.301	2.15

A difference of 3.3% is noted between the maximum (GELANc) and minimum (GELANe + CBAM) mAP@0.50, while it is inferred to be equivalent (75.2%) for GELANe and GELANc + CBAM. Moreover, the precision reached 90.2% (GELANc + ECA), and recall is estimated between 66.9% (GELANe) and 77.7% (GELANc + CBAM, almost similar to GELANe + CBAM). A minor deviation (+0.2%) is observed in the recall of GELANc and GELANc + ECA. Furthermore, the computational time (in hours (h)) is an important assessment factor and is hence considered in this research work. Table 1 shows the execution time of each model, and it is inferred that GELANe + ECA exhibits the maximum execution time (1.973 h), while GELANc + ECA requires the minimum time (1.027 h). In addition, the amount of GPU consumed (1.95 GB–4.06 GB) by each model is indicated. Moreover, Table 1 illustrates the computed GFLOPS (giga floating-point operations per second), with the highest noted at 190.8 (GELANe) and the lowest at 102.5 (GELANc). GFLOPS measures the computational complexity of a model, indicating the number of floating-point operations (FLOPs) required to process a single forward pass of the model, expressed in billions (giga) of operations. The FLOPs signify the total number of multiplications and additions required by all the operations in the model. GFLOPS represents an estimate of the computational resources needed to run the model. Params refers to the number of trainable parameters in the model. These include weights (coefficients that are learned from the data during training) and biases (additional parameters in each layer that shift the activation function) across all layers in the network. In our experiments, the maximum (58.85 million (m)) and minimum (25.41 m) Params are computed by GELANe+CBAM and GELANc, respectively. Hence, GFLOPS and Params are critical factors for evaluating the model's performance as they provide insights into the computational and memory requirements, influencing the choice of model for specific applications.

Additionally, Figures 6 and 7 illustrate the estimated training and validation loss functions, respectively. Particularly the bounding box regression loss (box loss), distributional focal loss (dfl loss), and classification loss are computed for the suggested networks (GELANe + ECA, GELANe + CBAM, GELANc + ECA, and GELANc + CBAM). The x axis represents the number of epochs, whereas the y axis indicates the computed loss values during the training and validation stages.

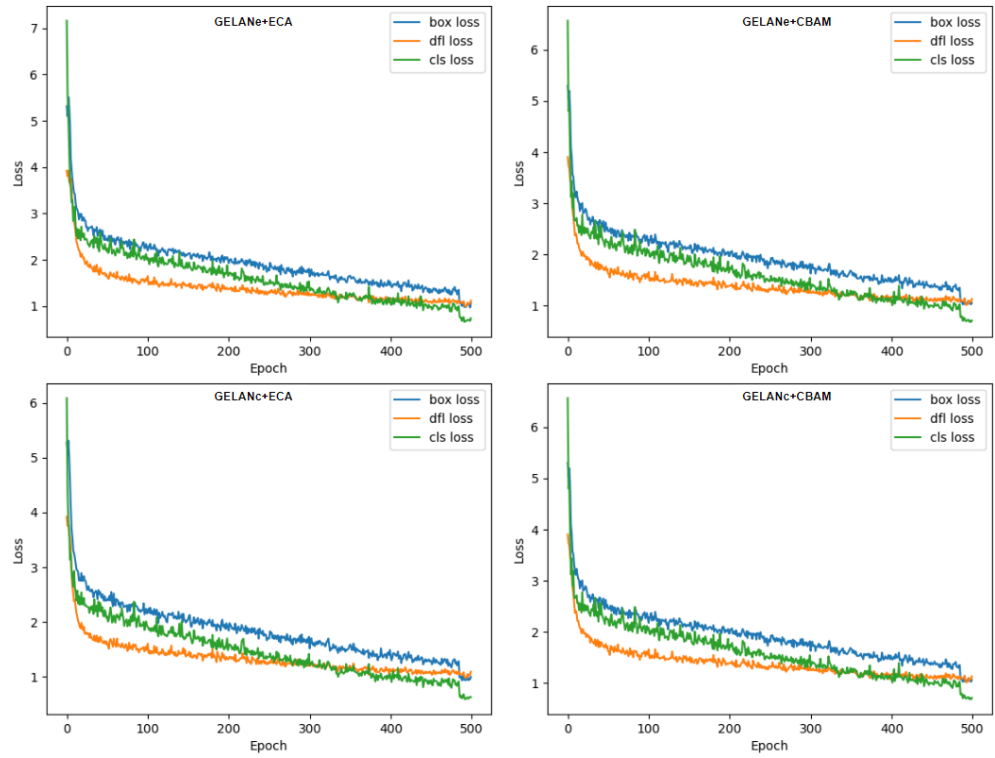


Figure 6. Results of the training loss function by the GELAN + attention model.

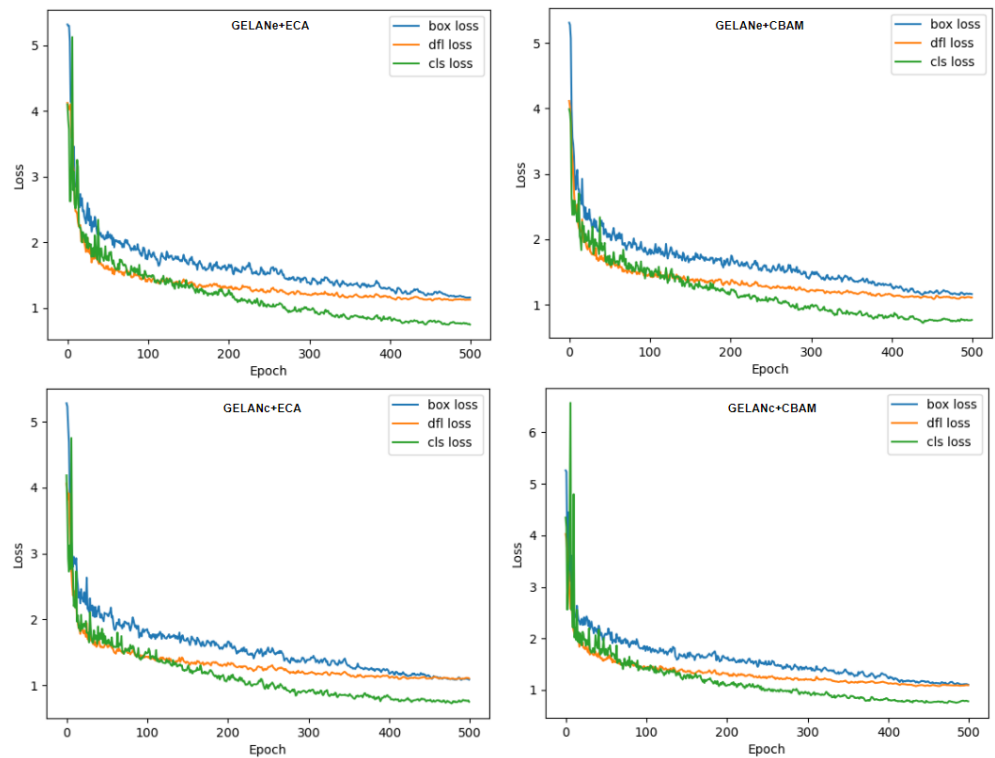


Figure 7. Results of the validation loss function by the GELAN + attention model.

4.2. Qualitative Evaluation

Visual analysis of the detected landslide by the proposed model is essential for validating model accuracy, interpreting results, and applying the findings in practical scenarios. It enhances the model’s reliability, supports effective communication and decision making, and is critical in both immediate disaster response and long-term risk mitigation strategies.

The overall impact of this study is significantly enhanced by integrating visual analysis into the workflow. Presenting the results as image patches offers significant advantages. Image patches enable a detailed and focused examination of specific areas, enhancing the clarity and resolution of the detected landslides. The results facilitate precise comparisons, highlight specific sites, and support communication efficiency with stakeholders by providing a concise and targeted representation of the results. Additionally, handling smaller patches is computationally efficient and aligns with the technical requirements of machine learning model training and validation. Therefore, using image patches alongside full-area visualizations provides a robust and comprehensive approach to landslide detection analysis. The YOLO models create the labels before target prediction. Figure 8 represents the generated labels for detecting landslides. The visual representation of the identified landslides by GELANc + CBAM and GELANe + CBAM are depicted through bounding boxes in Figure 8. The dataset comprises images with complex backgrounds/landscapes, along with cloud cover, where the manual interpretation is difficult and poses a challenge for landslide recognition; yet, the model effectively predicts them. The proposed algorithm demonstrates the ability to distinguish between landslides of varying extents. Due to the high detection correctness, the model proved its robustness, subsequently yielding enhanced prediction outcomes.

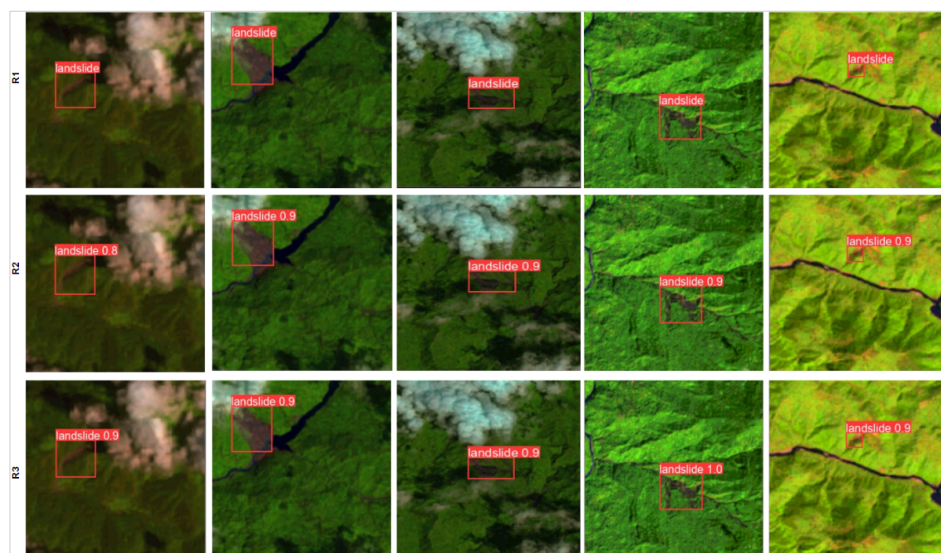


Figure 8. Detected landslides by the proposed method: Labels (R1), GELANc + CBAM (R2), GELANe + ECA (R3).

5. Discussion

The proposed deep learning-based algorithm identifies the presence of landslides successfully. However, to infer its suitability, we considered the additional benchmark dataset. Moreover, we examined the effectiveness of integrating an additional attention-based model into GELAN to assess its capabilities. Subsequently, we compared our results with previously proposed studies and examined the performance of state-of-the-art YOLO variants for recognizing landslides. Additionally, we underscore the importance of the patience factor. Finally, the limitations, prospects, and applications of this study are outlined.

5.1. Model's Performance on Benchmark Dataset

To assess the effectiveness of the suggested model, we included an additional dataset with geological and geomorphological characteristics distinct from those of the Himalayan terrain, specifically the popular benchmark Bijie landslide detection database. The quantitative outcomes (f-score, precision, and recall) are shown in Table 2. A competitive f-score is achieved for GELANc (98.0%) and GELANe (98.5%). By integrating ECA within GELANc (improvement = +1.5%) and GELANe (improvement = +1.0%), the computed

f-score reached 99.5%. Similarly, GELANe+CBAM (99.0%) and GELANc + CBAM (98.0%) demonstrated a challenging f-score. Recently, Du et al., 2024 [29] and Yang et al., 2024 [33] employed a similar dataset for their experiments based on YOLO models. Comparing our results with them (considering the average f-score), we noted a substantial improvement of +5.15% compared to Du et al., 2024 [29] and a remarkable progress of +10.25% in the case of Yang et al., 2024 [33]. Moreover, a noteworthy improvement of +19.75% is estimated when compared with the results of Tanatipuknon et al., 2021 [40]. This comparison unveils the novelty of the proposed algorithms.

Table 2. Results of the additional dataset (Bijie landslide detection database).

Reference	Model	F-Score (%)	P (%)	R (%)
Our work	GELANe	98.5	97.9	99.1
	GELANe + ECA	99.5	99.9	99.1
	GELANe + CBAM	99.0	98.1	99.9
	GELANc	98.0	98.1	98.0
	GELANc + ECA	99.5	99.9	99.1
	GELANc + CBAM	98.0	98.0	98.1
Tanatipuknon et al., 2021 [40]	Faster R-CNN	79.0	86.0	77.0
Du et al., 2024 [29]	LBE_YOLO	88.5	90.6	86.5
	YOLOv4 + VGG16	94.0	96.1	91.9
Yang et al., 2024 [33]	YOLOv4 + ECA	92.4	93.5	91.4
	YOLOv4 + CBAM	93.1	94.0	92.2
	YOLOv4 + LPRF	94.9	95.5	94.2

5.2. Additional Attention Model

We further explored an additional attention model to identify the most suitable one for landslide identification. Particularly, the global attention mechanism (GAM) [41], a widely recognized model, enhances the utility of global dimension-interactive features to reduce information dispersion. GAM innovatively transforms the channel and spatial components of CBAM. GELANc + GAM (f-score = 76.7%, mAP@0.50 = 73.2%) demonstrated better performance when compared with GELANe + GAM (f-score = 70.3%, mAP@0.50 = 67.4%). The recall computed by GELANc + GAM and GELANe + GAM is 71.8% and 60.2%, respectively. However, the estimated precision of GELANe + GAM is estimated to be higher (+2.2 units) than GELANc + GAM (P = 82.4%). Further, the computational efficiency of GELANc + GAM (time = 1.301 h, memory = 2.15 GB, GFLOPS = 120.6, and Params = 31.06) is also noted to be better than GELANe + GAM (time = 2.095 h, memory = 4.21 GB, GFLOPS = 208.9, and Params = 63.65). In addition, YOLOv9-C+GAM revealed a competitive f-score (77.5%) and mAP@0.50 (74.3%). The accurateness computed by the GAM is assessed to be less than the CBAM and ECA modules.

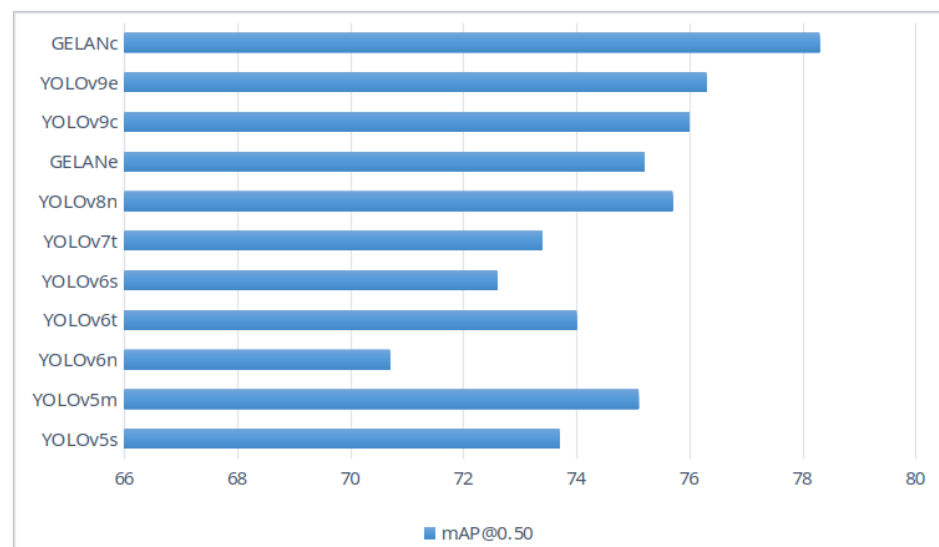
5.3. Comparative Analysis

To exhibit the effectiveness of the attention mechanism integrated into GELAN, the results are compared with previously proposed methods for landslide detection from the Nepal region, particularly Bragagnolo et al., 2021 [34], Chen et al., 2018 [36], and Yu et al., 2020 [35]. Comparison and evaluation are based on computed precision, recall, and f-score of our study's best model. Table 3 indicates the notable improvement in f-score (+14.5% to +55.5%), precision (+24.7 to +55.7 %), and recall (+3.7% to +54.7%), thus affirming its novelty and efficacy.

Table 3. Comparison of our model with previous studies.

Reference	F-Score (%)	P (%)	R (%)
GELANc + CBAM (our work)	81.5	85.7	77.7
Bragagnolo et al., 2021 [34]	67.0	61.0	74.0
Improvement	+14.5	+24.7	+3.7
Yu et al., 2020 [35]	60.0	55.0	65.0
Improvement	+21.5	+30.7	+12.7
Chen et al., 2018 [36]	26.0	30.0	23.0
Improvement	+55.5	+55.7	+54.7

We also evaluated the performance of YOLOv9 with seven algorithms, specifically YOLOv5 small (s), YOLOv5 medium (m), YOLOv6 nano (n), YOLOv6s, YOLOv6m, YOLOv7 tiny (t), and YOLOv8n. The obtained mAP at the intersection over the union threshold of 0.5 (mAP@0.50) for all models is depicted in Figure 9. Notably, GELANc achieved the highest mAP@0.50 at 78.3% followed by YOLOv9e (76.3%), which is almost equivalent to YOLOv9e (76.0%). In addition, the mAP@0.50 of GELANe reached 75.2%. The computed mAP@0.50 of the considered YOLO variants for comparison is significantly lower (+2.6 to +4.5) than YOLOv9. However, the marginal difference (0.5 units) is noted in the case of YOLOv8n and GELANe. This comparison highlights the effectiveness of the YOLOv9 network.

**Figure 9.** Comparison of YOLOv9 with the state-of-the-art algorithms.

The results of the proposed model (GELANe + ECA, GELANe + CBAM, GELANc + ECA, GELANc + CBAM) are compared with state-of-the-art models, particularly YOLOv6n, YOLOv6t, YOLOv6s, and YOLOv7t, considering mAP@0.50–0.95 matrices (refer to Figure 10). The maximum improvement of +7.3% is noted in the case of GELANc+ECA when compared with YOLOv6n. In addition, the progress of +6.6% and +6.1% is estimated when compared with YOLOv6s and YOLOv6t, respectively. Further, the mAP@0.50–0.95 improved by +3.4% in comparison with YOLOv7t. GELANc+CBAM shows an improvement of +4.9% over YOLOv6n, and the enhancements are +4.2%, +3.7%, and +1.0% when compared to YOLOv6s, YOLOv6t, and YOLOv7t, respectively. Additionally, GELANe+ECA improves mAP@0.50–0.95 by +4.4% over YOLOv6n. The improvements are +3.7%, +3.2%, and +0.5% compared to YOLOv6s, YOLOv6t, and YOLOv7t, respectively. Furthermore, enrichment of +3.1%, +2.4%, and +1.9% is noted when the mAP@0.50–0.95 of GELANe+CBAM is compared with YOLOv6n, YOLOv6s, and YOLOv6t, respectively, while a marginal difference (0.8%) is observed between the mAP@0.50–0.95 of

GELANc+CBAM and YOLOv7t. These comparisons highlight the significant enhancements in mAP@0.50–0.95 achieved by the proposed models, demonstrating their superior performance over the popular object detection models.

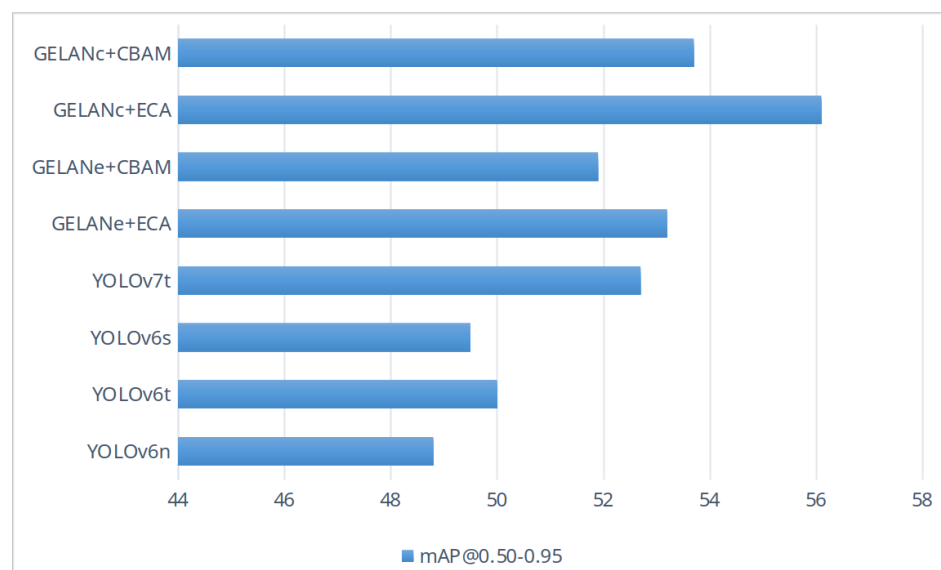


Figure 10. Comparison of the proposed method with the state-of-the-art models.

5.4. Significance of Early Stopping Factor

During the execution and implementation of YOLO algorithms, early stopping is a technique utilized to improve the efficiency of the network. It involves continuously monitoring evaluation metrics during training and halting the training process if no improvement in accuracy is observed after a certain number of epochs, known as the patience parameter. Patience is a user-defined integer that specifies the number of iterations or epochs the training process can endure without any advancement in evaluation metrics before terminating early. This supports maintaining a balance between training time and network capability. Choosing a suitable patience value is crucial for optimizing outcomes. Setting the patience value too low may prematurely end model training, potentially missing out on performance improvements. On the other hand, setting it too high can unnecessarily prolong training, resulting in ineffective use of computational resources/hardware and time. There is no standard formula for determining the optimal patience value; it relies on factors such as dataset, model complexity, speed, and machine configuration. Finding the optimal patience value requires trial experimentation and fine-tuning. The default patience value for the YOLOv9 network is 100, as utilized in this research work (patience = 100). Patience parameters are flexible and can be disabled (set patience = 0) and enabled easily. In our experiments, only GELANe terminated early (at 468 epochs), saving the best results at 367 epochs, and the other five models (GELANc, GELANe + ECA, GELANc + ECA, GELANe + CBAM, and GELANc + CBAM) were trained for the defined number of epochs.

5.5. Limitations, Prospects, and Applications

The development of intelligent and fully automated landslide event recognition models by deep learning algorithms is still ongoing, presenting opportunities for advancement. Hence, we suggest an attention-based GELAN for detecting landslide hazards of varying sizes from complex backgrounds. We explore the implications of different attention modules and compare their performance with state-of-the-art YOLO models to showcase their detection capabilities. However, our study is constrained by the dataset and computational resources. The experimentation database comprises a limited number of satellite images, which impacts the accuracy of the suggested network. Additionally, the applied deep learning models are executed on a machine with limited GPU access. For future research

work, we aim to enhance the accuracy of the model network by increasing the database size. Furthermore, the model can be evaluated using unmanned aerial vehicle images. The proposed approach holds the potential for quick and accurate mapping of landslide scenes, supporting effective rescue and recovery actions. Additionally, the findings of this study can be valuable in time-constrained landslide mapping for up-to-date inventory generation. Moreover, accurately identifying the spatial characteristics of landslides is essential for developing a landslide forecasting system.

6. Conclusions

This research work introduces an attention-embedded GELAN algorithm for detecting landslide events. Specifically, CBAM and ECA are discretely integrated into the neck of the original GELAN architecture. The proposed algorithm addresses the challenges associated with the correct landslide information extraction from fragile and complex landscapes, mainly the Himalayas. The results demonstrate that GELANc + CBAM yields the best performance. Overall, the findings are novel and encouraging, indicating the model's suitability for prompt hazard mapping to facilitate fast recovery and response efforts. Future research directions may explore the perspective of GELAN and attention models considering unmanned aerial vehicle-based images for landslide identification. This study offers an advanced solution to overcome the challenges allied with near-real-time landslide hazard mapping.

Author Contributions: N.C.: conceptualization, model execution, and manuscript writing; H.V.: implementation and manuscript editing; S.S.: manuscript editing; and S.R.M.: manuscript writing and editing. All authors have read and agreed to the published version of the manuscript.

Funding: This research work is supported by the Department of Science and Technology, Science and Engineering Research Board, New Delhi, India, under Grant No: EEQ/2022/000812.

Data Availability Statement: The data presented in this study can be downloaded from <https://zenodo.org/record/3675410#.ZBv3UZBxD8> and http://gpcv.whu.edu.cn/data/Bijie_pages.html.

Acknowledgments: The authors would like to thank the directors of the Wadia Institute of Himalayan Geology (WIHG), Dehradun, for their continuous encouragement. The contribution number for this research work is WIHG/0372.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Casagli, N.; Intrieri, E.; Tofani, V.; Gigli, G.; Raspini, F. Landslide detection, monitoring and prediction with remote-sensing techniques. *Nat. Rev. Earth Environ.* **2023**, *4*, 51–64. [CrossRef]
2. Cheng, L.; Li, J.; Duan, P.; Wang, M. A small attentional YOLO model for landslide detection from satellite remote sensing images. *Landslides* **2021**, *18*, 8. [CrossRef]
3. Mohan, A.; Singh, A.K.; Kumar, B.; Dwivedi, R. Review on remote sensing methods for landslide detection using machine and deep learning. *Trans. Emerg. Telecommun. Technol.* **2021**, *32*, e3998. [CrossRef]
4. Ma, Z.; Mei, G. Machine Learning for Landslides Prevention: A Survey. *Neural Comput. Appl.* **2020**, *30*, 10881–10907. [CrossRef]
5. Xu, Y.; Ouyang, C.; Xu, Q.; Wang, D.; Zhao, B.; Luo, Y. CAS Landslide Dataset: A Large-Scale and Multisensor Dataset for Deep Learning-Based Landslide Detection. *Sci. Data* **2024**, *11*, 12. [CrossRef] [PubMed]
6. Ma, Z.; Mei, G. Deep learning for geological hazards analysis: Data, models, applications, and opportunities. *Earth-Sci. Rev.* **2021**, *223*, 103858. [CrossRef]
7. Liu, P.; Wei, Y.; Wang, Q.; Xie, J.; Chen, Y.; Li, Z.; Zhou, H. A Research on Landslides Automatic Extraction Model Based on the Improved Mask R-CNN. *Isprs Int. J. Geo-Inf.* **2021**, *10*, 168. [CrossRef]
8. Han, Z.; Fang, Z.; Li, Y.; Fu, B. A novel Dynahead-Yolo neural network for the detection of landslides with variable proportions using remote sensing images. *Front. Earth Sci.* **2023**, *10*, 1077153. [CrossRef]
9. Lu, P.; Stumpf, A.; Kerle, N.; Casagli, N. Object-Oriented Change Detection for Landslide Rapid Mapping. *IEEE Geosci. Remote Sens. Lett.* **2011**, *8*, 4. [CrossRef]
10. Blaschke, T.; Feizizadeh, B.; Holbling, D. Object-Based Image Analysis and Digital Terrain Analysis for Locating Landslides in the Urmia Lake Basin, Iran. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2014**, *7*, 4806–4817. [CrossRef]
11. Keyport, R.N.; Oommen, T.; Martha, T.R.; Sajinkumar, K.S.; Gierke, J.S. A comparative analysis of pixel- and object-based detection of landslides from very high-resolution images. *Int. J. Appl. Earth Obs. Geoinf.* **2018**, *64*, 1–11. [CrossRef]

12. Ma, Y.; Wu, H.; Wang, L.; Huang, B.; Ranjan, R.; Zomaya, A.; Jie, W. Remote sensing big data computing: Challenges and opportunities. *Future Gener. Comput. Syst.* **2015**, *51*, 47–60. [CrossRef]
13. Zhang, X.; Zhou, Y.; Luo, J. Deep learning for processing and analysis of remote sensing big data: A technical review. *Big Earth Data* **2021**, *6*, 527–560. [CrossRef]
14. Lombardo, L.; Mai, P.M. Presenting logistic regression-based landslide susceptibility results. *Eng. Geol.* **2018**, *244*, 14–24. [CrossRef]
15. Huang, Y.; Zhao, L. Review on landslide susceptibility mapping using support vector machines. *Catena* **2018**, *165*, 520–529. [CrossRef]
16. Hu, Q.; Zhou, Y.; Wang, S.; Wang, F.; Wang, H. Improving the Accuracy of Landslide Detection in “Off-site” Area by Machine Learning Model Portability Comparison: A Case Study of Jiuzhaigou Earthquake, China. *Remote Sens.* **2019**, *11*, 2530. [CrossRef]
17. Fang, Z.; Wang, Y.; Duan, G.; Peng, L. Landslide Susceptibility Mapping Using Rotation Forest Ensemble Technique with Different Decision Trees in the Three Gorges Reservoir Area, China. *Remote Sens.* **2021**, *13*, 238. [CrossRef]
18. Zhao, Z.-Q.; Zheng, P.; Xu, S.; Wu, X. Object Detection with Deep Learning: A Review. *arXiv* **2019**, arXiv:1807.05511. Available online: <http://arxiv.org/abs/1807.05511> (accessed on 7 July 2024).
19. Minaee, S.; Boykov, Y.Y.; Porikli, F.; Plaza, A.J.; Kehtarnavaz, N.; Terzopoulos, D. Image Segmentation Using Deep Learning: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 3523–3542. [CrossRef]
20. Zeng, D.; Liao, M.; Tavakolian, M.; Guo, Y.; Zhou, B.; Hu, D.; Pietikäinen, M.; Liu, L. Deep Learning for Scene Classification: A Survey. *arXiv* **2021**, arXiv:2101.10531. Available online: <http://arxiv.org/abs/2101.10531> (accessed on 7 July 2024).
21. Ullo, S.; Mohan, A.; Sebastianelli, A.; Ahamed, S.; Kumar, B.; Dwivedi, R.; Sinha, G.R. A New Mask R-CNN-Based Method for Improved Landslide Detection. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 3799–3810. [CrossRef]
22. Devara, M.; Maurya, V.K.; Dwivedi, R. Landslide extraction using a novel empirical method and binary semantic segmentation U-NET framework using sentinel-2 imagery. *Remote Sens. Lett.* **2024**, *15*, 326–338. [CrossRef]
23. Ghorbanzadeh, O.; Crivellari, A.; Ghamisi, P.; Shahabi, H.; Blaschke, T. A comprehensive transferability evaluation of U-Net and ResU-Net for landslide detection from Sentinel-2 data (case study areas from Taiwan, China, and Japan). *Sci. Rep.* **2021**, *11*, 14629. [CrossRef] [PubMed]
24. Meena, S.R.; Nava, L.; Bhuyan, K.; Puliero, S.; Soares, L.P.; Dias, H.C.; Floris, M.; Catani, F. HR-GLDD: A globally distributed dataset using generalized DL for rapid landslide mapping on HR satellite imagery. *Earth Syst. Sci. Data Discuss.* **2022**, *15*, 3283–3298. [CrossRef]
25. Diwan, T.; Anirudh, G.; Tembhurne, J.V. Object detection using YOLO: Challenges, architectural successors, datasets and applications. *Multimed. Tools Appl.* **2023**, *82*, 9243–9275. [CrossRef] [PubMed]
26. Li, B.; Li, J. Methods for landslide detection based on lightweight YOLOv4 convolutional neural network. *Earth Sci. Inform.* **2022**, *15*, 765–775. [CrossRef]
27. Mo, P.; Li, D.; Liu, M.; Jia, J.; Chen, X. A Lightweight and Partitioned CNN Algorithm for Multi-Landslide Detection in Remote Sensing Images. *Appl. Sci.* **2023**, *13*, 8583. [CrossRef]
28. Liu, Q.; Wu, T.; Deng, Y.; Liu, Z. SE-YOLOv7 Landslide Detection Algorithm Based on Attention Mechanism and Improved Loss Function. *Land* **2023**, *12*, 1522. [CrossRef]
29. Du, Y.; Xu, X.; He, X. Optimizing Geo-Hazard Response: LBE-YOLO’s Innovative Lightweight Framework for Enhanced Real-Time Landslide Detection and Risk Mitigation. *Remote Sens.* **2024**, *16*, 534. [CrossRef]
30. Niu, Z.; Zhong, G.; Yu, H. A review on the attention mechanism of deep learning. *Neurocomputing* **2021**, *452*, 48–62. [CrossRef]
31. Ji, S.; Yu, D.; Shen, C.; Li, W.; Xu, Q. Landslide detection from an open satellite imagery and digital elevation model dataset using attention boosted convolutional neural networks. *Landslides* **2020**, *17*, 1337–1352. [CrossRef]
32. Jia, L.; Leng, X.; Wang, X.; Nie, M. Recognizing landslides in remote sensing images based on enhancement of information in digital elevation models. *Remote Sens. Lett.* **2024**, *15*, 224–232. [CrossRef]
33. Yang, Y.; Miao, Z.; Zhang, H.; Wang, B.; Wu, L. Lightweight Attention-Guided YOLO with Level Set Layer for Landslide Detection from Optical Satellite Images. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2024**, *17*, 3543–3559. [CrossRef]
34. Bragagnolo, L.; Rezende, L.R.; da Silva, R.V.; Grzybowski, J.M.V. Convolutional neural networks applied to semantic segmentation of landslide scars. *Catena* **2021**, *201*, 105189. [CrossRef]
35. Yu, B.; Chen, F.; Xu, C. Landslide detection based on contour-based deep learning framework in case of national scale of Nepal in 2015. *Comput. Geosci.* **2020**, *135*, 104388. [CrossRef]
36. Chen, F.; Yu, B.; Li, B. A practical trial of landslide detection from single-temporal Landsat8 images using contour-based proposals and random forest: A case study of national Nepal. *Landslides* **2018**, *15*, 453–464. [CrossRef]
37. Wang, C.-Y.; Yeh, I.-H.; Liao, H.-Y.M. YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information. *arXiv* **2024**, arXiv:2402.13616. Available online: <http://arxiv.org/abs/2402.13616> (accessed on 7 July 2024).
38. Woo, S.; Park, J.; Lee, J.-Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. *arXiv* **2018**, arXiv:1807.06521. Available online: <http://arxiv.org/abs/1807.06521> (accessed on 7 July 2024).
39. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks. *arXiv* **2020**, arXiv:1910.03151. Available online: <http://arxiv.org/abs/1910.03151> (accessed on 7 July 2024).

40. Tanatipuknon, A.; Aimmanee, P.; Watanabe, Y.; Murata, K.T.; Wakai, A.; Sato, G.; Hung, H.V.; Tungpimolrut, K.; Keerativittayanun, S.; Karnjana, J. Study on Combining Two Faster R-CNN Models for Landslide Detection with a Classification Decision Tree to Improve the Detection Performance. *J. Disaster Res.* **2021**, *16*, 4. [[CrossRef](#)]
41. Liu, Y.; Shao, Z.; Hoffmann, N. Global Attention Mechanism: Retain Information to Enhance Channel-Spatial Interactions. *arXiv* **2021**, arXiv:2112.05561. Available online: <http://arxiv.org/abs/2112.05561> (accessed on 7 July 2024).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.