Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2024.0429000

AINet: Integrating Mamba and CBAM for Enhanced Camouflage Object Detection

HENRY O. VELESACA[1,4], ANDREA MERO P.[1,5], ABEL REYES-ANGULO[3], and ANGEL D. SAPPA[1,2]
[1]ESPOL Polytechnic University, FIEC, CIDIS, Campus Gustavo Galindo, 090902, Guayaquil, Ecuador (e-mail: hvelesac@espol.edu.ec, anmero@espol.edu.ec, asappa@espol.edu.ec)
[2]Computer Vision Center, Universitat Autònoma de Barcelona, 08193-Bellaterra, Barcelona, Spain (e-mail: asappa@cvc.uab.es)
[3]Michigan Technological University, Houghton, MI, USA (e-mail: areyesan@mtu.edu)
[4]Software Engineering Department, Research Center for Information and Communication Technologies (CITIC-UGR), University of Granada, 18071, Granada, Spain (e-mail: hvelesaca@correo.ugr.es)
[5]Università della Svizzera Italiana, Via Giuseppe Buffi 13, 6900, Lugano, Switzerland (e-mail: andrea.mero@usi.ch)

Corresponding author: Henry O. Velesaca (e-mail: hvelesac@espol.edu.ec).

This work is supported in part by the Air Force Office of Scientific Research Under Award FA9550-24-1-0206; in part by the ESPOL project "Advancing Camouflaged Object Detection with a cost-effective Cross-Spectral vision system (ACODCS)" (CIDIS-003-2024).

ABSTRACT This paper introduces AINet, a novel deep learning architecture designed for detecting camouflaged objects in complex and diverse environments. The objective of this work is to design an end-to-end camouflaged object detection architecture that simultaneously captures long-range dependencies and refines subtle camouflage cues, improving segmentation accuracy and boundary delineation across both standard COD benchmarks and real-world agricultural scenarios. AINet leverages the strengths of Mamba, an efficient sequential state model for capturing long-range dependencies, and the Convolutional Block Attention Module (CBAM) for feature refinement through attention mechanisms. Detecting camouflaged objects is a significant challenge across a wide range of real-world applications, including surveillance, security, medical imaging, and autonomous systems, where objects of interest may blend into their backgrounds and evade conventional detection methods. To demonstrate its effectiveness, AINet is evaluated on multiple datasets, including standard camouflaged object detection benchmarks such as CAMO, COD10K, and NC4K, as well as domain-specific datasets (such as pest and fruit detection). Experimental results show that AINet outperforms existing state-of-the-art models. The implementation is publicly available on GitHub for reproducibility: https://github.com/hvelesaca/AINet.

INDEX TERMS Camouflaged object detection, fruit harvest, pest detection, precision agriculture.

I. INTRODUCTION

Detecting camouflaged objects remains a persistent and challenging problem in computer vision [48]. Camouflaged objects are characterized by their ability to blend seamlessly into their surroundings, making them difficult to distinguish using conventional detection methods [30]. This challenge is prevalent in a wide array of real-world scenarios, including surveillance, security, medical imaging, autonomous vehicles, and environmental monitoring, where accurate identification of hidden or obscured objects is critical for decision-making and safety [21], [43]. Moreover, camouflaged object detection not only demands robustness to variations in illumination, scale, and background clutter, but also the capacity to infer subtle semantic and structural differences that may not be immediately apparent at the pixel level, thereby pushing current detection paradigms to their limits.

Traditional object detection algorithms often struggle with camouflaged targets due to the minimal contrast and ambiguous boundaries between the object and its background [9], [44]. Recent advances in deep learning have led to significant improvements in object detection and segmentation; however, the unique nature of camouflage still poses obstacles [6]. Addressing these challenges requires models capable of capturing both global context and fine-grained details, as well as mechanisms to focus attention on subtle cues that differentiate camouflaged objects from their environments [4].

This paper presents AINet, a novel deep learning architecture specifically designed to improve the detection of camouflaged objects. AINet integrates two powerful components: Mamba [11], an efficient sequential state model capable of modeling long-range dependencies, and the Convolutional Block Attention Module (CBAM) [47], which refines feature representations using spatial and per-channel attention mechanisms. By combining these elements, AINet can effectively highlight and segment camouflaged objects, even in highly complex scenes.

The key contributions of this work include:

- A COD-oriented encoder–decoder architecture, referred to as AINet, is proposed; it integrates selective state-space modeling (Mamba) for efficient long-range dependency modeling with attention-based feature refinement (CBAM) to emphasize subtle camouflage cues.
- A comprehensive evaluation of COD standard benchmarks (CAMO, COD10K, NC4K) and agricultural case studies is offered, including quantitative, qualitative, and ablation analyses to validate the contribution of each component.
- An extensive ablation studies is provided, it demonstrates that the synergy of Mamba and CBAM modules, as well as multi-level deep supervision, is critical for optimal segmentation accuracy and robust edge delineation.
- A validation process across multiple datasets and evaluation metrics, demonstrating substantial improvements in accuracy, boundary precision, and overall segmentation mask quality.

The manuscript is organized as follows. Section II introduces related work, recent SOTA COD techniques, and methods that address the problem of the COD approach. Section III presents the proposed architecture. Then, Section IV shows the experimental results on different datasets, and an in-depth ablation study is presented in Section V. Finally, discussion and conclusions are given in Section VI and Section VII respectively.

II. RELATED WORKS

This section reviews key computer vision-based methods for camouflaged object detection (COD), analyzing their contributions and limitations across diverse application domains.

Detecting camouflaged objects is a persistent challenge in computer vision, as these objects are characterized by their ability to blend seamlessly into complex backgrounds, making them difficult to distinguish using conventional detection techniques. Recent advances in deep learning have significantly improved image segmentation and object detection, enabling more accurate and efficient identification of camouflaged targets in a variety of real-world scenarios, such as surveillance, security, medical imaging, environmental monitoring, and autonomous systems.

One of the pioneering state-of-the-art COD techniques is the Search Identification Network (SINet) [10], which introduces a two-stage framework inspired by predator hunting behavior, consisting of a Search Module (SM) and an Identification Module (IM). SINet presents a simple yet effective end-to-end architecture based on the richly annotated COD10K dataset, achieving visually appealing results compared to existing baselines. Building on this, SINet-V2 [9] enhances the original design by incorporating densely connected layers and a receptive field component to better capture multi-level features. This improved architecture demonstrates competitive performance and broader applicability, including potential use in military, security, and wildlife conservation, where the detection of concealed objects is essential [17].

Another notable contribution is the SegMaR technique [20], which introduces a multistage iterative refinement framework for camouflaged object detection, simulating the coarse-to-fine detection process of the human visual system. The framework consists of three core steps: *Segment*, *Magnify*, and *Reiterate*. Initially, a camouflaged segmentation network generates a preliminary mask for the object. The *Magnify* module then adaptively enlarges the object region using attention-based sampling, making it more distinct within the image. Finally, the *Reiterate* module refines the segmentation through iterative feedback, progressively capturing finer details, especially for small or highly camouflaged objects. The architecture also incorporates a distraction module to disentangle foreground and background features, and employs parallel decoders to focus on key object regions and contours. However, SegMaR cannot be trained end-to-end [14], which restricts its adaptability in certain scenarios.

COD approaches face two primary challenges: (1) intrinsic similarity (IS), where objects visually resemble the background, and (2) edge disruption (ED), which results in unclear boundaries. Biologically inspired methods like SINet [10] and SegMaR [20] attempt to address these by mimicking predator hunting behavior or human visual cognition through multistage processing. However, these approaches often struggle with complex camouflage patterns and may fail to capture the subtle cues needed to resolve IS and ED effectively. To address these limitations, the FEature Decomposition and Edge Reconstruction (FEDER) technique [12] adopts a targeted two-stage strategy. It uses learnable wavelets to decompose image features and identify the most informative frequency bands via a frequency attention and feature aggregation module. To address ED, it introduces an auxiliary edge reconstruction task inspired by differential equations, improving boundary precision and overall detection accuracy. Despite promising results, FEDER's reliance on edge reconstruction may be less effective when camouflaged objects have very similar textures to their surroundings, leading to potential false negatives.

The field of COD has seen remarkable progress in recent years. Foundational work by Qin et al. [40] on salient object detection has profoundly influenced the evolution of COD methodologies. In 2022, Chen et al. [2] proposed a context-aware cross-level fusion approach that enhanced camouflaged object identification accuracy, while Chen et al. [3] introduced a boundary-guided network to improve edge and feature detection. Liu et al. [29] further advanced the field by addressing aleatoric uncertainty modeling in COD. More recently, high-resolution iterative feedback networks [16], edge-aware networks [41], and deep gradient learning techniques [18] have contributed to significant improvements in detection efficiency and accuracy. The scope of COD research continues to expand, with recent work such as PlantCamo [52] targeting specialized applications in plant camouflage detection.

While much of the research has focused on general-purpose COD, these advances have also been adapted to address domain-specific challenges. For example, in agricultural environments, camouflage plays a critical role in both pest and fruit detection, complicating monitoring and management tasks [26]. Recent studies have explored the use of advanced deep learning architectures for these applications. Meng et al. [37] addressed camouflaged pest instance segmentation by combining Pyramid Vision Transformer (PVT) and Mask R-CNN, leveraging PVT's hierarchical feature extraction and Mask R-CNN's instance-level segmentation capabilities. Similarly, Evangelista et al. [6] introduced FCNet, a transformer-based, context-aware segmentation framework for detecting camouflaged fruits in complex orchard environments. These works demonstrate the potential of adapting general COD methodologies to agriculture, though challenges remain regarding generalization to diverse conditions and the need for large labeled datasets.

Collectively, these advances have established more robust and accurate camouflaged object detection systems, setting new benchmarks in the field and paving the way for future research and practical implementations. In the following, we highlight how these general advances have been adapted and extended to address specific challenges in agricultural environments, such as pest and fruit detection.

III. PROPOSED AINET

Consistent with recent advances in COD [9], [10], [22], [38], [57], an encoder-decoder pipeline is adopted for the proposed AINet architecture. The framework is designed to be end-to-end trainable, as illustrated in Fig. 1.

A. OVERALL ARCHITECTURE

AINet is constructed by integrating Mamba and CBAM modules within an encoder-decoder design tailored for COD. The selective state space modeling capabilities of Mamba are combined with the feature refinement provided by the Convolutional Block Attention Module (CBAM), enabling effective identification of objects that visually blend into their surroundings.

Encoder. A PVTv2-B2 backbone [45] is employed as the encoder, providing hierarchical feature representations at multiple scales. From the input RGB image $I \in R^{H \times W \times 3}$, feature maps at four different resolutions are extracted, capturing both low-level details and high-level semantic information necessary for camouflaged object detection.

PVTv2-B2 is adopted as the encoder because COD demands both fine-grained, multi-scale detail extraction and global context modeling to distinguish targets that blend into the background. In addition, several representative COD pipelines that rely on ResNet50-family backbones, as reported in Table 4 (e.g., LSR, MGL, PFNet, TINet, UGTR), achieve lower benchmark accuracy than the proposed AINet on CAMO, COD10K, and NC4K. Although overall performance depends on the full architecture rather than the backbone alone, these results support using a modern hierarchical encoder for this task.

Decoder. A progressive refinement strategy is utilized in the decoder, where Decoder Blocks upsample features from the previous stage and aggregate them with skip connections from the encoder. Feature Aggregation is performed using 1×1 and 3×3 convolutions. Multiple segmentation heads at different decoder levels are incorporated to provide deep supervision, enhancing gradient flow and feature learning.

Loss Function. The loss function proposed by [46] is adopted, following established practices [9], [42]. The predictions generated by the decoder are denoted as $\{P_i\}_{i=0}^{3}$. During training, each prediction Pi is resized to the original input size and supervised using a combination of binary cross-entropy (BCE) loss [5] and intersection-over-union (IoU) loss [35]. The total loss is computed by summing the losses from all decoder stages, as follows, where GT denotes the ground truth annotation:

$$\mathcal{L}(P, GT) = \sum_{i=0}^{3} \mathcal{L}_{BCE}(P_i, GT) + \mathcal{L}_{IoU}(P_i, GT). \quad (1)$$

To further enhance feature learning at different scales, deep supervision with multiple segmentation heads is employed. Each decoder level produces a segmentation map supervised by the ground truth, and the final prediction is obtained by averaging the outputs from all levels. This multi-level supervision strategy enables the learning of robust features for detecting camouflaged objects across varying scales and concealment levels.

B. MAMBA AND CBAM BLOCKS

A core component of AINet is the Mamba Block, in which residual learning is combined with state space modeling [11], following the approach in [33]. The Mamba Block is composed of Residual Blocks (two consecutive blocks with instance normalization and LeakyReLU activation to enhance feature representation while preserving spatial information); Dual-Branch Processing, including a Mamba Branch (processing features via linear projection, 1D convolution, and the Mamba state space model to capture long-range dependencies) and an Activation Branch (applying SiLU activation for local information preservation); and a Hadamard Product, where outputs from both branches are combined through element-wise multiplication for adaptive feature fusion. Figure 1 illustrates the Mamba Block.

On the other hand, the Convolutional Block Attention Module (CBAM) [47] is also incorporated to refine features by applying both channel and spatial attention. Channel attention is computed using average and max pooling, followed by a shared MLP to generate channel-wise attention weights, while spatial attention is generated through channel pooling and convolution to produce a spatial attention map.

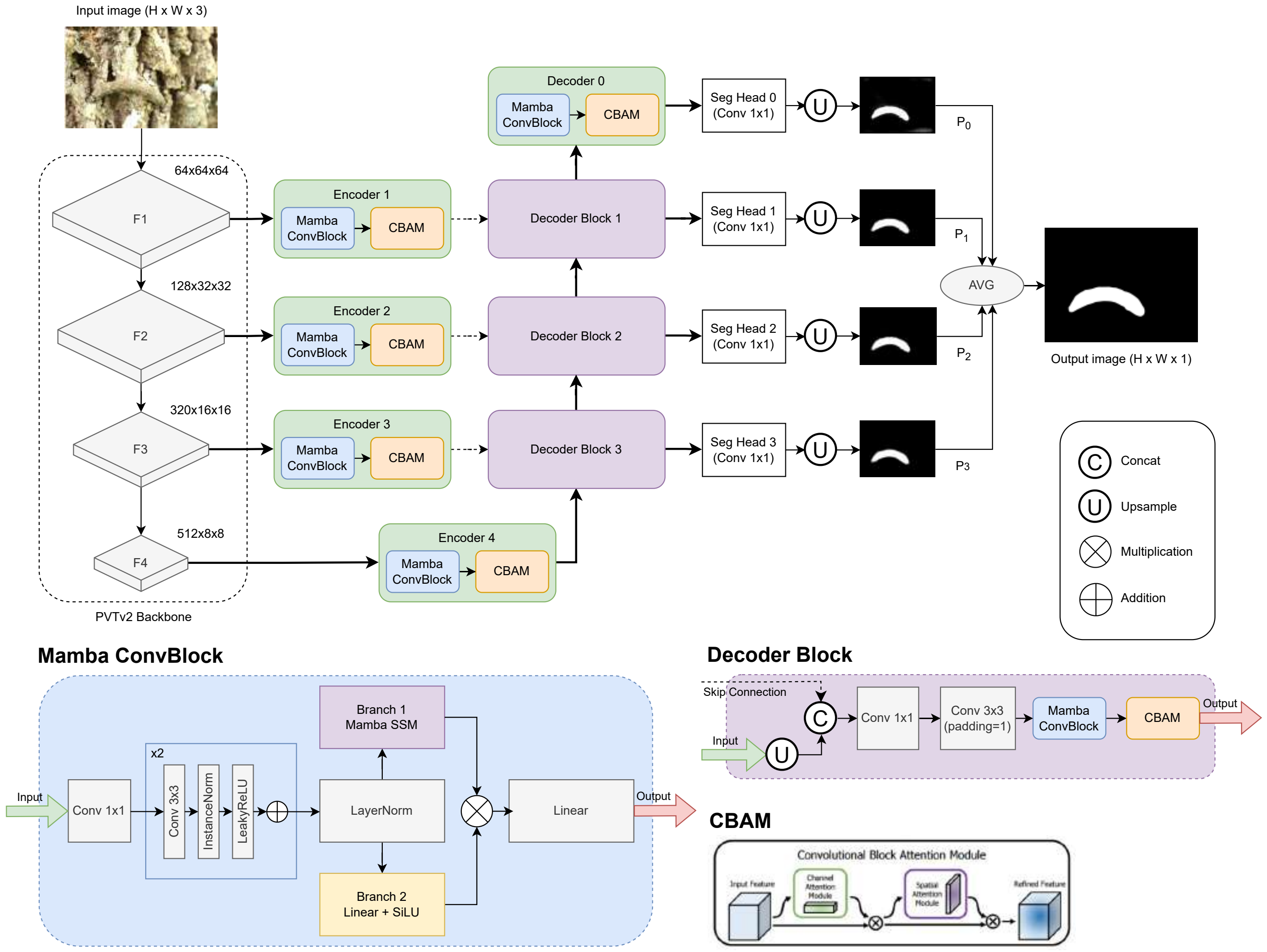

FIGURE 1. The overall architecture of the proposed AINet.

C. IMPLEMENTATION DETAILS

AINet has been implemented using the PyTorch library. The encoder is initialized with a PVTv2-B2 backbone [45] pre-trained on ImageNet. AdamW is used for optimization, with a weight decay of $1e^{-4}$. The initial learning rate is set to $1e^{-4}$ and is scheduled using cosine annealing. Input images are resized to 352×352 for both training and inference. The model is trained end-to-end for 100 epochs with a batch size of 16 on an NVIDIA TESLA P100 GPU. All experiments are conducted on the Kaggle platform[1].

Datasets. To comprehensively evaluate the effectiveness and generalization of AINet, experiments have been conducted on both standard benchmark datasets and domain-specific agricultural datasets. For general COD evaluation, three widely recognized benchmark datasets have been utilized: CAMO [25], [49], which contains 2,500 images balanced between camouflaged and non-camouflaged objects; COD10K [9], [10], which offers 5,066 camouflaged, 3,000 background, and 1,934 non-camouflaged images; and NC4K [32], contributing 4,121 images exclusively for testing.

Type	Dataset	Number of images		
		Train	Val	Test
Benchmark	CAMO [25], [49]	797	203	250
	COD10K [9], [10]	2435	605	2026
	NC4K [32]	-	-	4121
Agriculture	Cotton Bollworm [37]	856	161	56
	Mango [24]	1944	486	230

TABLE 1. Distribution of the datasets used in experimental results

Following established protocols by [10], [38], we use 1,000 images from CAMO and 3,040 images from COD10K for model development (train+val). Specifically, as summarized in Table 1, CAMO uses 797/203/250 images for train/val/test, and COD10K uses 2,435/605/2,026 images for train/val/test. NC4K is used exclusively for testing (4,121 images).

To further assess the applicability of AINet, two case stud-

[1] https://www.kaggle.com/

ies in agricultural scenarios have been selected. Hence, two domain-specific datasets are included: the Cotton Bollworm dataset [37], which contains 1,073 camouflaged pest images, and the Mango dataset [24], comprising 2,660 patch images for fruit detection. Figure 2 presents sample images from benchmark and agricultural datasets. The distribution of all datasets used in this study is summarized in Table 1.

Metrics. This study employs five widely recognized evaluation metrics to evaluate COD performance. These metrics provide a comprehensive assessment criterion for analyzing detection accuracy and effectiveness across different models. The Structure-measure (S_α) [7], weighted F-measure (F_β^w) [34], Mean Absolute Error (M) [39], E-measure (E_ϕ) [8], and F-measure (F_β) [1]. The S_α metric quantifies the structural similarity between prediction and GT maps. The F_β^w represents an enhanced evaluation metric that extends the traditional F_β by incorporating spatial weights, providing a better assessment of segmentation quality with emphasis on boundary accuracy and location-based importance of detected pixels. The M metric focuses on pixel-level error evaluation between the normalized prediction and GT. The E_ϕ metric simultaneously evaluates the global and local accuracy of COD based on human visual perception mechanisms. The F_β provides a synthetic measure that considers both precision and recall components. For both F-measure and E-measure metrics, different scores can be obtained according to different precision-recall pairs. This leads to the computation of mean F-measure (F_β^{mean}). Similarly, the E-measure utilizes mean variants, denoted as E_ϕ^{mean}, which are also employed as evaluation metrics.

Training details. This subsection summarizes the training configuration used across all COD techniques used to compare with AINet. Table 2 reports the optimizer, learning rate, batch size, number of epochs, scheduler, and loss functions employed per model.

IV. RESULTS

This section presents a comprehensive evaluation of AINet, focusing first on standard SOTA COD benchmarks and subsequently on agricultural datasets as a specialized case study. Both quantitative and qualitative analyses are provided, along with ablation studies to assess the contribution of key architectural components.

A. QUANTITATIVE EVALUATION

Table 3 introduces the deployment-oriented efficiency evaluation, reporting FLOPs, inference time, and parameters. Under a consistent input resolution, these results provide a practical view of AINet's computational footprint relative to representative COD baselines.

To validate the level of generalization, the proposed AINet has been compared with 26 SOTA COD models, which are listed in Table 4 (*1st column*). All prediction results for CAMO [25], [49], COD10k [9], [10], and NC4K [32] datasets are either directly obtained from the original papers for fair evaluation. As shown in Table 4, the proposed AINet architecture outperforms all 26 SOTA techniques across all three benchmark datasets and evaluation metrics. Taking into account the top 3 best techniques in different metrics, AINet surpasses techniques such as SINet-V2 [9], SegMaR [20], BGNet [3], OCENet [29], DGNet [18], FEDER [12], and UJSCOD-V2 [27]. Among the SOTA techniques, AINet outperforms recent approaches such as GenSAM [15], which employs cross-modal chains of thought prompting to generate visual prompts (e.g., CLIP and BLIP2) and progressively produces masks, iteratively refining detection results. Another recent method that AINet surpasses is UCOS-DA [55], which implements source-free unsupervised domain adaptation using DINO through a foreground-background contrastive self-adversarial approach.

Unlike CAMO, COD10k or NC4K, the prediction results for the Cotton Bollworm and Mango datasets are generated using architectures with codes and weights available from the official author pages; for this case, the default configuration, training and inference code published by the authors are followed.

AINet demonstrates exceptional performance for pest detection applications, as evidenced by Table 5, which shows experimental results from the proposed AINet architecture compared to existing SOTA COD techniques on the Cotton Bollworm dataset. AINet achieves remarkable metrics with a structure measure (S_α) of 0.8787, F-measure (F_β^w) of 0.8428, and the lowest MAE (M) at 0.0132. Notably, our approach excels in enhanced boundary detection with the highest E-measure scores (E_ϕ^{adp} of 0.9691 and E_ϕ^{mean} of 0.9727) and F-measure values (F_β^{adp} of 0.8330 and F_β^{mean} of 0.8453). These results surpass well-established methods like BASNet [40], SINet-v2 [9], BGNet [3], and PCNet [52], positioning AINet as the most effective solution for cotton bollworm detection, with significant improvements in both accuracy and boundary precision.

Finally, Table 6 on the Mango dataset used for the harvesting operation. This advanced architecture achieves superior results, securing first place in both S_α (0.8662) and F_β^w (0.8280) metrics, while maintaining second place in F_β^{adp} (0.9545) and E_ϕ^{adp} (0.9558). AINet's consistent top-tier performance across multiple evaluation criteria makes it particularly well-suited for automated fruit detection and harvesting systems, where reliable object identification is crucial for agricultural robotics. The results confirm that AINet's architecture provides the robust and consistent performance required for real-world agricultural scenarios.

B. QUALITATIVE EVALUATION

Figure 3 presents the results of the evaluation of our AINet and six previous SOTA techniques on classical COD benchmark datasets-i.e., CAMO-Test, COD10K-Test, and NC4K. To visually identify which technique presents the best performance, the GT is compared with the predicted mask for each image. Successful matches between GT and predicted masks are painted with white color; False positive regions

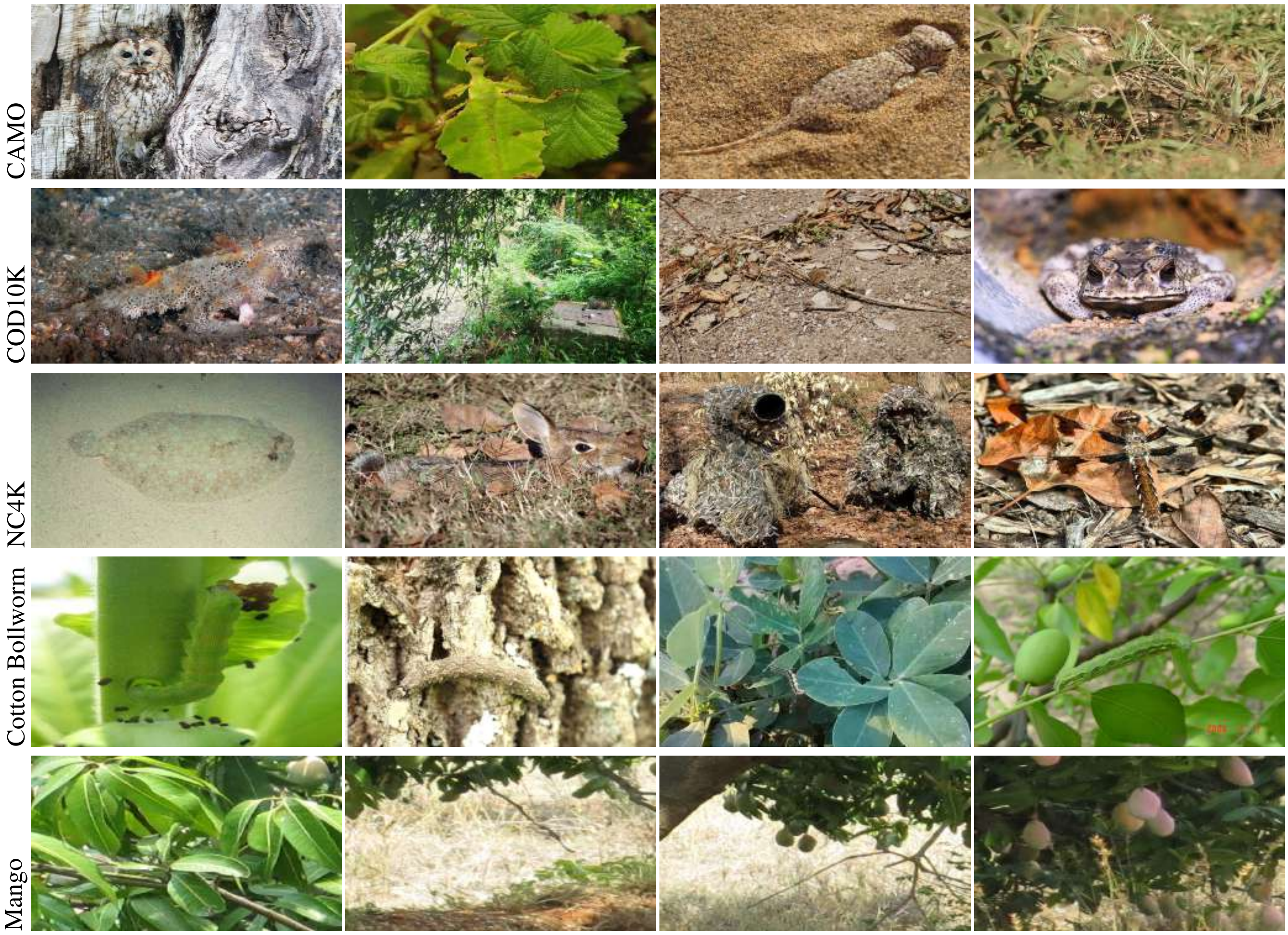

FIGURE 2. Example images of the benchmark datasets (CAMO [25], [49], COD10K [9], [10], NC4K [32]) and agricultural datasets (Cotton Bollworm [37] and Mango [24]) used as case studies.

TABLE 2. Details of the training parameters used in evaluated SOTA COD techniques. Learning rate (LR); Batch size (BS).

Technique	Optimizer	LR	BS	Epochs	Scheduler	Loss function
BASNet [40]	Adam	1e-3	8	1000	ReduceLROnPlateau	BCE + SSIM + IOU (multi-stage fusion)
SINet-v2 [9]	Adam	1e-4	16	150	Custom (Adjust LR)	Structure loss (weighted BCE + weighted IOU)
BGNet [3]	Adam	1e-4	12	100	Custom (Poly LR)	Structure loss (weighted BCE + weighted IOU) + Dice loss (edge)
C^2F-Net [2]	AdaXW	1e-4	32	50	Custom (Poly LR)	Structure loss (weighted BCE + weighted IOU)
OCENet [29]	Adam	1e-5	4	50	StepLR	Uncertainty aware structure loss (weighted BCE + weighted IOU)
DGNet [18]	AdamW	5e-5	16	150	CosineAnnealingLR	Hybrid loss (weighted BCE + weighted IOU) + MSE loss (grad)
PCNet [52]	AdamW	1e-4	8	150	Custom (Adjust LR)	Structure loss (weighted BCE + weighted IOU)
AINet (Ours)	AdamW	1e-4	16	150	CosineAnnealingLR	Structure loss (weighted BCE + weighted IOU)

TABLE 3. Comparison of architectural and efficiency characteristics of competing methods, including publication venue (source and type), input resolution, computational cost (FLOPs), inference time, and number of parameters.

Technique	Source	Source Type	Image Size (px)	FLOPs(G)	Inference time (ms)	# Params. (M)
BASNet [40]	CVPR	Conference	256×256	18.29	58.13	87.06
SINet-v2 [9]	TPAMI	Journal	352×352	4.96	41.20	24.93
BGNet [3]	IJCAI	Conference	416×416	16.93	51.28	77.80
C^2F-Net [2]	TCSVT	Conference	352×352	5.30	38.10	26.36
OCENet [29]	WACV	Conference	352×352	12.22	60.67	58.17
DGNet [18]	MIR	Journal	352×352	2.63	33.27	8.30
PCNet [52]	arXiv	-	352×352	5.65	65.94	27.66
AINet (Ours)	IEEE Access	Journal	352×352	7.78	55.92	36.05

TABLE 4. Experimental results for SOTA COD techniques and the proposed AINet architecture on benchmark datasets. The best three performing results are highlighted using color: First , Second , and Third respectively.

Technique	Pub/Year	Backbone	CAMO-Test (250)				COD10K-Test (2,026)				NC4K (4,121)			
			$S_\alpha \uparrow$	$M \downarrow$	$F_\beta^{mean} \uparrow$	$E_\phi^{mean} \uparrow$	$S_\alpha \uparrow$	$M \downarrow$	$F_\beta^{mean} \uparrow$	$E_\phi^{mean} \uparrow$	$S_\alpha \uparrow$	$M \downarrow$	$F_\beta^{mean} \uparrow$	$E_\phi^{mean} \uparrow$
BASNet [40]	CVPR-19	ResNet34	0.615	0.124	0.503	0.671	0.661	0.071	0.486	0.729	0.696	0.095	0.610	0.762
C^{2}F-Net [42]	IJCAI-21	Res2Net50	0.796	0.080	0.762	0.854	0.813	0.036	0.723	0.890	0.838	0.049	0.795	0.897
LSR [31]	CVPR-21	ResNet50	0.708	0.105	0.645	0.755	0.760	0.045	0.658	0.831	0.797	0.061	0.758	0.854
MGL [53]	CVPR-21	ResNet50	0.775	0.088	0.726	0.812	0.814	0.035	0.711	0.852	0.833	0.052	0.782	0.867
PFNet [36]	CVPR-21	ResNet50	0.782	0.085	0.746	0.841	0.800	0.040	0.701	0.877	0.829	0.053	0.784	0.887
TINet [59]	AAAI-21	ResNet50	0.781	0.087	0.728	0.836	0.793	0.042	0.679	0.861	0.829	0.055	0.773	0.879
UGTR [51]	ICCV-21	ResNet50	0.785	0.086	0.738	0.823	0.818	0.035	0.712	0.853	0.839	0.052	0.787	0.874
UR-SINet [23]	ACMMM-21	ResNet50	0.741	0.091	0.649	0.804	0.775	0.041	0.643	0.869	0.806	0.057	0.731	0.873
BGNet [3]	IJCAI22	Res2Net50	0.812	0.073	0.789	0.870	0.831	0.033	0.753	0.901	0.851	0.044	0.820	0.907
C^{2}F-Net-V2 [2]	TCSVT-22	Res2Net50	0.799	0.077	0.770	0.859	0.811	0.036	0.725	0.887	0.840	0.048	0.802	0.896
ERRNet [19]	PR-22	ResNet50	0.779	0.085	0.729	0.842	0.786	0.043	0.675	0.867	0.827	0.054	0.778	0.887
FAP-Net [58]	TIP-22	Res2Net50	0.815	0.076	0.776	0.865	0.822	0.036	0.731	0.888	0.851	0.047	0.810	0.899
FindNet [28]	TIP-22	Res2Net	0.803	0.077	0.763	0.862	0.811	0.036	0.706	0.883	0.841	0.048	0.802	0.895
OCENet [29]	WACV-22	ResNet50	0.802	0.080	0.766	0.852	0.827	0.033	0.741	0.894	0.853	0.045	0.818	0.902
PreyNet [54]	ACMMM-22	ResNet50	0.790	0.077	0.757	0.842	0.813	0.034	0.736	0.881	0.834	0.050	0.803	0.887
SegMaR [20]	CVPR-22	ResNet50	0.815	0.071	0.795	0.874	0.833	0.034	0.757	0.899	0.841	0.046	0.821	0.896
SINet-v2 [9]	TPAMI-22	Res2Net50	0.820	0.070	0.782	0.882	0.815	0.037	0.718	0.887	0.847	0.048	0.805	0.903
DGNet [18]	MIR-23	EfficientNet	0.839	0.057	0.806	0.901	0.822	0.033	0.728	0.896	0.857	0.042	0.814	0.911
FEDER [12]	CVPR-23	ResNet50	0.807	0.069	0.785	0.873	0.823	0.032	0.740	0.900	0.846	0.045	0.817	0.905
LSR-V2 [30]	TCSVT-23	ResNet50	0.789	0.079	0.751	0.840	0.805	0.037	0.711	0.880	0.840	0.048	0.801	0.896
MRR-Net [50]	TNNLS-23	ResNet50	0.811	0.076	0.772	0.869	0.822	0.036	0.730	0.889	0.848	0.049	0.801	0.898
PUENet [56]	TIP-23	ResNet50	0.794	0.080	0.762	0.857	0.813	0.035	0.727	0.887	0.836	0.050	0.798	0.892
UCOS-DA [55]	ICCVW-23	DINO	0.701	0.127	0.646	0.784	0.689	0.086	0.546	0.740	0.755	0.085	0.689	0.819
UJSCOD-V2 [27]	arXiv-23	ResNet50	0.803	0.071	0.768	0.858	0.817	0.033	0.733	0.895	0.856	0.040	0.824	0.913
WS-SAM [13]	NeurIPS-23	ResNet50	0.759	0.092	0.742	0.818	0.803	0.038	0.719	0.878	0.829	0.052	0.802	0.886
GenSAM [15]	AAAI-24	CLIP, BLIP2	0.719	0.113	0.659	0.775	0.775	0.067	0.681	0.838	—	—	—	—
AINet (Ours)	-	PVTv2	0.840	0.057	0.815	0.909	0.835	0.028	0.757	0.913	0.868	0.036	0.838	0.926

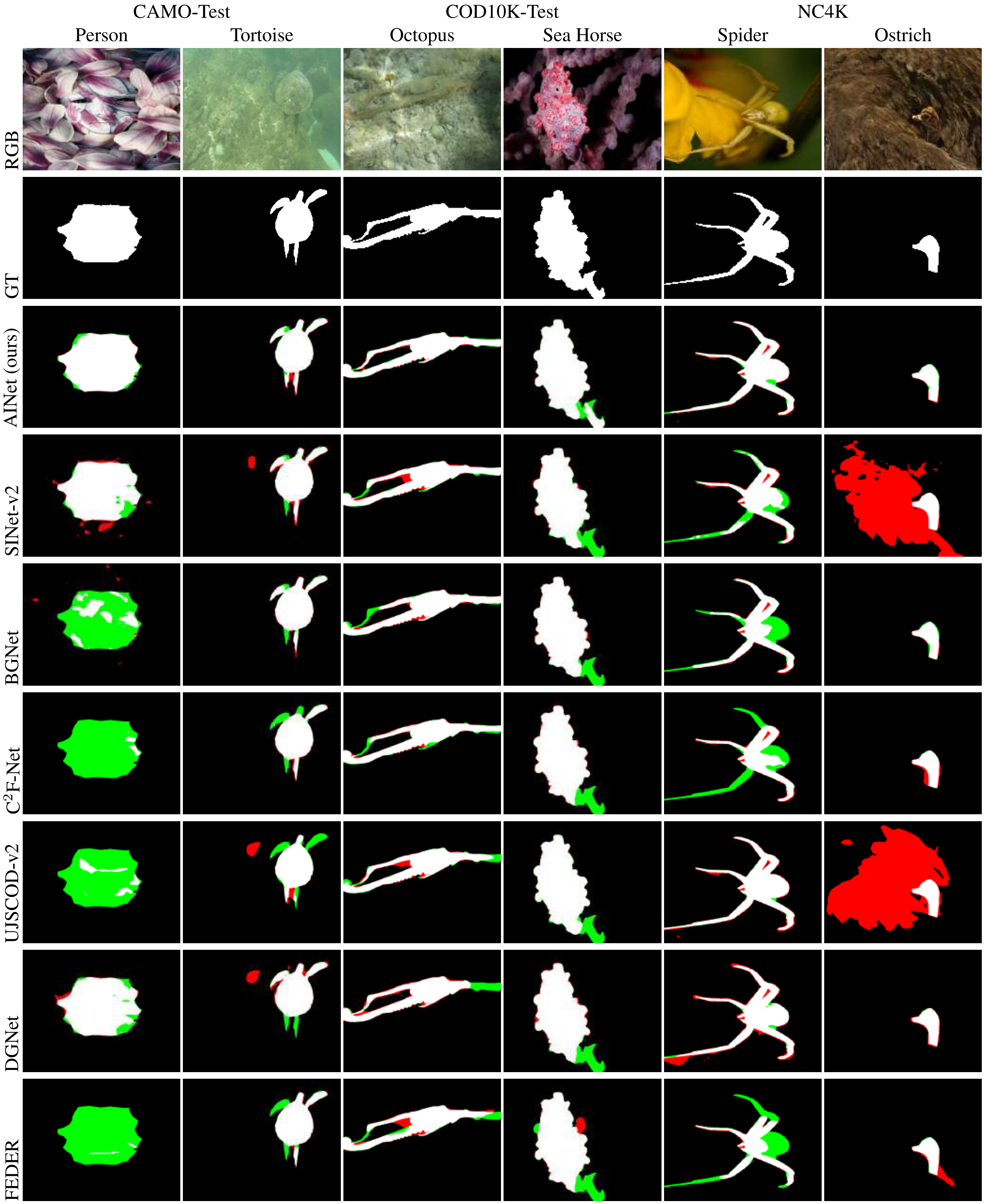

FIGURE 3. Quantitative results for six SOTA COD techniques and the proposed AINet architecture, evaluated on example images from benchmark datasets. Successful matches between GT and predicted masks (white areas); False positive regions (red areas, over-segmentation); and false negative regions (green areas, miss-segmentation).

with red color (over-segmentation); and false negative regions with green color (miss-segmentation). As can be seen in the example images, AINet presents exceptional results in terms of the delimitation of the camouflaged object area, as well as not presenting excessive over-segmentation or miss-segmentation, clearly surpassing recent SOTA COD techniques.

Figure 4 shows the comparative results on the Cotton Bollworm dataset [37]. The third column shows the results of AINet, which demonstrates significantly more accurate detection compared to other methods, correctly identifying the shape and location of cotton bollworms in various environments. AINet is observed to produce sharper and more complete masks, with better edge delineation and fewer false positives. Particularly in difficult cases where the insect is camouflaged with the plant background, AINet maintains high segmentation accuracy, highlighting its effectiveness for this specific application of agricultural pest detection.

On the other hand, the qualitative results on the Mango dataset strongly demonstrate the superiority of AINet as a leading architecture for automated fruit harvesting applications. AINet exhibits exceptional performance in accurately detecting multiple mangoes, achieving defined contours and minimal false positives compared to established methods such as PCNet, DGNet, and other SOTA approaches as shown in Fig. 5.

Finally, AINet's robustness is evident in its ability to maintain consistent and accurate detections under diverse lighting conditions, from bright natural light to shadowed areas, while producing significantly cleaner detection maps with fewer artifacts and noise than its competitors. The edges detected by AINet are noticeably smoother and more precise. While other methods show considerable variability in their results, AINet maintains superior stability across different scenarios, establishing itself as the most reliable and effective solution for real-world applications.

V. ABLATION STUDY

To systematically assess the contribution of each component within AINet, an extensive ablation analysis is performed on three standard COD benchmarks (CAMO, COD10K, and NC4K) as well as two domain-specific agricultural datasets (Cotton Bollworm and Mango). On the benchmark datasets, the study examines: (i) the effect of integrating Mamba and CBAM (module combinations), (ii) the impact of multi-level deep supervision through different output-fusion strategies, and (iii) model interpretability via Grad-CAM visualizations computed from the segmentation heads.

CAMO ablations. Table 7 reports the performance obtained with different module combinations on CAMO. The complete design (Mamba + CBAM) provides the best overall results, improving segmentation accuracy and reducing the pixel-wise error compared with single-module variants. This indicates that long-range dependency modeling (Mamba) and attention-based refinement (CBAM) play complementary roles when dealing with low-contrast camouflage. The qualitative examples in Fig. 6 support this observation, where Mamba + CBAM yields cleaner masks and sharper boundaries than using only Mamba or only CBAM.

Table 8 analyzes different output-fusion settings derived from the multi-level segmentation heads. Although the simple averaging baseline (Avg) attains a relatively high S_α, it is less consistent across complementary measures. In contrast, fusing multiple supervised outputs (notably M1/M2) leads to a more balanced behavior, improving the F-measure while reducing MAE (e.g., M2 achieves $F_\beta^w = 0.7898$ and $M = 0.0566$). Figure 7 visually reinforces these findings: incorporating multiple outputs produces more complete object regions and more coherent contours, which is particularly beneficial for small targets and ambiguous boundaries.

Beyond accuracy, we examine where the network focuses when segmenting camouflaged objects. Figure 8 shows Grad-CAM responses for the segmentation heads on CAMO, where the activation maps consistently concentrate on the true target regions rather than on background clutter. In addition, different heads exhibit complementary attention patterns, with some emphasizing coarse localization and others highlighting finer structures and boundaries. This behavior is consistent with the intended multi-level design and supports the effectiveness of the proposed feature refinement strategy.

COD10K ablations. Table 9 reports the same module-combination study on COD10K. The Mamba + CBAM configuration remains the strongest, outperforming both only-Mamba and only-CBAM across the evaluated metrics. Qualitative comparisons in Fig. 9 confirm that the combined design better preserves object completeness and reduces missed regions on highly camouflaged samples, demonstrating that the observed synergy extends to large-scale benchmark data.

Table 10 evaluates the role of deep supervision through different output-fusion strategies. While Avg achieves a high S_α, richer fusion with multi-level supervision (particularly M1) yields improved segmentation quality and lower error (M1: $F_\beta^w = 0.7371$, $M = 0.0283$ vs. Avg: $F_\beta^w = 0.7145$, $M = 0.0316$). Figure 10 illustrates that multi-level fusion provides more reliable masks, especially when the target shares texture and color with the background.

Figure 11 provides Grad-CAM visualizations for the segmentation heads on COD10K. The heatmaps remain well aligned with the true object locations even under strong camouflage. Moreover, the heads again show complementary attention: earlier heads tend to capture broader localization cues, whereas deeper heads emphasize boundary-sensitive details, which is consistent with the role of multi-level supervision in improving robustness.

NC4K ablations. Table 11 reports the module ablation on NC4K. As in the previous benchmarks, integrating Mamba + CBAM achieves the best overall performance compared to only-Mamba and only-CBAM. The qualitative examples in Fig. 12 further show that the full configuration improves boundary delineation while reducing both missing parts and spurious detections in complex natural scenes.

Table 12 analyzes different output-fusion variants on

TABLE 5. Quantitative results for SOTA COD techniques and AINet on testing set of the Cotton Bollworm dataset [37]. The best three performing results are highlighted using color: First, Second, and Third respectively.

Technique	$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$M \downarrow$	$F_\beta^{adp} \uparrow$	$F_\beta^{mean} \uparrow$	$E_\phi^{adp} \uparrow$	$E_\phi^{mean} \uparrow$
BASNet [40]	0.8335	0.7500	0.0242	0.7538	0.7702	0.9270	0.9049
SINet-v2 [9]	0.9010	0.8677	0.0136	0.8483	0.8685	0.9668	0.9662
BGNet [3]	0.8953	0.8769	0.0133	0.8757	0.8864	0.9708	0.9635
C^{2}F-Net [2]	0.8861	0.7758	0.0178	0.8369	0.8571	0.9483	0.9341
OCENet [29]	0.9071	0.8692	0.0108	0.8431	0.8674	0.9586	0.9622
DGNet [18]	0.9001	0.8725	0.0104	0.8525	0.8747	0.9615	0.9590
PCNet [52]	0.9000	0.8652	0.0125	0.8532	0.8657	0.9651	0.9647
AINet (Ours)	0.9208	0.9197	0.0076	0.9092	0.9156	0.9838	0.9820

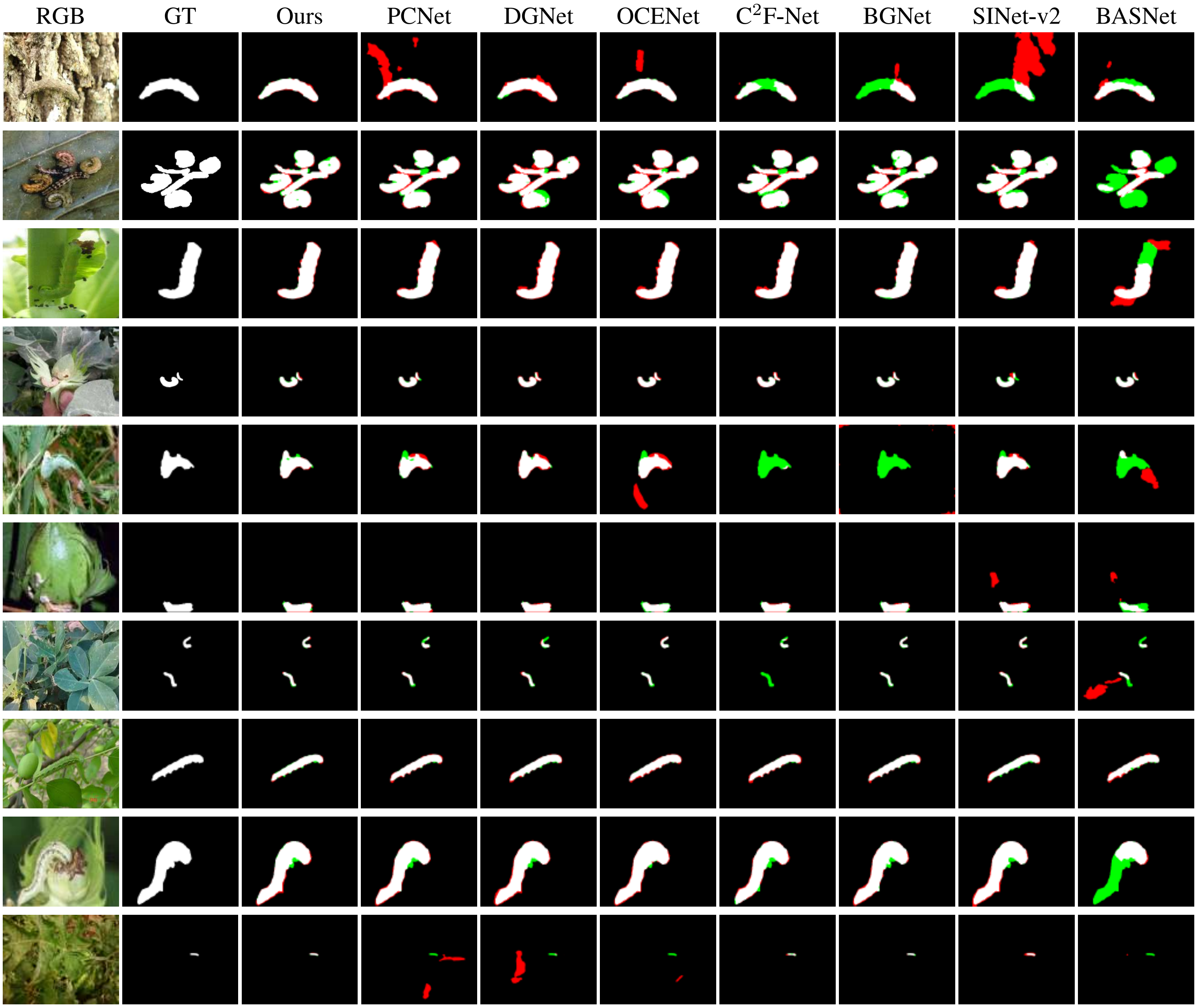

FIGURE 4. Qualitative results of seven SOTA COD techniques and AINet, evaluated on some images from the Cotton Bollworm dataset [37]. Successful matches between GT and predicted masks (white areas); False positive regions (red areas, over-segmentation); and false negative regions (green areas, miss-segmentation).

NC4K. Although Avg obtains a higher S_α, multi-level fusion (M1) offers a better trade-off across metrics by improving F-measure and reducing MAE (M1: $F_\beta^w = 0.8204$, $M = 0.0361$ vs. Avg: $F_\beta^w = 0.8007$, $M = 0.0404$). As shown in Fig. 13, multi-level fusion tends to produce more complete and stable segmentations, particularly for thin structures and low-contrast regions where reduced supervision may lead to fragmented masks.

TABLE 6. Quantitative results for SOTA COD techniques and AINet on testing set of the Mango dataset [24]. The best three performing results are highlighted using color: First, Second, and Third respectively.

Technique	$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$M \downarrow$	$F_\beta^{adp} \uparrow$	$F_\beta^{mean} \uparrow$	$E_\phi^{adp} \uparrow$	$E_\phi^{mean} \uparrow$
BASNet [40]	0.8508	0.8011	0.0119	0.7954	0.8224	0.9422	0.9336
SINet-v2 [9]	0.8543	0.8194	0.0106	0.7854	0.8290	0.9400	0.9562
BGNet [3]	0.8578	0.8278	0.0117	0.8200	0.8366	0.9513	0.9535
C^{2}F-Net [2]	0.8659	0.8234	0.0112	0.8167	0.8348	0.9545	0.9532
OCENet [29]	0.8505	0.8094	0.0111	0.8078	0.8301	0.9532	0.9452
DGNet [18]	0.8595	0.8000	0.0116	0.7621	0.8106	0.9284	0.9499
PCNet [52]	0.8582	0.8205	0.0103	0.7996	0.8310	0.9487	0.9558
AINet (Ours)	0.8662	0.8280	0.0100	0.8222	0.8285	0.9545	0.9558

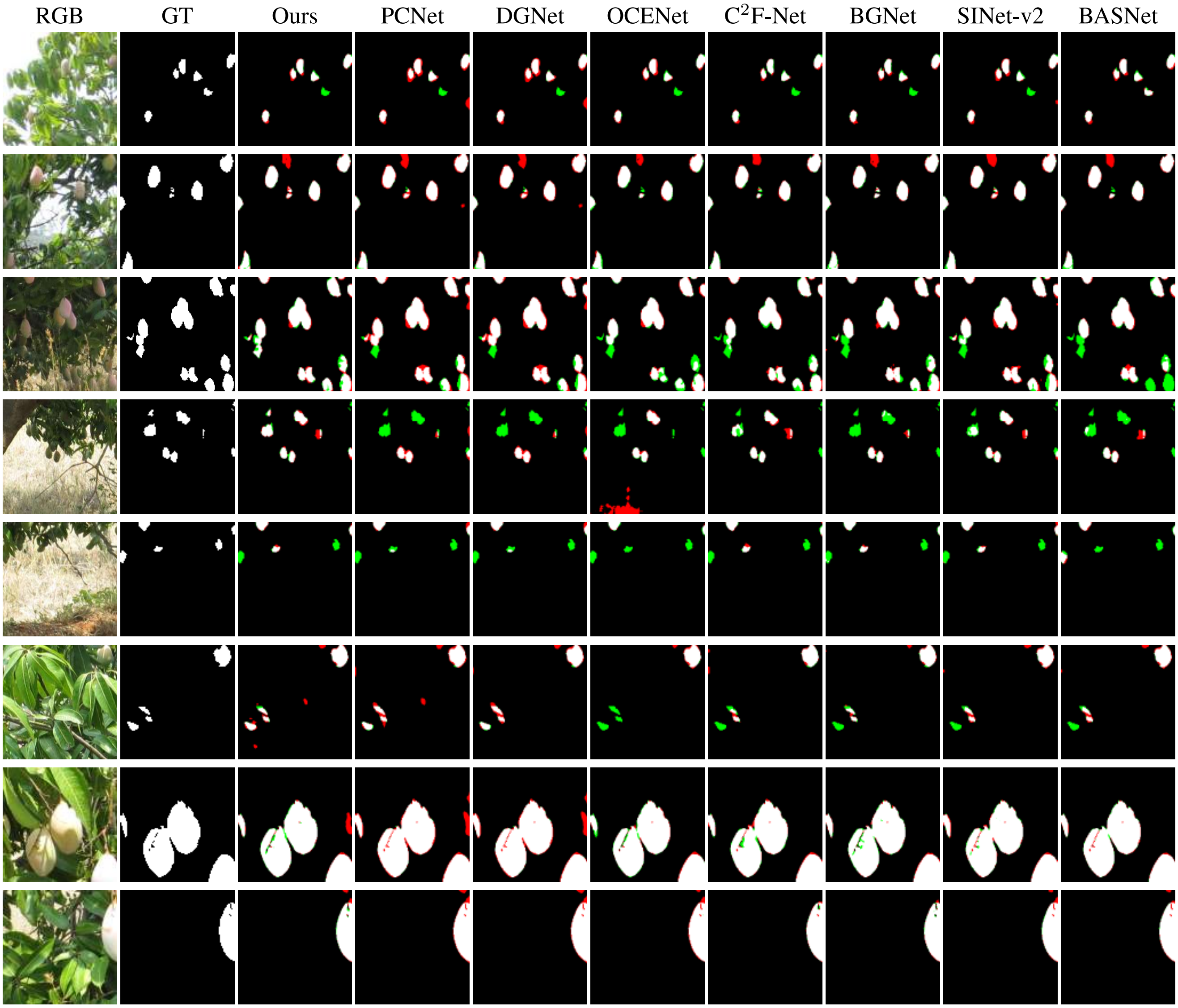

FIGURE 5. Qualitative results of seven SOTA COD techniques and AINet, evaluated on some images from the Mango dataset [24]. Successful matches between GT and predicted masks (white areas); False positive regions (red areas, over-segmentation); and false negative regions (green areas, miss-segmentation).

Figure 14 visualizes Grad-CAM responses for the segmentation heads on NC4K. The activations consistently align with the camouflaged targets across diverse scenes, indicating that predictions are driven by meaningful object cues rather than background artifacts. The complementary evidence across heads further supports the multi-level decoding strategy as a mechanism to aggregate distinct spatial cues for improved final segmentation.

Cotton Bollworm ablations. Table 13 reports the effect of different module combinations on the Cotton Bollworm dataset. The full configuration (Mamba + CBAM) achieves the strongest performance, whereas single-module variants are consistently weaker (e.g., only Mamba: $S_\alpha = 0.9131$; only CBAM: $S_\alpha = 0.9118$). This gap indicates that Mamba and CBAM contribute complementary benefits in agricultural scenes: Mamba strengthens long-range contextual modeling, while CBAM refines discriminative features to better separate camouflaged pests from visually similar plant textures. The qualitative examples in Fig. 15 reinforce this outcome, where Mamba + CBAM produces sharper boundaries and fewer false positives under challenging camouflage conditions.

Table 14 analyzes the role of multi-level deep supervision through different output-fusion settings. Although Avg attains a slightly higher S_α (0.9261), it is less consistent across the remaining metrics. In contrast, multi-output fusion (e.g., M_1: $P_0 + P_1 + P_2 + P_3 + Avg$) provides a more stable trade-off, achieving the best overall balance ($S_\alpha = 0.9208$, $F_\beta^w = 0.9197$). As illustrated in Fig. 16, multi-level fusion tends to yield more complete and coherent pest regions, whereas reduced-supervision variants (e.g., M_3, M_4) more often lead to fragmented or partially missing segmentations.

Figure 17 presents Grad-CAM visualizations for the segmentation heads on the Cotton Bollworm dataset. The activation maps consistently align with the pest regions, even when insects are partially occluded or strongly blended with the plant background, suggesting that predictions are driven by meaningful object cues rather than background artifacts. In addition, the heads exhibit complementary attention patterns, where some heads emphasize coarse localization and others highlight fine structures and boundary-sensitive areas, supporting the intended multi-level decoding behavior.

Mango ablations. Table 15 reports the module-combination study on the Mango dataset. The combined design (Mamba + CBAM) again achieves the strongest performance (e.g., $S_\alpha = 0.8662$, $F_\beta^w = 0.8280$), outperforming single-module alternatives. The qualitative results in Fig. 18 further show that integrating both modules improves mask completeness and preserves fruit contours under variable illumination and cluttered foliage.

Table 16 evaluates multi-level deep supervision through different output-fusion strategies on Mango. Multi-output fusion provides a clear advantage over simple averaging, improving robustness and yielding stronger overall segmentation quality (e.g., best $S_\alpha = 0.8662$ vs. Avg with $S_\alpha = 0.8478$). Figure 19 corroborates this trend, where multi-level fusion reduces both over-segmentation and under-segmentation in difficult cases with strong background similarity and partial occlusions.

Figure 20 provides Grad-CAM visualizations for the segmentation heads on the Mango dataset. The heatmaps remain well aligned with fruit regions across challenging scenarios, indicating that the model focuses on target-relevant cues. Similar to the other datasets, the heads show complementary attention behavior: earlier heads capture broader localization patterns, while deeper heads emphasize finer details and boundary regions, which is consistent with the purpose of multi-level supervision and staged decoding.

In summary, the ablation study demonstrates that (i) integrating Mamba and CBAM and (ii) employing multi-level deep supervision are both key to achieving strong and stable performance. Their combined effect yields more accurate, reliable, and generalizable camouflaged object segmentation on standard benchmarks and in challenging agricultural imagery.

VI. DISCUSSION

The experimental results demonstrate that AINet achieves state-of-the-art performance on both standard COD benchmarks and specialized agricultural datasets. The model's consistent superiority across diverse datasets underscores the effectiveness of integrating sequential state modeling (Mamba) with attention-based feature refinement (CBAM). This synergy enables AINet to capture long-range dependencies and focus on subtle visual cues, which are essential for detecting camouflaged objects in complex backgrounds.

The strong results on agricultural datasets highlight AINet's adaptability to real-world applications beyond generic COD tasks. Accurate detection of camouflaged pests and fruits is vital for precision agriculture, where reliable object identification supports automated monitoring and harvesting systems. The case study results suggest that AINet can be effectively deployed in such domains, offering both high accuracy and robustness under challenging conditions.

Despite these advances, some challenges remain. The reliance on large annotated datasets for training may limit the applicability of AINet in domains with scarce labeled data. Furthermore, while the model demonstrates strong generalization, future work should explore its performance in even more diverse and dynamic environments, as well as investigate strategies for reducing data requirements and improving real-time inference.

Despite the strong overall performance, AINet can still fail in extremely challenging camouflage scenarios. Figure 21 shows representative failure cases in which the target object exhibits near-identical texture/color to the background or appears very small and fragmented, making boundary cues ambiguous. In such cases, the model may produce false negatives (missed regions) due to insufficient discriminative contrast, or false positives when background structures mimic object-like patterns. These observations indicate that COD remains fundamentally difficult when both intrinsic similarity and edge disruption are simultaneously extreme. Addressing these cases may require incorporating additional cues (e.g., multi-modal data, stronger boundary priors, or specialized augmentation) and further improving robustness under scarce-label regimes.

In summary, AINet sets a new benchmark for camouflaged object detection, offering a versatile and high-performing solution for both academic research and practical deployment in fields such as agriculture, surveillance, and environmental monitoring.

TABLE 7. Metric evaluation with different module combinations (e.g., Mamba and CBAM) on testing set of the CAMO dataset [25], [49]. The best results by each metric are highlighted in bold.

Module	$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$M \downarrow$	$F_\beta^{adp} \uparrow$	$F_\beta^{mean} \uparrow$	$E_\phi^{adp} \uparrow$	$E_\phi^{mean} \uparrow$
Mamba + CBAM	**0.8380**	**0.7896**	**0.0567**	**0.8149**	**0.8144**	**0.9104**	**0.9080**
only Mamba	0.8366	0.7861	0.0595	0.8120	0.8116	0.9079	0.9057
only CBAM	0.8285	0.7782	0.0622	0.8112	0.8113	0.9008	0.8983

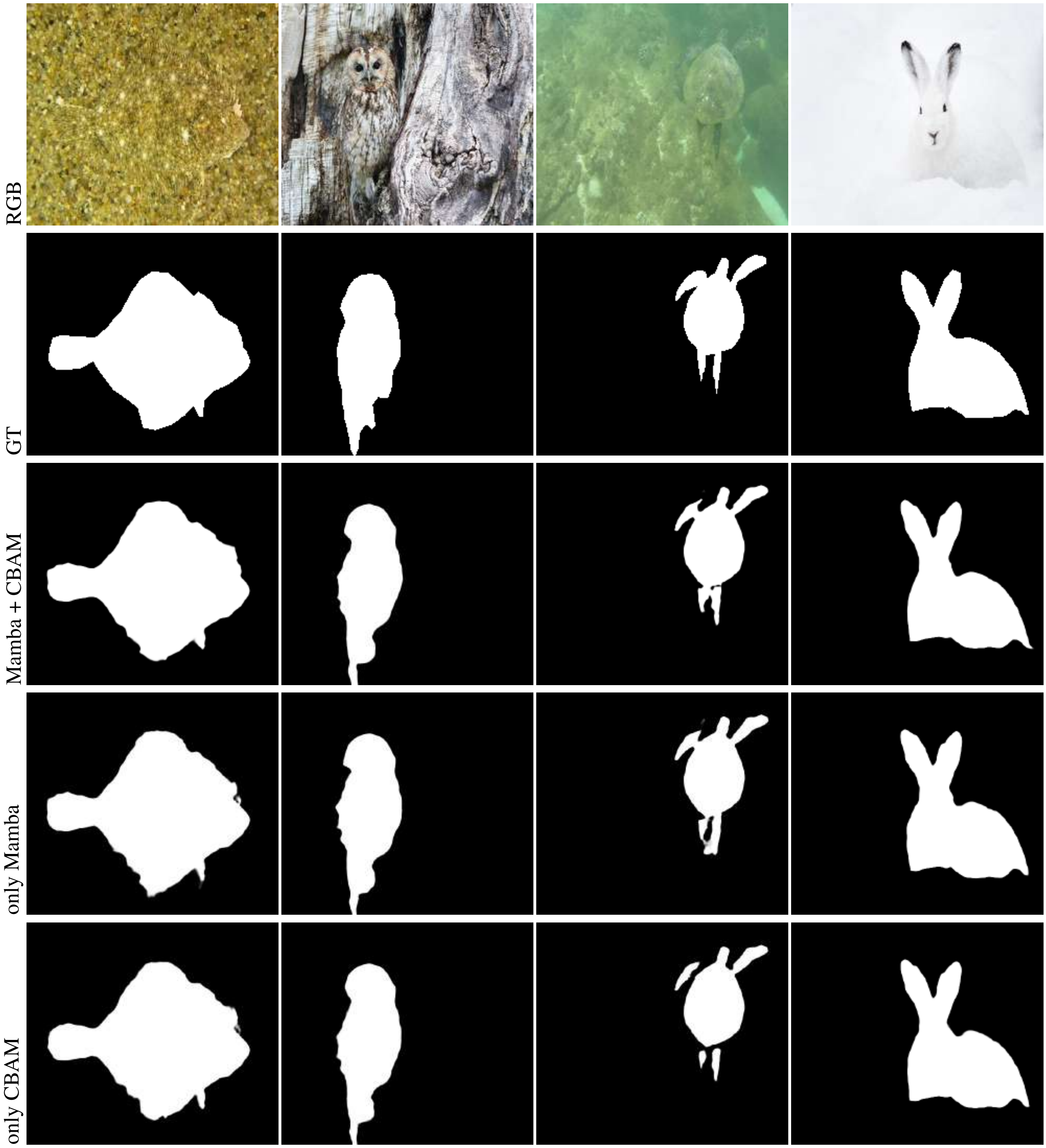

FIGURE 6. Qualitative results of different module combinations used, evaluated on some images from the CAMO dataset [25], [49].

TABLE 8. Metric evaluation with different outputs on testing set of the CAMO dataset [25], [49]. The best three performing results are highlighted using color: First, Second, and Third respectively.

Name	Output	$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$M \downarrow$	$F_\beta^{adp} \uparrow$	$F_\beta^{mean} \uparrow$	$E_\phi^{adp} \uparrow$	$E_\phi^{mean} \uparrow$
Avg	Avg	0.8489	0.7669	0.0632	0.7962	0.7990	0.8964	0.8899
M_1	$P_0 + P_1 + P_2 + P_3 + Avg$	0.8380	0.7896	0.0567	0.8149	0.8144	0.9104	0.9080
M_2	$P_1 + P_2 + P_3 + Avg$	0.8404	0.7898	0.0566	0.8147	0.8150	0.9101	0.9079
M_3	$P_2 + P_3 + Avg$	0.8419	0.7879	0.0570	0.8125	0.8139	0.9086	0.9069
M_4	$P_3 + Avg$	0.8445	0.7835	0.0581	0.8080	0.8111	0.9055	0.9038

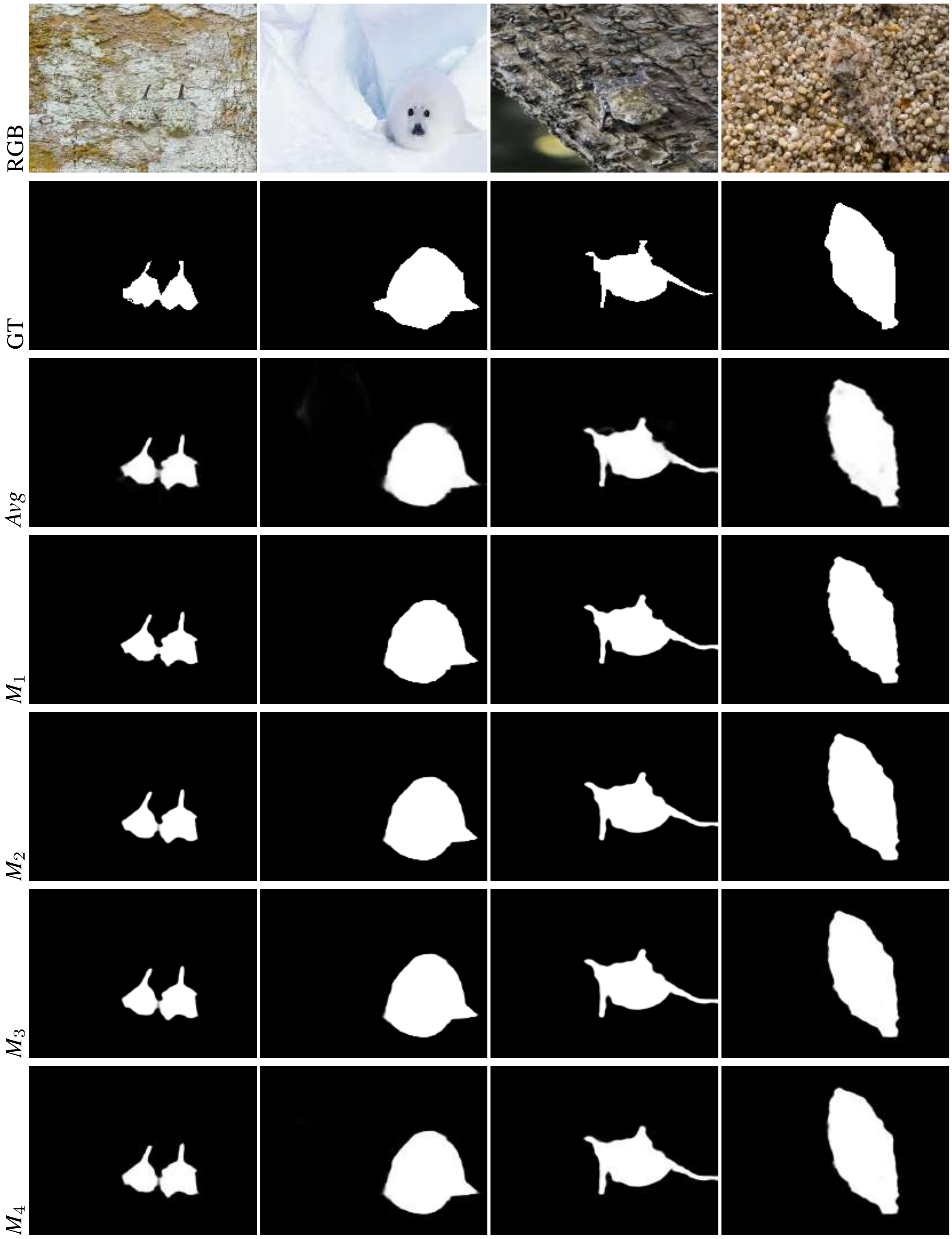

FIGURE 7. Qualitative results of different deep supervision outputs, evaluated on some images from the CAMO dataset [25], [49].

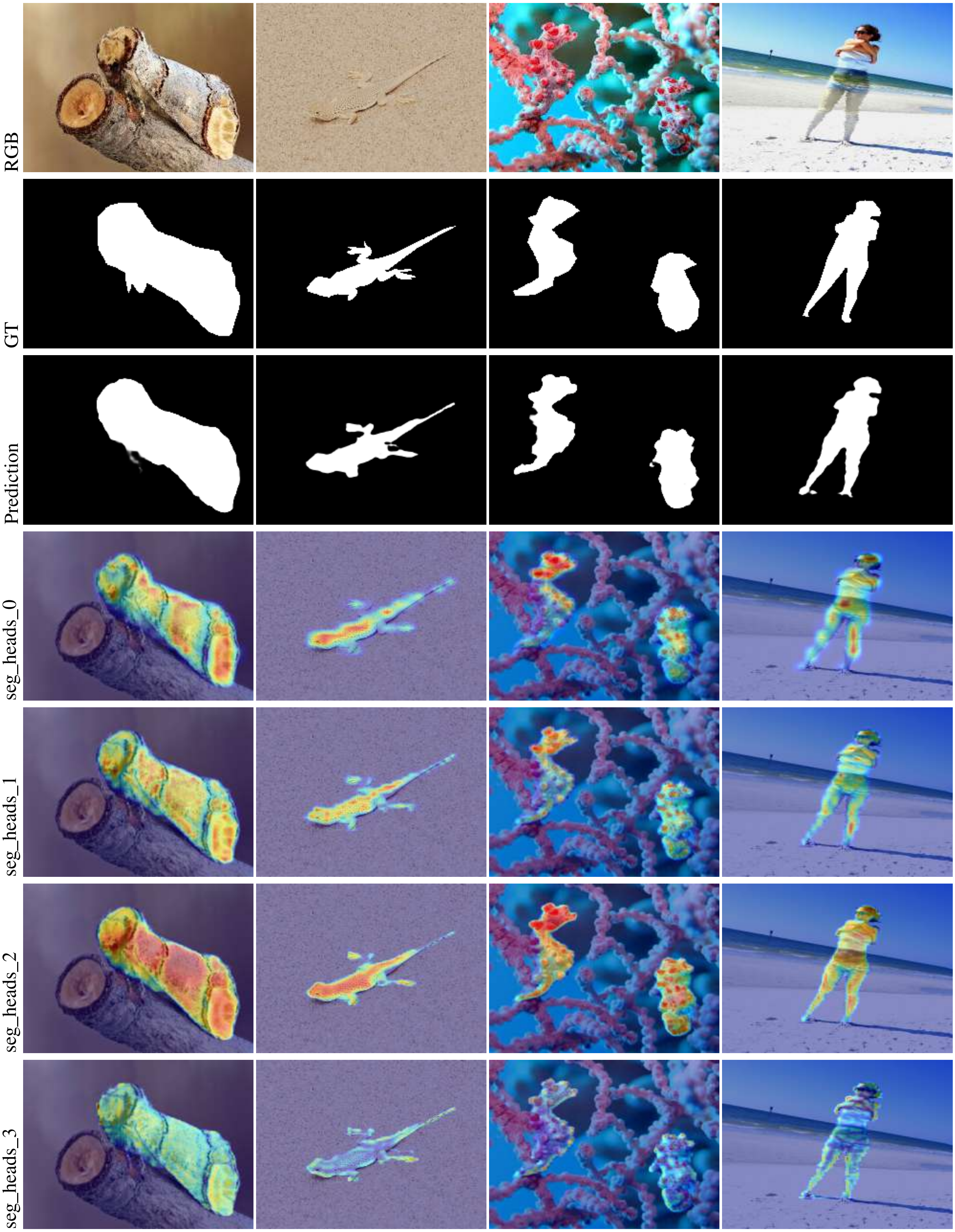

FIGURE 8. Grad-CAM visualization using segmentation heads outputs, evaluated on some images from the CAMO dataset [25], [49].

TABLE 9. Metric evaluation with different module combinations (e.g., Mamba and CBAM) on testing set of the COD10K dataset [9], [10]. The best results by each metric are highlighted in bold.

Module	$S_\alpha \uparrow$	$F^w_\beta \uparrow$	$M \downarrow$	$F^{adp}_\beta \uparrow$	$F^{mean}_\beta \uparrow$	$E^{adp}_\phi \uparrow$	$E^{mean}_\phi \uparrow$
Mamba + CBAM	**0.8352**	**0.7371**	**0.0283**	**0.7522**	**0.7568**	**0.9110**	**0.9131**
only Mamba	0.8267	0.7222	0.0316	0.7374	0.7420	0.8990	0.9010
only CBAM	0.8289	0.7242	0.0307	0.7430	0.7499	0.9030	0.9070

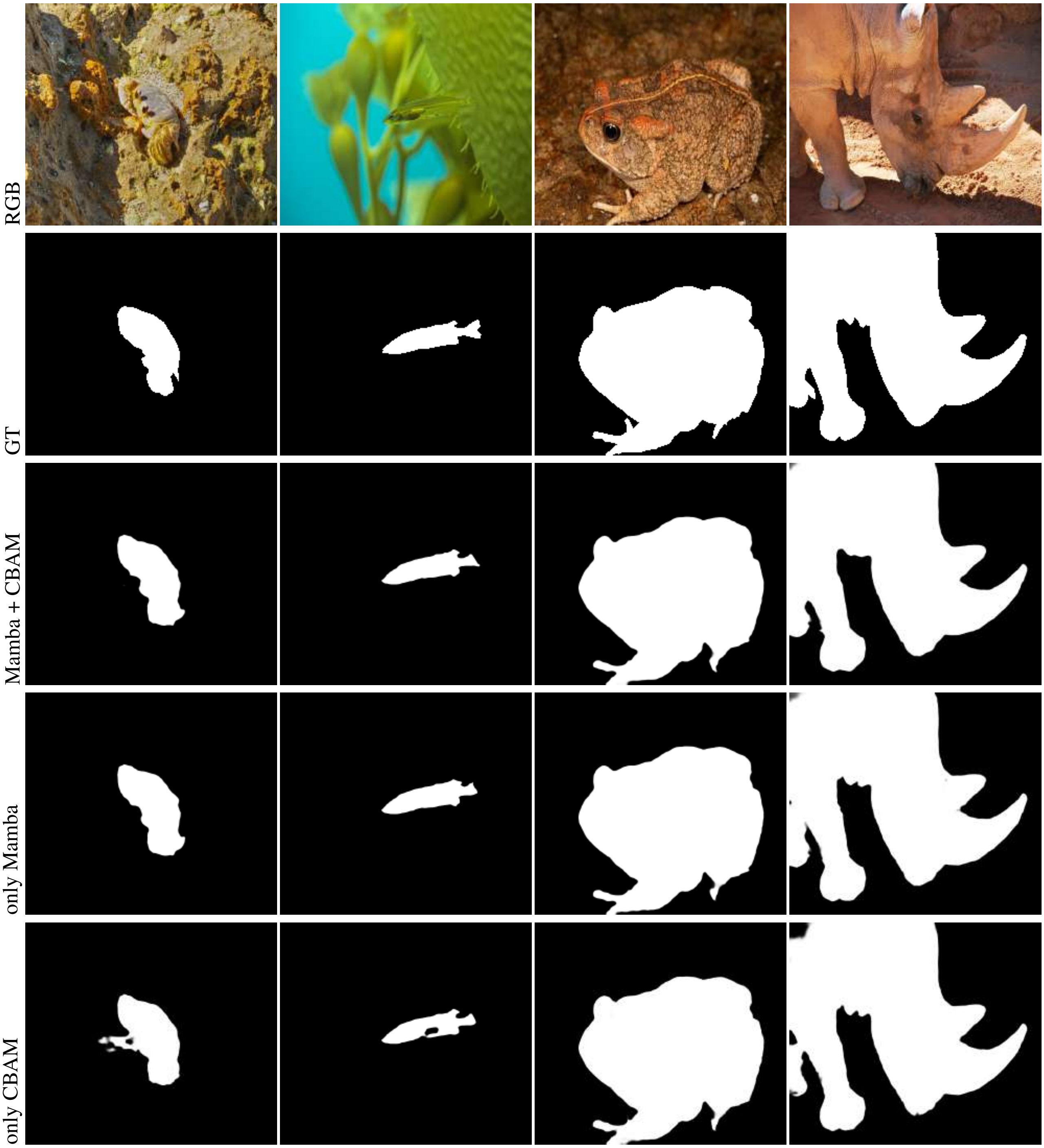

FIGURE 9. Qualitative results of different module combinations used, evaluated on some images from the COD10K dataset [9], [10].

TABLE 10. Metric evaluation with different outputs on testing set of the COD10k dataset [9], [10]. The best three performing results are highlighted using color: First , Second , and Third respectively.

Name	Output	$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$M \downarrow$	$F_\beta^{adp} \uparrow$	$F_\beta^{mean} \uparrow$	$E_\phi^{adp} \uparrow$	$E_\phi^{mean} \uparrow$
Avg	Avg	0.8430	0.7145	0.0316	0.7135	0.7453	0.8794	0.8992
M_1	$P_0 + P_1 + P_2 + P_3 + Avg$	0.8352	0.7371	0.0283	0.7522	0.7568	0.9110	0.9131
M_2	$P_1 + P_2 + P_3 + Avg$	0.8364	0.7356	0.0285	0.7465	0.7554	0.9078	0.9127
M_3	$P_2 + P_3 + Avg$	0.8368	0.7320	0.0289	0.7395	0.7527	0.9032	0.9111
M_4	$P_3 + Avg$	0.8389	0.7275	0.0295	0.7293	0.7505	0.8952	0.9084

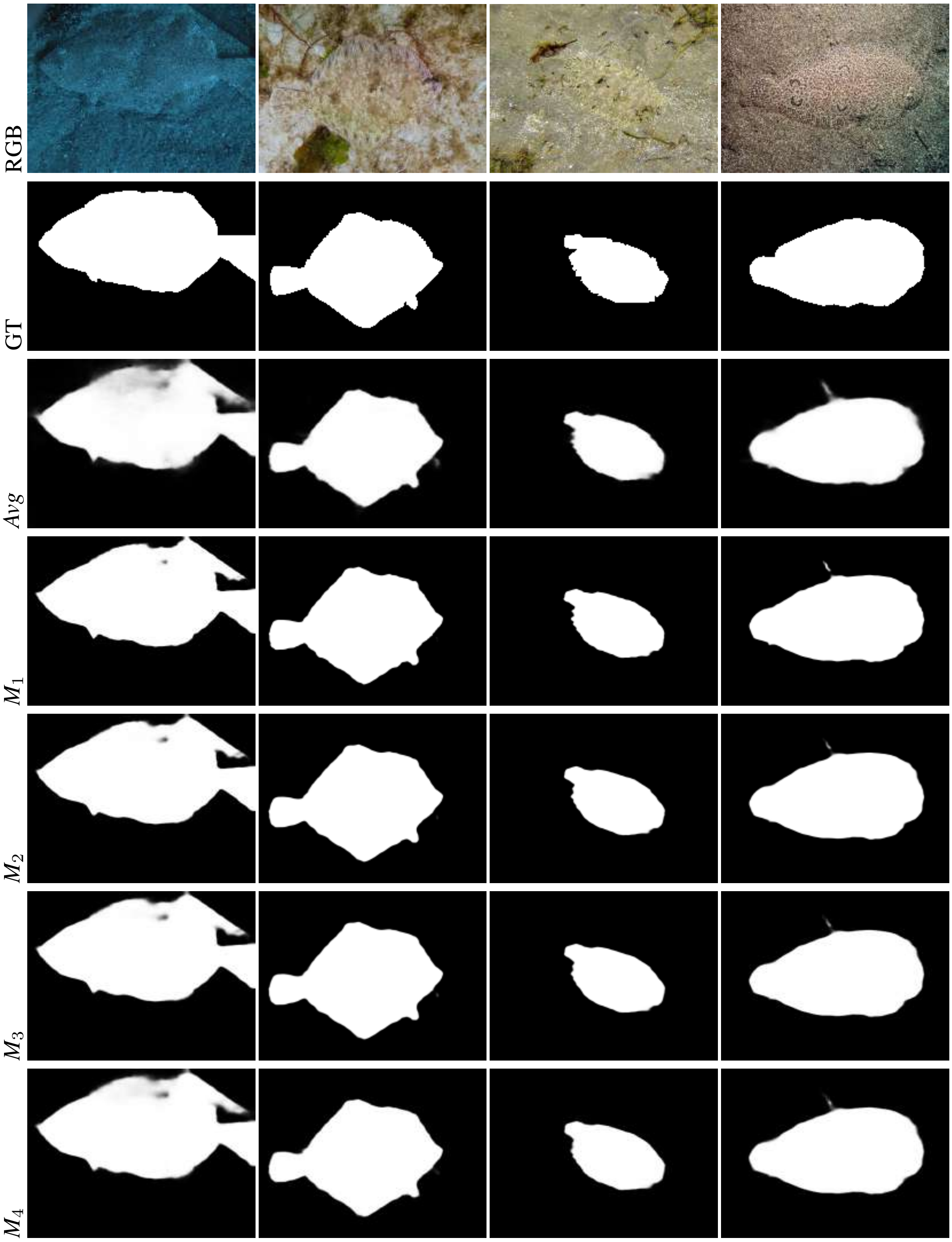

FIGURE 10. Qualitative results of different deep supervision outputs, evaluated on some images from the COD10K dataset [9], [10].

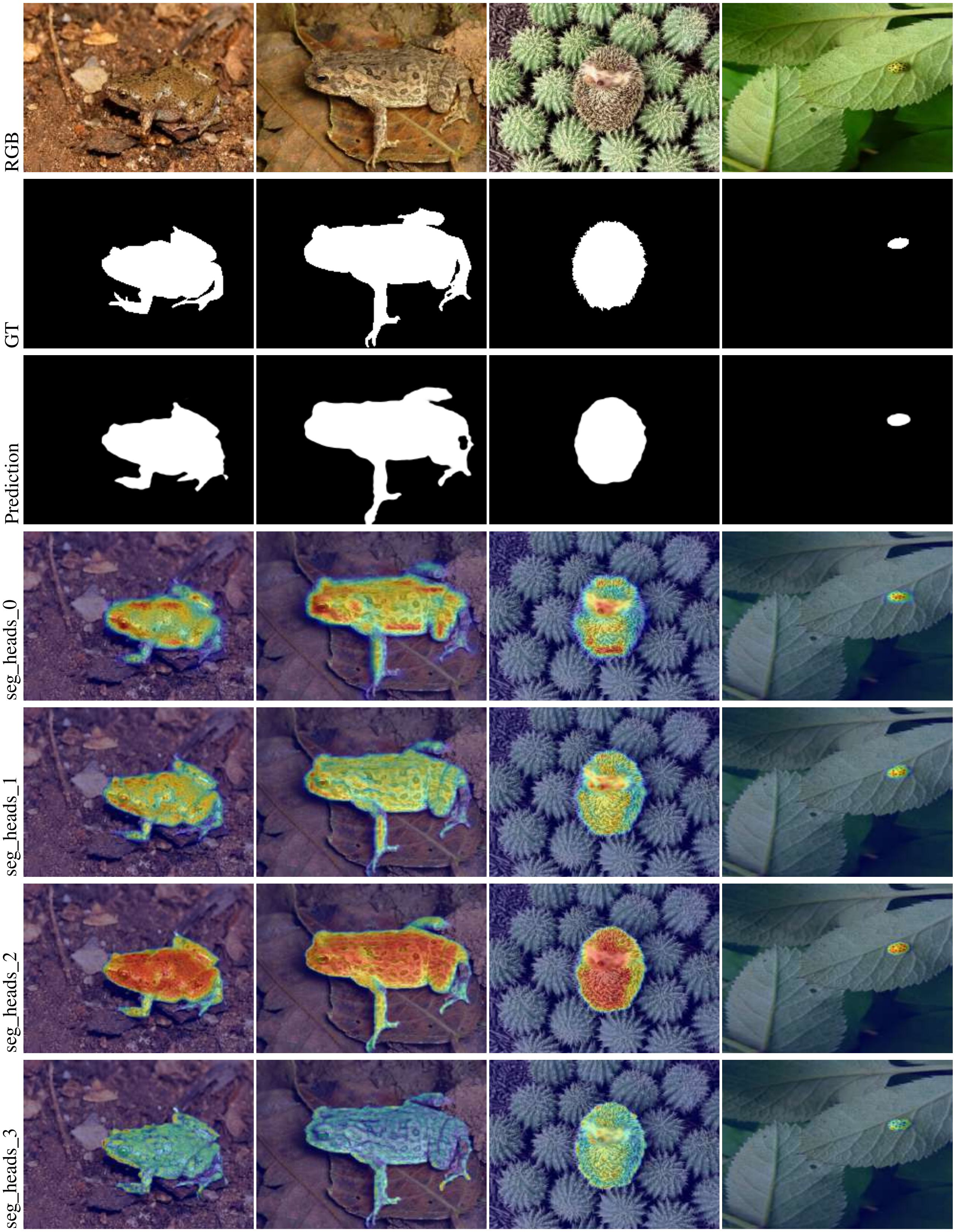

FIGURE 11. Grad-CAM visualization using segmentation heads outputs, evaluated on some images from the COD10K dataset [9], [10].

TABLE 11. Metric evaluation with different module combinations (e.g., Mamba and CBAM) on testing set of the NC4K dataset [32]. The best results by each metric are highlighted in bold.

Module	$S_\alpha \uparrow$	$F^w_\beta \uparrow$	$M \downarrow$	$F^{adp}_\beta \uparrow$	$F^{mean}_\beta \uparrow$	$E^{adp}_\phi \uparrow$	$E^{mean}_\phi \uparrow$
Mamba + CBAM	**0.8672**	**0.8204**	**0.0361**	**0.8387**	**0.8392**	**0.9287**	**0.9276**
only Mamba	0.8611	0.8093	0.0390	0.8270	0.8280	0.9207	0.9199
only CBAM	0.8605	0.8078	0.0394	0.8306	0.8323	0.9220	0.9210

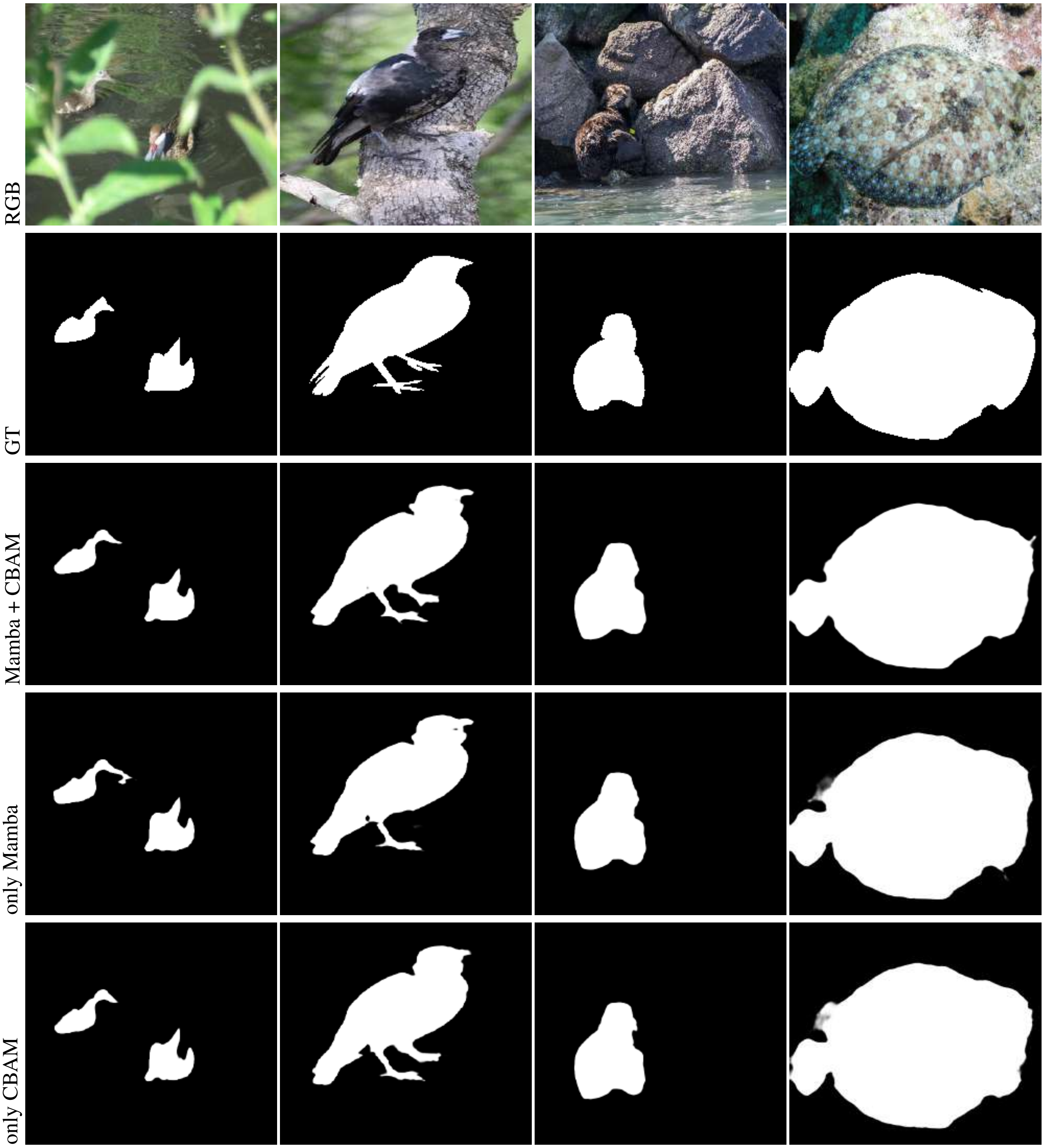

FIGURE 12. Qualitative results of different module combinations used, evaluated on some images from the NC4K dataset [32].

TABLE 12. Metric evaluation with different outputs on testing set of the NC4K dataset [32]. The best three performing results are highlighted using color: First , Second , and Third respectively.

Name	Output	$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$M \downarrow$	$F_\beta^{adp} \uparrow$	$F_\beta^{mean} \uparrow$	$E_\phi^{adp} \uparrow$	$E_\phi^{mean} \uparrow$
Avg	Avg	0.8756	0.8007	0.0404	0.8165	0.8273	0.9136	0.9151
M_1	$P_0 + P_1 + P_2 + P_3 + Avg$	0.8672	0.8204	0.0361	0.8387	0.8392	0.9289	0.9276
M_2	$P_1 + P_2 + P_3 + Avg$	0.8681	0.8188	0.0365	0.8358	0.8381	0.9275	0.9269
M_3	$P_2 + P_3 + Avg$	0.8689	0.8163	0.0369	0.8322	0.8363	0.9257	0.9259
M_4	$P_3 + Avg$	0.8711	0.8125	0.0377	0.8270	0.8341	0.9225	0.9238

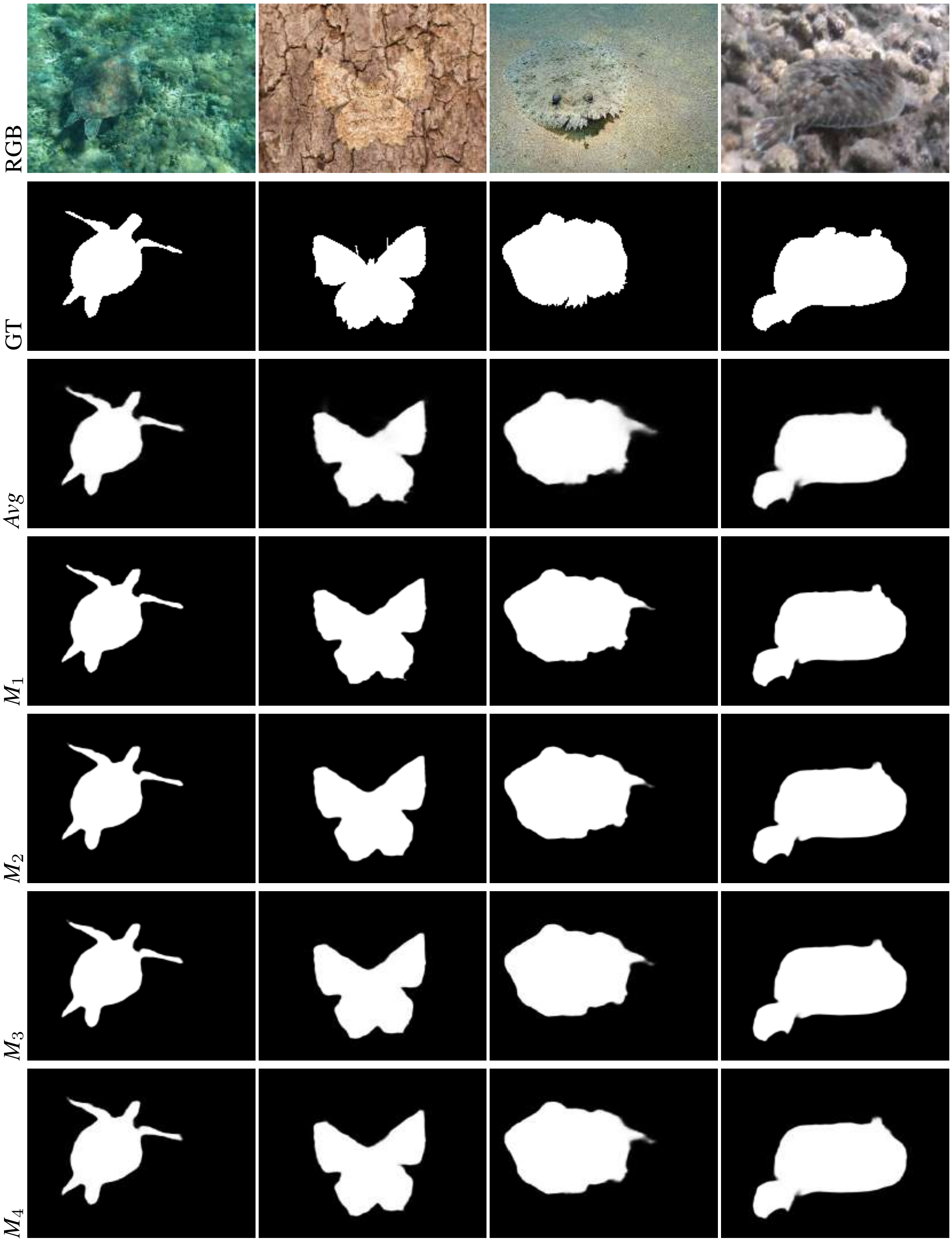

FIGURE 13. Qualitative results of different deep supervision outputs, evaluated on some images from the NC4K dataset [32].

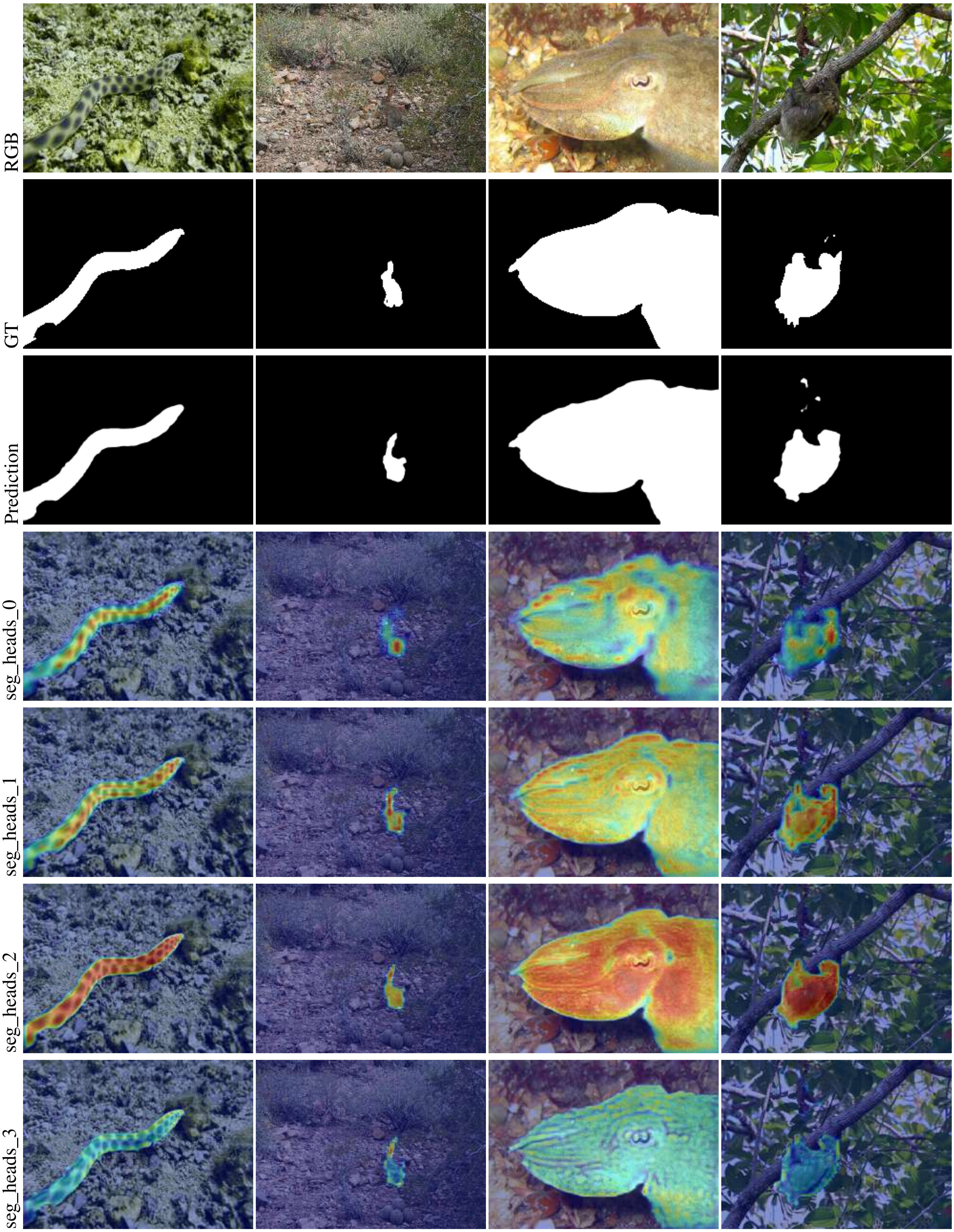

FIGURE 14. Grad-CAM visualization using segmentation heads outputs, evaluated on some images from the NC4K dataset [32].

TABLE 13. Metric evaluation with different module combinations (e.g., Mamba and CBAM) on testing set of the Cotton Bollworm dataset [37]. The best results by each metric are highlighted in bold.

Module	$S_\alpha \uparrow$	$F^w_\beta \uparrow$	$M \downarrow$	$F^{adp}_\beta \uparrow$	$F^{mean}_\beta \uparrow$	$E^{adp}_\phi \uparrow$	$E^{mean}_\phi \uparrow$
Mamba + CBAM	**0.9208**	**0.9197**	**0.0076**	**0.9092**	**0.9156**	**0.9838**	**0.9820**
only Mamba	0.9131	0.8953	0.0084	0.8877	0.8932	0.9788	0.9784
only CBAM	0.9118	0.8950	0.0087	0.8877	0.8923	0.9748	0.9744

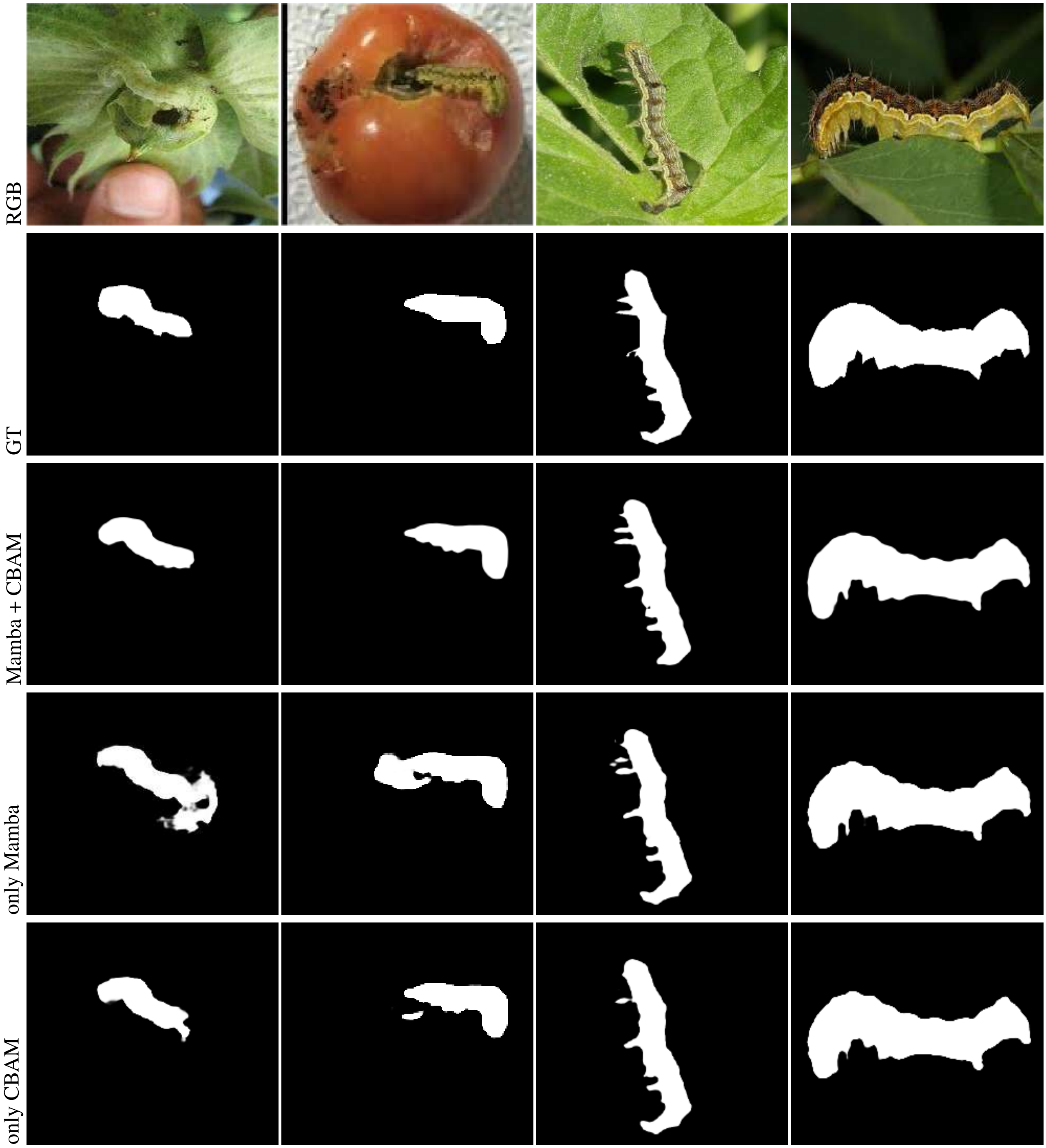

FIGURE 15. Qualitative results of different module combinations used, evaluated on some images from the Cotton Bollworm dataset [37].

TABLE 14. Metric evaluation with different outputs on testing set of the Cotton Bollworm dataset [37]. The best three performing results are highlighted using color: First , Second , and Third respectively.

Name	Output	$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$M \downarrow$	$F_\beta^{adp} \uparrow$	$F_\beta^{mean} \uparrow$	$E_\phi^{adp} \uparrow$	$E_\phi^{mean} \uparrow$
Avg	Avg	0.9261	0.9137	0.0082	0.8885	0.9121	0.9795	0.9799
M_1	$P_0 + P_1 + P_2 + P_3 + Avg$	0.9208	0.9197	0.0076	0.9092	0.9156	0.9838	0.9820
M_2	$P_1 + P_2 + P_3 + Avg$	0.9206	0.9177	0.0077	0.9054	0.9132	0.9829	0.9817
M_3	$P_2 + P_3 + Avg$	0.9210	0.9161	0.0078	0.9017	0.9117	0.9821	0.9813
M_4	$P_3 + Avg$	0.9215	0.9131	0.0080	0.8946	0.9093	0.9804	0.9807

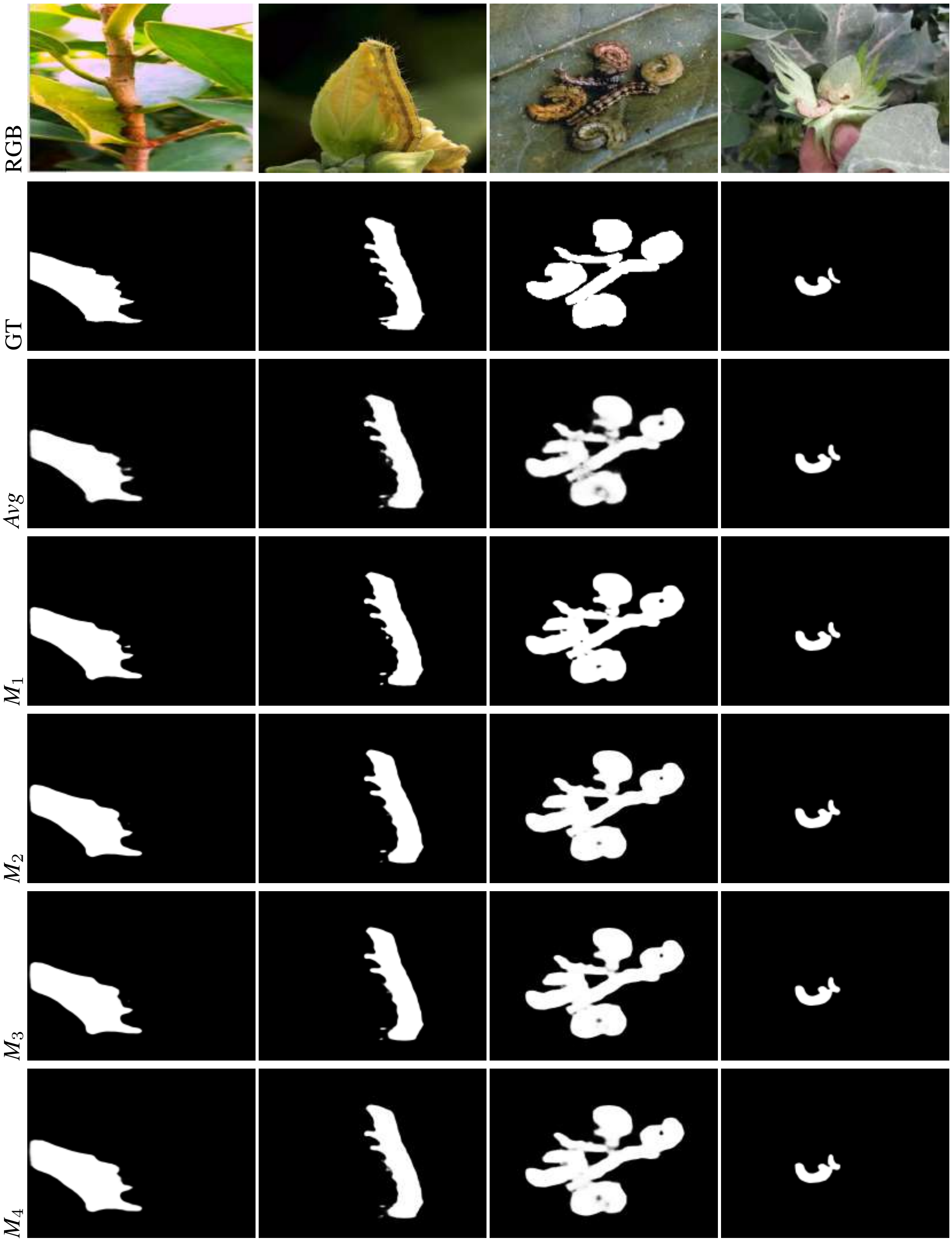

FIGURE 16. Qualitative results of different deep supervision outputs, evaluated on some images from the Cotton Bollworm dataset [37].

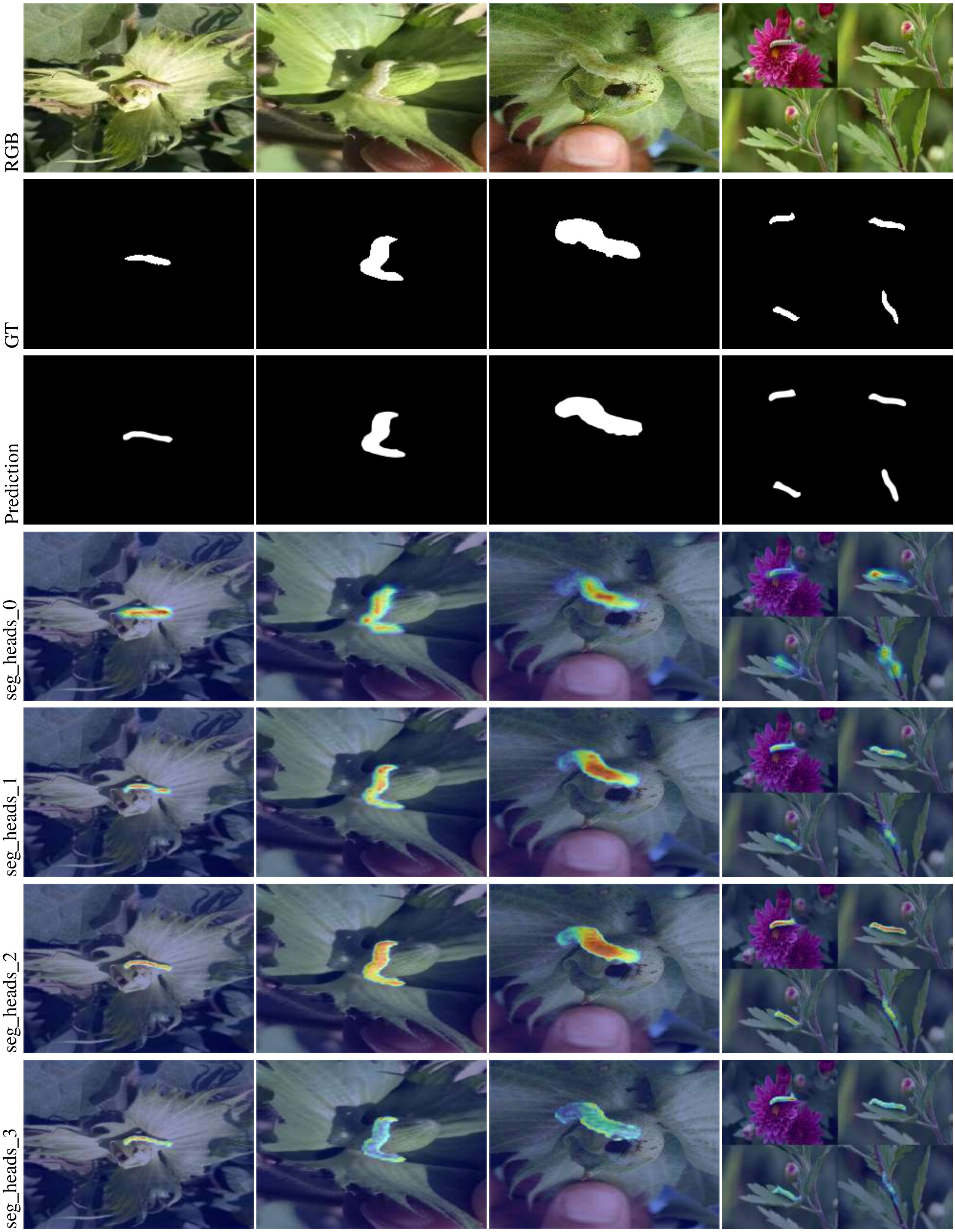

FIGURE 17. Grad-CAM visualization using segmentation heads outputs, evaluated on some images from the Cotton Bollworm dataset [37].

TABLE 15. Metric evaluation with different module combinations (e.g., Mamba and CBAM) on testing set of the Mango dataset [24]. The best results by each metric are highlighted in bold.

Module	$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$M \downarrow$	$F_\beta^{adp} \uparrow$	$F_\beta^{mean} \uparrow$	$E_\phi^{adp} \uparrow$	$E_\phi^{mean} \uparrow$
Mamba + CBAM	**0.8662**	**0.8280**	**0.0100**	**0.8222**	**0.8285**	**0.9545**	**0.9558**
only Mamba	0.8646	0.8131	0.0108	0.8113	0.8174	0.9494	0.9507
only CBAM	0.8511	0.7936	0.0119	0.7921	0.8013	0.9457	0.9497

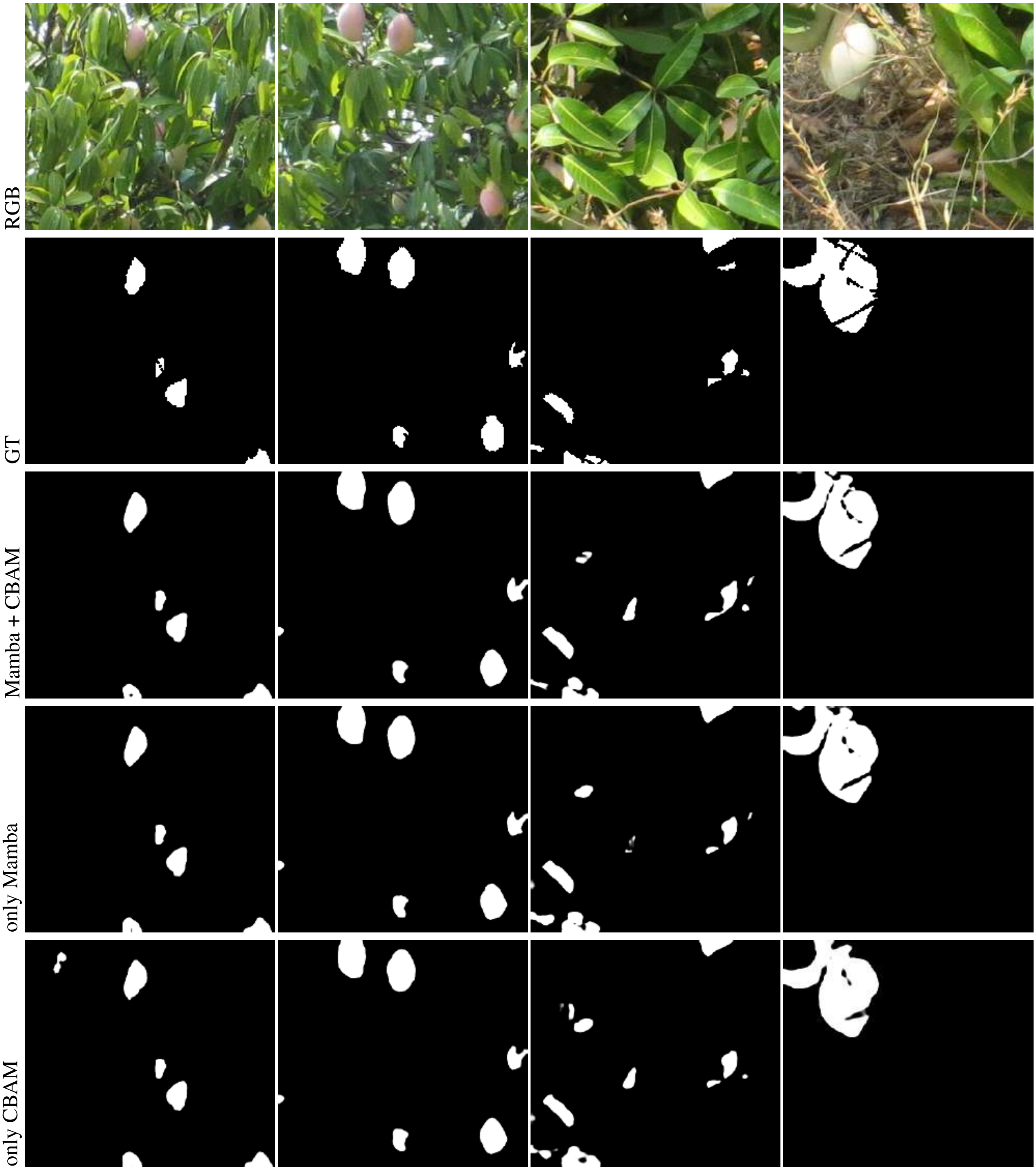

FIGURE 18. Qualitative results of different module combinations used, evaluated on some images from the Mango dataset [24].

TABLE 16. Metric evaluation with different outputs on testing set of the Mango dataset [24]. The best three performing results are highlighted using color: First, Second, and Third respectively.

Name	Output	$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$M \downarrow$	$F_\beta^{adp} \uparrow$	$F_\beta^{mean} \uparrow$	$E_\phi^{adp} \uparrow$	$E_\phi^{mean} \uparrow$
Avg	Avg	0.8478	0.8080	0.0113	0.7703	0.8199	0.9260	0.9491
M_1	$P_0 + P_1 + P_2 + P_3 + Avg$	0.8662	0.8280	0.0100	0.8222	0.8285	0.9544	0.9557
M_2	$P_1 + P_2 + P_3 + Avg$	0.8671	0.8263	0.0101	0.8146	0.8263	0.9510	0.9548
M_3	$P_2 + P_3 + Avg$	0.8663	0.8235	0.0103	0.8059	0.8240	0.9471	0.9539
M_4	$P_3 + Avg$	0.8634	0.8167	0.0107	0.7913	0.8208	0.9396	0.9541

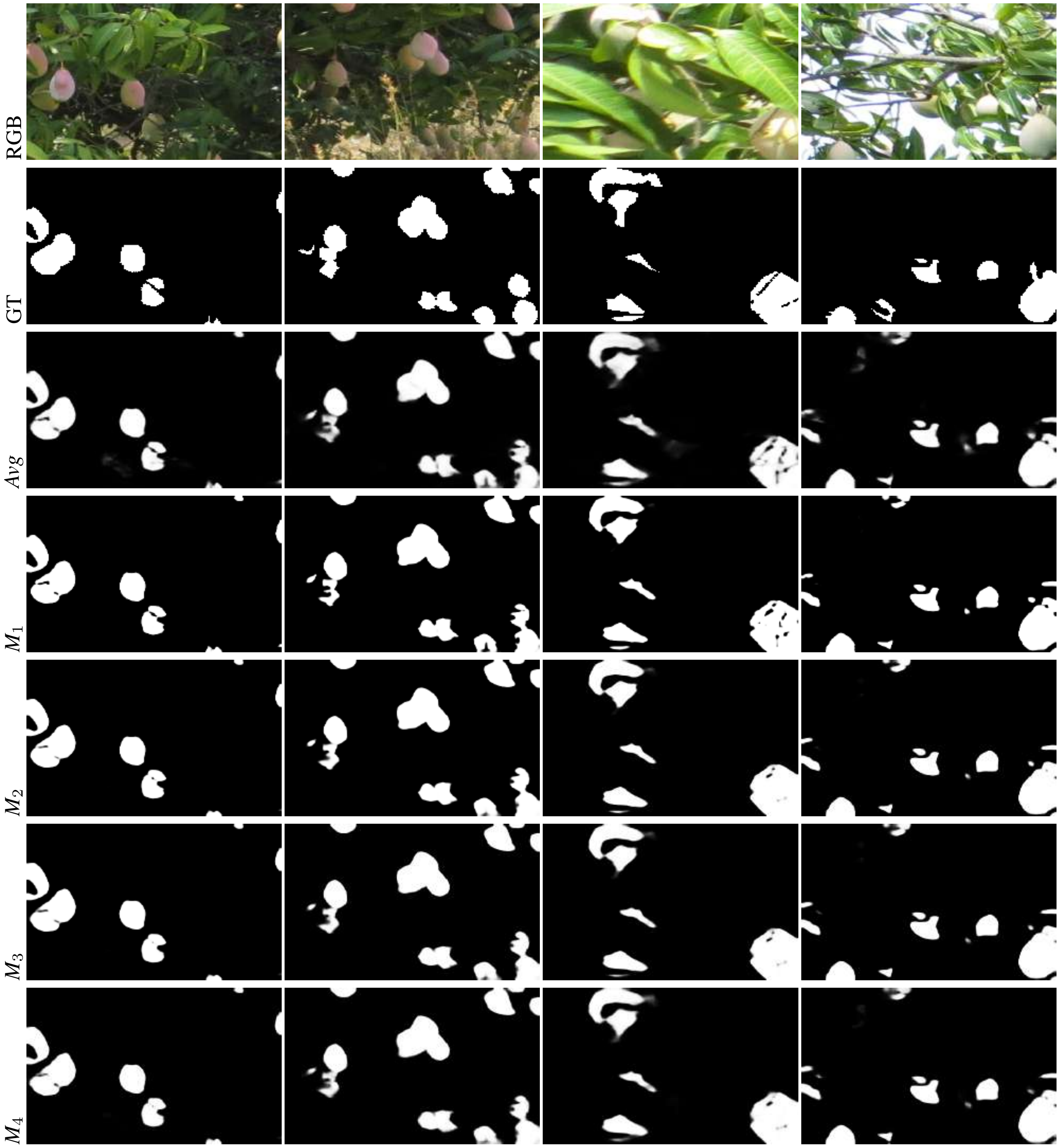

FIGURE 19. Qualitative results of different deep supervision outputs, evaluated on some images from the Mango dataset [24].

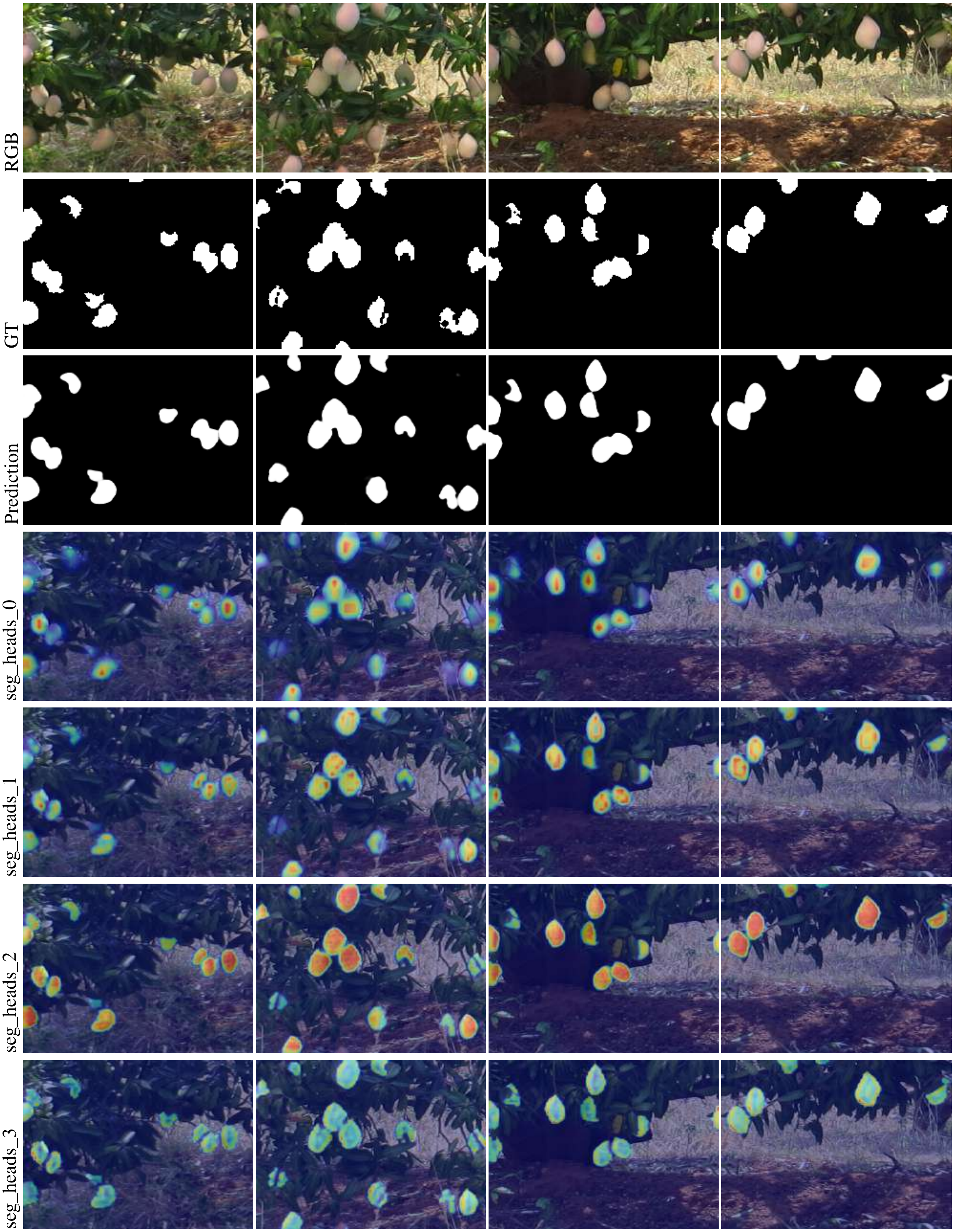

FIGURE 20. Grad-CAM visualization using segmentation heads outputs, evaluated on some images from the Mango dataset [24].

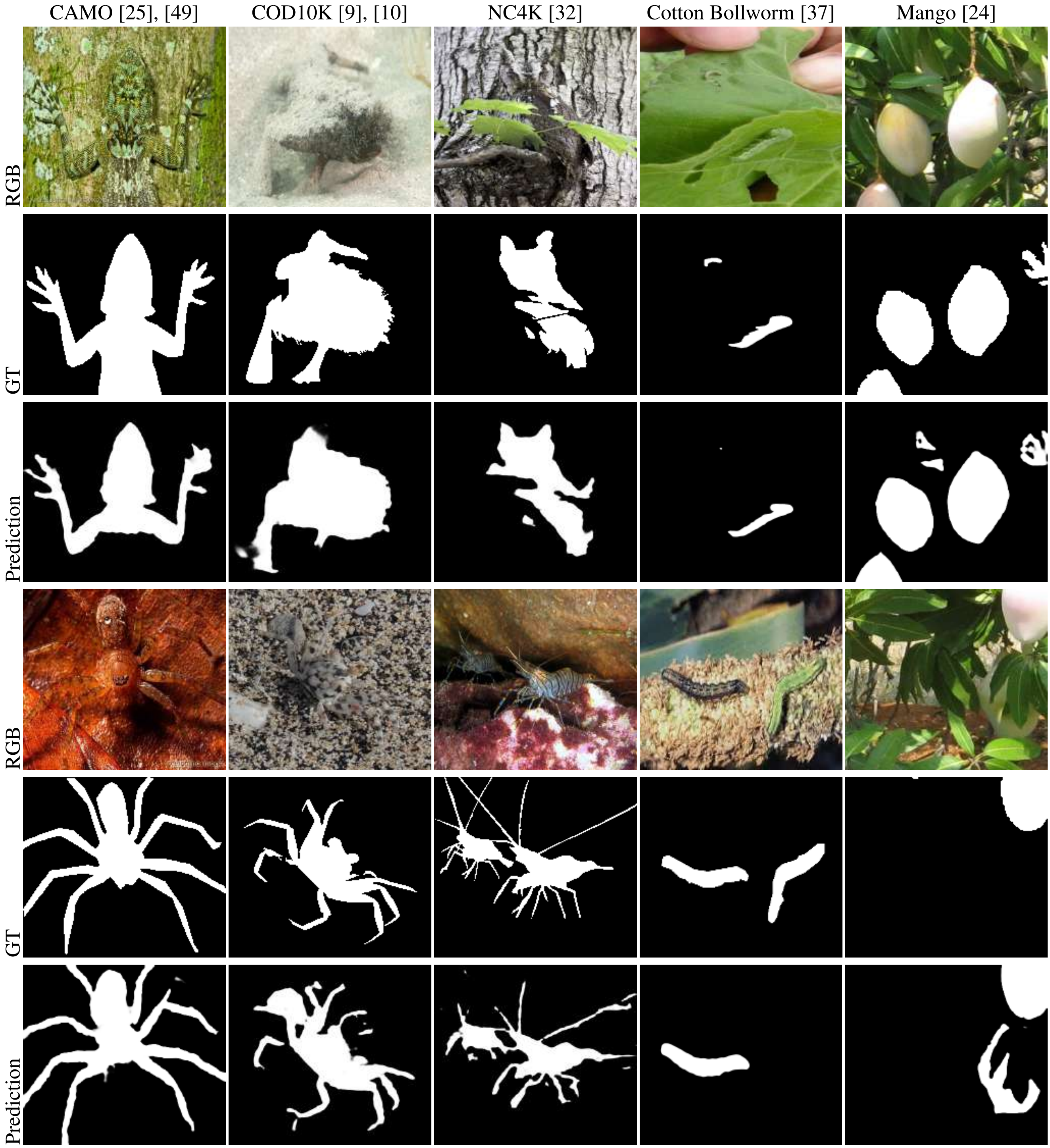

FIGURE 21. Analysis of AINet failure cases using two example images on standard COD benchmarks (CAMO [25], [49], COD10K [9], [10] and NC4K [32] datasets), as well as Cotton Bollworm [37] and Mango [24] datasets, summarizing common error patterns.

VII. CONCLUSIONS

This work introduces AINet, a novel architecture for camouflaged object detection that sets a new standard across both general benchmarks and agricultural case studies. On widely recognized benchmark datasets, AINet consistently achieves state-of-the-art results, surpassing existing methods in all evaluated metrics. The model demonstrates superior segmentation accuracy and edge delineation, effectively detecting a wide range of camouflaged objects and establishing itself as a leading solution for challenging COD scenarios.

Beyond general benchmarks, AINet's effectiveness is further validated through comprehensive case studies on agricultural datasets. On the Cotton Bollworm dataset, AINet excels in pest detection, delivering highly accurate identification and producing sharp, reliable segmentation masks even in complex backgrounds where pests are difficult to distinguish from vegetation. Similarly, on the Mango dataset, AINet achieves outstanding fruit detection performance, maintaining precise contours and minimal false positives under diverse illumination conditions—key requirements for automated fruit harvesting and monitoring systems.

The results demonstrate that AINet not only advances the state of the art in generic camouflaged object detection but also offers significant benefits for precision agriculture. Its robust performance across both benchmark and agricultural datasets highlights its versatility and potential impact, supporting more effective pest management, reducing crop losses, and enabling the development of intelligent, automated systems for modern farming.

In summary, AINet emerges as a powerful and generalizable solution for camouflaged object detection, delivering substantial improvements in accuracy and boundary precision across diverse real-world applications.

ACKNOWLEDGEMENTS

This work is supported in part by the Air Force Office of Scientific Research Under Award FA9550-24-1-0206; in part by the ESPOL project "Advancing Camouflaged Object Detection with a cost-effective Cross-Spectral vision system (ACODCS)" (CIDIS-003-2024).

REFERENCES

[1] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk. Frequency-tuned salient region detection. In *Conf. on Computer Vision and Pattern Recognition*, pages 1597–1604. IEEE, 2009.

[2] G. Chen, S.-J. Liu, Y.-J. Sun, G.-P. Ji, Y.-F. Wu, and T. Zhou. Camouflaged object detection via context-aware cross-level fusion. *Transactions on Circuits and Systems for Video Technology*, 32(10):6981–6993, 2022.

[3] T. Chen, J. Xiao, X. Hu, G. Zhang, and S. Wang. Boundary-guided network for camouflaged object detection. *Knowledge-based systems*, 248:108901, 2022.

[4] L. M. Dang, S. Danish, A. Khan, N. Alam, M. Fayaz, D. K. Nguyen, H.-K. Song, and H. Moon. An efficient zero-labeling segmentation approach for pest monitoring on smartphone-based images. *European Journal of Agronomy*, 160:127331, 2024.

[5] P.-T. De Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein. A tutorial on the cross-entropy method. *Annals of Operations Research*, 134:19–67, 2005.

[6] I. R. Evangelista, A. Bandala, and E. Dadios. Fcnet: A transformer-based context-aware segmentation framework for detecting camouflaged fruits in orchard environments. *Technologies*, 13(8):372, 2025.

[7] D.-P. Fan, M.-M. Cheng, Y. Liu, T. Li, and A. Borji. Structure-measure: A new way to evaluate foreground maps. In *Int. Conf. on Computer Vision*, pages 4548–4557, 2017.

[8] D.-P. Fan, C. Gong, Y. Cao, B. Ren, M.-M. Cheng, and A. Borji. Enhanced-alignment measure for binary foreground map evaluation. *arXiv*, 2018.

[9] D.-P. Fan, G.-P. Ji, M.-M. Cheng, and L. Shao. Concealed object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[10] D.-P. Fan, G.-P. Ji, G. Sun, M.-M. Cheng, J. Shen, and L. Shao. Camouflaged object detection. In *CVPR*, 2020.

[11] A. Gu and T. Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

[12] C. He, K. Li, Y. Zhang, L. Tang, Y. Zhang, Z. Guo, and X. Li. Camouflaged object detection with feature decomposition and edge reconstruction. In *Conf. on Computer Vision and Pattern Recognition*, pages 22046–22055, 2023.

[13] C. He, K. Li, Y. Zhang, G. Xu, L. Tang, Y. Zhang, Z. Guo, and X. Li. Weakly-supervised concealed object segmentation with sam-based pseudo labeling and multi-scale feature grouping. *Advances in Neural Information Processing Systems*, 36:30726–30737, 2023.

[14] C. He, K. Li, Y. Zhang, Y. Zhang, Z. Guo, X. Li, M. Danelljan, and F. Yu. Strategic preys make acute predators: Enhancing camouflaged object detectors by generating camouflaged objects. *arXiv preprint arXiv:2308.03166*, 2023.

[15] J. Hu, J. Lin, S. Gong, and W. Cai. Relax image-specific prompt requirement in sam: A single generic prompt for segmenting camouflaged objects. In *AAAI Conf. on Artificial Intelligence*, volume 38, pages 12511–12518, 2024.

[16] X. Hu, S. Wang, X. Qin, H. Dai, W. Ren, D. Luo, Y. Tai, and L. Shao. High-resolution iterative feedback network for camouflaged object detection. In *Conf. on Artificial Intelligence*, volume 37, pages 881–889, 2023.

[17] C. A. Javalagi, K. M. Medha, N. T. Patil, S. Itagalli, U. Kulkarni, and S. Chikkamath. Comparative Study of CNNs for Camouflaged Object Detection. In V. K. Gunjan and J. M. Zurada, editors, *Int. Conf. on Recent Trends in Machine Learning, IoT, Smart Cities and Applications*, pages 207–220, Singapore, 2024. Springer Nature Singapore.

[18] G.-P. Ji, D.-P. Fan, Y.-C. Chou, D. Dai, A. Liniger, and L. Van Gool. Deep gradient learning for efficient camouflaged object detection. *Machine Intelligence Research*, 20(1):92–108, 2023.

[19] G.-P. Ji, L. Zhu, M. Zhuge, and K. Fu. Fast camouflaged object detection via edge-based reversible re-calibration network. *Pattern Recognition*, 123:108414, 2022.

[20] Q. Jia, S. Yao, Y. Liu, X. Fan, R. Liu, and Z. Luo. Segment, magnify and reiterate: Detecting camouflaged objects the hard way. In *Conf. on Computer Vision and Pattern Recognition*, pages 4713–4722, 2022.

[21] P. Jiang, Y. Chen, B. Liu, D. He, and C. Liang. Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks. *IEEE Access*, 7:59069–59080, 2019.

[22] X. Jiang, W. Cai, Z. Zhang, B. Jiang, Z. Yang, and X. Wang. MAGNet: A camouflaged object detection network simulating the observation effect of a magnifier. *Entropy*, 24(12):1804, 2022.

[23] N. Kajiura, H. Liu, and S. Satoh. Improving camouflaged object detection with the uncertainty of pseudo-edge labels. In *ACM Int. Conference on Multimedia in Asia*, pages 1–7, 2021.

[24] R. Kestur, A. Meduri, and O. Narasipura. MangoNet: A deep semantic segmentation architecture for a method to detect and count mangoes in an open orchard. *Engineering Applications of Artificial Intelligence*, 77:59–69, 2019.

[25] T.-N. Le, T. V. Nguyen, Z. Nie, M.-T. Tran, and A. Sugimoto. Anabranch network for camouflaged object segmentation. *Journal of Computer Vision and Image Understanding*, 184:45–56, 2019.

[26] S. Lev-Yadun, A. Dafni, M. A. Flaishman, M. Inbar, I. Izhaki, G. Katzir, and G. Ne'eman. Plant coloration undermines herbivorous insect camouflage. *BioEssays*, 26(10):1126–1130, 2004.

[27] A. Li, J. Zhang, Y. Lv, T. Zhang, Y. Zhong, M. He, and Y. Dai. Joint salient object detection and camouflaged object detection via uncertainty-aware learning. *arXiv preprint arXiv:2307.04651*, 2023.

[28] P. Li, X. Yan, H. Zhu, M. Wei, X.-P. Zhang, and J. Qin. Findnet: Can you find me? boundary-and-texture enhancement network for camouflaged object detection. *IEEE Transactions on Image Processing*, 31:6396–6411, 2022.

[29] J. Liu, J. Zhang, and N. Barnes. Modeling aleatoric uncertainty for camouflaged object detection. In *Winter conference on applications of computer vision*, pages 1445–1454, 2022.

[30] Y. Lv, J. Zhang, Y. Dai, A. Li, N. Barnes, and D.-P. Fan. Toward deeper understanding of camouflaged object detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(7):3462–3476, 2023.

[31] Y. Lv, J. Zhang, Y. Dai, A. Li, B. Liu, N. Barnes, and D.-P. Fan. Simultaneously localize, segment and rank the camouflaged objects. In *Conf. on Computer Vision and Pattern Recognition*, pages 11591–11601, 2021.

[32] Y. Lyu, J. Zhang, Y. Dai, A. Li, B. Liu, N. Barnes, and D.-P. Fan. Simultaneously localize, segment and rank the camouflaged objects. In *Conf. on Computer Vision and Pattern Recognition*, 2021.

[33] J. Ma, F. Li, and B. Wang. U-Mamba: Enhancing Long-range Dependency for Biomedical Image Segmentation. *arXiv:2401.04722*, 2024.

[34] R. Margolin, L. Zelnik-Manor, and A. Tal. How to evaluate foreground maps? In *Conf. on Computer Vision and Pattern Recognition*, pages 248–255, 2014.

[35] G. Máttyus, W. Luo, and R. Urtasun. Deeproadmapper: Extracting road topology from aerial images. In *Int. Conf. on Computer Vision*, pages 3438–3446, 2017.

[36] H. Mei, G.-P. Ji, Z. Wei, X. Yang, X. Wei, and D.-P. Fan. Camouflaged object segmentation with distraction mining. In *Conf. on Computer Vision and Pattern Recognition*, pages 8772–8781, 2021.

[37] K. Meng, K. Xu, P. Cattani, and S. Mei. Camouflaged cotton bollworm instance segmentation based on PVT and Mask R-CNN. *Computers and Electronics in Agriculture*, 226:109450, 2024.

[38] Y. Pang, X. Zhao, T.-Z. Xiang, L. Zhang, and H. Lu. Zoom in and out: A mixed-scale triplet network for camouflaged object detection. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pages 2160–2170, 2022.

[39] F. Perazzi, P. Krähenbühl, Y. Pritch, and A. Hornung. Saliency filters: Contrast based filtering for salient region detection. In *Conf. on Computer Vision and Pattern Recognition*, pages 733–740. IEEE, 2012.

[40] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, and M. Jagersand. BASNet: Boundary-Aware Salient Object Detection. In *Conf. on Computer Vision and Pattern Recognition*, June 2019.

[41] D. Sun, S. Jiang, and L. Qi. Edge-aware mirror network for camouflaged object detection. In *Int. Conf. on Multimedia and Expo*, pages 2465–2470. IEEE, 2023.

[42] Y. Sun, G. Chen, T. Zhou, Y. Zhang, and N. Liu. Context-aware Cross-level Fusion Network for Camouflaged Object Detection. In *IJCAI*, pages 1025–1031, 2021.

[43] H. Velesaca, A. Mero, R. Rivadeneira, G. Castillo, and A. Sappa. AVNet: Cross-Spectral Attention-Vision Model for Camouflaged Object Detection in Ecological Conservation. In *Int. Conf. on Computer Vision Theory and Applications*, pages 1–10. INSTICC, SciTePress, 2026.

[44] H. Velesaca, H. Villegas, and A. Sappa. Exploring Camouflaged Object Detection Techniques for Invasive Vegetation Monitoring. In *Int. Conf. on Data Science, Technology and Applications*, pages 1–8. INSTICC, SciTePress, 2025.

[45] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao. Pvt v2: Improved baselines with pyramid vision transformer. *Computational Visual Media*, 8(3):415–424, 2022.

[46] J. Wei, S. Wang, and Q. Huang. F^{3}Net: fusion, feedback and focus for salient object detection. In *AAAI Conf. on Artificial Intelligence*, volume 34, pages 12321–12328, 2020.

[47] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.

[48] F. Xiao, S. Hu, Y. Shen, C. Fang, J. Huang, C. He, L. Tang, Z. Yang, and X. Li. A survey of camouflaged object detection and beyond. *arXiv preprint arXiv:2408.14562*, 2024.

[49] J. Yan, T.-N. Le, K.-D. Nguyen, M.-T. Tran, T.-T. Do, and T. V. Nguyen. Mirrornet: Bio-inspired camouflaged object segmentation. *IEEE Access*, 9:43290–43300, 2021.

[50] X. Yan, M. Sun, Y. Han, and Z. Wang. Camouflaged object segmentation based on matching–recognition–refinement network. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[51] F. Yang, Q. Zhai, X. Li, R. Huang, A. Luo, H. Cheng, and D.-P. Fan. Uncertainty-guided transformer reasoning for camouflaged object detection. In *Int. Conf. on Computer Vision*, pages 4146–4155, 2021.

[52] J. Yang, Q. Wang, F. Zheng, P. Chen, A. Leonardis, and D.-P. Fan. Plant-Camo: Plant Camouflage Detection. *arXiv*, 2024.

[53] Q. Zhai, X. Li, F. Yang, C. Chen, H. Cheng, and D.-P. Fan. Mutual graph learning for camouflaged object detection. In *Conf. on Computer Vision and Pattern Recognition*, pages 12997–13007, 2021.

[54] M. Zhang, S. Xu, Y. Piao, D. Shi, S. Lin, and H. Lu. Preynet: Preying on camouflaged objects. In *Int. Conf. on Multimedia*, pages 5323–5332, 2022.

[55] Y. Zhang and C. Wu. Unsupervised camouflaged object segmentation as domain adaptation. In *Int. Conf. on Computer Vision*, pages 4334–4344, 2023.

[56] Y. Zhang, J. Zhang, W. Hamidouche, and O. Deforges. Predictive uncertainty estimation for camouflaged object detection. *IEEE Transactions on Image Processing*, 32:3580–3591, 2023.

[57] Y. Zhong, B. Li, L. Tang, S. Kuang, S. Wu, and S. Ding. Detecting camouflaged object in frequency domain. In *Conf. on Computer Vision and Pattern Recognition*, pages 4504–4513, 2022.

[58] T. Zhou, Y. Zhou, C. Gong, J. Yang, and Y. Zhang. Feature aggregation and propagation network for camouflaged object detection. *IEEE Transactions on Image Processing*, 31:7036–7047, 2022.

[59] J. Zhu, X. Zhang, S. Zhang, and J. Liu. Inferring camouflaged objects by texture-aware interactive guidance network. In *AAAI Conf. on Artificial Intelligence*, volume 35, pages 3599–3607, 2021.

HENRY O. VELESACA received his degree in Computer Science Engineering from the Escuela Superior Politécnica del Litoral (ESPOL), Ecuador, in 2014. He obtained an Honors Diploma for Best Graduate of the Computer Science Engineering Degree in 2013. In 2017, he obtained a Master's in Software Development at the University of Granada. In the same year, he received Second Place in the First Agro-Food Innovation Contest of Grupo La Caña, Spain - Motril. He is currently a Ph.D. student in the Doctoral Program in Information and Communication Technologies at the University of Granada. Also works as a research technician at ESPOL. His research interests include deep learning, computer vision, and robust model design, with applications in agricultural, industrial automation, and sports analytics.

ANDREA MERO P. received the B.Sc. degree in computer science from the Escuela Superior Politécnica del Litoral (ESPOL), Guayaquil, Ecuador, in 2024. She is currently pursuing the M.Sc. degree in Artificial Intelligence with the Università della Svizzera italiana (USI), Lugano, Switzerland. She has held multiple research roles at ESPOL, including a position as a Computer Vision Research Assistant with the Centro de Investigación, Desarrollo e Innovación de Sistemas Computacionales (CIDIS). Her research contributions span computer vision, deep learning, and data analysis, with a focus on cross-spectral imagery, super-resolution, and camouflaged object detection.

ABEL REYES-ANGULO received the Ph.D. degree in Computational Science and Engineering from Michigan Technological University, Houghton, MI, USA, in 2025. He is currently a Senior Data Scientist with the Pricing and Assortment team at Walmart Global Tech and continues to collaborate as a researcher with academic partners in medical imaging and computer vision. His research interests include efficient deep learning architectures, medical image segmentation, representation learning, and robust model design, with applications in healthcare, remote sensing, and sports analytics. He has co-authored several peer-reviewed publications in these areas and is actively involved in the AI community, including service and mentoring roles in Latin American AI initiatives.

ANGEL D. SAPPA (Senior Member, IEEE) received the degree in electromechanical engineering from the National University of La Pampa, Argentina, in 1995, and the Ph.D. degree in industrial engineering from the Polytechnic University of Catalonia, Spain, in 1999. He is currently a Senior Scientist with the Computer Vision Center, Universitat Autónoma de Barcelona, Spain; and a full-time Professor with ESPOL Polytechnic University, Ecuador, where he leads the Computer Vision Team, CIDIS Research Center. He has over 25 years of experience in computer vision and artificial intelligence, with more than 300 publications in top-ranked journals and leading international conferences. From 2022 to 2024, he was included in Stanford University's top 2% most-cited researchers in the subfield of artificial intelligence (career-long ranking). He serves as an Editor in Chiev of ELCVIA Journal, an Associate Editor for Pattern Recognition and a Guest Editor for Robotics and Autonomous Systems and Journal of Computational Science.