

CLASS-IMBALANCE-AWARE DEEP SEMANTIC SEGMENTATION FOR *VARROA DESTRUCTOR* DETECTION UNDER 780-NM NARROW-SPECTRUM ILLUMINATION

Asad Khan^{*1}, Izaz Ullah², Inzamam Ul Haq³, Imadullah⁴, Wasim Jan⁵, Abdul Rashid⁶

^{*1,4,5,6}Government Collage of Management Sciences, Charsadda affiliated with Bacha Khan University.

^{2,3}Government Postgraduate College Charsadda affiliated with Bacha Khan University.

DOI: <http://doi.org/10.5281/zenodo.22187750>

Keywords

Varroa destructor; honey bee; 780 nm; narrow-spectrum illumination; near-infrared imaging; semantic segmentation; U-Net; U-Net++; Attention U-Net; DeepLabV3+; precision apiculture.

Article History

Received: 01 January 2026

Accepted: 16 March 2026

Published: 31 March 2026

Copyright @Author

Corresponding Author: *

Asad Khan

Abstract

Varroa destructor remains a major challenge for managed honey bee colonies, and an imaging system that can locate mites directly on bees could support continuous, non-destructive monitoring. The study 'Towards Varroa destructor mite detection using a narrow spectra illumination' established a useful direction by combining controlled illumination with U-Net segmentation and by identifying 780-nm infrared images as a promising condition for mite analysis. The present work builds on that foundation, but changes the research question from whether 780-nm imaging can support segmentation to how the segmentation pipeline can be made more sensitive to the small and highly imbalanced mite class. A reproducible experiment was assembled from the supplied Colab implementation using 647 image-mask pairs, a fixed 80/10/10 split (517 training, 65 validation, and 65 test images), and four semantic-segmentation architectures: U-Net, Attention U-Net, U-Net++, and DeepLabV3+. All models were trained for 30 epochs with Adam at a learning rate of 1e-4, batch size 1, and a 0.5 image scale. The final objective combined weighted CrossEntropy loss with a Varroa-specific Dice loss, using class weights of 0.25, 0.75, and 3.0 for background, bee, and mite, respectively. Checkpoints were selected by validation mite F1-score rather than by overall loss alone. On the held-out test set, U-Net++ achieved the best pixel-level segmentation among the current architectures, with 85.91% precision, 86.62% recall, 86.26% F1-score, and 75.84% IoU. Attention U-Net achieved 85.70% F1 and 74.97% IoU, while the conventional U-Net control reached 84.09% F1 and 72.55% IoU. DeepLabV3+ was much smaller and faster, requiring 3.77 million parameters and 10.25 ms per forward pass, but its mite recall fell to 32.14%. For historical context, the U-Net result reported in the base study corresponds to approximately 90.14% precision, 76.30% recall, 82.65% F1, and 70.42% IoU when calculated from its published confusion counts. The experiments therefore support class-imbalance-aware optimization as a practical way to move the system toward a more balanced sensitivity regime and identify U-Net++ as the strongest pixel-level architecture in the present benchmark. The findings are intentionally presented as an extension of the base study rather than as a controlled replication, because the dataset split, three-class formulation, loss function, checkpoint criterion, and annotation reconstruction pipeline differ.

1. INTRODUCTION

Healthy honey bee colonies are central to agricultural pollination and ecosystem stability, yet their productivity is affected by a combination of pathogens, parasites, environmental stress, and management practices. Among these pressures, the ectoparasitic mite *Varroa destructor* is particularly important because mites are associated with weakened hosts and broader colony-level health problems. In practical beekeeping, infestation assessment is often carried out through periodic inspection or indirect counting procedures. Those methods can provide useful information, but they do not naturally provide continuous, automated observation of bees moving through a hive entrance. A computer-vision system that can recognize a mite while a bee is passing through a controlled imaging region could therefore add a valuable layer of monitoring.

The visual problem, however, is not straightforward. A *Varroa* mite is small in relation to the bee body, and the available evidence may be limited by occlusion, body texture, orientation, blur, and illumination. When the target occupies only a small fraction of an image, an ordinary pixel-wise objective can be dominated by background and bee pixels. A model may consequently achieve a low average loss while still missing a substantial portion of the mite. This is particularly undesirable for an infestation-monitoring application, where a false negative can be more consequential than a small number of false-positive pixels.

Several research groups have approached the problem through different sensing strategies. Multispectral entrance-monitoring systems have shown that controlled illumination can improve the visibility of biological features and can be combined with deep learning for colony-level *Varroa* assessment. Bjerger et al., for example, investigated a computer-vision system installed at the hive entrance and reported a mite-detection F1-score of 0.91 in live-bee video monitoring [5]. Other studies have explored hyperspectral imaging to exploit spectral differences between mites and bees. Duma et al. reported that a limited number of informative wavelengths could

be sufficient for parasite discrimination, suggesting that a practical custom-band system may sometimes be preferable to a full hyperspectral sensor [3]. Ghezal et al. also demonstrated the usefulness of hyperspectral information for segmentation and counting in hive-board imagery [4].

The base study for the present manuscript, 'Towards *Varroa destructor* mite detection using a narrow spectra illumination,' is especially relevant because it takes a hardware-light route to the problem. The authors developed the Fčielka-Thor 3000 entrance-monitoring concept using three illumination conditions: approximately 500 nm turquoise light, 780 nm infrared light, and cold white light. A Raspberry Pi HQ camera with the infrared filter removed was used to acquire the corresponding images. The study found that the mite was not sufficiently visible under the turquoise condition and therefore concentrated its U-Net segmentation experiment on 780-nm images [1].

The base study also provides an important diagnostic observation: false negatives were a major limitation of its U-Net segmentation. Its reported confusion counts contained 1,893 true-positive mite pixels, 588 false-negative pixels, and 207 false-positive pixels. These values correspond to approximately 90.14% precision and 76.30% recall. The published work later considered a 20-pixel filtering condition, under which the reported mite recall decreased to 55%, reinforcing the point that missed mite detections remained the dominant weakness [1].

This observation motivates the present study. Rather than introducing another illumination device, we ask whether the learning pipeline can be made more sensitive to the sparse mite class while staying within the 780-nm narrow-spectrum imaging concept. The supplied Colab experiment therefore adds an explicit class-imbalance-aware objective and compares four segmentation architectures under a common training and evaluation framework. The focus is not to claim that U-Net++ or another network is universally superior, but to identify a strong architecture and a better training strategy for this particular imaging condition.

The work makes four practical contributions. First, it turns the supplied experiment into a reproducible held-out-test benchmark using 647 image-mask pairs and a fixed 80/10/10 partition. Second, it introduces a weighted CrossEntropy plus mite-specific Dice objective intended to reduce the tendency of the loss to be dominated by background and bee pixels. Third, it compares U-Net, Attention U-Net, U-Net++, and DeepLabV3+ on the same test set. Fourth, it reports both pixel-level segmentation measures and the connected-component metric implemented in the original code, while also documenting the important limitations of the annotation reconstruction, image-level splitting, and non-hyperspectral input representation.

2. Related Work

Automated Varroa monitoring can be viewed as a progression from indirect counting toward direct image-based analysis. Each stage reduces manual work but introduces different technical constraints. A useful way to understand the current study is to separate three strands: controlled multispectral or narrow-band imaging, richer hyperspectral sensing, and deep-learning-based detection or segmentation.

2.1. Controlled illumination and entrance monitoring

Entrance monitoring is attractive because it offers the possibility of observing bees as they move between the colony and the outside environment. Bjerger et al. demonstrated a camera-and-illumination system intended to estimate Varroa infestation from video at the hive entrance. Their reported detection performance showed that deep learning can be effective when image acquisition is carefully controlled [5]. The base paper used a related philosophy, but instead of relying only on visible appearance, it deliberately varied the illumination spectrum. This approach treats lighting as part of the sensing pipeline rather than as an uncontrolled environmental factor.

2.2. Hyperspectral and multispectral approaches

Hyperspectral systems can measure many closely spaced wavelengths and can therefore reveal differences that are difficult to see in ordinary RGB images. Duma et al. showed that Varroa and honey bee responses could be separated using spectral information and discussed reduced-band configurations as a route toward simpler devices [3]. Ghezal et al. investigated hyperspectral images of Varroa-contaminated hive debris and used conventional machine-learning methods after spectral processing, reporting strong segmentation performance [4]. These studies support the broader idea that the spectral domain contains discriminative information. At the same time, full hyperspectral cameras can be more expensive and operationally complicated than a single-band system.

2.3. Deep learning for Varroa detection

Deep learning has been applied to Varroa through both object detection and segmentation. A useful distinction is whether the image contains a mite by itself, such as on a sticky board, or a mite attached to a live or recently handled bee. Sticky-board detection is generally easier because the target is spatially separated from the host. Divasón et al. reported strong performance for deep-learning analysis of smartphone images of hive debris, with a validation mAP of 0.9073 [7]. By contrast, detecting a mite on the bee body is a nested-object problem. Kriouile et al. used a modified Mask R-CNN formulation to detect bees while segmenting internal Varroa regions and showed the value of architecture design for this more difficult setting [6].

2.4. Why narrow-spectrum segmentation remains relevant

Narrow-spectrum imaging occupies an interesting middle ground. It does not attempt to capture the entire spectrum, but it also does not assume that ordinary white-light RGB images contain the best possible contrast. If a small number of wavelengths can produce a repeatable appearance difference between mite and bee, then a simple camera plus controlled illumination may offer a

practical sensing configuration. The present study keeps that idea and concentrates on the learning side: how to train a semantic-segmentation model when the target class is small, the data set is limited, and the goal is reliable mite localization.

3. Research Gap, Questions, and Hypotheses

The base study already identified the most important practical weakness of the initial U-Net experiment: false negatives. The supplied manuscript and code also make clear that the mite class is sparse and that the original training objective was not specifically designed to protect a rare target class. Based on those observations, the present study addresses three linked gaps.

- Rare-target optimization: the contribution of background and bee pixels to the loss can overwhelm the contribution of the mite unless the objective is deliberately rebalanced.
- Architecture choice: the base study established U-Net as a useful baseline, but it did not provide the same controlled four-way architecture comparison used here.
- Deployment trade-offs: practical monitoring systems need more than a high segmentation score. They also need acceptable latency and model size, especially if inference is eventually moved to edge hardware.

The study is organized around two research questions:

- RQ1. Does explicit weighting of the mite class, combined with a mite-specific Dice term, produce a more useful sensitivity/precision balance than the original CrossEntropy-driven U-Net training?
- RQ2. Among U-Net, Attention U-Net, U-Net++, and DeepLabV3+, which architecture gives the best balance of mite segmentation quality and computational cost under the same narrow-spectrum input condition?

The corresponding hypotheses are:

- H1. A loss function that explicitly increases the influence of the Varroa class will improve mite recall and F1 relative to the original training behavior.
- H2. U-Net-family architectures with richer skip connectivity or attention will provide better

small-object sensitivity than the compact DeepLabV3+ implementation used in this study.

4. Materials and Methods

4.1. Relation to the 780-nm imaging concept

This study should be interpreted as a machine-learning extension of the 780-nm imaging concept rather than as a new optical-hardware contribution. The base Fčielka-Thor 3000 concept used three illumination rows corresponding to turquoise (~500 nm), infrared (~780 nm), and cold white light. The camera was an IR-filter-removed Raspberry Pi HQ camera. The original work observed that the mite was not sufficiently visible in the turquoise condition and therefore used the 780-nm images for U-Net segmentation [1].

The present Colab pipeline selects image files whose names or parent directories identify them as infrared or 780 nm. Those stored image files are read with OpenCV using a conventional image interface. In the neural network, the input is therefore a three-channel stored image representation, not a multi-band hyperspectral cube. This distinction is important: the study evaluates 780-nm narrow-spectrum image segmentation and should not be described as direct hyperspectral deep learning.

4.2. Dataset and image inventory

The supplied code identifies the data source as the Kaggle dataset `masuel/bee-hyperspectral-dataset`. The code searches the downloaded directory recursively, identifies image files, selects the subset associated with IR/780-nm naming, and attempts to construct image-mask pairs. The final run reconstructed 647 usable image-mask pairs, which is also the number of photographs described in the base study [1].

The base paper describes the image collection as containing bee, mite, and bee-with-mite examples and reports an image size of 1116 × 300 pixels, with images collected from multiple views and experimental conditions [1]. The current implementation uses those image files as the source for the segmentation benchmark. Because the supplied code performs its own file discovery and annotation reconstruction, the exact

provenance of each pair should be archived with the final repository rather than assumed from the

filename alone.

Table 1. Fixed image-level partition used by the final experiment (random seed = 0).

Split	Number of images	Percentage
Training	517	79.9%
Validation	65	10.0%
Test	65	10.0%

The partition was created by two successive random train-test operations using seed 0. This gives the intended 80/10/10 distribution, but the code does not group images by bee, acquisition session, or biological subject. The test set should therefore be interpreted as an image-level hold-out. Before publication, the sample identifiers should be audited to determine whether related views of the same bee occur across splits.

4.3. Annotation reconstruction

The implementation contains several annotation pathways. It first attempts direct mask pairing, then Label Studio JSON/JSONL parsing, then a YOLO fallback, and finally loose filename matching. The supplied result package contains generated files with names ending in `\_yolo\_mask.png`, which indicates that the executed run used the YOLO annotation path. In that path, YOLO class 0 is mapped to bee and class 1 to mite. Each bounding box is rasterized as a filled rectangle and assigned to the corresponding semantic class.

The resulting label image contains three classes: 0 for background, 1 for bee, and 2 for Varroa mite. This is a convenient representation for training a three-class semantic-segmentation network, but it is not equivalent to a manually traced contour. Pixels inside a bounding box can include neighboring bee tissue or background. The reported Dice and IoU should therefore be interpreted as agreement with the reconstructed mask rather than as exact measurement of the biological mite boundary. This distinction is central to a responsible interpretation of the numerical results.

4.4. Image preprocessing

Each image is loaded with OpenCV, converted from BGR to RGB, and resized to 50% of its original width and height using area interpolation. The associated mask is resized with nearest-neighbor interpolation so that class labels are not mixed during scaling. Image intensity values are divided by 255, producing input values in the range [0,1]. The supplied code contains no random crop, rotation, mirroring, color augmentation, or other synthetic augmentation step. After preprocessing, the network receives a three-channel tensor in channel-first order.

4.5. Network architectures

Four models were implemented directly in PyTorch. All models produce three output channels representing background, bee, and mite. This common output structure means that architecture comparison is carried out under the same semantic-label formulation.

4.5.1. U-Net

U-Net is the reference encoder-decoder architecture in this study [2]. The implementation uses four downsampling stages and four corresponding upsampling stages. Each convolutional block contains two 3×3 convolutions with batch normalization and ReLU activation. Max pooling performs spatial reduction, while bilinear interpolation is used for upsampling. Encoder features are concatenated with decoder features at matching spatial scales, preserving local information that can be lost in the deeper layers.

4.5.2. Attention U-Net

Attention U-Net modifies the skip connections by inserting attention gates before feature concatenation. The purpose is to modulate the spatial contribution of encoder features using the decoder context. For this problem, the intuition is straightforward: a model should be able to reduce the influence of irrelevant bee texture and concentrate the retained high-resolution features around regions that may contain a mite.

4.5.3. U-Net++

U-Net++ introduces nested, denser skip pathways between encoder and decoder representations. In the supplied implementation, intermediate feature maps are repeatedly combined and upsampled before reaching the final decoder output. The design provides multiple feature-routing paths through which fine spatial information can reach the output. This is potentially valuable for Varroa because the mite occupies a small region relative to the full image and may benefit from preservation of local detail.

4.5.4. DeepLabV3+

The supplied DeepLabV3+ implementation uses a compact convolutional encoder followed by an atrous spatial pyramid pooling module. The ASPP block contains parallel branches with dilation rates of 6, 12, and 18, together with a $1 \times$

1 branch. The multi-scale representations are then merged with a low-level feature path and decoded to the input resolution. The model has substantially fewer trainable parameters than the U-Net-family networks in this experiment.

4.6. Class-imbalance-aware training objective

The key methodological change from the original experiment is the loss function. The target mite class is sparse, which means a standard unweighted CrossEntropy objective can be satisfied to a large extent by predicting background and bee correctly. The final run therefore uses a weighted CrossEntropy term together with a Dice loss computed specifically for the Varroa class.

The training objective is written as:

$$L = L_WCE + \lambda L_Dice_mite$$

The experiment sets $\lambda = 1.0$ and uses class weights [0.25, 0.75, 3.0] for background, bee, and Varroa, respectively. The weighting reduces the contribution of the dominant background class and increases the contribution of the mite class. The mite Dice term then measures overlap between the predicted probability for the mite class and the binary ground-truth mite mask. This combination is intended to discourage a model from obtaining a deceptively low loss by largely ignoring the rare target.

4.7. Training configuration and checkpoint selection

Table 2. Final experimental configuration used by the supplied Colab implementation.

Setting	Value
Dataset pairs	647
Train / validation / test	517 / 65 / 65
Epochs	30
Optimizer	Adam
Learning rate	1e-4
Batch size	1
Image scale	0.5
Loss	Weighted CrossEntropy + Varroa Dice
Class weights	[0.25, 0.75, 3.0]
Checkpoint criterion	Highest validation mite F1; validation loss as tie-breaker
Random seed	0
SBM predicted-component threshold	20 pixels

The best checkpoint for each architecture is selected using the highest validation mite F1-score. If two epochs achieve effectively the same F1, the checkpoint with the lower validation loss is retained. The test set is only evaluated after model selection. This criterion is deliberately aligned with the research objective because the main failure mode of interest is missed mites rather than aggregate loss alone.

4.8. Evaluation metrics

Performance is calculated from the accumulated three-class confusion matrix, with class 2 treated as the target Varroa class. Precision measures the fraction of predicted mite pixels that are truly mite pixels, while recall measures the fraction of ground-truth mite pixels that are recovered. F1 summarizes the two quantities using their harmonic mean. Dice is identical to the F1 expression for the binary target class, while IoU uses the standard intersection-over-union formulation.

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP})$$

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN})$$

$$\text{F1} = 2\text{TP} / (2\text{TP} + \text{FP} + \text{FN})$$

$$\text{IoU} = \text{TP} / (\text{TP} + \text{FP} + \text{FN})$$

Pixel accuracy is also reported over all three classes. It is included for completeness, but it is not used as the primary ranking statistic because background and bee pixels greatly outnumber mite pixels. A model can therefore obtain high pixel accuracy while still providing poor mite recall.

4.9. Connected-component SBM evaluation

The code also implements a connected-component metric referred to as the Satisfied Bee Metric (SBM), following the terminology of the base work. Predicted and ground-truth mite masks are converted into connected components using 8-connectivity. Predicted components smaller than 20 pixels are removed. A predicted component counts as a hit if it overlaps at least one ground-truth component, and a ground-truth component counts as detected if it overlaps at least one prediction. The accumulated counts are then converted to precision, recall, and F1.

This metric is useful because it asks a practical question: did the segmentation produce any spatial evidence of a mite? It is, however, more permissive than standard object detection because a single-pixel overlap is sufficient for a hit. For this reason, SBM is treated as a supplementary object-presence measure rather than as a replacement for pixel-level IoU or standardized detection metrics such as mAP.

5. Results

5.1. Main architecture comparison

The primary comparison is intentionally organized around the published U-Net values of the base study. This preserves a clean historical reference while avoiding the mistake of presenting the current U-Net control as if it were the U-Net result from the earlier publication. For the base study, TP = 1,893, FN = 588, and FP = 207; the corresponding precision and recall are 90.14% and 76.30%, while F1 and IoU are derived from the same counts.

Table 3. Historical comparison with the published U-Net baseline and current held-out-test results.

Model source /	Mite Precision (%)	Mite Recall (%)	Mite F1 (%)	Mite IoU (%)	SBM Precision (%)	SBM Recall (%)	SBM F1 (%)
Published U-Net baseline [1]	90.14	76.30	82.65	70.42	—	—	—
Attention U-Net	89.36	82.33	85.70	74.97	97.94	100.00	98.96
U-Net++	85.91	86.62	86.26	75.84	95.00	100.00	97.44
DeepLabV3+	99.45	32.14	48.58	32.08	100.00	57.89	73.33

Among the current architectures, U-Net++ produced the highest mite F1-score and IoU. Its 85.91% precision and 86.62% recall are relatively balanced, which is reflected in the 86.26% F1-score. Attention U-Net was a close second, with slightly better precision (89.36%) but somewhat lower recall (82.33%). The conventional U-Net control achieved the highest recall of the three U-Net-family models at 87.29%, but its lower precision reduced its overall F1-score to 84.09%. DeepLabV3+ behaved differently. Its mite precision was 99.45%, which means that very few of the pixels it labelled as mite were false positives. The difficulty was sensitivity: only 32.14% of the ground-truth mite pixels were recovered. The resulting F1-score of 48.58% and IoU of 32.08% show that the model was conservative in assigning the mite class. This is a useful reminder that high precision alone is not sufficient for an infestation-monitoring problem.

5.2. Comparison with the base-paper U-Net

The best current model, U-Net++, has a derived F1-score of 86.26%, compared with

approximately 82.65% for the U-Net reported in the base paper. Its IoU of 75.84% is also higher than the approximately 70.42% derived for the published baseline. The current U-Net++ recall of 86.62% is substantially higher than the published U-Net recall of 76.30%, although the current precision is lower than the base-paper precision of 90.14%.

These differences should not be described as a direct head-to-head improvement because the experimental conditions are not identical. The present study uses a three-class formulation, an 80/10/10 held-out test split, a weighted CrossEntropy plus mite Dice objective, validation-F1 checkpointing, and a YOLO-derived mask reconstruction path. The base study used a different training formulation and evaluation setup. The most defensible conclusion is therefore that the present pipeline reaches a numerically stronger F1/IoU operating point on its own held-out test set while remaining directionally consistent with the base paper's narrow-spectrum U-Net findings.

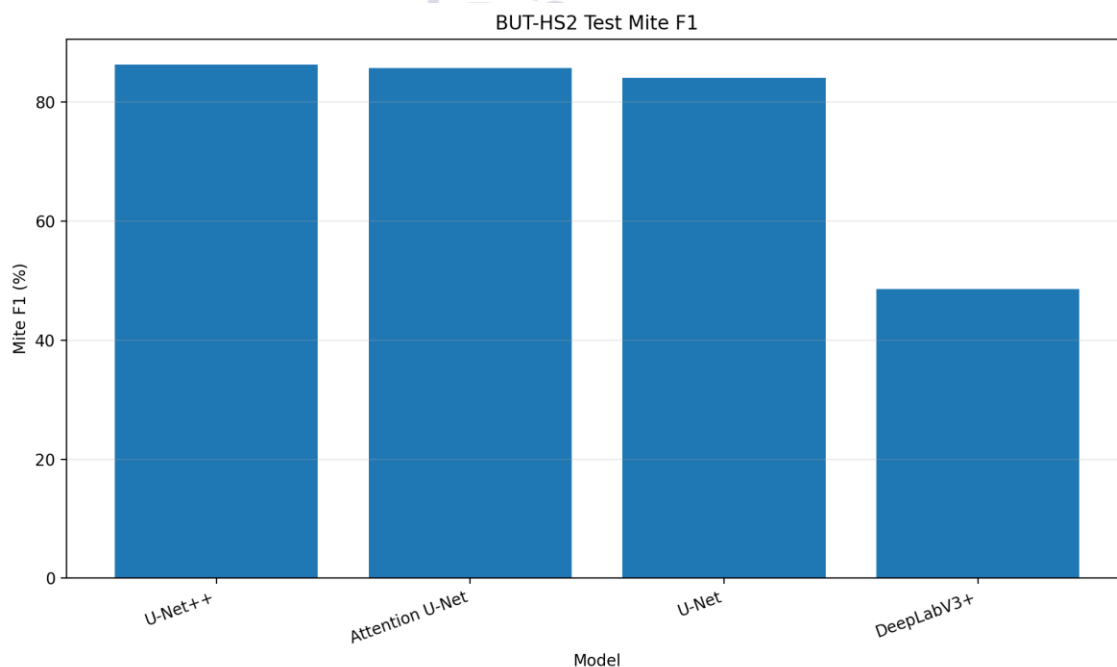


Figure 1. Mite F1-score comparison for the four architectures evaluated on the held-out test set. The historical U-Net value reported in the base study is discussed separately; the figure shows the current architecture benchmark.

Table 4. Full current held-out-test results from the supplied output package.

Model	Pixel Acc.	Precision	Recall	F1	IoU	SBM P	SBM R	SBM F1	Parameters	Inference (ms)
U-Net++	98.96	85.91	86.62	86.26	75.84	95.00	100.00	97.44	36,622,467	83.57
Attention U-Net	98.85	89.36	82.33	85.70	74.97	97.94	100.00	98.96	31,909,615	34.11
U-Net control	98.31	81.12	87.29	84.09	72.55	92.23	100.00	95.96	31,384,963	32.39
DeepLabV3+	98.21	99.45	32.14	48.58	32.08	100.00	57.89	73.33	3,773,123	10.25

5.3. Computational characteristics

The models also differ substantially in computational cost. DeepLabV3+ contains only 3.77 million trainable parameters and has the shortest measured forward-pass time at 10.25 ms. U-Net and Attention U-Net require about 32–34

ms per image, whereas U-Net++ requires 83.57 ms. Thus, U-Net++ is the strongest model for segmentation quality, while Attention U-Net provides a more attractive speed-quality balance if the application later requires near-real-time or embedded processing.

Table 5. Current model size and measured forward-pass latency.

Model	Parameters	Inference (ms)	Interpretation
Attention U-Net	31,909,615	34.11	High F1 with moderate latency
U-Net++	36,622,467	83.57	Best current F1 and IoU
U-Net control	31,384,963	32.39	Strong recall and simple baseline
DeepLabV3+	3,773,123	10.25	Smallest and fastest, but low recall

5.4. Effect of the class-imbalance-aware objective

The most revealing result in the project comes from comparing the original U-Net training behavior with the final U-Net control. In the original run, which used plain CrossEntropy and validation-loss checkpointing, U-Net reached 100.00% mite precision but only 10.42% recall, resulting in an F1-score of 18.88%. The model was therefore highly conservative: when it predicted a mite pixel it was usually correct, but it missed most of the target pixels.

After replacing the loss with weighted CrossEntropy plus a Varroa-specific Dice term and selecting checkpoints by validation mite F1, the same U-Net architecture reached 81.12% precision, 87.29% recall, 84.09% F1, and 72.55% IoU. This is a substantial change in the operating point. Recall increased by 76.87 percentage points, while F1 increased by 65.21 percentage points. The result strongly supports the study's first hypothesis and provides practical evidence that target-class weighting was not merely a cosmetic adjustment.

Table 6. Internal optimization ablation for U-Net. These values are included as a training-study analysis and are not substituted for the published U-Net historical baseline in Table 3.

U-Net training setup	Precision (%)	Recall (%)	F1 (%)	IoU (%)
Original CrossEntropy / loss-based checkpoint	100.00	10.42	18.88	10.42
Final weighted CE + mite Dice / F1 checkpoint	81.12	87.29	84.09	72.55

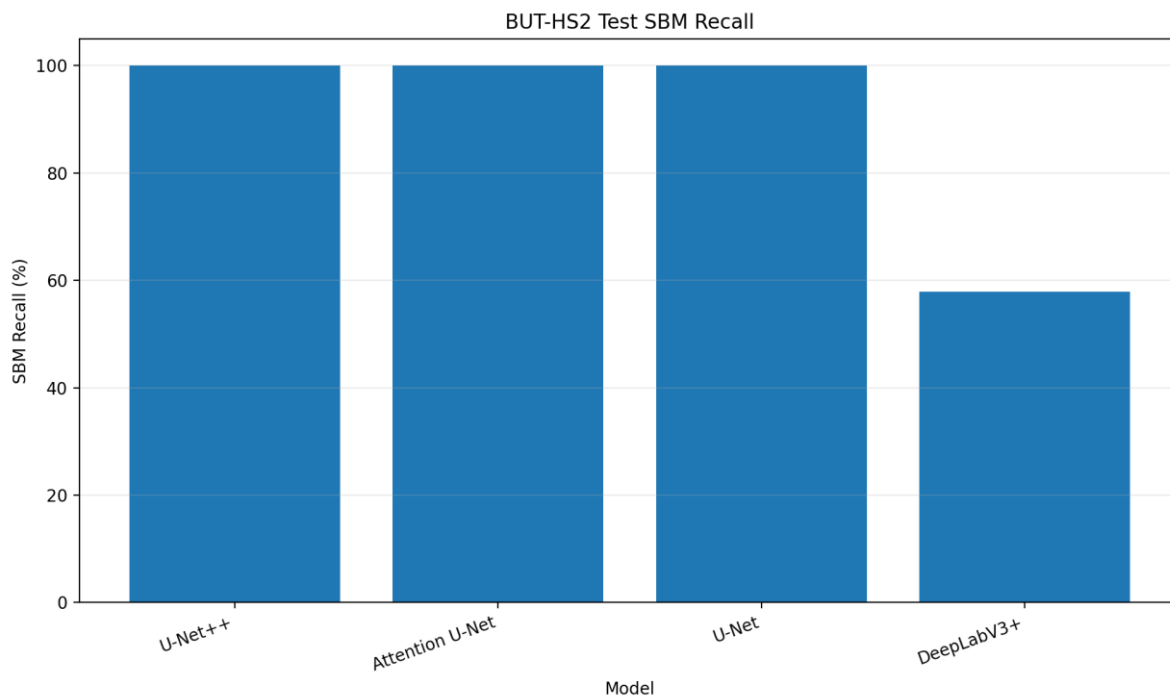


Figure 2. Connected-component SBM recall for the four current architectures on the held-out test set. The measure is based on overlap between predicted and ground-truth mite components.

5.5. Training convergence

Training and validation loss curves for all four models are shown in Figures 3-6. The curves show rapid loss reduction during the early epochs followed by a comparatively stable training regime. Validation loss is more variable, particularly for U-Net++ and DeepLabV3+, which

is consistent with the small validation set of 65 images. Because the final checkpoint was selected using validation mite F1 rather than loss alone, these curves are interpreted as convergence diagnostics rather than as the sole basis for model selection.

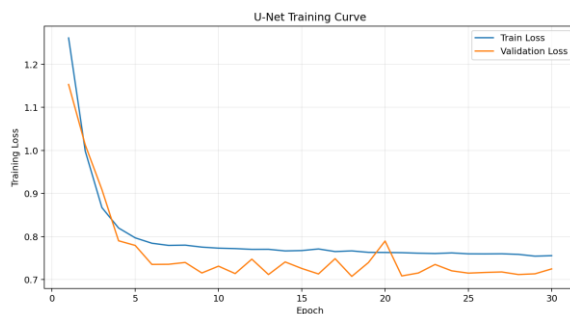


Figure 3. U-Net training and validation loss curves over 30 epochs.

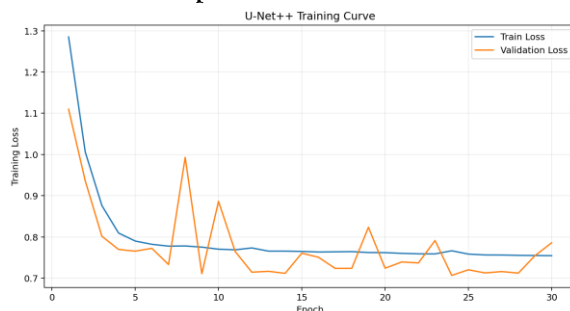


Figure 5. U-Net++ training and validation loss curves over 30 epochs.

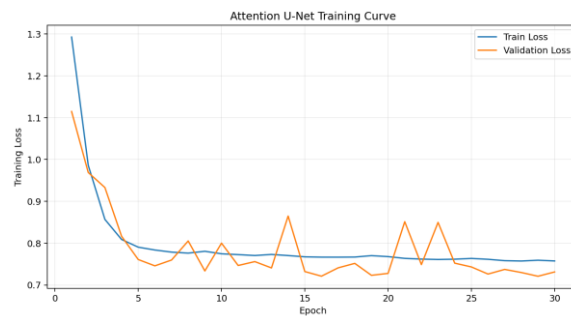


Figure 4. Attention U-Net training and validation loss curves over 30 epochs.

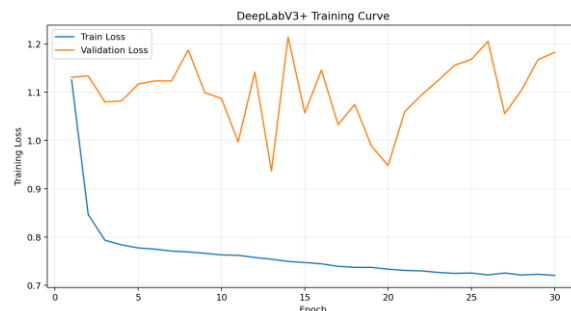


Figure 6. DeepLabV3+ training and validation loss curves over 30 epochs.

6. Discussion

6.1. What the results mean for the narrow-spectrum approach

The central message of the experiment is not that a particular network architecture solves Varroa detection. Instead, the results show that a useful narrow-spectrum image can still be wasted by an objective that treats the rare target as just another small portion of the image. Once the loss is modified to protect the mite class, the same U-Net architecture moves from an extremely conservative prediction regime to a much more balanced sensitivity regime.

This finding fits the practical limitation described in the base study. The earlier system demonstrated that 780-nm images can carry enough information for U-Net segmentation, but the false-negative burden remained high. The

current optimization result suggests a direct learning-side response to that limitation: the model can be encouraged to value the mite class more strongly without changing the underlying illumination condition.

6.2. Why U-Net++ is the best current model

U-Net++ achieves the highest F1 and IoU in the current benchmark, but the margin over Attention U-Net is small. The advantage is therefore better interpreted as a consistent, moderate improvement rather than a dramatic architectural breakthrough. A plausible explanation is that nested skip pathways preserve and repeatedly refine fine-scale features. A Varroa mite is a small target embedded in a much larger bee image, so preserving local detail across multiple decoder paths is intuitively useful.



Figure 7. Representative U-Net++ qualitative output showing the 780-nm input, reconstructed ground-truth mask, ground-truth mite region, and predicted mite region for a held-out test image.

The experiment does not contain an ablation that removes individual U-Net++ skip pathways, so the study cannot prove that the nested connections alone caused the improvement. The architectural explanation is consequently an interpretation of the observed pattern rather than a causal demonstration.

6.3. Attention U-Net as a practical deployment option

Attention U-Net produces 85.70% F1 and 74.97% IoU, only 0.56 percentage points below U-Net++. At the same time, its measured latency is 34.11 ms, compared with 83.57 ms for U-Net++. This difference is large enough to matter in an eventual embedded system. If the final application values speed almost as much as segmentation quality, Attention U-Net may be the more balanced engineering choice, even though U-Net++ remains the preferred model when maximum pixel-level performance is the primary objective.

6.4. Why DeepLabV3+ underperforms on the mite class

DeepLabV3+ is the smallest model in the comparison, yet its lower recall indicates that compactness comes with a sensitivity cost in the present setting. The model is not suffering from an explosion of false-positive predictions; its precision of 99.45% is actually the highest among the current models. Its weakness is the opposite: it is too selective when assigning the mite class. In a monitoring application, such conservatism can be undesirable because missed mites directly reduce detection sensitivity.

6.5. Interpreting the SBM results correctly

U-Net++, Attention U-Net, and the U-Net control all obtain 100% SBM recall in the current test output. This is encouraging because it means that, under the overlap-based object criterion used by the code, every ground-truth mite component intersects at least one predicted component. However, SBM recall does not say how much of each mite is covered. A prediction that touches the edge of a mite can count as a hit. The pixel-level F1 and IoU values are therefore

still necessary to distinguish a useful localization from a minimal overlap.

6.6. Implications for an eventual field system

If the eventual goal is live-bee monitoring at a hive entrance, several layers of robustness will be required beyond the present benchmark. The camera will encounter motion, variable bee orientation, changing body occlusion, and potentially changing illumination due to contamination or hardware aging. The current experiment uses a controlled dataset and therefore answers a narrower question: which learning strategy is promising under the available 780-nm image condition? It does not yet establish performance on live bees in uncontrolled operation.

7. Limitations and Threats to Validity

- Single split and single seed. All numerical results come from one fixed random partition and one random seed. The paper therefore reports a point estimate rather than a distribution across repeated runs. Future experiments should use repeated seeds or grouped cross-validation.
- Annotation reconstruction. The executed output indicates YOLO-derived rectangular masks. Such masks may include neighboring bee and background pixels, which affects the interpretation of Dice and IoU. Verified pixel-level annotations are needed for a stronger segmentation claim.
- Potential related-image leakage. The code splits at the image level and does not enforce grouping by bee, acquisition event, or hive. A future version should explicitly group related observations before splitting.
- Three-channel stored image input. Although the source dataset is described as hyperspectral, the implemented models ingest the selected 780-nm image file as a conventional three-channel image. The study therefore does not demonstrate direct hyperspectral neural-network processing.
- Custom SBM definition. The connected-component measure is permissive because any overlap counts as a hit. It should be reported as the study's custom object-presence metric rather

than equated with mAP or a standard IoU-thresholded detection score.

- Limited illumination comparison. Only the 780-nm condition is used in the final training pipeline. The experiment therefore cannot establish that 780 nm is globally optimal relative to 500 nm, white, or multi-band input.
- Laboratory/dataset setting. The current images are associated with the base dataset conditions and do not represent a complete test of live bees moving through a hive entrance. Motion blur and uncontrolled illumination remain open validation problems.
- No statistical significance analysis. The 0.56-point difference in F1 between U-Net++ and Attention U-Net is small. Without repeated trials, a significance claim would not be justified.

8. Reproducibility and Data Availability

The supplied project contains the Colab training pipeline, saved model checkpoints, training histories, test predictions, final comparison tables, experiment configuration, and plots. The final run records the following reproducibility settings: dataset identifier ``masuel/bee-hyperspectral-dataset``, 647 image-mask pairs, a 517/65/65 split, 30 epochs, Adam with a learning rate of $1e-4$, batch size 1, 0.5 image scale, class weights [0.25, 0.75, 3.0], a mite Dice weight of 1.0, validation mite F1 checkpoint selection, and seed 0.

For journal submission, the project should be placed in a permanent repository containing the exact notebook or Python script, package versions, dataset citation, split file, annotation files or annotation-generation procedure, model checkpoints or checksums, and the metric implementation. The annotation source should be described exactly as executed; the present output package contains YOLO-generated masks and should not be presented as manually traced contours unless the original annotation files confirm that claim.

9. Conclusion

This study extends the 780-nm narrow-spectrum Varroa detection direction established by Bielik and Bilik by concentrating on the learning problem rather than introducing new

illumination hardware. The principal change was to recognize the mite as a sparse target class and make that imbalance explicit in the training objective. A weighted CrossEntropy loss combined with a mite-specific Dice term transformed the behavior of the U-Net baseline from extremely conservative predictions to a much more useful sensitivity regime.

Among the four current architectures, U-Net++ produced the strongest pixel-level segmentation result, reaching 85.91% precision, 86.62% recall, 86.26% F1, and 75.84% IoU. Attention U-Net was only slightly behind on segmentation quality while requiring far less inference time, making it an attractive practical alternative. DeepLabV3+ was by far the smallest and fastest model, but its 32.14% mite recall makes it less suitable when missed mites are a primary concern.

The published U-Net results from the base study remain the appropriate historical reference rather than a value to substitute into the current model table. The present benchmark should be read as a follow-on study that retains the 780-nm imaging idea while changing the data split, semantic formulation, loss, model-selection rule, and architecture set. Within those conditions, the results support the view that class-imbalance-aware optimization and richer U-Net-family feature routing can make narrow-spectrum Varroa segmentation substantially more useful.

The next experimental steps are clear. The most important are to verify pixel-level ground truth, perform grouped data splitting, repeat training across multiple seeds, adopt a standardized object-level metric, and test the system on live bees under realistic entrance conditions. A controlled comparison of 500-nm, 780-nm, white, and multi-band inputs would also directly test whether the narrow-spectrum choice itself contributes to the observed performance. Those additions would turn the present benchmark into a stronger evidence base for a deployable precision-apiculture system.

10. Declarations

Funding. To be completed by the authors.

Conflicts of interest. To be completed by the authors.

Ethics statement. The study concerns honey bees and Varroa mites. The authors should insert the animal-handling and institutional statement applicable to the underlying image collection.

Author contributions. To be completed by the authors.

Data availability. The supplied code identifies the dataset as `masuel/bee-hyperspectral-dataset`. A permanent citation and repository link should be added before submission.

Code availability. The complete experimental pipeline, training histories, saved checkpoints, and evaluation outputs should be archived in a permanent repository for reproducibility.

REFERENCES

- [1] S. Bielik and S. Bilik, "Towards Varroa destructor mite detection using a narrow spectra illumination," *Proceedings of the 31st Conference STUDENT EEICT 2025: General Papers*, pp. 197–200, 2025. [arXiv:2504.06099](#). DOI/repository record: [hdl:11012/255279](#).
- [2] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015.
- [3] Z.-S. Duma, T. Zemcik, S. Bilik, T. Sihvonen, P. Honc, S.-P. Reinikainen, and K. Horak, "Varroa destructor detection on honey bees using hyperspectral imagery," *Computers and Electronics in Agriculture*, vol. 224, article 109219, 2024. doi:10.1016/j.compag.2024.109219.
- [4] A. Ghezal, C. J. L. Peña, and A. König, "Varroa Mite Counting Based on Hyperspectral Imaging," *Sensors*, vol. 24, no. 14, article 4437, 2024. doi:10.3390/s24144437.
- [5] K. Bjerger, C. E. Frigaard, P. H. Mikkelsen, T. H. Nielsen, M. Misbih, and P. Kryger, "A computer vision system to monitor the infestation level of Varroa destructor in a honeybee colony," *Computers and Electronics in Agriculture*, vol. 164, article 104898, 2019. doi:10.1016/j.compag.2019.104898.
- [6] Y. Kriouile, C. Ancourt, K. Węgrzyn-Wolska, and L. Bougueroua, "Nested object detection using mask R-CNN: application to bee and varroa detection," *Neural Computing and Applications*, vol. 36, pp. 22587–22609, 2024. doi:10.1007/s00521-024-10393-x.
- [7] J. Divasón, A. Romero, F. J. Martinez-de-Pison, M. Casalongue, M. A. Silvestre, P. Santolaria, and J. L. Yániz, "Analysis of Varroa Mite Colony Infestation Level Using New Open Software Based on Deep Learning Techniques," *Sensors*, vol. 24, no. 12, article 3828, 2024. doi:10.3390/s24123828.