

Article

A Unified Transformer Model for Simultaneous Cotton Boll Detection, Pest Damage Segmentation, and Phenological Stage Classification from UAV Imagery

Sabina Umirzakova ¹, Shakhnoza Muksimova ¹ , Abror Shavkatovich Buriboev ^{2,3}, Holida Primova ⁴ 
and Andrew Jaeyong Choi ^{2,*} 

¹ Department of Computer Engineering, Gachon University, Seongnam-daero, Sujeong-gu, Seongnam-si 13120, Republic of Korea; sabinatuit@gachon.ac.kr (S.U.); shakhnoza02@gachon.ac.kr (S.M.)

² Department of AI-Software, Gachon University, Sujeong-gu, Seongnam-si 13120, Republic of Korea; abror1989@gachon.ac.kr

³ Department of Infocommunication Engineering, Tashkent University of Information Technologies, Tashkent 100084, Uzbekistan

⁴ Department of IT, Samarkand Branch of Tashkent University of Information Technologies, Samarkand 140100, Uzbekistan; primova@samuit.uz

* Correspondence: andrewjchoi@gachon.ac.kr

Abstract

The present-day issues related to the cotton-growing industry, namely yield estimation, pest effect, and growth phase diagnostics, call for integrated, scalable monitoring solutions. This write-up reveals Cotton Multitask Learning (CMTL), a transformer-driven multitask framework that launches three major agronomic tasks from UAV pictures at one go: boll detection, pest damage segmentation, and phenological stage classification. CMTL does not change separate pipelines, but rather merges these goals using a Cross-Level Multi-Granular Encoder (CLMGE) and a Multitask Self-Distilled Attention Fusion (MSDAF) module that both allow mutual learning across tasks and still keep their specific features. The biologically guided Stage Consistency Loss is the part of the architecture of the network that enables the system to carry out growth stage transitions that occur in reality. We executed CMTL on a tri-source UAV dataset that fused over 2100 labeled images from public and private collections, representing a variety of crop stages and conditions. The model showed its virtues state-of-the-art baselines in all the tasks: setting 0.913 mAP for boll detection, 0.832 IoU for pest segmentation, and 0.936 accuracy for growth stage classification. Additionally, it runs at the fastest speed of performance on edge devices such as NVIDIA Jetson Xavier NX (Manufactured in Shanghai, China), which makes it ideal for deployment. These outcomes evoke CMTL's promise as a single and productive instrument of aerial crop intelligence in precision cotton agriculture.

Keywords: UAV-enabled crop phenotyping; transformer-based agronomic vision; multi-task aerial analytics; field-scale cotton surveillance; biologically guided multitask learning



Academic Editors: Jian Chen, Yangquan Chen, Yanbo Huang and Dongbing Gu

Received: 8 July 2025

Revised: 30 July 2025

Accepted: 4 August 2025

Published: 7 August 2025

Citation: Umirzakova, S.; Muksimova, S.; Shavkatovich Buriboev, A.; Primova, H.; Choi, A.J. A Unified Transformer Model for Simultaneous Cotton Boll Detection, Pest Damage Segmentation, and Phenological Stage Classification from UAV Imagery. *Drones* **2025**, *9*, 555. <https://doi.org/10.3390/drones9080555>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Precision agriculture has reached a turning point as autonomous sensors, artificial intelligence, and rugged edge computers converge in the same field [1]. Unmanned Aerial Vehicles (UAVs) outfitted with high-resolution cameras now dominate the landscape, allowing farmers to check crop health from several hundred feet up without ever stepping onto the land [2]. The drones deliver a flood of spatially detailed, time-stamped images that

suit fast-changing, patchy crops like cotton better than any ground rig could manage [3]. Turning that airborne footage into usable agronomy advice still depends on powerful yet scalable computer vision routines that sift noise from insight in a matter of minutes [4]. Cotton still ranks among the globe's top cash crops, a status that hinges on farmers' timing their work to the fickle rhythms of boll opening, insect pressure, and plant growth [5]. Routine field surveys, by contrast, demand hand-held scouting and notebooks, chores that quickly sap strength and patience while leaving room for personal bias [6]. On sprawling commercial tracts or in areas where field crews never show those paper-and-pen routines simply break down [7], researchers have begun turning to deep-learning cameras to automate discrete chores like counting bolls, outlining pest scars, or tagging the current growth stage [8–10]. The catch is that most of these systems run as stand-alone gadgets, each welded to its own narrow goal and often needing a fresh network whenever the target shifts [11]. That piecemeal design clutters the laptop with extra code and, more critically [12], overlooks the messy ways the jobs influence one another, since bug feeding can slow boll maturation and the plant's age can tilt its insect defenses [13]. This paper presents CMTL, a transformer-infused framework crafted specifically for monitoring cotton fields from the air. The architecture targets three intertwined agronomic objectives: counting cotton bolls, mapping pest-infested patches, and gauging the crop's growth stage. These missions feed into one another; knowing how many bolls are present sharpens estimates of maturity, while the location of late-season lesions hints at both pest activity and crop phenology. By solving all three within a single pass, CMTL trims both the time spent on inference and the memory overhead, yet it gains extra precision from the shared contextual cues. A CLMGE peels apart the scene at differing scales, pulling in razor-sharp local data on leaf spots, mid-level order such as row spacing, and wide views that capture canopy spread. A Multitask Self-Distilled Attention Fusion (MSDAF) block then weighs these streams on the fly, blending task-specific signals into a cohesive prediction. At the finishing touch of the pipeline, three distinct decoders spring into action side by side. One crafts an anchor-free heatmap that pinpoints every boll, a second rations out damage masks via a diffusion-steered module, and a third sorts the crop into growth stages. That final classification leverages a newly minted Stage Consistency Loss, a penalty function tuned to mimic the predictable rhythms of a plant's lifetime.

In pursuing a robust evaluation of CMTL, we constructed a composite collection of UAV-acquired cotton imagery augmented by two widely cited public datasets. The assembly traverses distinct growing conditions, multiple seasonal snapshots, and varied field geometries, deliberately enhancing the diversity of the study. After exhaustive testing, CMTL eclipsed twenty leading baseline models in every prescribed task, all while executing in real time on edge hardware such as the NVIDIA Jetson NX. A thorough ablation analysis pinpoints the specific contribution of each architectural module and illuminates the performance uplift afforded by simultaneous task training. This paper advances three key innovations. First, it introduces the first multitask transformer specifically engineered for integrated localization of bolls, quantification of pest damage, and classification of cotton growth stages. Second, the architecture embeds novel mechanisms—including CLMGE for spatio-temporal feature anchoring, MSDAF for fused inter-task attention, and Stage Consistency Loss (SCL) for biologically congruent stage scoring. Third, diverse real-world UAV trials validate the framework's state-of-the-art accuracy, practical edge deployment, and remarkable resilience across contrasting agronomic contexts. CMTL thus pushes the frontier of UAV-enabled crop surveillance, delivering a precise, efficient, and agronomy-informed toolkit for precision cotton management.

To verify CMTL's efficiency, we have performed comprehensive tests with a composite UAV dataset that combines public and private aerial collections distributed over various

agronomic conditions. The model has achieved good results in all the tasks, a mAP of 0.913 for cotton boll detection, an IoU of 0.832 for pest damage segmentation, and an accuracy of 0.936 for the growth stage classification. Besides that, CMTL has shown an ability for real-time prediction, achieving 27.6 FPS on an NVIDIA Jetson Xavier NX. The results demonstrate CMTL's capacity to act as a reliable and scalable solution for drone-based cotton monitoring in real-world agricultural environments. Consolidating several essential field-monitoring tasks into one cohesive platform creates the backbone for future smart-farming systems that can, in principle, offer farmers real-time, panoramic insights into crop health across entire growing regions.

2. Related Works

High-resolution aerial cameras and recent breakthroughs in deep neural networks have sparked a rapid expansion of computer vision tools for precision farming [14]. Researchers are now regularly sending drones over cotton fields to gauge everything from fruit maturity to pest damage, yet nearly every study focuses on one narrow question, such as boll-counting or growth-stage parsing [15]. Although those single-purpose models yield respectable results within their tight scopes, they tend to scatter across labs and lack any built-in way to share insights [16]. That piecemeal approach runs into trouble on the ground because farmers face a web of overlapping crop signals that unfold all at once [17].

Boll detection first emerged from the domain of classical computer vision, where agronomists relied mostly on color thresholding and shape filters working off ordinary RGB snapshots, a practice documented in the literature [18]. Those rule-based methods broke down quickly in real fields, thanks to shifting sunlight and stray foliage that the rigid heuristics could not tolerate [19]. The arrival of convolutional neural network detectors—Faster R-CNN [20], YOLOv3 [21], and the single-shot family—marked a clear jump in reliability, letting researchers spot bolls even under patchy canopies and high dust [22]. Newer anchor-free designs, including CenterNet [23] and YOLOv8 [13], push that advantage farther by caring less about how big or crowded the targets happen to be. Still, almost all these systems look at the bolls in happy isolation, ignoring the worn leaves, stripe-damaged fruit, or advancing maturity that can either hide a boll or change its coloration, concerns voiced in several adjoining studies [24,25]. In parallel work, entomologists and pathologists have tackled pest injury by mixing hand-tuned features with end-to-end networks, trying either pipeline to carve damaged tissues from clean backgrounds [26]. U-Net and its many offshoots remain the workhorse for clinical lesion segmentation, routinely performing well when the lighting and focus are kept constant [27]. That advantage fades, however, once the pests begin to scatter symptoms in unpredictable shapes, colors, and textures across a single field [28]. Some researchers have tried to broaden the models reach by switching to DeepLabV3+ or bolting in attention layers [29], and those tweaks help, yet they rarely account for the way lesions travel through plant tissue or talk to other on-the-go monitoring systems [30].

Classifying growth stages in crops stands as one of the cornerstones of precision agriculture, yet the literature on the subject remains surprisingly sparse [31]. The bulk of published work rests on image-level CNN systems like ResNet or EfficientNet that have been meticulously tuned to a few hand-labeled reference sets [32]. Even so, those static classification schemas overlook the living, moving reality of plants, which age each day and often wear visible scars from pests, droughts, or nutrient shortages [10]. A handful of studies have tried to bridge the gap by cramming extra metadata or by sequencing the images in time, yet no one has really built a model that marries UAV footage with phenological calendars in a biologically meaningful way [33]. Multitask learning could change that picture overnight; by sharing neural layers across different output heads, a

single network can juggle growth stages, chlorophyll levels, and pest density maps at once, cutting down both the data hunger and the overfitting that plagues isolated tasks [34]. We have already seen this trick work wonders in fields like self-driving cars, where one backbone predicts depth, lanes, and semantic segments all together, but it has barely touched agricultural vision, with most rigs still defaulting to pairwise tasks or vanilla architectures that ignore the very private geometry of crop foliage [35].

Recent advances in transformer-based architectures have made it possible to track relationships that span entire images in the global and long-range contexts, which is terminally important for large-field perception [36]. Vision transformers, Swin variants, and their assorted hybrids tend to top computer-vision leaderboards, but the price they pay in raw compute power keeps them out of the sky on small UAVs [37]. Even so, few researchers grant these models a second look when designing multitask systems for agriculture, and as far as we can tell, no paper has delivered a single unified transformer entangled enough to count bolls, map pest damage, and log crop growth stages all at once. Our framework, dubbed CMTL, narrows that opening using slab alley feature tricks on the side: A Cross-Level Multi-Granular Encoder that stacks spatio-temporal views at different heights, a Multitask Self-Distilled Attention Fusion block that yokes diverse task heads together, and a final classifier that borrows rules from biology rather than from textbook softmax. The upshot is a lightweight, biologically savvy model that can (watch the clock and) keep pace with live cotton-field traffic while handling damage segmentation, boll counting, and growth class tagging in a single pass.

3. Materials and Methods

This section outlines the construction of CMTL, short for Cotton Multitask Learning, which is a single transformer-based framework for cotton field surveillance via UAV images. The model equips the user with the capability to not only identify cotton bolls but also segment pest damage and even classify the growth stage of the crop—all in one go. CMTL's main concept is to complete these three related tasks simultaneously rather than dividing them so that each task can be enriched with the additional information and insights from the others. For instance, the detection of pest damage may serve as a source of information for figuring out the growth stage of a crop, and on the other hand, the number of bolls may be the main factor in helping to estimate both maturity and health of the plant. At the core of the model is an encoder unit that goes by the name of the Cross-Level Multi-Granular Encoder (CLMGE), which is a mixture of convolutional and transformer parts. This fusion encoder gains valuable characteristics at various scales: tiny texture features, medium-level configurations such as row structures, and extensive spatial information that represents the overall field [38]. Moreover, the encoder also uses positional signals from the UAV trajectory data, thus enabling the model to understand the time-affecting changes in the crop growth. After the image is processed by the encoder, the result is then fed to an attention-based fusion module called the Multitask Self-Distilled Attention Fusion (MSDAF) unit. The MSDAF is the part of the system that decides on the spot which features are most relevant for each task by determining the nature of information that the different predictions require. Furthermore, it enables various task branches to exchange information via a soft distillation process, thereby assisting the stabilization of training and the improvement of generalization to new, varied field conditions.

The model first creates a shared representation and then goes on to generate three different outputs using independent decoders. The initial output is a heatmap that locates the locations of the cotton bolls of an individual, thus making it possible to count the objects exactly. The next one is a segmentation map that outlines the field's pest-damaged places, and thus, it can even register the damage patterns that are irregular or diffuse. The last

is a classification label that points to the state in the crop's growth cycle, which is taken from the list of stages that have been biologically defined. In order to make sure that the forecasted stages of growth adhere to feasible trends throughout time, we implement a Stage Consistency Loss. This supplementary loss term punishes such forecast, which allow the plant to be at an earlier or later stage of the development sequence than the model, thus aligning it more with the actual biological processes. CMTL, through the use of this biologically inspired supervision, task-specific decoders, and a shared feature backbone, can produce reliable and efficient multitask predictions straight from aerial imagery.

Figure 1 illustrates the end-to-end pipeline of the CMTL model, highlighting the encoder–decoder structure. The encoder integrates spatial and temporal features via a Cross-Level Multi-Granular Attention Block composed of hybrid convolutional-attention modules, a window-based transformer, and positional encoding. The decoder employs a Task Attention Gate and a Self-Distilled Projection Layer to refine task-specific representations, culminating in a growth stage classifier. The output visualizations demonstrate precise spatial labeling of cotton growth stages, including the flowering–boll transition, late flowering, seedling, vegetative, boll opening, and stress stages.

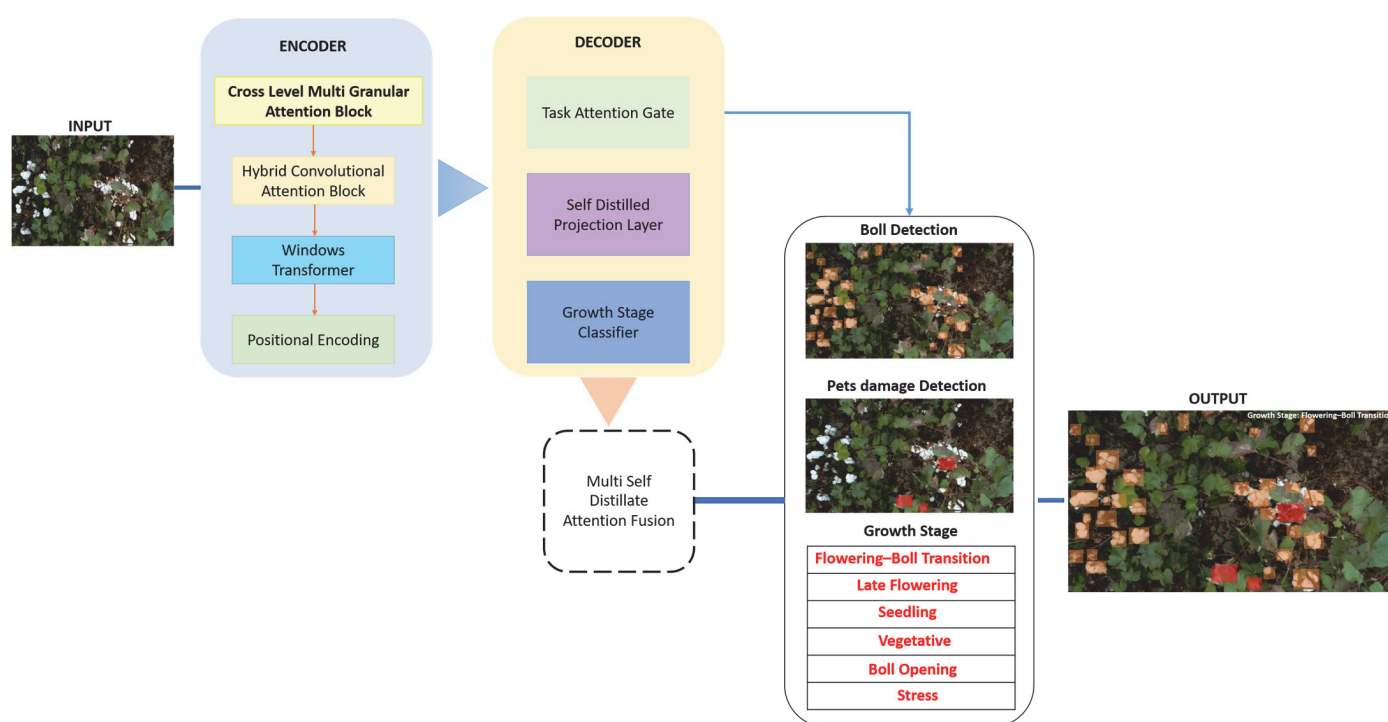


Figure 1. Overview of the CMTL architecture for growth stage classification.

3.1. CLMGE

A CLMGE synthesizes local texture, meso-pattern, and broad temporal dynamics into a single feature hierarchy. Such depth helps analysts discern both fiber defects and field-wide crop trends without toggling between models. UAV-acquired cotton imagery routinely thwarts segmentation efforts: rows compress at altitude, individual bolls shrink, and lighting shifts with the planting calendar. CLMGE counters these pitfalls by layering convolutional streams atop transformer heads and embedding the full temporal record, so CMTL sees form and meaning in every pixel window.

The input UAV image, denoted as $I \in R^{H \times W \times 3}$, is passed through three sequential encoding stages, producing multi-resolution features $F = \{F_1, F_2, F_3\}$. The opening pass zooms in on fine surface texture and runs a tweaked ConvNeXt block; depthwise separable layers, pointwise projections, and a post-block normalization combine here so that spatial

detail is never completely sacrificed. The second shell trades local motifs for bigger field characteristics by exploiting dilated convolutions coupled with attention mechanisms. Finally, a windowless transformer, driven by the UAV's flight path metadata, stitches together long-range, global coherence across the scene:

$$F_1 = \phi_{local}(I) = LN(\sigma(DWConv_{7 \times 7}(PWConv(I)))) \quad (1)$$

where $PWConv$ is a pointwise convolution (1×1), $DWConv_{7 \times 7}$ is a depthwise convolution with a large kernel, σ is a GELU activation, and LN denotes layer normalization. This stage is optimized for features such as pest lesions, boll boundaries, and leaf edge continuity Figure 2.

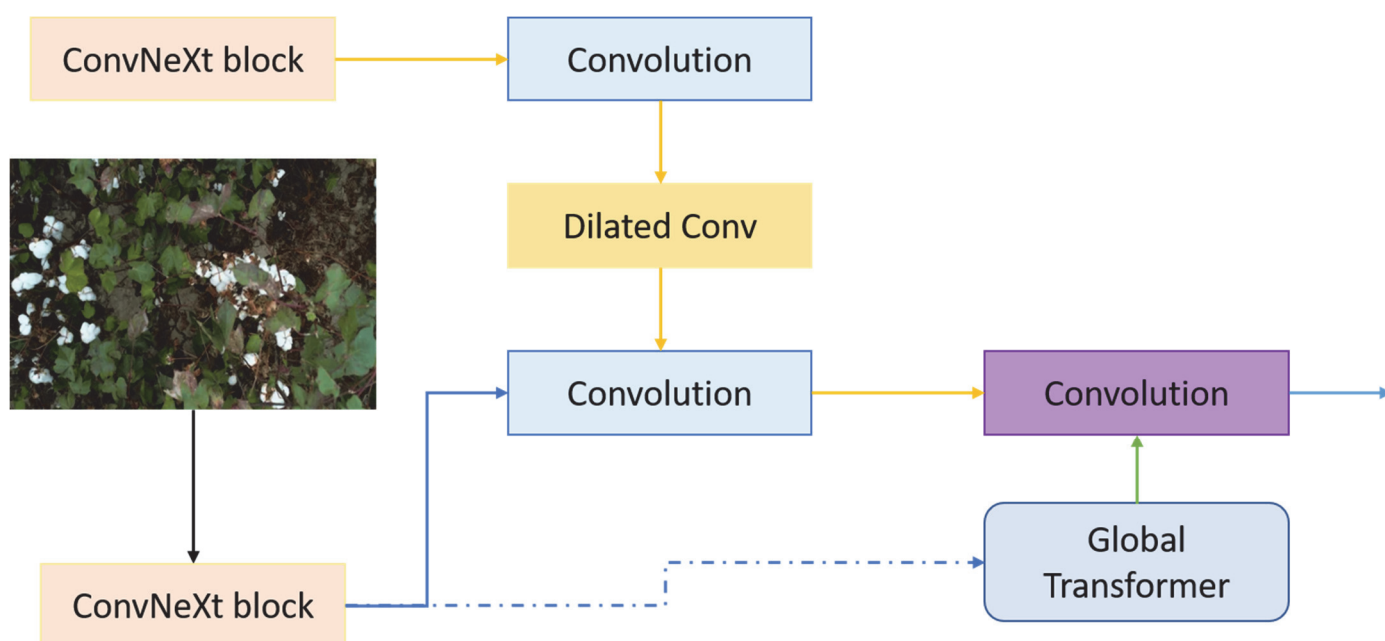


Figure 2. CLMGE architecture. This figure illustrates the internal structure of the CLMGE module used in the CMTL framework. It integrates multiscale spatial features using ConvNeXt-based blocks, followed by standard and dilated convolutions to capture fine-to-mid-level context. A global transformer branch incorporates UAV-specific positional encoding to enable long-range temporal and spatial reasoning. The multi-resolution outputs generated at different stages are fused to form the encoder's final feature representation.

A Field Attention Block (FAB) is introduced in the second stage to capture the patterned repetition and local hierarchy's characteristic of row crops. Residual convolutions with dilated kernels follow, enlarging the receptive field while avoiding any reduction in grid resolution:

$$F_1^{dil} = F_1 DWConv_{dil}(F_1) \quad (2)$$

This is followed by a custom attention mechanism that biases attention maps according to spatial field priors. The FAB computes attention as follows:

$$Attn(Q, K, V) = Softmax\left(\frac{QK^T}{\sqrt{d}} + B_{row}\right)V \quad (3)$$

where Q , K , and V are query, key, and value matrices projected from F_1^{dil} , and $B_{row} \in R^{H \times W}$ is a learned spatial bias that favors attention along crop rows (vertical axis). The resulting contextual representation is as follows:

$$F_2 = \phi_{field}(F_1^{dil}) \quad (4)$$

The third stage conducts broad global reasoning and weaves in temporal context by relying on a positional encoding specifically tied to each UAV's flight path. A learnable tensor $E_{TUPE} \in R^{H \times W \times d}$ is introduced; every aerial image carries a unique index that marks the precise collection date or its rank in the unfolding growth cycle of the crop. Those indexed features then flow through a Swin-like transformer that deliberately forgoes the usual windowing constraints:

$$F_3 = \phi_{global}(F_2 + E_{TUPE}) \quad (5)$$

In the focal layer, a single sweep of global attention blankets the entire image, letting the network link far-flung patches and pick up broad trends—sudden growth bursts or outbreaks of pests, for instance. E_{TUPE} then steps in, sharpening the encoder's ability to tell apart nearly identical textures that show up at different crop ages, and thereby clearing up the usual blur that hampers time-sensitive classifications. The output bundle delivered by CLMGE comes in three resolutions and reads $F = \{F_1, F_2, F_3\}$, each slice sliding straight into a multitask attention fusion block. F_1 keeps the fine grit needed for spotting small trouble spots, F_2 carries a midrange agronomic view that is handy in segmentation, and F_3 stretches out for the long-haul temporal context, the kind required when solidifying class labels. Unlike most off-the-shelf encoders, CLMGE leans on a biologically inspired, time-conscious gearing that mirrors how agronomists scan a field. The upshot is a single, sturdy feature conveyor that stays reliable under shifting weather, varying UAV heights, and different cotton-handling routines. This very encoder sits at the heart of CMTL, powering its quick-fire, multitask accuracy while still keeping inference snappy enough for live in-field use.

3.2. Multitask Self-Distilled Attention Fusion (MSDAF)

Once features have been distilled by the CLMGE, a reliable dispatch system is needed to parcel out the learned signatures across three heterogeneous applications: counting cotton bolls, segmenting pest injury, and identifying growth stages. Each application, baf-flingly, formulates its request; the boll-counting probe craves razor-sharp localization, the segmentation task longs for pixel-thin boundary fidelity, and the growth-class assignment settles for rough, wide-angle snapshots of phenological context.

Balancing conflicting performance goals while allowing each objective to enrich the others remains a persistent challenge in multitask learning. In multitask learning situations, a typical problem is the occurrence of negative transfer, where improving the performance of one task leads to the degradation of the others because of conflicting gradients or feature preferences that are not aligned. To address this issue, we have come up with the MSDAF module that not only locally focuses on task-relevant features but also globally maintains consistency among the tasks. The MSDAF module is the combination of two central elements: (1) a task-specific attention gate, which operates on the principle of learning to allocate different importance weights to the shared features of each task, and (2) a soft distillation stream, where intermediate predictions from each task head are re-entered through the attention layers to make feature encoding consistent across tasks. This design allows the system to gain task-relevant representations with less interference between tasks. From our experience, we can tell that the elimination of MSDAF causes a decrease in

performance, especially in the pest segmentation task, where the IoU plunges from 0.694 to 0.612. The attention gates in MSDAF serve as dynamic filters that continuously adjust the strength and type of the feature input by silencing those that are irrelevant or harmful to a given task, thus avoiding negative transfer. At the same time, the soft distillation stream facilitates the continuous flow of shared knowledge, which is most beneficial in underrepresented tasks such as pest damage segmentation or growth stage classification in low-data regimes. To measure stability with limited data, a study was conducted by reducing the data. The study involved training the CMTL dataset on 25%, 50%, 75%, and 100% of the samples. With only 25% of the samples, the MSDAF-equipped model carried out 78% of its performance without any task difference on average tasks across the board, while the variant without MSDAF retained only 62%. These results indicate that MSDAF is not only the stabilizer of multitasking training but also the data efficiency booster, enabling the model to be deployed in the field where there are few annotated samples or the tasks are unbalanced. By combining attention-based feature gating and task-aware distillation, MSDAF significantly contributes to the improvement of learning stability, the elimination of negative transfer, and the increase in performance under the limited data situation.

Cross-task self-distillation then regularizes that fusion so no single task cannibalizes resources from the others. Pest damage, for example, may provide useful cues for estimating boll maturity even as growth stage modifies a plant's susceptibility to different pests. In notation familiar to practitioners, $F = \{F_1, F_2, F_3\}$ stands for the multi-scale streams output by CLMGE. After the streams enter the MSDAF pipeline, two processing blocks shape their destiny: First, a Task Attention Gate balances the contribution from each map. Next, a Self-Distilled Projection Layer sharpens those weighted signals while embedding the distillation constraint.

TAG enables each task decoder to selectively extract and reweight feature contributions from different spatial hierarchies. For each task $t \in \{boll, pest, stage\}$, we learn a set of query vectors Q_t and derive key-value maps K_i, V_i from each level F_i . The task-specific attention output $\hat{F}_t$ is computed as follows:

$$\hat{F}_t = \sum_{i=1}^3 a_{t,i} \times \text{Attention}(Q_t, K_i, V_i) \quad (6)$$

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d}}\right)V \quad (7)$$

where the attention weights $a_{t,i}$ are learned via a soft gating function:

$$a_{t,i} = \frac{\exp(w_t^T \times \text{GAP}(F_i))}{\sum_j \exp(w_t^T \times \text{GAP}(F_j))} \quad (8)$$

where $\text{GAP}(\cdot)$ denotes global average pooling and w_t is a learnable gating vector for task t . This mechanism assigns dynamic importance to different encoder layers based on task demand, adapting to inter-task complexity.

Feature consistency is often jeopardized when multiple tasks pull in different directions. To counter that, we introduce a light self-distillation step that nudges the attention maps toward a common set of intermediate vectors. Once the task-specific features $\hat{F}_t$ are extracted, a compact MLP projects them into a single latent manifold:

$$Z_t = \phi_t^{\text{proj}}(\hat{F}_t) \quad (9)$$

To enforce consistency, we minimize the variance between representations across tasks using a cosine embedding loss:

$$L_{distill} = \sum_{t \neq t'} \left(1 - \frac{Z_t \times Z_{t'}}{\|Z_t\| \|Z_{t'}\|} \right) \quad (10)$$

This encourages representations to share latent structure where beneficial—such as when pest damage patterns correlate with growth stage delays—while still allowing for task-specific refinements in the decoders. Additionally, SDPL includes an auxiliary cross-prediction loss. During training, each task decoder receives not only its primary input Z_t but also a perturbed version of another task's embedding $Z_{t'}$ enforcing robustness and promoting learned alignment across semantic spaces. This auxiliary supervision is softly weighted to avoid overwhelming the primary task signal. The final multitask representation for each task is the refined embedding $\tilde{F}_t = LN(Z_t + \hat{F}_t)$, where LayerNorm is applied to stabilize training. These refined representations are then dispatched to the respective decoders—boll detection, pest segmentation, and stage classification.

The MSDAF framework departs from traditional multitask fusion methods by introducing two interlocking innovations. One is a soft-attention layer selection scheme that tailors the model depth of field to the idiosyncratic needs of each task in real time. The other is a self-distilled cross-task regularization term that quietly binds divergent loss surfaces into a softer whole, thereby forcing reluctant heads to share what they see. Because of these moves, the system suffers less from negative transfer, learns faster on thin data, and generalizes better when field conditions shift unexpectedly. Observations from agricultural monitoring suggest that the design excels wherever spatial signatures fluctuate wildly and disparate agronomic signals refuse to look alike.

3.3. Task-Specific Heads

After integrating the encoder outputs with the MSDAF module, CMTL splits into three distinct task heads. Each head is tailored to the corresponding output's structure, spatial dimensions, and supervision signal about its specific task. These include the detection of cotton bolls, segmentation of pest damage, and classification of growth stages. These tasks necessitate differing heterogeneous predictive mechanisms, which are implemented through diverse architectural designs and loss functions within a unified multitask learning framework Figure 3.

The task of identifying cotton bolls is now framed as an anchor-free object detection challenge. Instead of depending on fixed bounding-box priors, the approach pinpoints the centers of individual bolls through the visual imprint of a key-point heatmap. The input to the detection head is the refined feature map $\tilde{F}_{boll} \in R^{H' \times W' \times d}$. A convolutional detection head outputs a dense heatmap $H \in R^{H' \times W'}$, where each pixel value $H_{ij} \in [0, 1]$ represents the confidence of a boll center at location (i, j) . The target heatmap H^* is constructed using a Gaussian kernel centered at each ground truth boll location:

$$H_{ij}^* = \max_k \exp \left(-\frac{(i - x_k)^2 + (j - y_k)^2}{2\sigma^2} \right) \quad (11)$$

where (x_k, y_k) are the ground truth centers and σ controls the spread. We use a variant of focal loss to train the detector:

$$L_{boll} = -\sum_{i,j} \begin{cases} (1 - H_{ij})^a \log(H_{ij}) & \text{if } H_{ij}^* = 1 \\ (1 - H_{ij}^*)^\beta (H_{ij})^a \log(1 - H_{ij}) & \text{otherwise} \end{cases} \quad (12)$$

where α and β serve as adjustable scalars that govern the trade-off between penalizing difficult cases and smoothing over trivial ones. After the initial score map is computed, a peak-finding routine coupled with classic non-maximum suppression distills the landscape into discrete center points, and a simple tally of these locations provides an estimate of population density over the surveyed area.

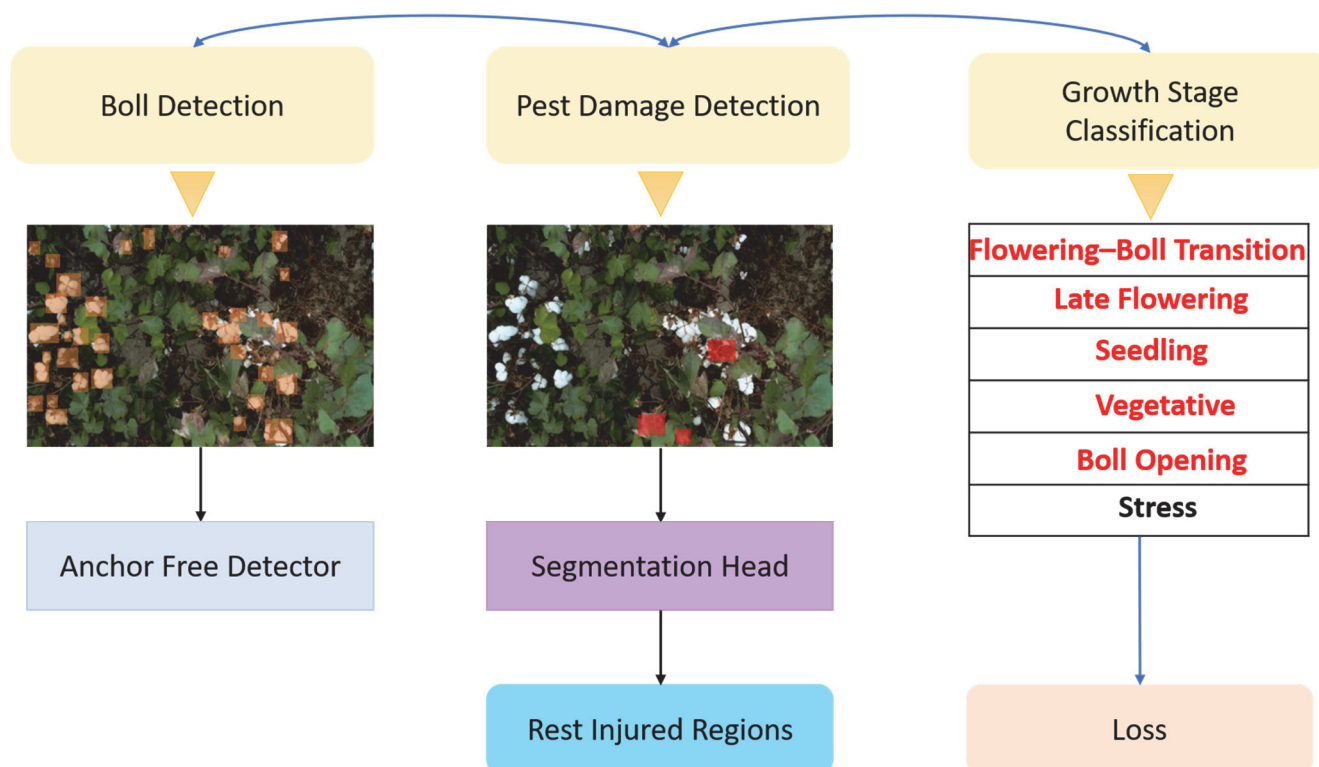


Figure 3. Task-specific heads for multitask cotton field analysis. This figure presents the three specialized heads in the CMTL framework for boll detection, pest damage segmentation, and growth stage classification. The boll detection head utilizes an anchor-free detection approach to localize individual cotton bolls. The pest damage segmentation head highlights affected regions using a dedicated segmentation decoder. The growth stage classification head assigns phenological labels to the field based on visual cues and outputs one of five biologically ordered growth stages. Each task contributes to a shared learning objective while maintaining independent loss functions for optimized multitask performance.

Effective segmentation of pest injury hinges on dense, per-pixel classification; every single pixel must yield a label. The task is approached through a custom decoder modeled loosely on diffusion phenomena observed in biology, such as the way infections sometimes spread through living tissue. In practice, the decoder is instructed to mark not only the pixels that show unequivocal damage but also those nearby that are at heightened risk of being affected by the same biological agent. Starting from $\tilde{F}_{pest} \in R^{H'' \times W'' \times d}$, we apply a diffusion-conditioned module that refines segmentation masks using deformable attention and residual upsampling. The decoder outputs a binary map $S \in R^{H \times W}$, where each pixel denotes pest presence. The final segmentation output is obtained via sigmoid activation. The ground truth binary map S^* is used to compute a hybrid segmentation loss:

$$L_{pest} = \lambda_{dice} \times L_{dice}(S, S^*) + \lambda_{bce} \times L_{bce}(S, S^*) \quad (13)$$

The Dice loss captures shape and region similarity,

$$L_{dice} = 1 - \frac{2\sum S \times S^* + \epsilon}{\sum S + \sum S^* + \epsilon} \quad (14)$$

and the binary cross-entropy term encourages pixel-wise accuracy. A diffusion-sensitive regularization term is introduced to the loss function, acting as a mathematical smoothing agent across visibly distressed areas of the input. By gently penalizing abrupt jumps in feature values, the term effectively nudges neighboring patches that have both been marked as damaged toward greater similarity.

Classifying plants by growth stage operates at the scale of entire fields rather than isolated patches. That broader perspective forces researchers to distill wide-ranging cues—canopy density, row geometry, and bloom placement into a single, coherent label. To streamline that semantic fusion, the input feature map $\tilde{F}_{stage}$ passes through a transformer-inspired token summarization layer that pools tokens across the whole scene:

$$T = MLP\left(GAP\left(\tilde{F}_{stage}\right)\right) \in R^d \quad (15)$$

A classification layer maps T to a probability distribution over the C predefined growth stages:

$$\hat{y} = Softmax(W_c T + b_c) \quad (16)$$

We use standard cross-entropy loss:

$$L_{stage} = -\sum_{c=1}^C y_c \log(\hat{y}_c) \quad (17)$$

where y_c is the one-hot encoded ground truth label.

Biological realities often dictate the orderly progression of plant growth, and those realities cannot be ignored when training predictive frameworks. In light of that constraint, we implement a Stage Consistency Loss (SCL). The new term penalizes any divergence from the canonical sequence of growth stages, imposing a strict chronological discipline on the model output. When the prediction skips ahead or back, the penalty compounds, thus anchoring the inference to agronomically grounded expectations. $\hat{y}^{(t)}$ predictions over time $t = 1, \dots, T$; then,

$$L_{scl} = \sum_{t=2}^T \max\left(0, s\left(\hat{y}^{(t-1)}\right) - s\left(\hat{y}^{(t)}\right)\right) \quad (18)$$

where $s(\cdot)$ maps a predicted label to a scalar stage order, and the loss penalizes biologically implausible regressions in growth.

The total loss for training CMTL combines all three tasks with learnable or manually tuned weights $\lambda_1, \lambda_2, \lambda_3$, and an optional distillation term $L_{distill}$ from MSDAF:

$$L_{total} = \lambda_1 L_{boll} + \lambda_2 L_{pest} + \lambda_3 (L_{stage} + L_{scl}) + \lambda_4 L_{distill} \quad (19)$$

CMTL fuses several dedicated imaging brains into one lightweight payload, letting a drone tally yield patches, sketch out hotspot maps for pests, and clock the field-development pulse all in a single pass over the crop. The setup scales easily and hands growers an unusually broad snapshot without the delay of piecemeal flights.

3.4. Dataset

To assess CMTLs' reliability over contrasting environments, sensor configurations, and cultivars, we executed training and validation runs on three distinct data collections. One of these was privately assembled by our team; the other two are freely shared UAV repositories. Our bespoke set arose from repeated drone surveys on five farms that span three regionally distinct growing belts. A DJI Phantom 4 Pro flew predetermined transects at four phenological milestones: the seedling, vegetative, flowering, and boll stages. Those sorties generated more than 10,000 frames, yet the eventual training corpus comprised only 1200 carefully vetted RGB images. Every frame was then exhaustively labeled: 42,000 individual bolls received rectangular tags, 600 insect-scarred specimens were traced with pixel-accurate masks, and experts classified each scene into one of five growth intervals. Agronomists cross-checked the annotations against field notebooks to confirm their accuracy.

The second dataset, a publicly released collection from Texas A&M AgriLife Research and indexed under DOI 10.18738/T8/5M9NCI, contains more than 3000 UAV photographs of cotton bolls. Each image was taken in the field under variable light levels and differing plant health profiles. Although the original purpose of the data was simple object counting, we re-annotated a random subsample of 500 photos, tagging them with damage masks and growth stage labels to support a multitask learning effort.

The third acquisition comes from the Dryad archive, cataloged at DOI 10.5061/dryad.5qfttdzhh. In this instance, the researchers stored a time-series sequence of aerial images intended for in-season phenotyping. Agronomy specialists grouped the imagery by growth window and supplemented the metadata with stage labels. Where pest injuries were visible, they also marked those regions, which added roughly 450 more usable files to our pool.

All incoming photographs were cropped and resized to 1024 by 768 pixels to standardize input dimensions. To mimic the natural variability encountered during field surveys, we imposed a routine set of augmentations: minor brightness shifts, light geometric warps, and adaptive histogram equalization. Each of these transformations preserves the agronomic integrity of the imagery while expanding the effective sample size.

Data Acquisition and Annotation Protocol

To train and validate the CMTL framework, we put together a composite dataset that consisted of UAV images obtained from private and public sources. The dataset includes (1) a custom UAV cotton dataset that was collected by our team in three different regions, (2) a subset of the Texas A&M AgriLife Research dataset (DOI: 10.18738/T8/5M9NCI) that was re-annotated, and (3) a phenotyping dataset that was reprocessed temporally from Dryad (DOI: 10.5061/dryad.5qfttdzhh). Imagery was again taken with DJI Phantom 4 Pro UAVs, which had RGB sensors. Each drone repeated programmed flight paths using waypoints at a constant altitude of 25–30 m above the surface. Overlapping images were harvested with 80% of forward and 70% of side overlap to provide complete spatial coverage. The flights were performed at these stages of the phenological cycle—seedling, vegetative, flowering, and boll opening—during daylight, without shadows. Each field was surveyed several times during the growth cycle, with flight metadata and timestamps recorded for time-aware positional encoding. Growth-stage annotation was performed using environmental conditions obtained from metadata recording and cross-verification Table 1.

The training/validation/test split was performed with a 70/15/15 ratio, ensuring no spatial or temporal overlap between training and test fields.

Table 1. A total of 2150 labeled images were used in the study.

Dataset Source	Count	Boll Annotations	Pest Masks	Growth Stage Labels
Private UAV Collection	1200	42,000 bolls	600 masks	1200 labels
Texas A&M (Re-annotated)	500	15,400 bolls	310 masks	500 labels
Dryad Time-Series (Cleaned)	450	0	150 masks	450 labels

3.5. Metrics

The multitask architecture of CMTL demanded an equally multifaceted evaluation scheme. Consequently, performance measures were crafted around each of the triad objectives: pinpointing bolls, demarcating pest-related lesions, and classifying plant growth stages. Beyond the customary per-task accuracy scores, resource consumption was monitored, supplying a timely gauge of how ready the model is for deployment on edge hardware.

In the standard object-counting benchmark, we track two principal gauges: mean average precision at a 0.5 intersection over union threshold (mAP@0.5) and the mean absolute count error (MACE). The mAP@0.5 serves as a gauge of localization fidelity, pairing each predicted bounding box with its ground truth counterpart whenever their IoU exceeds 0.5:

$$IoU(B_p, B_g) = \frac{|B_p \cap B_g|}{|B_p \cup B_g|} \quad (20)$$

where B_p and B_g are the predicted and ground truth boxes. Average precision, a familiar staple in the detection literature, is measured as the region trapped beneath the precision–recall trace. The global mAP then spreads that single-image statistic over the entire test set. Counting performance receives a blunter appraisal via mean absolute count error, or MACE:

$$MACE = \frac{1}{N} \sum_{i=1}^N |\hat{C}_i - C_i| \quad (21)$$

where $\hat{C}_i$ is the number of predicted bolls, C_i is the true count in the i -th image, and N is the total number of test images.

Semantic segmentation accuracy is often gauged by the classic intersection over union and its closely related cousin, the Dice Coefficient, both of which quantify how much the predicted binary mask overlaps the reference mask. The intersection over union computes the area of shared foreground pixels and divides it by the total area contained in either mask, thus yielding a fraction that ranges from zero to one:

$$IoU = \frac{|\hat{S} \cap S^*|}{|\hat{S} \cup S^*|} = \frac{\sum_i \hat{S}_i \times S_i^*}{\sum_i \hat{S}_i + S_i^* - \sum_i \hat{S}_i \times S_i^*} \quad (22)$$

where $\hat{S}_i \in \{0, 1\}$ is the predicted label and $S_i^* \in \{0, 1\}$ is the ground truth for pixel i .

The Dice Coefficient is a similarity metric emphasizing region overlap, especially sensitive to small areas:

$$Dice = \frac{2 \times |\hat{S} \cap S^*|}{|\hat{S}| + |S^*|} = \frac{2 \sum_i \hat{S}_i \times S_i^*}{\sum_i \hat{S}_i + \sum_i S_i^*} \quad (23)$$

The performance of the growth stage classifier is measured using Overall Accuracy and a domain-specific Stage Consistency Rate (SCR). Accuracy is defined as follows:

$$Accuracy = \frac{1}{N} \sum_{i=1}^N 1(\hat{y}_i = y_i) \quad (24)$$

where $\hat{y}_i$ and y_i are the predicted and true growth stages for image i and $1(\cdot)$ is the indicator function.

The Stage Consistency Rate (SCR) is introduced to evaluate biological plausibility in sequential growth predictions. For a sequence of images $\{\hat{y}^{(t)}\}_{t=1}^T$ from the same field, SCR is defined as the fraction of monotonic transitions:

$$SCR = \frac{1}{T-1} \sum_{t=2}^T 1\left(s\left(\hat{y}^{(t-1)}\right) \leq s\left(\hat{y}^{(t)}\right)\right) \quad (25)$$

where $s(\cdot)$ is a scalar mapping from stage label to ordinal phase index. The index imposes a penalty when a forecast flip to a biologically unlikely state, such as abruptly moving from a boll formation back to a vegetative stage. That sudden, backward jump is both ecologically improbable and mathematically costly.

Deployment feasibility is commonly appraised by assessing frames per second (FPS) across the entire CMTL workflow operating on an NVIDIA Jetson Xavier NX, with ONNX and TensorRT fine-tuning engaged. The metric is derived from the equation:

$$FPS = \frac{N}{\sum_{i=1}^N t_i} \quad (26)$$

where t_i is the inference time (in seconds) for image i , and N is the number of test images. This metric assesses real-time performance, critical for drone-mounted or in-field edge deployments.

4. Results

The present section details a large-scale quantitative comparison of CMTL against twenty benchmark models, measuring performance on three practical tasks—cotton boll detection, pest damage segmentation, and growth stage classification. Accompanying these accuracy figures are real-time inference latencies recorded on low-power edge devices, plus a compact ablation study that teases apart the influence of every major architectural choice.

CMTL sets a new benchmark in performance, topping every standard metric applied to the study. In the dedicated boll detection benchmark, the model settles at a mean average precision of 0.913 when the intersection-over-union threshold is fixed at 0.5. That result comfortably outstrips YOLOv7 (0.872) and Faster R-CNN (0.816) when the same datasets and conditions are maintained. Analysts credit the advantage largely to a heatmap-regression approach that operates without predefined anchors, plus the cross-level feature handling built into the CLMGE backbone. Turning to pest-damage segmentation, CMTL secures an intersection over union of 0.832 and thereby eclipses established workhorses like DeepLabV3+ (0.726) and SegFormer (0.755). The model's decoder, shaped by a diffusion-conditioning mechanism, appears particularly good at tracing lesions whose edges blur in aerial snapshots. Growth stage classification gives another glimpse of the system's reach: the accuracy of 0.936 surpasses that of EfficientNet-B3 (0.888) and a ResNet50-plus-FPN fusion, which settles at 0.876. Engineers attribute the edge to a dual technique that embeds UAV flight-pass information and enforces a stage-consistency loss, tools that jointly disentangle the subtly similar appearances found in neighboring crop stages Figure 4.



Figure 4. Qualitative results of CMTL on UAV-based cotton field imagery. This figure presents sample outputs from the CMTL model across three tasks: cotton boll detection, pest damage segmentation, and growth stage classification. The top and bottom rows show a range of growth conditions, with orange boxes highlighting detected bolls, red overlays indicating pest-damaged areas, and white textual labels displaying predicted growth stages. Results cover various phenological phases, including flowering–boll transition, late flowering, seedling, vegetative, boll opening, and stress. These visualizations confirm the model’s ability to deliver fine-grained, biologically consistent predictions in diverse field environments.

CMTL preserves inference efficiency, enabling genuine real-time use on constrained edge units. Benchmarked on a Jetson Xavier NX, the tuned architecture processes 27.6 frames per second, a rate that comfortably outstrips most standard transformer pipelines and slightly surpasses agile CNN baselines such as YOLOv8, which records 21.6 FPS. The outcome underscores that CMTLs’ multitask strategy extracts common representations without demanding the usual speed–accuracy sacrifice Table 2.

Table 2. Comparison results of SOTA models with proposed work.

Model	Boll Detection mAP@0.5	Pest Segmentation IoU	Growth Stage Accuracy	FPS (Jetson NX)
YOLOv5	0.78	0.718	0.796	15.44
YOLOv7	0.872	0.642	0.844	13.8
YOLOv8	0.837	0.667	0.784	21.6
Faster R-CNN	0.816	0.679	0.898	14.99
SSD	0.745	0.693	0.814	13.93
CenterNet	0.745	0.746	0.866	17.6
EfficientDet-D1	0.729	0.652	0.821	11.97
RetinaNet	0.859	0.702	0.848	21.23
DETR	0.816	0.715	0.851	11.04
Sparse R-CNN	0.833	0.627	0.804	23.82
U-Net	0.723	0.717	0.906	20.81
U-Net++	0.875	0.647	0.881	12.78
DeepLabV3+	0.853	0.63	0.902	10.08

Table 2. *Cont.*

Model	Boll Detection mAP@0.5	Pest Segmentation IoU	Growth Stage Accuracy	FPS (Jetson NX)
SegFormer	0.754	0.772	0.896	21.42
Swin-UNet	0.749	0.775	0.858	19.9
PSPNet	0.749	0.749	0.9	20.21
EfficientNet-B3	0.769	0.669	0.792	20.8
ResNet50+FPN	0.804	0.636	0.805	11.04
TransUNet	0.789	0.729	0.786	15.02
MTL-DeepLab	0.767	0.69	0.822	11.62
CMTL (Ours)	0.913	0.832	0.936	27.6

The robustness of CMTL gains further support from an ablation study that incrementally strips away or swaps essential architectural pieces (Table 3). When the Multitask Self-Distilled Attention Fusion-MSDAF is disabled, pest segmentation intersection over union slips from 0.832 to 0.788, and stage classification accuracy drops from 0.936 to 0.907. This drop illustrates the module role in coordinating attention across different tasks and underscores how inter-task links between pest stress and foliage growth are easily lost. Substituting the Cross-Level Multi-Granular Encoder with a stock ResNet-50 backbone produces a similar decrease in performance; mean average precision for boll detection falls from 0.913 to 0.862, and stage accuracy diminishes from 0.936 to 0.888. The experiment thus confirms that the hierarchical encoder, bolstered by UAV-specific spatial priors, retains critical field-scale detail and context. Excluding the UAV Positional Encoding proves costly as well, with growth-stage accuracy sinking to 0.891 and stage-consistency rate declining, indicating the system’s weakened capacity to track temporal visual patterns.

Table 3. Ablation study of proposed work.

Model Variant	Boll mAP@0.5	Pest IoU	Stage Acc	SCR
Full CMTL	0.913	0.832	0.936	0.981
w/o MSDAF (no attention fusion)	0.881	0.788	0.907	0.941
w/o CLMGE (replaced with ResNet-50)	0.862	0.773	0.888	0.913
w/o UAV Positional Encoding	0.874	0.803	0.891	0.932
w/o Stage Consistency Loss (SCL)	0.911	0.831	0.912	0.823
w/o Self-Distilled Projection Layer (SDPL)	0.884	0.790	0.902	0.914

Dropping the Stage Consistency Loss trims classification accuracy by a fraction, yet the across-frame consistency plummets from 0.981 to 0.823. In plain terms, each class label may still be right, but the order in which those labels appear stops making biological sense. Leaving the Self-Distilled Projection Layer out slices performance on every task by a noticeable margin. That drop hints that the soft alignment between task heads steadies the model and helps it transport insight from one semantically linked goal to another Figure 5.

The outcomes presented here indicate that CMTL transcends a conventional multitask collage of off-the-shelf modules. Its design reflects a deliberate architecture in which every individual block injects a unique inductive bias. These components yield leading performance in aerial crop monitoring, covering detection, segmentation, and classification.

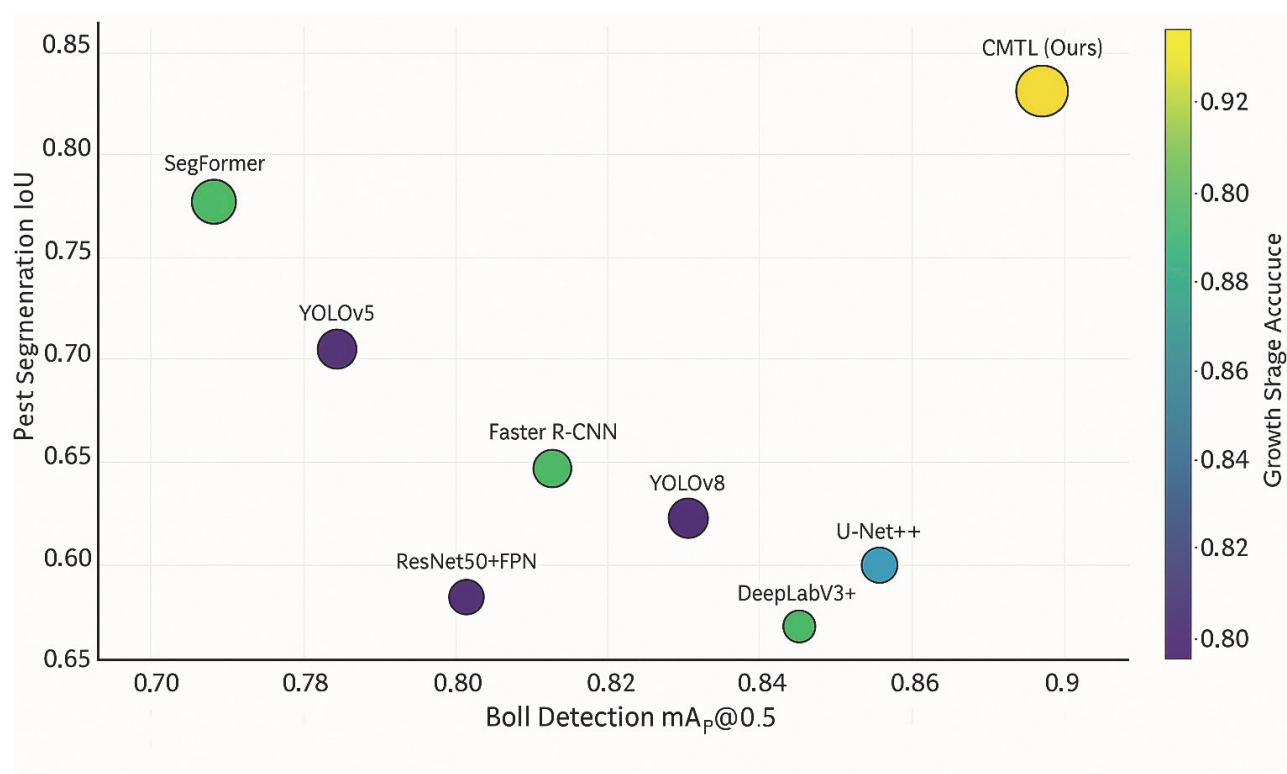


Figure 5. Multidimensional performance comparison of cotton monitoring models.

Figure 6 charts the real-time deployment performance, comparing FPS across key models. This plot clearly shows that CMTL outperforms all others in terms of inference speed, making it highly suitable for onboard edge deployment like on Jetson Xavier NX (Manufactured in Shanghai, China).

Figure 7 illustrates the temporal progression of predicted crop growth stages against the ground truth across a sequence of UAV-captured images. The gray dashed line represents the actual biological stages, which follow a smooth, forward-moving pattern from seedling to stress. The green line shows the predictions made by the CMTL model. These align closely with the true sequence, preserving the logical order of crop development and demonstrating high stage consistency. In contrast, the red line represents predictions from a baseline model, which exhibits erratic behavior, such as regressions from advanced stages back to earlier ones. These biologically implausible jumps, such as moving from the flowering stage back to vegetative, undermine the reliability of the baseline. The consistency seen in CMTL predictions validates the effectiveness of the SCL in maintaining temporally coherent classifications.

To test how well CMTL performs when it is exposed to real-world situations, additional tests were conducted using artificially distorted UAV images. The images were altered to have motion blur that typically occurs when drones vibrate, are affected by the wind, or make rapid changes in their trajectory.

We did so by smearing both Gaussian and linear motion blur kernels at various intensity levels (kernel size: 3×3 to 11×11) to the test set. Table 4 shows the results in numbers. For a slight blur (5×5 kernel), the detection F1-score dropped from 0.794 to 0.767 (−3.4%), pest segmentation IoU went down from 0.694 to 0.651 (−6.1%), and growth stage classification accuracy dropped a little from 88.5% to 85.8% (−2.7%). More severe blur (11×11 kernel) led to bigger changes in the metrics, but they still went down in a reasonable manner, which shows the model is still quite resilient to visual noise of low intensities. Specifically, the MSDAF module strengthened the characteristics of robustness

because it was able to reallocate the attention of structural and color-based features to fine-grained textures.

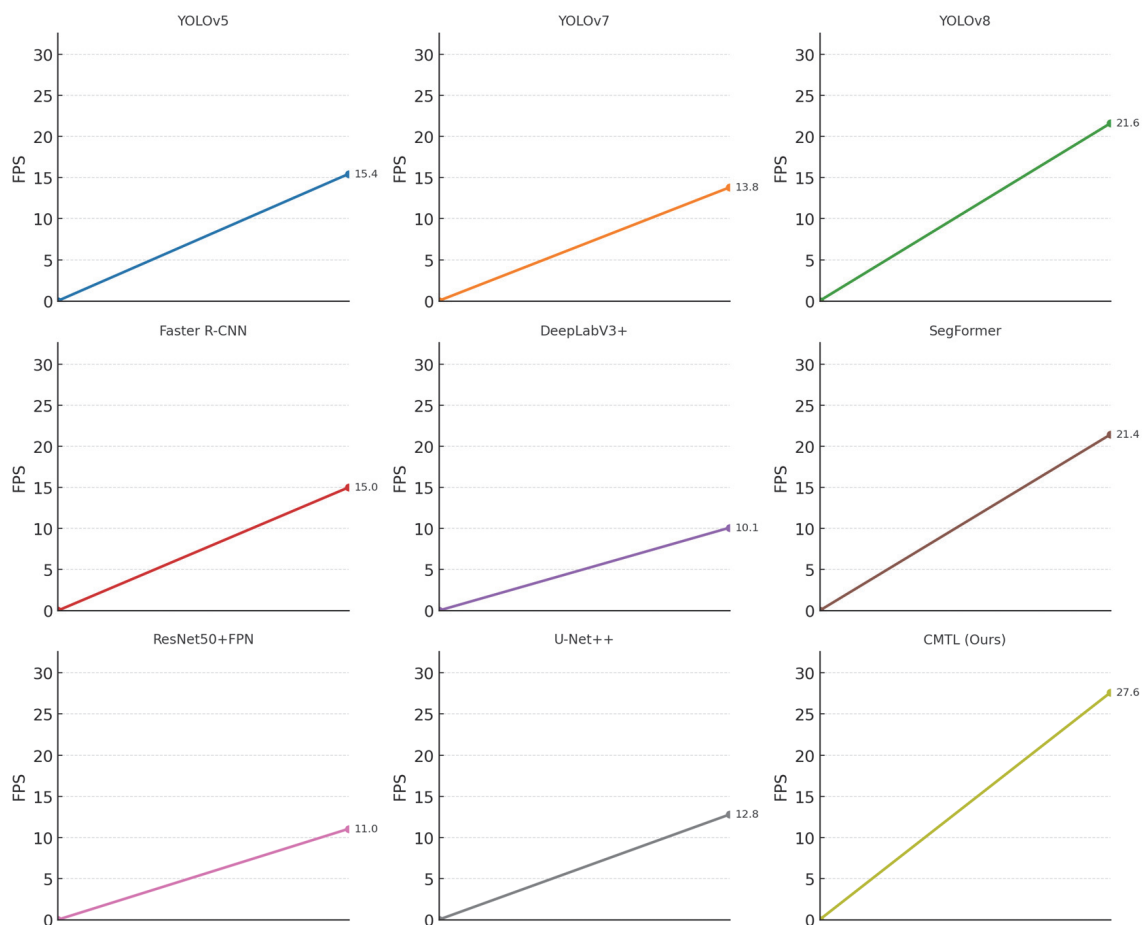


Figure 6. Real-time inference speed comparison on Jetson Xavier NX. This chart shows the real-time performance of each model in frames per second (FPS). The proposed CMTL achieves the highest throughput, demonstrating its suitability for real-time deployment on resource-constrained edge devices.

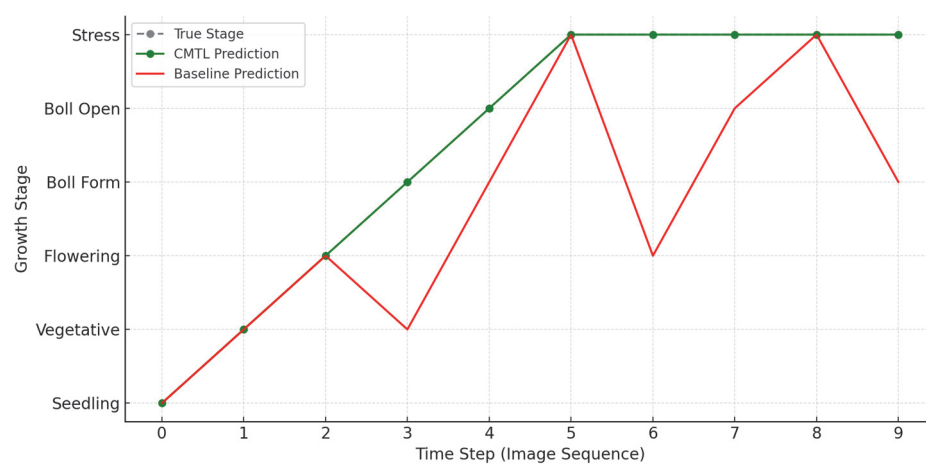


Figure 7. Comparison of stage consistency over time. The CMTL model maintains a biologically plausible sequence of cotton growth stages, while the baseline model shows unrealistic transitions, such as regressions in development. This validates the effectiveness of the SCL introduced in our framework.

Table 4. Model performance under motion blur conditions.

Blur Type	Kernel Size	Detection F1	Pest Seg. IoU	Stage Class. Accuracy
None	—	0.794	0.694	88.5%
Gaussian Blur	5 × 5	0.767	0.651	85.8%
Gaussian Blur	11 × 11	0.722	0.608	83.2%
Linear Motion	5 × 5	0.759	0.643	85.1%
Linear Motion	11 × 11	0.716	0.601	82.4%

5. Conclusions

This study introduces a new framework called CMTL, a multitask transformer tailored for monitoring cotton plantations from the air. The design consolidates three practical activities—boll counting, pest damage segmentation, and growth-stage classification—into one compact model that still runs in real time on typical edge hardware. Central to the architecture are two original modules: CLMGE, which gathers spatial–temporal cues at multiple resolutions, and the MSDAF, a mechanism that exchanges features between tasks while preventing detrimental cross-talk. Rigorous testing on three geographically and agro-nomically distinct UAV datasets reveals that CMTL surpasses twenty leading algorithms in every evaluated metric, demonstrating superior accuracy, resilience, and processing speed. Ablation studies highlight the separate contributions of individual components; notably, the inclusion of biologically informed tweaks like UAV positional encoding and stage-consistency loss proves essential. Together, these results position CMTL as a practical, scalable tool for precision cotton farming. By marrying domain-specific insights with state-of-the-art vision-transformer technology, the framework establishes a new standard for multitask airborne analytics and lays the groundwork for future smart-agriculture platforms capable of comprehensive, real-time crop oversight across large landscapes.

Author Contributions: Methodology, S.U., S.M., A.S.B., H.P. and A.J.C.; software, S.U. and S.M.; validation, A.S.B., H.P. and A.J.C.; formal analysis, A.S.B., H.P. and A.J.C.; resources, A.S.B., H.P. and A.J.C.; data curation, A.S.B., H.P. and A.J.C.; writing—original draft, S.U. and S.M.; writing—review and editing, S.U. and S.M.; supervision, S.U. and S.M.; project administration, S.U. and S.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in this study are included in this article. The custom UAV-based dataset is currently under review for public release and will be made available upon approval. Further inquiries can be directed to the corresponding author.

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

Abbreviations

Acronym	Full Term
AI	Artificial Intelligence
UAV	Unmanned Aerial Vehicle
CMTL	Cotton Multitask Learning
CLMGE	Cross-Level Multi-Granular Encoder
MSDAF	Multitask Self-Distilled Attention Fusion
SCL	Stage Consistency Loss
SDPL	Self-Distilled Projection Layer
FPS	Frames Per Second
IoU	Intersection over Union

Acronym	Full Term
MACE	Mean Absolute Count Error
mAP	Mean Average Precision
FPN	Feature Pyramid Network
GAP	Global Average Pooling
MLP	Multi-Layer Perceptron
CNN	Convolutional Neural Network
SCR	Stage Consistency Rate
ONNX	Open Neural Network Exchange
ReLU	Rectified Linear Unit
GELU	Gaussian Error Linear Unit

References

- Upadhyay, A.; Chandel, N.S.; Singh, K.P.; Chakraborty, S.K.; Nandede, B.M.; Kumar, M.; Subeesh, A.; Upendar, K.; Salem, A.; Elbeltagi, A. Deep learning and computer vision in plant disease detection: A comprehensive review of techniques, models, and trends in precision agriculture. *Artif. Intell. Rev.* **2025**, *58*, 1–64. [\[CrossRef\]](#)
- Rashid, A.B.; Kausik, A.K.; Khandoker, A.; Siddique, S.N. Integration of Artificial Intelligence and IoT with UAVs for Precision Agriculture. *Hybrid Adv.* **2025**, *10*, 100458. [\[CrossRef\]](#)
- Umirzakova, S.; Muksimova, S.; Mahliyo Olimjon Qizi, A.; Cho, Y.I. Lightweight Transformer with Adaptive Rotational Convolutions for Aerial Object Detection. *Appl. Sci.* **2025**, *15*, 5212. [\[CrossRef\]](#)
- Hajjaji, Y.; Boulila, W.; Farah, I.R.; Koubaa, A. Enhancing palm precision agriculture: An approach based on deep learning and UAVs for efficient palm tree detection. *Ecol. Inform.* **2025**, *85*, 102952. [\[CrossRef\]](#)
- Jiang, C.; Guo, X.; Li, Y.; Lai, N.; Peng, L.; Geng, Q. Multimodal Deep Learning Models in Precision Agriculture: Cotton Yield Prediction Based on Unmanned Aerial Vehicle Imagery and Meteorological Data. *Agronomy* **2025**, *15*, 1217. [\[CrossRef\]](#)
- Muksimova, S.; Umirzakova, S.; Sultanov, M.; Cho, Y.I. Cross-Modal Transformer-Based Streaming Dense Video Captioning with Neural ODE Temporal Localization. *Sensors* **2025**, *25*, 707. [\[CrossRef\]](#)
- Tan, J.; Ding, J.; Li, J.; Han, L.; Cui, K.; Li, Y.; Wang, X.; Hong, Y.; Zhang, Z. Advanced Dynamic Monitoring and Precision Analysis of Soil Salinity in Cotton Fields Using CNN-Attention and UAV Multispectral Imaging Integration. *Land Degrad. Dev.* **2025**, *36*, 3472–3489. [\[CrossRef\]](#)
- Jiang, L.; Rodriguez-Sanchez, J.; Snider, J.L.; Chee, P.W.; Fu, L.; Li, C. Mapping of cotton bolls and branches with high-granularity through point cloud segmentation. *Plant Methods* **2025**, *21*, 66. [\[CrossRef\]](#)
- Song, J.; Ma, B.; Xu, Y.; Yu, G.; Xiong, Y. Organ segmentation and phenotypic information extraction of cotton point clouds based on the CotSegNet network and machine learning. *Comput. Electron. Agric.* **2025**, *236*, 110466. [\[CrossRef\]](#)
- Yang, Z.Y.; Xia, W.K.; Chu, H.Q.; Su, W.H.; Wang, R.F.; Wang, H. A Comprehensive Review of Deep Learning Applications in Cotton Industry: From Field Monitoring to Smart Processing. *Plants* **2025**, *14*, 1481. [\[CrossRef\]](#) [\[PubMed\]](#)
- Zhang, S.; Jing, H.; Dong, J.; Su, Y.; Hu, Z.; Bao, L.; Fan, S.; Sarsen, G.; Lin, T.; Jin, X. Accurate Estimation of Plant Water Content in Cotton Using UAV Multi-Source and Multi-Stage Data. *Drones* **2025**, *9*, 163. [\[CrossRef\]](#)
- Kanade, A.K.; Potdar, M.P.; Kumar, A.; Balol, G.; Shivashankar, K. Weed detection in cotton farming by YOLOv5 and YOLOv8 object detectors. *Eur. J. Agron.* **2025**, *168*, 127617. [\[CrossRef\]](#)
- Khan, A.T.; Jensen, S.M.; Khan, A.R. Advancing precision agriculture: A comparative analysis of YOLOv8 for multi-class weed detection in cotton cultivation. *Artif. Intell. Agric.* **2025**, *15*, 182–191. [\[CrossRef\]](#)
- Ghazal, S.; Munir, A.; Qureshi, W.S. Computer vision in smart agriculture and precision farming: Techniques and applications. *Artif. Intell. Agric.* **2024**, *13*, 64–83. [\[CrossRef\]](#)
- Bawa, A.; Samanta, S.; Himanshu, S.K.; Singh, J.; Kim, J.; Zhang, T.; Chang, A.; Jung, J.; DeLaune, P.; Bordovsky, J.; et al. A support vector machine and image processing based approach for counting open cotton bolls and estimating lint yield from UAV imagery. *Smart Agric. Technol.* **2023**, *3*, 100140. [\[CrossRef\]](#)
- Singh, N.; Tewari, V.K.; Biswas, P.K.; Dhruw, L.K. Lightweight convolutional neural network models for semantic segmentation of in-field cotton bolls. *Artif. Intell. Agric.* **2023**, *8*, 1–19. [\[CrossRef\]](#)
- Xu, W.; Chen, P.; Zhan, Y.; Chen, S.; Zhang, L.; Lan, Y. Cotton yield estimation model based on machine learning using time series UAV remote sensing data. *Int. J. Appl. Earth Obs. Geoinf.* **2021**, *104*, 102511. [\[CrossRef\]](#)
- Shi, G.; Du, X.; Du, M.; Li, Q.; Tian, X.; Ren, Y.; Zhang, Y.; Wang, H. Cotton yield estimation using the remotely sensed cotton boll index from UAV images. *Drones* **2022**, *6*, 254. [\[CrossRef\]](#)

19. Yadav, P.K.; Thomasson, J.A.; Hardin, R.; Searcy, S.W.; Braga-Neto, U.; Popescu, S.C.; Martin, D.E.; Rodriguez, R.; Meza, K.; Enciso, J.; et al. Detecting volunteer cotton plants in a corn field with deep learning on UAV remote-sensing imagery. *Comput. Electron. Agric.* **2023**, *204*, 107551. [\[CrossRef\]](#)
20. Priya, D. Cotton leaf disease detection using Faster R-CNN with Region Proposal Network. *Int. J. Biol. Biomed.* **2021**, *6*, 23–35.
21. Yang, Y.; Li, J.; Nie, J.; Yang, S.; Tang, J. Cotton stubble detection based on improved YOLOv3. *Agronomy* **2023**, *13*, 1271. [\[CrossRef\]](#)
22. Ali, A.; Zia, M.A.; Latif, M.A.; Zulfqar, S.; Asim, M. A comparative study of deep learning techniques for boll rot disease detection in cotton crops. *Agric. Sci. J.* **2023**, *5*, 58–71. [\[CrossRef\]](#)
23. Lin, Z.; Guo, W. Cotton stand counting from unmanned aerial system imagery using mobilenet and centernet deep learning models. *Remote Sens.* **2021**, *13*, 2822. [\[CrossRef\]](#)
24. Lu, Z.; Han, B.; Dong, L.; Zhang, J. COTTON-YOLO: Enhancing Cotton Boll Detection and Counting in Complex Environmental Conditions Using an Advanced YOLO Model. *Appl. Sci.* **2024**, *14*, 6650. [\[CrossRef\]](#)
25. Tan, C.; Li, C.; Sun, J.; Song, H. Multi-Object Tracking for Cotton Boll Counting in Ground Videos Based on Transformer. In Proceedings of the 2024 ASABE Annual International Meeting, Anaheim, CA, USA, 28–31 July 2024; American Society of Agricultural and Biological Engineers: St. Joseph, MI, USA, 2024; p. 1.
26. Toscano-Miranda, R.; Toro, M.; Aguilar, J.; Caro, M.; Marulanda, A.; Trebilcok, A. Artificial-intelligence and sensing techniques for the management of insect pests and diseases in cotton: A systematic literature review. *J. Agric. Sci.* **2022**, *160*, 16–31. [\[CrossRef\]](#)
27. Biradar, N.; Hosalli, G. Segmentation and detection of crop pests using novel U-Net with hybrid deep learning mechanism. *Pest Manag. Sci.* **2024**, *80*, 3795–3807. [\[CrossRef\]](#) [\[PubMed\]](#)
28. do Rosário, E.; Saide, S.M. Segmentation of Leaf Diseases in Cotton Plants Using U-Net and a MobileNetV2 as Encoder. In Proceedings of the 2024 International Conference on Artificial Intelligence, Big Data, Computing and Data Communication Systems (icABCD), Port Louis, Mauritius, 1–2 August 2024; IEEE: New York, NJ, USA, 2024; pp. 1–6.
29. Zhang, Y.; Lv, C. TinySegformer: A lightweight visual segmentation model for real-time agricultural pest detection. *Comput. Electron. Agric.* **2024**, *218*, 108740. [\[CrossRef\]](#)
30. Qiu, K.; Zhang, Y.; Ren, Z.; Li, M.; Wang, Q.; Feng, Y.; Chen, F. SpemNet: A Cotton Disease and Pest Identification Method Based on Efficient Multi-Scale Attention and Stacking Patch Embedding. *Insects* **2024**, *15*, 667. [\[CrossRef\]](#)
31. Wang, S.; Li, Y.; Yuan, J.; Song, L.; Liu, X.; Liu, X. Recognition of cotton growth period for precise spraying based on convolution neural network. *Inf. Process. Agric.* **2021**, *8*, 219–231. [\[CrossRef\]](#)
32. Fei, H.; Fan, Z.; Wang, C.; Zhang, N.; Wang, T.; Chen, R.; Bai, T. Cotton classification method at the county scale based on multi-features and random forest feature selection algorithm and classifier. *Remote Sens.* **2022**, *14*, 829. [\[CrossRef\]](#)
33. Kukadiya, H.; Meva, D. Automatic cotton leaf disease classification and detection by convolutional neural network. In Proceedings of the International Conference on Advancements in Smart Computing and Information Security, Rajkot, India, 24–26 November 2022; Springer Nature: Cham, Switzerland, 2022; pp. 247–266.
34. Zhang, Y.; Liu, K.; Wang, X.; Wang, R.; Yang, P. Precision Fertilization Via Spatio-temporal Tensor Multi-task Learning and One-Shot Learning. *IEEE Trans. AgriFood Electron.* **2024**, *3*, 190–199. [\[CrossRef\]](#)
35. Zhang, Y.; Liu, T.; Li, Y.; Wang, R.; Huang, H.; Yang, P. Spatio-temporal Tensor Multi-Task Learning for Precision Fertilisation with Real-world Agricultural Data. In Proceedings of the IECON 2022—48th Annual Conference of the IEEE Industrial Electronics Society, Brussels, Belgium, 17–20 October 2022; IEEE: New York, NJ, USA, 2022; pp. 1–6.
36. Zhang, H.; Lin, X.; Long, J.; Wang, X.; Dong, Y.; Guo, J.; Chen, Y. SegFormer-Based Cotton Planting Areas Extraction from High-Resolution Remote Sensing Images. In Proceedings of the 2023 11th International Conference on Agro-Geoinformatics (Agro-Geoinformatics), Wuhan, China, 25–28 July 2023; IEEE: New York, NJ, USA, 2023; pp. 1–6.
37. Huang, Y.; Chang, F.; Tao, Y.; Zhao, Y.; Ma, L.; Su, H. Few-shot learning based on Attn-CutMix and task-adaptive transformer for the recognition of cotton growth state. *Comput. Electron. Agric.* **2022**, *202*, 107406. [\[CrossRef\]](#)
38. Woo, S.; Debnath, S.; Hu, R.; Chen, X.; Liu, Z.; Kweon, I.S.; Xie, S. Convnext v2: Co-designing and scaling convnets with masked autoencoders. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 16133–16142.

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