Journal of Theoretical and Applied Information Technology
15th April 2026. Vol.104. No.7

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

STRUCTURAL DEFENSE-ORIENTED GCR-SCAPS METHOD WITH SPARSE CAPSULE ENCODING AND CONSISTENCY-BASED LEARNING

HEMASHREE P[1] , N VALLIAMMAL [2]

[1]Research Scholar, Department of Computer Science, Avinashilingam Institute for Home Science and Higher Studies for Women, Coimbatore, India
[2]Associate Professor, Department of Computer Science, Avinashilingam Institute for Home Science and Higher Studies for Women, Coimbatore, India
E-mail: [1]shreehema256@gmail.com, [2]valliammal_cs@avinuty.ac.in

ABSTRACT

Adversarial perturbations compromise the structural reliability of remote sensing image classification, challenging the stability of decision-making systems in security-critical contexts. The proposed GCR-SCaps model introduces a Geometric Consistency Reinforced Sparse Capsule Network tailored to counter structural attacks through object-aware representations. Initial convolutional layers extract shallow spatial cues, which are then transformed into capsule vectors with enforced sparsity to promote discriminative feature isolation. Dynamic routing captures inter-capsule agreement, while final class capsules preserve semantic integrity. A parallel stream generates geometrically altered inputs to compute a consistency loss, aligning original and transformed predictions. This dual-path strategy ensures robustness against geometric distortions while maintaining classification fidelity. The joint optimization of structural margin and geometric consistency losses strengthens spatial reasoning, minimizing confusion caused by adversarial interference. GCR-SCaps establishes a structurally grounded defense architecture that enhances resilience, preserves spatial coherence, and ensures dependable classification performance across diverse aerial scenarios prone to manipulation or ambiguity.

Keywords: *Remote Sensing Classification, Capsule Networks, Geometric Consistency, Adversarial Defense, Sparse Routing, Structural Robustness*

1. INTRODUCTION

Adversarial attacks represent calculated manipulations applied to input data, designed to subtly alter the behavior of deep learning models. These attacks typically exploit model sensitivity by embedding minimal, visually imperceptible perturbations into the input, leading to misclassification or misleading outputs. Even though the changes introduced may be undetectable to human observers, their impact on the model's decision boundaries can be severe. Such attacks have been identified across multiple domains, including medical diagnosis, financial fraud detection, autonomous navigation systems, and biometric access control. The growing sophistication of adversarial strategies in these sectors has underscored the necessity of advancing adversarial attack prevention mechanisms as a critical research priority [1].

Adversarial prevention techniques are engineered to shield model predictions from tampered inputs by implementing strategies such as gradient masking, adversarial training, and noise robustness modules. Gradient suppression reduces sensitivity to perturbation directions, while filtering techniques isolate signal features from adversarial noise [2]. Structural consistency modules reinforce the alignment of spatial and contextual features, especially valuable for visual data. These defense mechanisms collectively strengthen model stability and promote accurate predictions even under hostile input conditions. Within the broader scope of computer vision, image classification remains a foundational task, involving the systematic categorization of visual data into predefined semantic classes. In remote sensing, this process becomes more intricate owing to spectral richness, varied spatial resolutions, and sensor-specific variations [3].

Remote sensing image classification involves assigning meaningful land cover categories or semantic attributes to aerial or satellite-captured imagery. These images often support critical

operations in precision agriculture, infrastructure surveillance, climate mapping, and disaster impact assessment. The complexity of remote sensing data makes classification models highly susceptible to adversarial interference. Even a minor spatial perturbation in the spectral domain can redirect prediction toward incorrect classes, leading to errors in land use assessment or risk prediction. Such misclassifications can have far-reaching consequences in policy formulation, resource distribution, and environmental sustainability initiatives [4].

Preserving spatial integrity in remote sensing imagery is key to countering adversarial threats. Features such as object contours, relative positioning, and inter-region consistency typically exhibit stability under genuine conditions but are highly vulnerable under adversarial influence. Defense strategies must integrate geometric consistency constraints into the learning process to reinforce the alignment between predicted outcomes and spatial coherence. By embedding spatial fidelity into model training, the classifier becomes more resilient against perturbation-based attacks. Models tuned to respect geometric boundaries exhibit stronger interpretability and enhanced reliability. This robustness is critical for maintaining decision integrity in domains where satellite data drives essential action, ensuring dependable insights amidst adversarial interference.[5], [6]. Integrating geometric constraints into model training promotes robustness by aligning predicted classes with underlying spatial structures. This approach enhances resilience against a broad spectrum of adversarial attacks, particularly those relying on spatial distortion or region-specific perturbation [6]. In remote sensing, preserving geometric fidelity ensures that the classifier maintains semantic accuracy while being resistant to synthetic disruptions, enabling more trustworthy outcomes in high-stakes domains [7].

1.1. Problem Statement

Remote sensing classification models are highly susceptible to adversarial attacks that target the geometric structure of spatial features such as boundaries, shapes, and object layouts. Adversarial perturbations designed to distort geometric attributes can cause models to misclassify terrain types, infrastructure zones, or natural resources. Current classification architectures often rely on convolutional features that lack explicit structural representation, making them vulnerable to spatial deformations. Techniques that preserve semantic content but alter feature geometry are particularly effective in evading detection. The absence of geometric regularization or consistency enforcement within model architectures has left classification outputs highly unstable under adversarial threats. Addressing this vulnerability requires a framework that incorporates geometric priors and enforces structural consistency throughout the learning process. Failure to defend against such threats can degrade model performance in practical deployments, including environmental surveys, land-use categorization, and agricultural mapping. A classification system that resists geometric distortion while maintaining edge-level semantic understanding is essential to ensure operational accuracy in adversarial environments.

1.2. Motivation

Geometric distortions introduced by adversarial perturbations challenge remote sensing classifiers by misaligning spatial structures. Classification tasks dependent on object shapes and boundaries become unreliable when geometric consistency is compromised. Ensuring resistance to spatial misrepresentation is essential, particularly in tasks involving building extraction, water body identification, or road mapping. Preserving shape, edge continuity, and relative spatial placement can improve classification trustworthiness under adversarial influence. Developing a method that emphasizes structural integrity can significantly reduce misclassification risks and enhance robustness against spatially deceptive inputs in real-world imagery.

1.3. Objective

The objective is to develop a classification framework that incorporates geometric consistency constraints to defend against adversarial spatial deformations. The model is intended to learn and preserve object shape and boundary structure, making it resilient to perturbations that distort geometry. This involves designing a spatial regularization mechanism that enforces consistency in object representation across layers. The aim is to detect and suppress adversarial perturbations that disrupt spatial alignment while maintaining the semantic granularity necessary for accurate classification. The framework should also support multi-resolution inputs to enhance structural awareness across scales. Performance will be assessed on tasks involving edge-sensitive classification, such as infrastructure mapping and ecological boundary detection. The objective includes ensuring that the proposed model retains

high accuracy across both perturbed and unperturbed inputs, with a focus on interpretability, robustness, and generalizability across diverse geographic contexts.

2. LITERATURE REVIEW

"Double Dynamic Graphs"[8] structured a dual-pathway graph network to capture inter-source and intra-source relations across satellite modalities. Each source-specific graph encoder utilized temporal-consistent attention and spatial correlation filters. Graph construction dynamically evolved through edge-weight learning based on feature similarity and contextual adjacency. A dual fusion bridge aligned graph outputs using distribution-aware normalization and hierarchical node merging. "Fusion-Label Classifier" [9] applied a multistage feature fusion mechanism where low-level textures and high-level semantics were jointly recalibrated using skip-connections. A confidence-adaptive pseudo-labeling scheme selectively added unlabeled data to training based on entropy-guided thresholds. Fusion modules reinforced category consistency by integrating inter-layer feature bridges during training. "LSC-Based Adaptor" [10] deployed a saturation-aware adjustment module for model calibration during inference. Test-time adaptation relied on entropy minimization over feature activations with class-wise low-confidence filtering. Confidence distribution was analyzed dynamically, allowing masking of unstable class responses during backpropagation-free refinement.

"Universal ECG Shield" [11] introduced a defense pipeline that operated without knowledge of adversarial data, relying instead on intrinsic signal priors. A denoising autoencoder reconstructed baseline ECG signals using frequency-suppressed encoding. Gradient filtering techniques suppressed adversarial patterns through a noise-aware regularization layer. "DP-Resilient Classifier" [12] developed a noise-robust classification scheme that preserved prediction fidelity despite the injection of differential privacy perturbations. A hybrid encoder was trained on gradient-clipped images to avoid overfitting on obfuscated features. An adaptive filtering mechanism tuned for privacy-noise patterns removed uncertainty-laden regions before classification. "IoMT Federated Shield" [13] examined vulnerabilities in Medical IoT devices through gradient poisoning, replay injection, and model inversion. A federated learning architecture with differential privacy noise was deployed across distributed devices. Local models updated the central server while maintaining zero raw data leakage.
"Hybrid Net Attack Detector" [14] implemented both conventional ML classifiers and deep networks across real-time packet flows. Feature engineering extracted statistical, temporal, and protocol-level signatures. A layer-wise hybrid detector stack used ensemble voting for initial screening, followed by CNN-LSTM pipelines for deep feature validation. "Sharpening-Aware Attack" [3] introduced pixel-level perturbations during pan-sharpening, aiming to distort object recognition tasks downstream. Perturbations were designed using wavelet-based gradient amplifiers. Loss constraints balanced spectral distortion with structural tampering. Object detectors trained on clean images were evaluated on perturbed outputs, revealing misalignment. "Fusion-Secure UAV Targeting" [15] deployed a transformer-based architecture for fusing multi-spectral and LiDAR data in forestry target classification. Temporal branches captured seasonal shifts, while spatial encoders handled canopy depth textures. Security-aware attention layers mask regions vulnerable to spoofing. Domain-aligned fusion preserved vegetation structure during target segmentation.

"Caption Attack Generator" [16] crafted adversarial examples that subtly altered image features to manipulate output captions. Attention-based optimization targeted encoder attention maps, redirecting focus away from critical image elements. Perturbations were constrained by visual imperceptibility thresholds. Decoder outputs were re-aligned to desired miscaption targets using semantic drift regularization. "Stepwise Attack Detector" [17] structured a multi-stage detection pipeline using weighted conditional adversarial networks. Feature maps were passed through conditional layers sensitive to gradient irregularities. Detection relied on stepwise propagation of attack traces using cumulative confidence scoring. "3D TopoAttack Engine" [18] developed spatially structured perturbations targeting point-cloud object trackers. Topology-aware encoding captured surface continuity and normal orientation. Universal attacks were crafted without target data, leveraging geometric invariance and tracker behavior priors. Loss optimization penalized point-shift misalignments across tracking frames. "RL Cluster Shield" [19] deployed a clustering framework to detect adversarial policy divergence in reinforcement learning agents. Feature traces from policy updates were embedded into vector spaces and clustered using dynamic thresholds. Outlier

detection revealed policy manipulation events. Transition trajectory inspection provided behavioral context to reinforce detection. "Ensemble Audio Adversary" [20] created a robust audio attack pipeline leveraging ensemble outputs from multiple speech recognition models. Gradient alignment techniques enhanced perturbation stability across acoustic models. Transferability improved through temporal masking and phoneme-preserving transformations.

"MSRF" [21] introduced a fusion strategy that combined multi-spectral images by aligning their gradient-based moment characteristics. The method captured structural patterns by computing local gradient distributions and matching their statistical properties. A moment-weighted optimization adjusted intensity variations across channels, preserving edges and textures. Fusion weights are adapted regionally using contrast-aware cues, ensuring consistent blending. Low-pass and high-pass filtering enhanced complementary details while reducing redundancy. Spectral fidelity was maintained by penalizing deviation from the original channels. "A3OD" [22] constructed adversarial perturbations targeting object detectors commonly used in remote sensing. The attack modified input pixels using optimized gradients, forcing detection modules to misidentify or ignore targets. Anchor box shifts and feature misalignment caused bounding box errors across networks like YOLO and Faster R-CNN. Perturbations were designed to remain imperceptible while maximizing detection confusion. Transferability was improved through shared feature-space disruption. The framework evaluated attack robustness across varied scenes, including buildings, vehicles, and vegetation.

3. GEOMETRIC CONSISTENCY REINFORCED SPARSE CAPSULE NETWORK (GCR-SCAPS)

GCR-SCaps Framework is a geometric consistency reinforced sparse capsule network designed to defend remote sensing classifiers against structural adversarial attacks by preserving object shape and spatial alignment through sparse capsule activations, dynamic routing, and transformation-aware loss optimization, ensuring stable and reliable decision-making under perturbation-influenced environments.

3.1. Remote Sensing Image Acquisition and Normalization

Remote sensing imagery typically involves the acquisition of high-resolution, multispectral data captured from aerial or satellite platforms. These images carry extensive spatial, spectral, and radiometric characteristics crucial for distinguishing terrain objects such as buildings, roads, vegetation, and vehicles. The raw data collected from remote sensing platforms are inherently heterogeneous, with significant variations in brightness, illumination, scale, and object density. Standardizing these inputs is essential to ensure stable downstream feature extraction and geometric consistency when passed into capsule-based architectures.

The input image $x \in R^{H\times W\times C}$ is initially defined in terms of its height H, width W, and channel dimension C. Each pixel in x holds a spectral intensity value ranging across visible or infrared bands. The first processing phase involves resizing the image to a fixed input dimension, such that every image maintains a uniform size across the training pipeline. Let the resized image be represented as Eq.(1).

$$x_r = Resize(x, H_0, W_0) \quad (1)$$

Where H_0 and W_0 represent the standard height and width selected for all inputs. This resizing operation facilitates consistent receptive fields during convolutional processing and aligns the structural scale of the objects under analysis.

The spectral channels in remote sensing imagery may exhibit varying dynamic ranges. Some sensors capture reflectance in raw radiometric units, while others apply onboard calibration. To stabilize inter-channel variance and suppress over-amplified pixel intensities, each channel $c \in C$ is normalized using min-max scaling. The operation across all pixels for a given channel is given as Eq(2).

$$x_r^{(c)} = \frac{x_r^{(c)} - min\left(x_r^{(c)}\right)}{max\left(x_r^{(c)}\right) - min\left(x_r^{(c)}\right)} \quad (2)$$

This operation scales the values to a bounded range of [0,1], preserving relative intensity distributions while preventing numerical instability in early layers of the network. The output $x_r^{(c)}$now represents a normalized spectral channel with reduced radiometric bias.

Remote sensing images often contain spatially redundant or low-informative areas, such as background vegetation or water bodies. These regions may not contribute to meaningful capsule activation in subsequent steps. To suppress such redundancies, a contrast enhancement step is incorporated through adaptive histogram equalization, which sharpens object boundaries and

enhances local spatial structures. The enhanced image x_e is obtained as shown in Eq.(3).

$$x_e = CLAHE(x_r) \quad (3)$$

Where CLAHE refers to Contrast Limited Adaptive Histogram Equalization, this technique segments the image into local tiles. It applies histogram flattening to improve spatial salience without introducing over-amplification in noise-prone regions.

Artifacts such as sensor noise, compression errors, or atmospheric haze frequently distort the spatial or spectral integrity of remote sensing data. A noise suppression filter is applied over the normalized image to eliminate high-frequency distortions. A bilateral filter or non-local means operation serves this purpose. The denoised image x_d is computed as illustrated in Eq.(4).

$$x_d = Denoise(x_e, \sigma_s, \sigma_r) \quad (4)$$

Where σ_s and σ_r represent the spatial and range parameters controlling the extent of smoothing based on spatial closeness and intensity similarity, respectively. This filtering maintains edge definitions while suppressing pixel-level fluctuations irrelevant to geometric integrity.
Since the subsequent stages in the capsule architecture require an object-centric perspective, a centralized spatial window is often extracted around the dominant region of interest (ROI). This spatial cropping focuses on the object geometry, preserving high-salience areas and reducing peripheral distractions. The object-centric patch x_c is localized using activation heatmaps or precomputed bounding indices, represented as Eq.(5).

$$x_c = Crop(x_d, x, y, h, w) \quad (5)$$

Here, (x, y) denotes the top-left coordinate of the ROI, and h, w specify the height and width of the cropped window. This extraction step enhances geometric focus, which is foundational for learning pose-preserving representations in sparse capsule layers.

With x_c prepared, the image maintains structural coherence, spectral stability, and spatial sharpness, ready to be passed through shallow convolutional encoders. The controlled variations and absence of disruptive noise in x_c ensure that the sparse capsules initialized later are aligned with authentic object boundaries. Moreover, the contrast-enhanced and normalized form of the image stabilizes routing activations and reduces the likelihood of false agreement among capsules, especially under adversarial geometric perturbations.

3.2. Shallow Feature Encoding via Convolution

Once the remote sensing image $x_c \in R^{H_0 \times W_0 \times C}$ is preprocessed for spatial uniformity, spectral normalization, contrast sharpening, and structural focus, the next computational operation involves shallow feature encoding. This stage utilizes a lightweight convolutional architecture to capture primitive visual cues such as edges, textures, and region transitions while preserving the structural geometry essential for capsule initialization.

The process begins by applying a convolutional layer that projects the input image into a lower-dimensional yet more expressive feature space. Let $W_1 \in R^{k \times k \times C \times d_1}$ denote the weight tensor of the first convolutional layer with a kernel size k and d_1 output channels. The output feature map F_1 is computed as Eq.(6)

$$F_1 = ReLU(Conv(x_c, W_1) + b_1) \quad (6)$$

where $b_1 \in R^{d_1}$ is the bias vector, and ReLU represents the rectified linear unit activation, promoting sparse and stable activations by nullifying negative responses. The tensor $F_1 \in R^{H_1 \times W_1 \times d_1}$ contains localized filters emphasizing line segments, edges, and texture gradients within the image.
To enhance the discriminative capacity, a second convolutional block is introduced to refine the initial encodings. Let the second set of filters be $W_2 \in R^{k \times k \times d_1 \times d_2}$. The updated feature map F_2 is produced as Eq.(7).

$$F_2 = ReLU(BN(Conv(F_1, W_2) + b_2)) \quad (7)$$

where $b_2 \in R^{d_2}$, and BN denotes batch normalization, which standardizes the activations across mini-batches to prevent internal covariate shift. The feature map $F_2 \in R^{H_2 \times W_2 \times d_2}$ reflects progressively more abstract visual patterns while maintaining the underlying object boundary geometry.

Given that remote sensing images often exhibit complex background textures, a spatial reduction operation is necessary to condense local context while preserving the dominant geometric layout. A max-pooling layer is employed to extract high-salience regions from F_2, resulting is expressed as Eq.(8).

$$F_3 = MaxPool(F_2, s)) \quad (8)$$

Where s indicates the pooling stride. This downsampling not only reduces computational overhead but also aids in retaining the strongest activations corresponding to object edges and contours. The output $F_3 \in R^{H_3 \times W_3 \times d_2}$ carries compact and high-contrast spatial signals.

To further enhance feature representation, a depthwise separable convolution layer is introduced, reducing parameter count while refining semantic quality. Let the depthwise weights be $W_{dw} \in R^{k \times k \times d_2}$, and pointwise weights be $W_{pw} \in R^{1 \times 1 \times d_2 \times d_3}$. The refined feature map $F_4$is given by Eq.(9).

$$F_4 = ReLU\big(Conv_{pw}\big(Conv_{dw}(F_3, W_{dw}), W_{pw}\big) + b_3\big) \quad (9)$$

where $b_3 \in R^{d_3}$ this operation separates channel-wise filtering and combination, capturing localized directional features critical for structural alignment under adversarial shifts.

Preservation of geometric layout across feature layers is crucial for initializing pose-aware capsules. To maintain this, the final feature map is refined through a dilated convolution layer that expands the receptive field without reducing spatial resolution. Let $W_d \in R^{k \times k \times d_3 \times d_4}$ and dilation rate r define the operation which is expressed in Eq.(10).

$$F_5 = ReLU(Conv_{dilated}(F_4, W_d, r) + b_4) \quad (10)$$

where $b_4 \in R^{d_4}$ the dilation rate r allows for multi-scale geometric abstraction by connecting distant features while retaining the resolution required for capsule formation.

The resulting tensor $F_5 \in R^{H_5 \times W_5 \times d_4}$ encapsulates shallow semantic and geometric properties of the input image, enriched through convolutional, pooling, and dilation operations. These features carry spatial alignment, object-part orientation, and texture variations attributes necessary for the succeeding sparse capsule vectorization phase.

3.3. Primary Capsule Vectorization

Feature tensor $F_5 \in R^{H_5 \times W_5 \times d_4}$, extracted from the final convolutional encoder, holds localized semantic and geometric information necessary for initiating capsule representations. This tensor encapsulates shallow spatial features such as object edges, regional symmetry, and boundary contrasts that are pivotal for pose-aware recognition. To convert these spatially grounded feature points into entity-specific vectors, a capsule-based vectorization strategy is required. This strategy enables the model to detect the presence and spatial orientation of object parts in a structured form rather than as individual scalar activations.

Primary capsules are built by reorganizing the flattened convolutional outputs into groups, where each group functions as a vector unit representing the existence and instantiation parameters of a visual entity. Let the tensor F_5 be partitioned into N_c capsule channels, where each channel maps to a primary capsule type. The initial reshaping can be formulated as Eq.(11).

$$U = Reshape(F_5, [N_c, H_c, W_c, d_c]) \quad (11)$$

Where N_c is the number of capsule types, H_c and W_c denote spatial dimensions, and d_c is the dimensionality of each capsule vector. This operation groups d_c neurons into a structured vector at each spatial location, laying the groundwork for directional encoding and entity transformation modeling.

Unlike scalar activations in CNNs, capsule vectors require nonlinear normalization to limit vector lengths within a bounded range without discarding their orientation. This is achieved using the squashing activation function that preserves directionality while controlling magnitude. The squashing process for a capsule vector s_i produces the output vector v_i as expressed in Eq.(12).

$$v_i = \frac{\|s_i\|^2}{1 + \|s_i\|^2} \cdot \frac{s_i}{\|s_i\|} \quad (12)$$

Here, $\|s_i\|$ denotes the Euclidean norm of the pre-activation vector, and v_i retains the pose direction while compressing magnitude to a range close to one for confident entities and near-zero for uncertain ones. This scaling ensures numerical stability and gradient consistency during routing.

Each capsule vector captures instantiation parameters that are sensitive to object features like location, scale, rotation, and texture. These pose-aware representations enable geometric part-whole reasoning, which forms the core principle in capsule networks. To refine the vector input s_i, a trainable transformation matrix W_{ij} is introduced. This matrix projects a lower-level capsule v_i into a higher-level prediction $\hat{u}_{j|i}$ for class capsule j which is represented mathematically in Eq.(13).

$$\hat{u}_{j|i} = W_{ij} \cdot v_i \quad (13)$$

The matrix $W_{ij} \in R^{d_t \times d_c}$ learns class-specific affine transformations such as scaling or translation, allowing lower-level capsules to predict how their associated part might appear under different object-

level poses. This alignment facilitates dynamic routing based on agreement between predicted poses and actual higher-level capsules.

Primary capsules must be spatially diverse but semantically consistent. To introduce regularity in their responses, a local capsule convolution is applied, where small spatial neighborhoods are processed collectively to build pose-preserving regional vectors. Let the capsule convolution kernel be represented as $K_c \in R^{k\times k\times d_c\times d_c}$, which performs localized matrix multiplication over vector fields. The convolved capsule output s_i is updated as Eq.(14).

$$s_i = \sum_{u,v\in N(i)} K_c(u,v)\cdot U_{i+u,j+v} \qquad 14)$$

Here, $N(i)$ defines the local region centered at the capsule position i, and $U_{i+u,j+v}$ are neighboring capsules. This convolutional capsule aggregation promotes orientation coherence and localized consistency.

To reduce redundancy and enhance expressive sparsity, a relevance gating mechanism filters out low-importance capsule vectors before routing. Each capsule is assigned a probability score $p_i \in [0,1]$ indicating its contextual importance, computed via a sigmoid-activated gating unit.

$$p_i = \sigma\left(w_g^{\mathrm{T}}\cdot v_i + b_g\right) \qquad (15)$$

In Eq.(15) the scalar p_i is used to mask weak capsule activations, promoting sparsity and robustness. The gating vector $w_g \in R^{d_c}$ and bias $b_g \in R$are trainable parameters. Capsules with low p_i are dropped, allowing only structurally meaningful pose vectors to participate in routing, which aligns with the model's goal of resisting structural perturbations caused by adversarial attacks.

Following gating, valid capsules are stacked into a flattened set of spatially aligned vector entities $V = \{v_1, v_2, \dots, v_n\}$, ready to be routed toward class capsules in subsequent steps. These vectors are now pose-sensitive, structurally gated, and dynamically transformable, capturing foundational building blocks for semantic entity recognition.

3.4. Sparsity Enforcement on Capsule Activations

Capsule networks inherently capture high-dimensional pose features that encode geometric information, but when all capsules are active simultaneously, the network becomes over-expressive and prone to adversarial vulnerability. Enforcing sparsity among capsule activations reduces this over-sensitivity, ensuring that only structurally meaningful capsules contribute to downstream routing. Sparsity introduces a competitive mechanism among capsules, helping the architecture to focus on a compact set of entities that are geometrically relevant and semantically robust, particularly for remote sensing images susceptible to structural perturbations.

Let the set of primary capsule outputs from the previous step be denoted as $V = \{v_1, v_2, \dots, v_n\}$ represents a vector encoding the presence and pose of a visual part. Each capsule's activation strength is inferred from the vector norm, which serves as an implicit confidence measure. To evaluate the overall activation State, define the capsule activation score as shown in Eq.(16).

$$a_i = \|v_i\|_2 \qquad (16)$$

This activation score $a_i \in [0,1]$ reflects the certainty of the capsule i being active. A higher magnitude indicates a strong presence and spatial consistency of the corresponding visual entity, whereas lower magnitudes signal noisy or irrelevant representations.

Sparsity can be introduced by ranking these activations and retaining only the top-performing capsules within a localized spatial context. Define a sparsity threshold $\tau \in R$ and let $S \subset V$ be the subset of capsules whose activations exceed τ. This threshold-based filtering is expressed as Eq.(17).

$$S = \{v_i \in V | a_i \geq \tau\} \qquad (17)$$

The subset S now contains only capsules with sufficiently strong pose confidence, enforcing a form of hard sparsity that discards capsules unlikely to align with valid object structures. This selective retention enhances structural alignment and reduces adversarial activation diffusion.

For improved flexibility and trainable selection, a soft sparsity regularization is applied using an entropy-based objective. Define normalized capsule probabilities based on activation magnitudes as shown in Eq.(18).

$$p_i = \frac{a_i}{\sum_{j=1}^{n} a_j} \qquad (18)$$

These probabilities indicate the relative importance of each capsule across the set. A low entropy across this distribution signifies a sharp focus on fewer capsules, while a high entropy indicates scattered and non-discriminative activation. To regularize capsule competition, the sparsity loss is computed as Eq.(19).

$$L_{sparse} = -\sum_{i=1}^{n} p_i log(p_i) \quad (19)$$

This entropy minimization objective penalizes broad activation distributions, encouraging the model to concentrate activation across fewer capsules with stronger geometric salience. The sparsity loss complements the hard thresholding approach, ensuring both binary and probabilistic suppression mechanisms co-exist during training.

To further stabilize capsule selection during adversarial perturbations, a margin-based reinforcement strategy is added. Capsules whose activations fall within a transition margin are softly gated. Let δ_1 and δ_2 define the lower and upper activation boundaries for soft selection. A continuous gating score $g_i \in [0,1]$ is introduced as Eq.(20).

$$g_i = min\left(1, max\left(0, \frac{a_i - \delta_l}{\delta_u - \delta_l}\right)\right) \quad (20)$$

This gating score allows gradual inclusion of capsules as their activation scores improve, thereby preventing abrupt inclusion or exclusion caused by hard cutoffs. The gated output $\tilde{v}_i$ is then defined using Eq.(21).

$$\tilde{v}_i = g_i \cdot v_i \quad (21)$$

This softly gated capsule preserves pose direction but attenuates or suppresses the contribution of less confident capsules, enabling smooth adaptation under minor input variations. The gating mechanism ensures robustness to noise and localized distortions in satellite or aerial imagery.

The final sparse capsule set passed to the routing layer is denoted as $\tilde{V} = \{\tilde{v}_1, \tilde{v}_2, \dots, \tilde{v}_m\}$, where $m \leq n$. The reduced dimensionality and enhanced focus reduce redundant agreement, streamline the routing process, and increase resistance to adversarial manipulation that targets low-activation or noisy capsules. By integrating top-k filtering, entropy regularization, and soft gating, the capsule network transitions from an overactive encoder to a sparsely activated, semantically guided structure. This controlled sparsity plays a foundational role in maintaining interpretability, enforcing geometric regularity, and suppressing adversarial interference in the class capsule formation and routing that follows.

3.5. Dynamic Routing with Agreement Mechanism

Sparse capsule vectors $\tilde{V} = \{\tilde{v}_1, \tilde{v}_2, \dots, \tilde{v}_m\}$, filtered through activation gating and entropy regulation, represent localized visual entities with pose-encoded characteristics. To aggregate these into higher-order semantic representations, a dynamic routing mechanism is employed. This mechanism selectively transfers information from lower-level capsules to class-level capsules based on their agreement in pose orientation and activation confidence. The routing process ensures that structurally consistent predictions reinforce each other, which is critical for preserving geometric coherence in remote sensing images undergoing adversarial spatial perturbations.

Each sparse capsule $\tilde{v}_i \in R^{d_c}$ contributes to a prediction vector $\hat{u}_{j|i} \in R^{d_{cls}}$ for each higher-level class capsule j. A learned projection matrix achieves this transformation $W_{ij} \in R^{d_{cls} \times d_c}$, enabling viewpoint invariance and spatial extrapolation which is expressed in Eq.(22).

$$\hat{u}_{j|i} = W_{ij} \cdot \tilde{v}_i \quad (22)$$

These predicted vectors form the basis for evaluating mutual agreement between lower-level part capsules and upper-level object capsules. A coupling coefficient $c_{ij} \in [0,1]$ is assigned to each connection between capsules i and capsule j, indicating the strength of the contribution from $\tilde{v}_i$to the aggregated output s_j of capsule j. The output is computed as a weighted sum as shown in Eq.(23).

$$s_j = \sum_{i=1}^{m} c_{ij} \cdot \hat{u}_{j|i} \quad (23)$$

The coupling coefficients c_{ij} are constrained to sum to one across all classes for each lower capsule i, allowing them to distribute their influence based on alignment. These coefficients are derived from routing logits $b_{ij} \in R$, initialized to zero and refined through iterative agreement. The softmax normalization applied to the logits yields the coefficients.

$$c_{ij} = \frac{exp(b_{ij})}{\sum_{k=1}^{C} exp(b_{ik})} \quad (24)$$

In Eq.(24) C denotes the total number of class capsules. This normalization ensures proper probabilistic interpretation of contribution strengths during agreement propagation. After computing the initial outputs s_j, a nonlinear squashing function is applied to produce the final class capsule outputs $v_j \in R^{d_{cls}}$, encapsulating both the presence and pose

of class-specific features which is mathematically expressed using Eq.(25).

$$v_j = \frac{\|s_j\|^2}{1+\|s_j\|^2} \cdot \frac{s_j}{\|s_j\|} \quad (25)$$

The squashing operation scales down low-activation capsules while preserving direction and amplifying stronger ones, which aligns well with the robustness requirement under structural attack conditions. Once the final class vectors v_j are computed, the agreement between predictions and actual outputs is evaluated. This agreement score a_{ij} is measured using dot product similarity and is expressed mathematically in Eq.(26).

$$a_{ij} = \hat{u}_{j|i} \cdot v_j \quad (26)$$

The score $a_{ij} \in R$ quantifies how closely the predicted pose from the capsule i matches the actual pose of the capsule j. A higher score indicates more substantial alignment and mutual support. These scores are then used to update the routing logits as expressed in Eq.(27).

$$b_{ij} \leftarrow b_{ij} + a_{ij} \quad (27)$$

This update increases the contribution of capsules that agree more, while suppressing those that do not align with the target structure. The entire routing loop is repeated for a fixed number of iterations R, typically three, allowing stable convergence of the agreement-driven aggregation. During each iteration, the routing coefficients are refined, enabling the class capsules to form a holistic understanding of object configuration based on part-level capsule consensus.

3.6. Final Class Capsule Representation

Aggregated outputs from dynamic routing, encoded in vectors $v_j \in R^{d_{cls}}$, serve as high-level class-specific capsules that synthesize part-based evidence into global object representations. These vectors capture not just class presence but also encode instantiation parameters such as spatial orientation, relative part alignment, and transformation invariance—traits vital for recognizing structures in remote sensing images that vary in perspective, size, and alignment. Each capsule vector v_j results from an agreement-driven routing process, selectively integrating transformed pose predictions from sparse lower capsules that demonstrate strong alignment with class-specific expectations.

Class capsule activation is interpreted based on the magnitude of v_j, where stronger norms imply higher confidence in the class presence and spatial consistency. The probability score p_j assigned to class j is computed by taking the vector norm as expressed in Eq.(28).

$$p_j = \|v_j\|_2 \quad (28)$$

This scalar score $p_j \in [0,1]$ functions as a continuous confidence estimate. Unlike softmax-based activations in conventional classifiers, capsule probability scores retain the interpretability of object presence alongside pose agreement. This alignment ensures that geometric misalignments, often introduced by adversarial spatial transformations, are reflected in reduced vector norms, effectively degrading the certainty in classification.

To ensure well-calibrated confidence predictions across multiple classes, a margin-based loss is introduced. Each class capsule is evaluated against a target presence indicator $T_j \in \{0,1\}$, which signifies whether the class j corresponds to the ground truth. The loss penalizes cases where the norm of the correct class capsule falls below a predefined upper margin m^+, and the norm of incorrect class capsules rises above a lower threshold m^-. The objective is captured using Eq.(29).

$$L_{margin} = \sum_j \Big[T_j \cdot max\left(0, m^+ - \|v_j\|^2 + \lambda \cdot (1-T_j) \cdot max(0, \|v_j\| - m^-)\right)^2 \Big] \quad (29)$$

Here, $\lambda \in [0,1]$ regulates the suppression of non-target class activations, enforcing the network to produce sparse and focused predictions. The upper and lower bounds m^+ and m^-serve to push the correct capsule norms closer to one while discouraging non-relevant capsules from exceeding the margin.

To promote consistency in semantic and geometric encoding, the final capsule outputs are regularized using a pose dispersion constraint. This mechanism prevents overlapping feature representations across different class capsules, enhancing class separability in the embedding space. The pairwise cosine similarity between capsule vectors is minimized using Eq.(30).

$$L_{disp} = \sum_{j \neq k} \frac{v_j \cdot v_k}{\|v_j\| \cdot \|v_k\|} \quad (30)$$

Lower dispersion loss encourages distinct orientation and directional encoding across classes, ensuring that objects such as buildings, roads, or vegetation patches occupy separate vector

subspaces, even if they share spatial or spectral similarity.

In scenarios where multiple classes may co-exist or spatial overlap exists in remote sensing scenes, a capsule attention modulation strategy is employed. This technique allows the model to weigh the importance of each capsule dynamically, adapting to multi-structure imagery. An attention weight $\alpha_j \in [0,1]$ is computed for each class capsule based on its vector magnitude and pose alignment certainty, which is represented in Eq.(31).

$$\alpha_j = sigmoid\left(w^{\top} \cdot v_j + b\right) \tag{31}$$

The trainable vector $w \in R^{r_{cls}}$ and scalar $b \in R$ learn to modulate capsule emphasis based on pose stability. These weights are then applied to derive refined outputs $\tilde{v}_j$ as specified in Eq.(32).

$$\tilde{v}_j = \alpha_j \cdot v_j \tag{32}$$

This modulation introduces an adaptive relevance gate, strengthening the contribution of capsules with stable structural agreement while downscaling ambiguous or noisy representations. The refined capsule set $\{\tilde{v}_1, \tilde{v}_2, \ldots., \tilde{v}_C\}$ reflects a well-discriminated and calibrated semantic structure.
For interpretability and tracking spatial relationships under adversarial perturbations, a geometric coherence matrix $G \in R^{C \times C}$ is constructed. Each entry G_{jk} represents alignment between class capsule pairs based on pose similarity and adjacency patterns, which is shown in Eq.(33)

$$G_{jk} = \left\|\tilde{v}_j - \tilde{v}_k\right\|_2^2 \tag{33}$$

This matrix helps in diagnosing misclassifications due to adversarial spatial shifts by analyzing deviations in expected pose relations. Capsules showing abnormal similarity under adversarial input can be identified and isolated through this coherence structure. The final set of class capsules, embedded with margin-regularized norms, dispersion penalties, and adaptive attention modulation, carries high-confidence, structurally grounded decision representations. These pose-aware outputs act not only as classification tokens but also as interpretable carriers of spatial semantics, supporting downstream modules such as geometric consistency enforcement and adversarial feedback comparison.

3.7. Structural Margin Loss Computation

Class capsule vectors $\tilde{v}_j \in R^{d_{cls}}$, refined through agreement-based routing and attentional modulation, represent both categorical confidence and spatial structure of detected entities. These vectors must be further regulated to encode accurate presence while maintaining structural alignment across possible adversarial deformations. In order to achieve this dual objective, a margin-based loss framework is imposed. This framework selectively pushes the correct class capsule vector norms towards unity while constraining all non-relevant capsule activations within a lower boundary. Structural integrity is preserved by enforcing a geometric margin between active and inactive class capsules.

Let each class capsule's activation strength be expressed as $p_j = \left\|\tilde{v}_j\right\|$, where higher norms signify confident detection and consistent pose inference. A binary indicator $T_j \in \{0,1\}$denotes the presence (1) or absence (0) of class j within the current input. The core margin loss objective prioritizes strong activation for ground truth capsules while penalizing spurious activations. The primary loss function is structured as Eq.(34)

$$L_{margin} = \sum_{j=1}^{C} \left[T_j \cdot max\left(0, m^{+} - p_j\right)^2 + \lambda \cdot \left(1 - T_j\right) \cdot max\left(0, p_j - m^{-}\right)^2\right] \tag{34}$$

The upper margin m^{+}encourages true class capsule norms to exceed a desired boundary, while the lower margin m^{-} suppresses norms of capsules unrelated to the current image. The regularization parameter $\lambda \in [0,1]$ balances penalty intensity for false positives, reducing the risk of over-suppression in multi-class or ambiguous contexts. This formulation aligns classification certainty with spatial coherence. To incorporate structural awareness into the margin formulation, a pose alignment sensitivity term is embedded into the objective. For each activated capsule, the pose orientation is compared against an expected pose direction $d_j \in R^{d_{cls}}$, derived from aggregated agreement across routing iterations. The angular divergence $\theta_j$between $\tilde{v}_j$ and d_j is computed as Eq.(35).

$$\theta_j = arccos\left(\frac{\tilde{v}_j \cdot d_j}{\left\|\tilde{v}_j\right\| \cdot \left\|d_j\right\|}\right) \tag{35}$$

This angle quantifies the directional deviation of capsule vectors from their ideal pose configurations. Smaller angles correspond to better alignment with structural expectations. Capsules that exhibit both low activation and poor alignment are likely false activations. A structural penalty term is defined by integrating angular deviation into the margin loss as Eq.(36).

$$L_{struct} = \sum_{j=1}^{C} \left(1 - cos(\theta_j)\right) \cdot max(0, p_j - m^-) \quad (36)$$

The penalty term amplifies the suppression of capsules that are not only active but geometrically inconsistent. This integration ensures that even if a non-target class capsule activates, it must demonstrate a valid pose structure to avoid penalization.

To further promote spatial discrimination among capsules, a mutual separation constraint is introduced across active capsule pairs. For each valid pair (j, k), where $T_j = T_k = 1$ and $j \neq k$, a margin-based divergence is enforced. The separation loss is formulated as Eq.(37).

$$L_{sep} = \sum_{j \neq k} max\left(0, m_s - \left\|\tilde{v}_j - \tilde{v}_k\right\|\right)^2 \quad (37)$$

Here, m_s is the minimum Euclidean separation distance required between active capsules. This constraint ensures that simultaneously active capsules are spatially and semantically distinct, avoiding class overlap in dense or multi-object imagery familiar in remote sensing environments.

To unify all structural constraints, the final loss composition includes the core margin term, the pose-alignment regularizer, and the separation objective is expressed in Eq.(38).

$$L_{structural\_margin} = L_{margin} + \alpha \cdot L_{struct} + \beta \cdot L_{sep} \quad (38)$$

The weights α and β control the influence of structural penalties. By tuning these hyperparameters, the network can be calibrated to prioritize clean margin enforcement, alignment fidelity, or spatial distinctiveness, depending on the complexity of terrain or adversarial vulnerability.

In scenes with hierarchical object arrangements or nested categories, class capsules often exhibit natural geometric correlations. To prevent suppression of capsules that may be partially overlapping but structurally valid, a confidence-sharpening term is introduced to reward capsules that both align with expected structure and exceed norm thresholds. Let $\gamma_j = exp(-\theta_j) \cdot ReLU(p_j - m^+)$. The sharpening term is defined as Eq.(39).

$$L_{sharpen} = -\sum_{j=1}^{C} \gamma_j \cdot log\left(p_j + \epsilon\right) \quad (39)$$

The additive constant ϵ prevents numerical instability in logarithmic computation. This sharpening term promotes strong, confident activation of geometrically consistent capsules, enhancing clarity in ambiguous regions.

3.8. Generate Geometrically Transformed Input

High-resolution remote sensing imagery often suffers from pose and orientation inconsistencies caused by platform movement, environmental drift, and terrain elevation variation. To simulate such spatial deviations and evaluate the structural reliability of capsule representations, deliberate geometric transformations are applied to the original input. These transformations produce a variant of the image that retains semantic content while altering its structural alignment, thereby allowing the capsule network to learn geometry-invariant behavior.

Let the original preprocessed input be denoted as $x_c \in R^{H_0 \times W_0 \times C}$, representing the object-aligned, normalized satellite image. A geometric transformation function T is defined to introduce controlled alterations such as translation, rotation, and scaling. The transformed input x_t is generated as Eq.(40).

$$x_t = T(x_c) \quad (40)$$

The transformation operator T is parameterized by a set of variables $\theta = \{\theta_r, \theta_s, \theta_t\}$, where θ_r denotes rotation angle, θ_s indicates scale factor, and $\theta_t = (t_x, t_y)$ corresponds to horizontal and vertical translations. These variables are randomly sampled from bounded intervals to avoid semantic corruption while preserving the physical layout of relevant regions.

To apply rotation, a spatial rotation matrix $R(\theta_r) \in R^{2\times2}$ is defined as Eq.(41).

$$R(\theta_r) = \begin{bmatrix} cos(\theta_r) & -sin(\theta_r) \\ sin(\theta_r) & cos(\theta_r) \end{bmatrix} \quad (41)$$

This matrix rotates each spatial coordinate about the image center, effectively simulating aerial platform tilt or angular misalignment. Rotation helps test the capsule network's ability to maintain pose coherence even when global orientation shifts.

For scaling, a transformation matrix $S(\theta_s) \in R^{2\times2}$ is applied to compress or expand the field of view. The matrix is defined as Eq.(42).

$$S(\theta_s) = \begin{bmatrix} \theta_s & 0 \\ 0 & \theta_s \end{bmatrix} \quad (42)$$

This operation uniformly enlarges or reduces object regions, mimicking resolution variability or altitude drift. Maintaining consistent capsule activations across scale changes requires that pose matrices within capsules generalize spatial size variations without distortion.

Translation introduces positional shifts along horizontal and vertical axes. The translated coordinates (x', y') of a pixel originally at (x, y) are computed by Eq.(43).

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \quad (43)$$

The translation vector $\theta_t = (t_x, t_y)$ ensures simulation of location noise or sensor jitter. Capsule responses must remain invariant to these shifts, especially when regions of interest are repositioned without alteration in content.

To implement compound transformations, an affine transformation matrix $A \in R^{2\times3}$ is constructed by combining rotation, scaling, and translation into a unified matrix, which is mathematically represented in Eq.(44).

$$A = \begin{bmatrix} \theta_s cos(\theta_r) & -\theta_s sin(\theta_r) & t_x \\ \theta_s sin(\theta_r) & \theta_s cos(\theta_r) & t_y \end{bmatrix} \quad (44)$$

Applying the affine transformation A to each pixel coordinate in $x_c$results in the transformed image x_t, containing the same semantic entities arranged in a different spatial layout. This geometric drift induces changes in relative part configuration, which challenges the capsule network to preserve object identity under structural rearrangement.

To ensure information consistency between x_c and x_t, both images are forwarded through identical convolutional encoders and capsule initialization pipelines. Let v_j denote the class capsule output from the original input, and v_j' be the capsule output is generated from the transformed input. A geometric alignment consistency check requires the pose vectors v_j' to be brought back to the original coordinate frame by applying the inverse transformation T^{-1}. The reconstructed pose vector $\hat{v}_j'$ is computed as Eq.(45).

$$\hat{v}_j' = T^{-1}(v_j') \quad (45)$$

This re-mapped vector $\hat{v}_j'$ should closely match v_j if the capsule has captured the structure-invariant essence of the object. Subsequent steps measure this alignment through vector distance or cosine similarity, enforcing geometric consistency constraints. In this setup, the transformed input x_t becomes a strategic probe, designed to validate pose stability, structural understanding, and routing alignment of capsule-based recognition under adversarial shifts.

To maintain feature-level consistency, both inputs are encoded in parallel, and their capsule activations are paired for comparative loss computation. This ensures that the training process encourages invariance while penalizing distortions in semantic prediction caused by geometric modification. The deliberate generation of x_t using controlled spatial transformation anchors the adversarial resilience strategy of the GCR-SCaps model. Capsule vectors that remain consistent across x_c and x_t are deemed structurally robust, while those that deviate significantly are subject to refinement through joint loss optimization. This simulation step builds the foundation for pose-aligned capsule training and guides later constraints focused on enforcing geometric similarity, spatial attention stability, and misalignment suppression.

3.9. Forward Propagation of Transformed Input

The geometrically transformed image $x_t \in R^{H_0\times W_0\times C}$, generated using the affine transformation matrix A, contains identical semantic content as x_c but spatially rearranged. To assess the pose stability and structural alignment within capsule representations, the transformed input is passed through the same capsule encoding pipeline that previously processed the original input. Every architectural component remains consistent—convolutional encoders, primary capsule vectorization, sparsity filters, and dynamic routing are identically applied to x_t, ensuring a fair structural comparison.

The initial stage involves encoding shallow features fromx_t. Let $F_t \in R^{h\times w\times d}$ denote the output from the feature encoder applied to the transformed input, which is expressed in Eq.(46).

$$F_t = ConvEncoder(x_t) \quad (46)$$

The feature tensor F_t preserves edge transitions, texture boundaries, and spatial gradients, but now exhibits modified orientations and locations due to the transformation applied in the previous step. These encoded features are reshaped and grouped into vector form for capsule generation.

From F_t, primary capsule vectors $\tilde{v}_i' \in R^{d_c}$ are derived using grouped convolution and squashing functions identical to those used in processing x_c. The vectorized set is denoted as Eq.(47).

$$\tilde{V}' = \{\tilde{v}_1', \tilde{v}_2', \ldots., \tilde{v}_m'\} \quad (47)$$

Each $\tilde{v}_i'$ captures pose-specific information such as part location, shape orientation, and spatial scale post-transformation. Since the transformation matrix A altered the geometric structure without affecting

object identity, the capsule system must now demonstrate stability across these structural deviations.

Capsule activation filtering is enforced again using the same sparsity gating and entropy suppression techniques. Active capsules from $\tilde{V}'$ are propagated through the dynamic routing module to produce prediction vectors for each class capsule. Let $\hat{u}'_{j|i} \in R^{d_{cls}}$ represent the prediction from the transformed capsule $\tilde{v}'_i$ toward the class capsule j is expressed in Eq.(48).

$$\hat{u}'_{j|i} = W_{ij} \cdot \tilde{v}'_i \quad (48)$$

The transformation matrix W_{ij} is shared with the original routing phase to maintain consistency in pose projection. The agreement mechanism accumulates these predictions, forming the routed output s'_j for each class capsule is illustrated mathematically in Eq.(49).

$$s'_j = \sum_i c'_{ij} \cdot \hat{u}'_{j|i} \quad (49)$$

The routing coefficients $c'_{ij} \in [0,1]$ are derived via softmax over routing logits b'_{ij}, updated through the inner product agreement as in the original capsule pass. The final capsule output from the transformed input is squashed into a bounded vector form, which is expressed in Eq.(50).

$$v'_j = \frac{\|s'_j\|^2}{1 + \|s'_j\|^2} \cdot \frac{s'_j}{\|s'_j\|} \quad (50)$$

The class capsule $v'_j \in R^{d_{cls}}$ represents the category presence and pose encoding obtained from the transformed image x_t. Structural variations introduced by T are now reflected in these outputs, and any significant deviation from the original capsule v_j indicates instability in structural encoding.

To enable comparison between $v_j$and v'_j, the pose from the transformed capsule is first re-aligned to the original spatial configuration. The inverse transformation matrix A^{-1} is applied to v'_j to yield the corrected pose vector $\hat{v}'_j$ is shown in Eq.(51).

$$\hat{v}'_j = A^{-1} \cdot v'_j \quad (51)$$

This corrected capsule output should approximate the pose of the original vector v_j if the capsule system preserves orientation and part-to-object relationships despite input variation. The capsule distance under alignment is computed using Eq.(52).

$$D_j = \|v_j - v'_j\|_2 \quad (52)$$

Lower values of D_j indicate strong geometric coherence and validate the pose-preserving ability of the network under spatial shifts. Higher values imply that pose encoding failed to maintain stability, suggesting the need for regularization.

In addition to Euclidean pose deviation, angular misalignment is evaluated to capture rotational inconsistency. The cosine similarity between capsule directions offers a rotational perspective of alignment quality is expressed in Eq.(53).

$$cos(\phi_j) = \frac{v_j \cdot \hat{v}'_j}{\|v_j\| \cdot \|\hat{v}'_j\|} \quad (53)$$

This similarity score is used in subsequent steps to compute geometric consistency losses, enhancing robustness under both global and local adversarial distortions. Capsules that maintain consistent norm and direction despite spatial perturbation are rewarded, reinforcing geometry-invariant learning in structurally volatile environments.

Through repeated application of identical layers, routing coefficients, and activation constraints, the capsule outputs from x_t offer a parallel pathway for evaluating model generalization. This symmetric propagation of structurally altered input supports the enforcement of dual-objective loss functions, capturing both classification performance and pose stability across spatial transformations. The ability of capsule networks to generalize across geometric variance is thus tested directly through the activation coherence between $v_j$and $\hat{v}'_j$, providing a strong basis for enforcing geometric consistency in the adversarial training pipeline.

3.10. Geometric Consistency Loss Formulation

Geometric consistency ensures that pose information encoded in capsule vectors remains stable when the input image undergoes structured spatial transformations. In remote sensing imagery, such transformations often arise from aerial rotation, altitude variation, or alignment shifts due to sensor orientation. Enforcing consistency between capsule representations of original and transformed views allows the model to develop an internal mechanism for recognizing structural identity across spatial alterations.

Let the final class capsule output be from the original input x_c be denoted by $v_j \in R^{d_{cls}}$, and the corresponding transformed and re-aligned capsule from $x_t$be represented by $\hat{v}'_j \in R^{d_{cls}}$, as computed

through inverse geometric transformation. A consistency objective aims to minimize deviations in both magnitude and orientation between these two capsule vectors.

The most direct alignment objective is based on Euclidean distance, ensuring that the spatial difference in pose encoding is constrained. The L2-based geometric consistency loss is given as Eq.(54).

$$L_{L2} = \sum_{j=1}^{C} \left\| v_j - \hat{v}_j' \right\|_2^2 \quad (54)$$

This formulation penalizes dissimilarity in capsule embeddings caused by rotation, translation, or scaling. It treats the original and transformed capsules as two instances of the same spatial entity and reinforces the preservation of structure during capsule routing.

While distance captures vector displacement, orientation similarity is also vital for maintaining part-to-whole geometric agreement. A cosine similarity-based formulation evaluates the angular alignment between v_j and $\hat{v}_j'$. The corresponding loss is defined as Eq.(55).

$$L_{cos} = \sum_{j=1}^{C} \left(1 - \frac{v_j \cdot \hat{v}_j'}{\|v_j\| \cdot \|\hat{v}_j'\|} \right) \quad (55)$$

This term becomes minimal when the vectors are directionally aligned, providing orientation-level robustness even under spatial warping. The inclusion of angular alignment encourages the capsule model to encode stable geometric directions, which strengthens inter-class boundary confidence under adversarial conditions.

To prevent overfitting to only norm or angle-based constraints, a hybrid consistency term is introduced. This combines L2 deviation and cosine misalignment in a weighted manner, ensuring both scale and directional preservation. The combined formulation is defined as Eq.(56).

$$L_{geo} = \lambda_1 \cdot L_{L2} + \lambda_2 \cdot L_{cos} \quad (56)$$

The hyperparameters λ_1 and λ_2 allow calibration between distance and angle preservation, supporting flexible model adaptation across different terrain complexities in remote sensing scenes. Higher values of λ_1 emphasize spatial stability, while elevated λ_2 values focus on rotational fidelity.

Since capsule norms also represent categorical confidence, an additional consistency loss is enforced over vector magnitudes. The goal is to retain class-level certainty despite geometric variation. Let $p_j = \|v_j\|$ and $p_j' = \|\hat{v}_j'\|$. The norm alignment loss is formulated as Eq.(57).

$$L_{norm} = \sum_{j=1}^{C} (p_j - p_j')^2 \quad (57)$$

This ensures that semantic activation intensity does not collapse or inflate during transformation, preserving consistent classification output under different input orientations. Norm preservation contributes directly to the interpretability of capsule responses in aerial domains where label assignment requires stable recognition confidence.

To enhance robustness against small structural perturbations, a margin-aware consistency adjustment is integrated. Capsules whose pose deviation exceeds a structural threshold δ are penalized more aggressively. Let the margin-based penalty be as shown in Eq.(58).

$$L_{margin\_geo} = \sum_{j=1}^{C} max\left(0, \left\| v_j - \hat{v}_j' \right\|_2 - \delta\right)^2 \quad (58)$$

This formulation activates only when the deviation between aligned capsules surpasses the allowed structural drift δ. The term discourages unstable behavior for certain classes without constraining naturally variable features like orientation-preserving backgrounds or non-rotational categories.

A final consistency term integrates confidence-weighted penalization. Capsules with higher activation scores should show more substantial alignment. Define weights $w_j = min(p_j, p_j')$, and compute a weighted consistency objective is expressed in Eq.(59).

$$L_{weighted} = \sum_{j=1}^{C} w_j \cdot \left\| v_j - \hat{v}_j' \right\|_2^2 \quad (59)$$

This term ensures that the network prioritizes consistency for more confident predictions, enhancing semantic reliability in presence of noise or alignment distortions. Capsules that encode weak or ambiguous class presence receive reduced penalization, allowing flexibility in uncertain regions.

The total geometric consistency loss combines all structural components into a unified expression Eq.(60).

$$L_{concistency} = L_{geo} + \alpha \cdot L_{norm} + \beta \cdot L_{margin_{geo}} + \gamma \cdot L_{weighted} \quad (60)$$

Hyperparameters α, β, γ govern the contribution of each structural term, allowing task-specific tuning for classification stability and geometric alignment. This total loss directly connects capsule pose

preservation to adversarial robustness by aligning capsule embeddings across spatial views.

The enforcement of geometric consistency across transformed inputs not only reduces classification drift under spatial attack but also refines the ability of capsule networks to encode stable, invariant structural features. Each consistency component anchors capsule learning to geometric reality, preventing over-sensitivity to perturbations and preserving interpretability in complex terrain classification tasks.

3.11. Joint Loss Optimization

The GCR-SCaps framework combines multiple objectives rooted in classification accuracy, structural fidelity, and pose alignment to guide the training process. These objectives, although independently significant, must be cohesively optimized to ensure robustness under adversarial geometric perturbations. A unified joint loss is formulated to consolidate these components, allowing the model to prioritize semantic correctness and geometric stability simultaneously. This multi-objective formulation guides the learning trajectory toward solutions that are both discriminative and transformation-invariant.

Let the structural margin function define the core classification loss L_{margin}, which regulates capsule activation norms across true and false classes. This term was previously introduced as Eq.(61).

$$L_{margin} = \sum_{j=1}^{C} \left[T_j \cdot max\left(0, m^+ - p_j\right)^2 + \lambda \cdot \left(1 - T_j\right) \cdot max\left(0, p_j - m^-\right)^2 \right] \quad (61)$$

Where $T_j \in \{0,1\}$ indicates target class presence, $p_j = \|v_j\|$ reflects the norm of the class capsule, and m^+, m^- are margin thresholds that delineate confidence bounds for positive and negative classes. This loss enforces categorical alignment by separating active capsules from inactive ones.

The geometric consistency between the original and transformed capsule outputs is regulated through the multi-term consistency loss $L_{concistency}$, defined as Eq.(62).

$$L_{concistency} = L_{geo} + \alpha \cdot L_{norm} + \beta \cdot L_{marin_{geo}} + \gamma \cdot L_{weighted} \quad (62)$$

Each term in this composition plays a role in enforcing either pose alignment, norm preservation, or distance-based threshold enforcement across capsule representations. The coefficients α, β, γ control the influence of each term and are tuned based on terrain variability and noise characteristics within the remote sensing data.

Sparsity control also contributes to the joint optimization framework. Excessive activation of capsules reduces interpretability and weakens part-based semantic assignment. To combat this, an entropy-based sparsity term L_{sparse} is included as shown in Eq.(63).

$$L_{sparse} = -\sum_{i=1}^{n} p_i log(p_i) \quad (63)$$

Where p_i denotes the normalized activation probability for the capsule i. Lower entropy reflects focused activation, while higher entropy suggests diffusion. This term enhances model robustness by discouraging the inclusion of uncertain or redundant pose vectors in the routing process.

To consolidate all critical training objectives into a unified optimization target, the total joint loss L_{total} is defined as Eq.(64).

$$L_{total} = L_{margin} + \lambda_1 \cdot L_{concsistency} + \lambda_2 \cdot L_{sparse} \quad (64)$$

The hyperparameters λ_1 and λ_2 are scalar weights that determine how much geometric consistency and sparsity regularization influence the final optimization. Larger values of λ_1 lead to stricter enforcement of transformation invariance, while higher λ_2 values increase structural filtering by sparsity gating.

During training, the model parameters $\Theta = \{W_{conv}, W_{caps}, W_{route}\}$ are updated via gradient descent to minimize L_{total}. Let the batch-wise gradient be defined as Eq.(65).

$$\Theta \leftarrow \Theta - \eta \cdot \nabla_{\Theta} L_{total} \quad (65)$$

Here, η represents the learning rate and the gradient term $\nabla_{\Theta} L_{total}$ aggregates all partial derivatives across loss components. The gradients originating from the consistency loss propagate through the capsule pose transformations, ensuring that even early convolutional layers adjust to preserve spatial identity under geometric transformation.

To maintain training stability across adversarial samples and geometric variations, gradient norm clipping is introduced. Let G represent the accumulated gradient, and τ be the clipping threshold. The clipped update is computed as in Eq.(66).

$$G \leftarrow \frac{\tau}{max\left(\tau, (\|G\|_2)\right)} \cdot G \quad (66)$$

This clipping restricts the update magnitude, preventing drastic parameter shifts when encountering high-variance gradients due to spatial perturbations in remote sensing inputs. Stable training behavior is fundamental in capsule networks, where routing operations can amplify noise-sensitive gradients.

Batch normalization updates and routing coefficient adjustments are conducted simultaneously during optimization. For each iteration, the routing logits b_{ij} are updated using Eq.(67).

$$b_{ij} \leftarrow b_{ij}\hat{u}_{j|i} \cdot v_j \quad (67)$$

The dot product agreement term ensures that gradient propagation reinforces only capsule interactions with aligned pose predictions, filtering out misaligned capsule pairs through the routing mechanism. This dynamic refinement of capsule communication channels, governed by the gradient updates from the total loss, strengthens the spatial resilience of the model.

Incorporating all components within L_{total}, the GCR-SCaps training process develops a balanced internal representation that reflects both semantic relevance and structural consistency. Capsule activations become sparse, class-aligned, and invariant to geometric deformation, satisfying the robustness criteria for adversarially perturbed remote sensing scenarios. Each parameter update reflects a trade-off between classification clarity and geometric coherence, gradually shaping a decision boundary that is both discriminative and structurally grounded.

3.12. Structural Adversarial Fine-Tuning

Remote sensing environments often experience targeted adversarial distortions where minor geometric manipulations mislead classification decisions. Such perturbations may not significantly alter the pixel-level appearance but subtly disrupt structural integrity, particularly in semantic boundaries or object alignments. To enhance robustness against these spatial attacks, structural adversarial fine-tuning is employed, involving gradient-guided perturbation on geometry-sensitive features to simulate realistic adversarial conditions. Capsule representations refined through this process become more resilient to structural deception.

Let the clean input $x_c \in R^{H_0 \times W_0 \times C}$ represent the normalized remote sensing image. An adversarially perturbed input x_{adv} is constructed by introducing structural noise along the gradient direction of a geometry-specific loss. This loss is derived from the previously defined geometric consistency objective $L_{concistency}$, ensuring perturbations target alignment-sensitive properties. The adversarial example is generated using the following update as shown in Eq.(68).

$$x_{adv} = x_c + \epsilon \cdot sign\left(\nabla_{x_c} L_{consistency}\right) \quad (68)$$

Here, ϵ controls the perturbation magnitude, while the gradient $\nabla_{x_c} L_{consistency}$ directs the transformation toward maximally disturbing the pose similarity between clean and transformed capsules. The sign operator ensures computational efficiency while emphasizing alignment instability.
To ensure that perturbations reflect spatial deformation rather than random pixel noise, a structure-preserving constraint is imposed on the perturbation map. A total variation penalty is introduced to restrict high-frequency fluctuations, preserving geometric semantics in the adversarial image. Let $\Delta x = x_{adv} - x_c$, and define the structural regularization term as Eq.(69).

$$L_{tv} = \sum_{i,j} \left(\left| \Delta x_{i+1,j} - \Delta x_{i,j} \right| + \left| \Delta x_{i,j+1} - \Delta x_{i,j} \right| \right) \quad (69)$$

This regularizer maintains continuity in the adversarial field, simulating distortions such as rotations, warps, or displacements rather than noise bursts. Capsule networks tuned under this constraint learn to capture deeper geometry rather than local appearance only.

The adversarial image x_{adv} undergoes forward propagation through the capsule network. Let $v_j^{adv} \in R^{d_{cls}}$ denote the final class capsule output from x_{adv}. A stability objective is defined to ensure capsule predictions remain consistent across the clean and perturbed variants. The L2 deviation is used as a primary alignment metric Eq.(70).

$$L_{align} = \sum_{j=1}^{C} \left\| v_j - v_j^{adv} \right\|_2^2 \quad (70)$$

Minimizing L_{align} encourages the network to produce capsule embeddings that retain directional consistency under targeted structural alterations. In addition to Euclidean misalignment, angular divergence is also penalized using Eq.(71).

$$L_{ang} = \sum_{j=1}^{C} \left(1 - \frac{v_j \cdot v_j^{adv}}{\|v_j\| \cdot \|v_j^{adv}\|}\right) \quad (71)$$

This term strengthens orientation invariance by aligning rotational pose features between both representations. Capsule networks trained under angular supervision are better suited for recognizing rotated or misaligned targets in surveillance and terrain monitoring.

To jointly optimize robustness and classification reliability under perturbation, a combined adversarial objective is formulated. This includes the standard margin loss from clean input L_{margin}, the adversarial stability constraints, and a regularization penalty for adversarial magnitude control. The combined loss is expressed as Eq.(72).

$$L_{adv\_total} = L_{margin} + \lambda_1 \cdot L_{align} + \lambda_2 \cdot L_{ang} + \lambda_3 \cdot L_{tv} \quad (72)$$

The coefficients $\lambda_1, \lambda_2, \lambda_3$ modulate the influence of each adversarial component, ensuring a balanced trade-off between geometric reliability and output consistency. The total loss L_{adv_total} is backpropagated through the entire capsule architecture, refining all layers involved in pose construction, part-whole routing, and class assignment.

To improve sample diversity and prevent overfitting to a single perturbation type, randomization is introduced during adversarial generation. Each batch is perturbed using a mix of structural biases, including random affine deformation, local translation maps, and directionally constrained noise. Let P be the adversarial perturbation space and $\delta \sim P$ be a sampled perturbation mask. The adversarial image is updated as in Eq.(73).

$$x_{adv} = x_c + \epsilon \cdot sign\left(\nabla_{x_c} L_{consistency} + \delta\right) \quad (73)$$

This formulation blends gradient-based perturbations with task-specific structural biases, promoting resilience across multiple adversarial strategies without collapsing performance on clean inputs.

Capsule alignment under these perturbations is further reinforced by adjusting routing agreement thresholds. The routing logits b_{ij} are updated using a resistance-scaled dot product to suppress unstable contributions from adversarially sensitive capsules. Let ρ_i be the inverse deviation score of the capsule i, and update the routing logit as shown in Eq.(74).

$$b_{ij} \leftarrow b_{ij} + \rho_i \cdot \left(\hat{u}_{j|i} \cdot v_j^{adv}\right) \quad (74)$$

The resistance score $\rho_i = exp\left(-\|\tilde{v}_i - \tilde{v}_i^{adv}\|_2\right)$ penalizes noisy capsule connections during routing, preserving stable paths within the capsule graph under adversarial pressure.

Each iteration of fine-tuning with structurally informed adversarial samples forces the capsule architecture to encode reliable object parts, route part-to-whole relationships with increased certainty, and output classification decisions that remain invariant to crafted spatial inconsistencies. This process builds an additional defense layer within the network, empowering the GCR-SCaps architecture to sustain high performance in domains with structural attack risk, including military aerial monitoring and critical infrastructure surveillance.

3.13. Prediction and Confidence Calibration

Final classification in GCR-SCaps is executed using class capsule vectors $v_j \in R^{d_{cls}}$, where each capsule encodes not only the presence of a class but also its pose, orientation, and part-to-whole relational integrity. Capsule magnitudes $\|v_j\|$ provide a direct representation of prediction confidence, making them naturally interpretable in terms of activation strength. However, direct usage of these magnitudes for decision-making without post-processing can lead to miscalibration, particularly under structural perturbations. To ensure reliable predictions in adversarially sensitive remote sensing tasks, a systematic calibration mechanism is required.

The raw prediction score for each class is computed as the L2 norm of its corresponding capsule vector, which is represented mathematically in Eq.(75).

$$p_j = \|v_j\|_2 \quad (75)$$

The scalar $p_j \in [0,1]$ reflects the model's confidence in the presence of class j, with higher values indicating more substantial semantic alignment and spatial consistency. Classification is achieved by selecting the class with the maximum capsule norm as expressed in Eq.(76).

$$\hat{y} = arg\max_j \|v_j\|_2 \quad (76)$$

While this selection mechanism provides interpretable outputs, the distribution of capsule norms may suffer from scale drift across batches or under geometric adversarial scenarios. Without calibration, identical scores may vary in reliability

across scenes, particularly in multi-object regions where overlapping semantics confuse the network.

To correct for such inconsistencies, a temperature scaling method is introduced. This approach smooths the confidence distribution by rescaling the capsule norms before applying a softmax-like transformation. Let $T \in R^+$ denote the temperature parameter, and the calibrated class probability $\tilde{p}_j$ is defined as Eq.(77).

$$\tilde{p}_j = \frac{exp\left(\|v_j\|_2/T\right)}{\sum_k exp(\|v_k\|_2/T)} \quad (77)$$

Lower values of T produce sharper distributions, while higher values induce smoother, less confident predictions. Calibration ensures that the probabilities are interpretable and aligned with actual classification correctness. This transformation preserves relative order but improves consistency across diverse conditions.

For scenarios involving overlapping or hierarchically similar classes, a margin-preserving confidence adjustment is incorporated. A prediction gap Δ_j between the top two capsule magnitudes is used to modify the top score's sharpness, as expressed in Eq.(78).

$$\Delta_j = \|v_{j^*}\|_2 - \max_{k \neq j^*} \|v_k\|_2 \quad (78)$$

Here, j^* denotes the predicted class. A small margin Δ_j indicates a less confident decision, prompting confidence suppression. The adjusted score $\hat{p}_{j^*}$ is computed as Eq.(79).

$$\hat{p}_{j^*} = \sigma\left(\frac{\Delta_j}{\tau}\right) \cdot \hat{p}_{j^*} \quad (79)$$

The sigmoid function $\sigma(\cdot)$ and scaling parameter τ modulate the contribution of the prediction gap to the confidence output. This formulation penalizes predictions where class distinctions are ambiguous, improving trust in the network's output.

To enhance region-level consistency, spatial attention weights derived from capsule contributions are incorporated. For each class capsule, a relevance score is computed by aggregating routing agreement values between primary and class capsules. Let $\alpha_j = \sum_i c_{ij}$, where c_{ij} represents the routing coefficient, which is expressed as Eq.(80).

$$\alpha_j = \sum_i c_{ij} \quad (80)$$

The score α_j reflects how strongly lower-level parts support the decision for the class j. A low α_j suggests weak structural support, even if the norm $\|v_j\|$ is high. The final calibrated confidence C_j integrates of this routing consensus are shown in Eq.(81).

$$C_j = \tilde{p}_j \cdot \alpha_j \quad (81)$$

This weighting adjusts prediction strength according to structural support, particularly beneficial in adversarial conditions where pose-preserving features may become sparse or misaligned.

For evaluation and interpretability, the entropy of the final calibrated distribution is computed to assess decision certainty. Let H denote the entropy across all class predictions which is represented mathematically in Eq.(82).

$$H = -\sum_j C_j log\left(C_j + \epsilon\right) \quad (82)$$

This entropy measure identifies uncertain predictions and highlights input cases that may require human intervention or deeper post-processing. Lower entropy signifies confident classification, while higher values suggest ambiguity, potentially arising from adversarial misalignment.

4. DATASET DESCRIPTION

EuroSat is a publicly available satellite image dataset built from Sentinel-2 sensor readings, comprising 27,000 RGB image tiles. Each sample measures 64x64 pixels and is annotated across 10 distinct land cover classes including River, Residential, Forest, and Highway. The dataset encompasses scenes from across Europe, capturing diverse spatial patterns influenced by various ecological and climatic zones. All images have been preprocessed to maintain uniform resolution, ensuring model readiness for deep learning applications. EuroSat is well-suited for tasks requiring robust classification under natural image noise and adversarial scenarios, offering rich contextual backgrounds and class overlaps. The presence of intricate textures, spectral variations, and fine boundary regions encourages the development of attention-based and adversarially resilient models. Each class contains a balanced number of images, allowing fair comparative evaluation. Given its high-quality annotation, spatial consistency, and multi-domain representation, the dataset has become essential for assessing classification accuracy, interpretability, and generalizability of remote sensing algorithms. The design supports research in land surveillance, environmental change tracking, and adversarial defense for earth observation analytics.

5. RESULTS AND DISCUSSIONS

Results and discussions provide an in-depth analysis of the experimental findings, emphasizing the comparative evaluation of classification models for remote sensing image classification under structural adversarial conditions. This section examines classification accuracy and F-measure, two pivotal metrics that reflect the predictive strength and consistency of a model in identifying spatial classes with minimal distortion. These performance indicators reveal how well the models can preserve structural semantics even in adversarially manipulated input scenarios.

Classification accuracy measures the overall percentage of correctly classified instances, including both true positives and true negatives, relative to the total test samples. A model with higher accuracy can effectively separate meaningful features from noise or perturbations. The MSRF model achieves a classification accuracy of 55.062%, showing moderate detection strength under clean conditions. A3OD improves upon this with 58.696%, reflecting a better ability to generalize across structurally complex samples. The GCR-SCaps model achieves a significant leap with 74.342%, demonstrating that enforcing geometric consistency and capsule sparsity enables a more stable classification boundary even in structurally deceptive contexts. The increase in accuracy indicates a strong capability to preserve object integrity under attack.

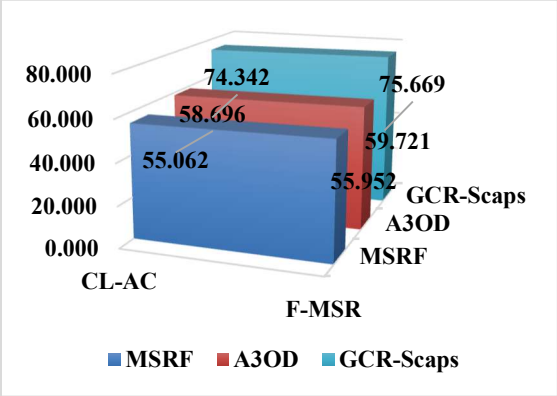

Fig.1 CL-AC And F-MSR

F-measure, calculated as the harmonic mean of precision and recall, evaluates how well the model balances true positive detections while minimizing false positives and false negatives. MSRF scores an F-measure of 55.952, indicating limited capability in preserving class-sensitive responses. A3OD shows modest improvement with a value of 59.721, benefiting from feature augmentation. GCR-SCaps, on the other hand, secures an F-measure of 75.669, highlighting a consistent and balanced classification response across various object deformations. This performance gain arises from the model's ability to route structurally critical features through sparse capsule connections, improving detection granularity.

Fig.1 visually supports the numerical outcomes, validating the superiority of GCR-SCaps in maintaining classification fidelity and consistency under adversarial conditions, making it a robust defense-oriented framework for remote sensing applications.

To evaluate model performance across core classification quality metrics, particularly focusing on Fowlkes–Mallows Index (FMI) and Matthews Correlation Coefficient (MCC). These metrics provide deeper insights into structural integrity preservation and classification balance, particularly in remote sensing contexts affected by adversarial deformations. While classification accuracy and F-measure assess overall and harmonic detection strength, FMI and MCC offer more rigorous evaluation by quantifying the correlation and concordance between predicted and true labels with a heightened sensitivity to imbalanced and perturbed distributions.

FMI reflects the geometric mean of precision and recall, serving as a robust measure of the balance between relevance and completeness in prediction. Higher FMI values indicate a model's capability to reduce both false positives and false negatives simultaneously, ensuring classification outputs closely reflect structural consistency. The MSRF model records an FMI of 56.239, signifying restricted detection fidelity in high-variance adversarial inputs. A3OD improves with an FMI of 59.821, attributing better feature sensitivity across spatially variant classes. GCR-SCaps achieves a substantial rise to 75.758, validating that geometric consistency and sparse capsule routing substantially enhance the model's ability to align predictions with true spatial structures.

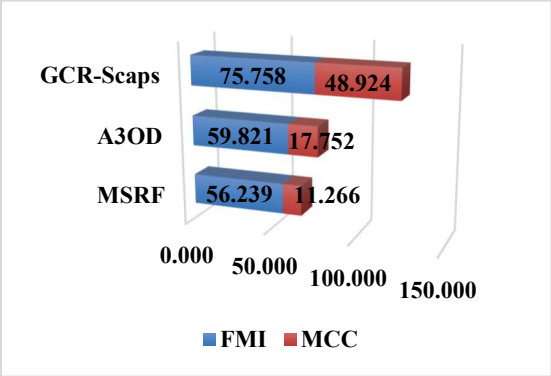

Fig.2 FMI And MCC

Fig.2 illustrates the outcomes of MCC and FMI in a pictorial format. MCC is a correlation-based metric that considers all four confusion matrix outcomes true positives, true negatives, false positives, and false negatives yielding a single, interpretable score even in class-imbalanced environments. MSRF records an MCC of 11.266, highlighting weak agreement between prediction and ground truth. A3OD shows slight improvement with 17.752, reflecting moderate structural recovery. GCR-SCaps registers 48.924, confirming its effective balancing of both major and minor class detections. This indicates the model's success in minimizing random or biased misclassifications through structurally reinforced capsule representations. The FMI and MCC results support the architectural advantage of GCR-SCaps in maintaining classification fidelity under adversarial stress, showcasing structural alignment and consistent detection behavior.

Precision and recall serve as core indicators for evaluating classification reliability, especially in structurally perturbed remote sensing images. Precision measures the proportion of correctly predicted positives among all predicted positives, offering a view into how well the model avoids false alarms. Recall, on the other hand, captures the ability of the model to identify all actual positives, emphasizing sensitivity to relevant features despite adversarial distortions.

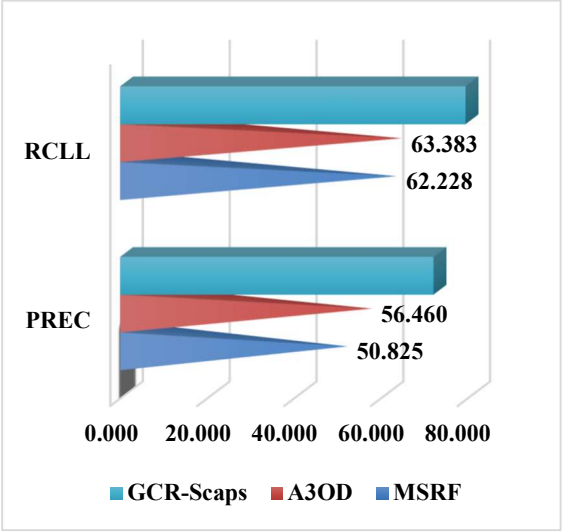

Fig.3 PREC And RCLL

Fig.3 exhibits the outcomes of GCR-SCaps under the metrics of precision and recall. The MSRF model records a precision of 50.825%, indicating moderate confidence in its positive predictions but a relatively higher susceptibility to false positives. A recall of 62.228% reflects its limited capability in fully recovering relevant spatial classes, particularly when under structural perturbation. A3OD exhibits improved precision at 56.460%, which signifies better alignment of predictions with genuine positive instances. Its recall, marginally higher at 63.383%, suggests slight gains in detecting relevant features across input variations.

GCR-SCaps significantly outperforms both with a precision of 72.162%, demonstrating a refined selection of true positives and minimal mislabeling. This improvement arises from the structural routing and sparsity-enforced capsules that preserve critical feature boundaries. Its recall peaks at 79.533%, affirming robust detection of structurally valid targets even in geometrically distorted inputs. The dual gain in precision and recall confirms that GCR-SCaps achieves a balanced and resilient classification behavior, maintaining integrity in both selective and exhaustive predictions. These outcomes validate the structural reinforcement and geometric alignment mechanisms embedded within the GCR-SCaps framework, ensuring enhanced adversarial resistance and semantic preservation across diverse remote sensing conditions.

5. CONCLUSION

The proposed GCR-SCaps framework introduces a structurally aware classification architecture that

addresses adversarial vulnerabilities in remote sensing image interpretation. By incorporating sparsity-driven capsule activations and geometric consistency regularization, the model establishes a defense-sensitive pathway that preserves object structures across spatial perturbations. The integration of sparse capsule routing enhances the model's ability to isolate class-relevant features, minimizing the influence of noisy or misleading patterns introduced by adversarial manipulations. Geometric transformation-based dual-path propagation and consistency loss computation further ensure that learned representations remain invariant to rotations, translations, and affine alterations, reinforcing spatial robustness. Experimental results validate the superiority of GCR-SCaps across key performance metrics, demonstrating substantial improvements in classification accuracy, F-measure, MCC, and FMI when compared to existing models such as MSRF and A3OD. The elevated recall and precision scores confirm the model's capacity to maintain both detection sensitivity and prediction reliability under structurally distorted inputs. The consistent performance across clean and adversarial samples highlights the framework's resilience in retaining topological fidelity. The modular design of GCR-SCaps, combining dynamic routing with structure-aware loss optimization, enables flexible deployment across diverse remote sensing scenarios where adversarial threats are prevalent. The model's ability to generalize across varying geometric alterations without compromising classification confidence underscores its potential for critical applications such as aerial surveillance, disaster mapping, and defense reconnaissance. Overall, the GCR-SCaps framework contributes a structurally grounded and performance-validated adversarial defense mechanism, advancing the reliability and interpretability of remote sensing image classifiers in high-risk operational environments.

REFERENCES:

[1] S. Liu, Z. Lian, S. Zhang, and L. Xiao, "Adversarial purification of information masking," *Neurocomputing*, vol. 621, p. 129214, 2025, doi: https://doi.org/10.1016/j.neucom.2024.129214.

[2] M. Jaber, S. Elmi, M. Nassar, and W. El Hajj, "Introducing Residual Networks to Vision Transformers for Adversarial Attacks.," *Procedia Comput Sci*, vol. 246, pp. 423–432, 2024, doi: https://doi.org/10.1016/j.procs.2024.09.421.

[3] X. Wei and M. Yuan, "Adversarial pan-sharpening attacks for object detection in remote sensing," *Pattern Recognit*, vol. 139, p. 109466, 2023, doi: https://doi.org/10.1016/j.patcog.2023.109466.

[4] M. Al-Fawa'reh, J. Abu-khalaf, N. Janjua, and P. Szewczyk, "Detection of on-manifold adversarial attacks via latent space transformation," *Comput Secur*, vol. 154, p. 104431, 2025, doi: https://doi.org/10.1016/j.cose.2025.104431.

[5] C. Shi, Y. Liu, M. Zhao, C.-M. Pun, and Q. Miao, "Attack-invariant attention feature for adversarial defense in hyperspectral image classification," *Pattern Recognit*, vol. 145, p. 109955, 2024, doi: https://doi.org/10.1016/j.patcog.2023.109955.

[6] G. Tang, W. Yao, C. Li, T. Jiang, and S. Yang, "Black-box adversarial patch attacks using differential evolution against aerial imagery object detectors," *Eng Appl Artif Intell*, vol. 137, p. 109141, 2024, doi: https://doi.org/10.1016/j.engappai.2024.109141.

[7] A. D. M. Ibrahum, M. Hussain, and J.-E. Hong, "Deep learning adversarial attacks and defenses in autonomous vehicles: a systematic literature review from a safety perspective," *Artif Intell Rev*, vol. 58, no. 1, p. 28, 2024, doi: 10.1007/s10462-024-11014-8.

[8] T. Yang, S. Xiao, and J. Qu, "D3GNN: Double dual dynamic graph neural network for multisource remote sensing data classification," *International Journal of Applied Earth Observation and Geoinformation*, vol. 139, p. 104496, 2025, doi: https://doi.org/10.1016/j.jag.2025.104496.

[9] J. Feng, H. Luo, and Z. Gu, "Improving semi-supervised remote sensing scene classification via Multilevel Feature Fusion and pseudo-labeling," *International Journal of Applied Earth Observation and Geoinformation*, vol. 136, p. 104335, 2025, doi: https://doi.org/10.1016/j.jag.2024.104335.

[10] Y. Liang, S. Cao, J. Zheng, X. Zhang, J. Huang, and H. Fu, "Low Saturation Confidence Distribution-based Test-Time Adaptation for cross-domain remote sensing image classification," *International Journal of Applied Earth Observation and Geoinformation*, vol. 139, p. 104463, 2025, doi: https://doi.org/10.1016/j.jag.2025.104463.

[11] S. Rahman, S. Pal, A. Habib, L. Pan, and C. Karmakar, "Attack-data independent defence mechanism against adversarial attacks on ECG signal," *Computer Networks*, vol. 258, p.

111027, 2025, doi: https://doi.org/10.1016/j.comnet.2024.111027.
[12] S. Chowdhury and M. Das, "A Performance-Preserving Scheme for Classifying Differentially Private Remote Sensing Images," *Procedia Comput Sci*, vol. 260, pp. 761–767, 2025, doi: https://doi.org/10.1016/j.procs.2025.03.256.
[13] K. M and T. Poongodi, "Investigation of Security Attacks in IoMT Devices and Federated Learning as a Mitigation Strategy," *Procedia Comput Sci*, vol. 258, pp. 3426–3435, 2025, doi: https://doi.org/10.1016/j.procs.2025.04.599.
[14] K. A. Dhanya, S. Vajipayajula, K. Srinivasan, A. Tibrewal, T. S. Kumar, and T. G. Kumar, "Detection of Network Attacks using Machine Learning and Deep Learning Models," *Procedia Comput Sci*, vol. 218, pp. 57–66, 2023, doi: https://doi.org/10.1016/j.procs.2022.12.401.
[15] H. Feng *et al.*, "Security of target recognition for UAV forestry remote sensing based on multi-source data fusion transformer framework," *Information Fusion*, vol. 112, p. 102555, 2024, doi: https://doi.org/10.1016/j.inffus.2024.102555.
[16] J. Li, M. Ni, Y. Dong, T. Zhu, Y. Gong, and W. Liu, "AICAttack: Adversarial Image Captioning Attack with Attention-based Optimization," *Machine Intelligence Research*, 2025, doi: 10.1007/s11633-024-1535-z.
[17] K. Barik, S. Misra, and L. Fernandez-Sanz, "Adversarial attack detection framework based on optimized weighted conditional stepwise adversarial network," *Int J Inf Secur*, vol. 23, no. 3, pp. 2353–2376, 2024, doi: 10.1007/s10207-024-00844-w.
[18] R. Cheng, X. Wang, F. Sohel, and H. Lei, "Topology-aware universal adversarial attack on 3D object tracking," *Visual Intelligence*, vol. 1, no. 1, p. 31, 2023, doi: 10.1007/s44267-023-00033-8.
[19] R. Majadas, J. García, and F. Fernández, "Clustering-based attack detection for adversarial reinforcement learning," *Applied Intelligence*, vol. 54, no. 3, pp. 2631–2647, 2024, doi: 10.1007/s10489-024-05275-7.
[20] F. Guo, Z. Sun, Y. Chen, and L. Ju, "Towards the transferable audio adversarial attack via ensemble methods," *Cybersecurity*, vol. 6, no. 1, p. 44, 2023, doi: 10.1186/s42400-023-00175-8.
[21] H. Fan and G. Wei, "Multi-spectral remote sensing image fusion method based on gradient moment matching," *Systems and Soft Computing*, vol. 6, p. 200108, 2024, doi: https://doi.org/10.1016/j.sasc.2024.200108.
[22] R. Huang, L. Chen, J. Zheng, Q. Zhang, and X. Yu, "Adversarial Attacks Against Object Detection in Remote Sensing Images," in *Artificial Intelligence Security and Privacy*, J. Vaidya, M. Gabbouj, and J. Li, Eds., Singapore: Springer Nature Singapore, 2024, pp. 358–367.