Determining Object Orientation

Computer Vision Project Report

Group 6 – Asim, Jan-Philipp

July 12, 2025

1. Introduction

Tracking and analyzing the behavior of individuals within dense, dynamic groups remains a fundamental challenge in computer vision [1], [2]. Honey bee colonies consist of thousands of similar individuals that move and interact rapidly on cluttered surfaces [1]. This presents difficulties for automated image-based analysis, particularly when it comes to estimating the position and orientation of individuals in crowded settings [1], [2].

Bozek et al. [1] introduced a convolutional neural network (CNN)-based segmentation approach to detect unmarked bees and estimate their positions, orientations, and within-cell states on a natural honeycomb background. Their method achieved accurate tracking of individuals and revealed collective colony dynamics over extended periods.

Building on this approach, we focus specifically on estimating the orientations of individual bees in dense hives using segmentation masks of the head and tail regions. We implement and compare two deep learning architectures – a U-Net variant inspired by [1] and a U-Net with a ResNet18 encoder – and benchmark their performance based on segmentation quality and orientation accuracy. Our results demonstrate that while improved segmentation quality leads to more accurate orientation estimates, substantial annotation noise and challenging visual conditions limit overall accuracy.

2. State of the Art

Bozek et al. [1], [2] proposed a method to detect and track unmarked bees in dense hives. Their approach used a network to jointly predict segmentation masks of bee bodies and their orientation angles, achieving an orientation error of approximately 10° [1]. A key insight of their work was to exploit temporal information from preceding video frames via a recurrent component in the U-Net architecture, which reduced the number of parameters by about 94 % [1], [2]. However, this temporal component is not available to us, as our experiments operate on static images only.

To explore alternative architectures, we considered works demonstrating the effectiveness of ResNet-encoded U-Nets in segmentation tasks. Mukasheva et al. [3] modified U-Net with a ResNet50 encoder and an Atrous Spatial Pyramid Pooling block to improve segmentation quality

on medical images (IoU from 0.86 to 0.91-0.93). Although their results were on medical data, such architectures may also help improve segmentation quality in our setting, which exhibits clutter and occlusions but also isolates individual bees through cropping.

Motivated by these findings, we compare two architectures: (i) a compact 3-level U-Net inspired by [1] and (ii) a U-Net with a ResNet18 encoder. The first provides a lightweight, proven baseline for bee detection and orientation estimation, while the second leverages pretrained features and greater capacity. In contrast to previous work, we specifically aim to segment head and tail regions separately to improve orientation estimates.

3. Methods

All code, data preprocessing scripts, and trained models are available at: github.com/Kenste/CV-BeeOrientation for full reproducibility.

3.1. Data

We use the Honeybee Segmentation and Tracking Dataset[1], introduced by Bozek et al. [1], focusing on the 30 fps grayscale video recordings. This dataset provides annotated frames indicating the positions, orientations, and within-cell states of the bees.

We extract individual training examples from these frames by cropping fully visible bees into 160×160 grayscale images. The crop size balances two competing goals: ensuring the bee is fully visible and centered, despite size differences and annotation noise, while minimizing the presence of neighboring bees. Some unavoidable background clutter may appear at the edges of the crop, but the annotated bee is centered and the only segmented individual. Other bees, if present, are treated as background noise.

For each cropped image, we generate a dense segmentation mask, initialized as background (label 0). An ellipse is placed at the center of the bee, aligned with the annotated orientation and approximating the bee's body shape. The half of the ellipse pointing in the orientation's direction is labeled head (label 1), and the opposite half is labeled tail (label 2). We also store the ground-truth orientation angle of the bee in a CSV file, along with the corresponding mask and crop filenames.

Processing all annotated frames yields approximately 130 266 samples for training, validation, and testing. Occasional annotation errors (e.g., incorrect orientations) were rare and treated as noise that is not expected to bias the training process.

We split the dataset into training (64 %), validation (16 %), and test (20 %) sets. We ensure reproducibility by fixing and reporting the random seed.

3.2. Method 1: U-Net3

Our first architecture is a compact, three-level U-Net inspired by the work of Bozek et al. [1], [2], and their publicly available implementation[2]. We reimplemented the architecture in PyTorch,

[1]Dataset available at groups.oist.jp/bptu/honeybee-tracking-dataset
[2]Available at github.com/kasiabozek/bee_tracking

while keeping the design as close as possible to the original TensorFlow implementation. The network consists of three downsampling blocks with increasing filter sizes (32, 64, and 128), a 256-filter bottleneck, and symmetric decoder blocks with skip connections. The final step is a 1x1 convolution that produces three-class logits (background, head, and tail).

We trained this model using an Adam optimizer with a learning rate of 10^{-3} and a weighted cross-entropy loss with class weights of [0.1, 1.0, 1.0] to mitigate the dominance of background pixels. The model contains 1 927 907 parameters and was trained for up to ten epochs. These hyperparameters were selected based on preliminary experiments and found to yield reasonable performance.

3.3. Method 2: U-ResUNet18

The second architecture integrates a ResNet18 encoder into a U-Net-like decoder structure, following Mukasheva et al.'s [3] idea of using a ResNet backbone with skip connections to improve segmentation quality. For the implementation, we followed the decoder design from the segmentation_models.pytorch repository [4].

The encoder is a pretrained ResNet18, and its intermediate feature maps are connected with upsampled decoder outputs via skip connections. The decoder mirrors the U-Net structure and ends with a 1×1 convolution, producing three-class logits.

As with the U-Net3, we used the Adam optimizer with the same learning rate and loss weights and trained the model for ten epochs. Due to the larger encoder, the ResUNet18 has substantially more parameters (14 239 721) than the U-Net3.

3.4. Experiments and Evaluation

We trained both models using the same dataset splits and identical training and validation pipelines. For evaluation purposes, we report on segmentation quality and orientation accuracy on the test set.

Segmentation quality is measured by the per-class Intersection-over-Union (IoU) and the mean IoU (mIoU) computed over the foreground classes (head and tail). IoU and mIoU are standard metrics widely used in segmentation benchmarks [5]–[8].

To estimate the bee orientation from a predicted segmentation mask, we compute the center of mass of head and tail pixels, form a vector from tail to head, and calculate the angle between this vector and the vertical upward direction (measured clockwise). Mathematically [9]:

$$\alpha = \operatorname{atan2}\left(x_{\text{head}} - x_{\text{tail}},\ y_{\text{tail}} - y_{\text{head}}\right)$$

where $(x_{\text{head}}, y_{\text{head}})$ and $(x_{\text{tail}}, y_{\text{tail}})$ are the respective centers of mass of the head and tail regions. The resulting angle α is compared to the ground truth angle from the CSV annotations, and the angular error (in degrees) is reported.

We also quantify the inherent base error present in the dataset by comparing the orientation derived from the ground truth masks to the annotated ground truth angles in the CSV. This accounts for discretization artifacts and annotation noise in the masks and serves as a lower bound on the achievable orientation error.

4. Results

We begin by quantifying the base error inherent in the ground-truth masks. We then compare the segmentation performance of our two architectures. Next, we evaluate the resulting orientation accuracy. Finally, we examine the relationship between segmentation quality and orientation error. All quantitative results are summarized in tables 2 and 3, while key qualitative examples are shown in figs. 2 and A.8.

4.1. Dataset & Baseline Error

To quantify the inherent noise in the annotations, we compared the ground truth angles obtained from the annotations with those derived from the ground truth segmentation masks (see Table 1). The small discrepancy (mean $\approx 0.25°$) arises from the discretisation of the ellipses into pixelated masks, and it represents the lower bound of the achievable orientation error. Figure A.5 shows the histogram and cumulative distribution function (CDF) of this baseline error.

Table 1: Orientation error between GT masks and GT CSV angles (baseline error).

Metric	**Error**
Mean	0.25°
Std. Dev.	0.20°
Median	0.22°
Percentiles	
50 %	0.22°
75 %	0.39°
90 %	0.54°
95 %	0.62°
99 %	0.74°
Max	0.85°

4.2. Segmentation Performance

First, we evaluate the segmentation quality of both models on the test set, reporting per-class IoU and foreground mIoU over the head and tail classes.

Table 2 summarizes the results: ResUNet18 substantially outperforms UNet3 across all classes, achieving a foreground mIoU of 0.71 compared to 0.48 for UNet3. ResUNet18 also achieves a notably lower segmentation loss and a higher background IoU.

Figure 1 shows the training and validation loss curves over 10 epochs. Both models converge quickly, but ResUNet18 starts to overfit after around four epochs, as indicated by an increase in validation loss. Nevertheless, ResUNet18 consistently achieves lower losses than UNet3 throughout training – UNet3 never reaches ResUNet18's loss levels, even at its best.

Figure 2 presents qualitative predictions for three test samples for each model. Each example shows the input image, the ground-truth mask and the predicted mask. ResUNet18's predictions

appear sharper and more accurate, with clear distinction between the head and tail, whereas UNet3's predictions are blurrier and more easily confused by noise from other bees or background clutter.

Table 2: Segmentation performance on the test set.

Model	**#Params**	**Test Loss**	**IoU$_0$**	**IoU$_1$**	**IoU$_2$**	**mIoU$_{1,2}$**
UNet3	1.93 M	0.3620	0.8959	0.4852	0.4827	0.4839
ResUNet18	14.24 M	0.1742	0.9482	0.7069	0.7205	0.7137

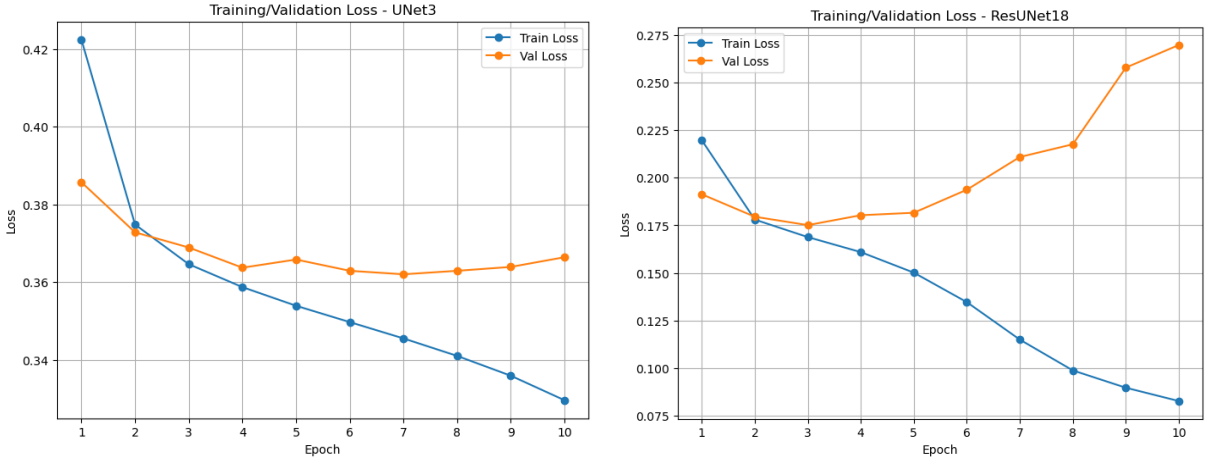

Figure 1: Training and validation loss curves over 10 epochs for both models. (**Left**) UNet3 shows steady convergence but plateaus at a higher loss. (**Right**) ResUNet18 achieves lower training and validation loss overall but begins to overfit after about 4 epochs, as indicated by the upward trend in validation loss.

4.3. Orientation Estimation

Next, we evaluate the models' ability to predict the bees' orientation angles based on the predicted head and tail masks. Table 3 summarizes the orientation error on the test set, reported as the mean, median and key percentiles.

ResUNet18 achieves slightly better orientation accuracy than UNet3: the mean error is reduced from 14.8° to 13.2°, and the median error from 6.5° to 5.7°. However, both models remain well above the baseline error of approximately 0.25° measured between the ground-truth masks and the ground-truth CSV angles. This suggests that imperfect segmentations rather than annotation noise are the main limiting factor.

Figure 3 shows the distribution of absolute orientation errors for each model, presented as both a histogram and a CDF. Both models exhibit a strong concentration of predictions with errors below 20°, with ResUNet18 producing slightly more low-error predictions than UNet3. The

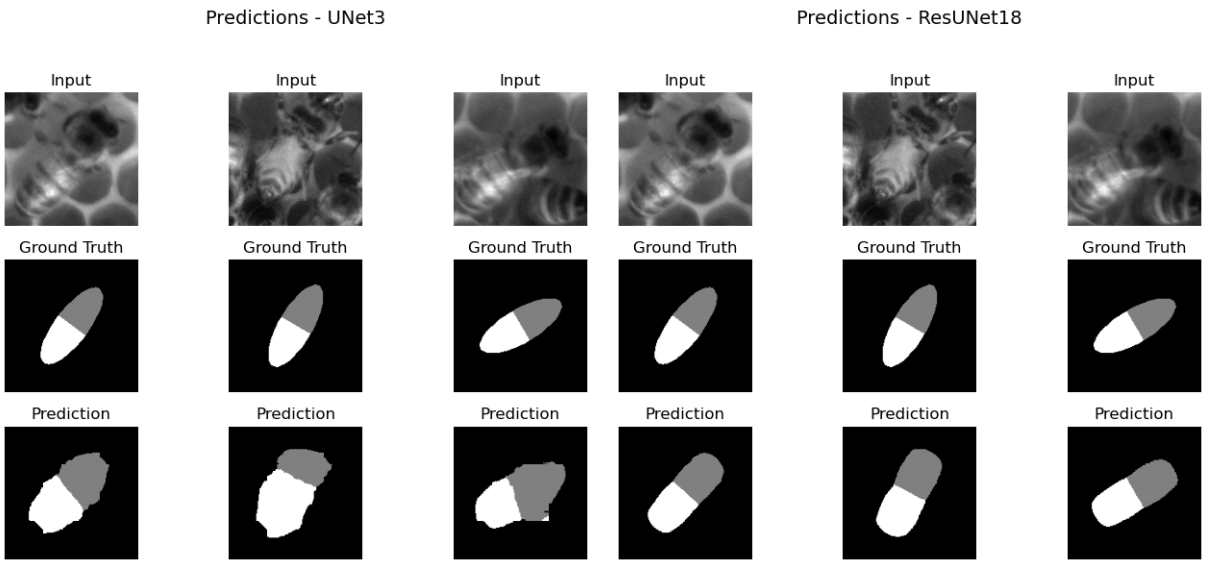

Figure 2: Example segmentation predictions on three test samples for each model. Each column shows (*top*) the input image, (*middle*) the ground-truth mask, and (*bottom*) the predicted mask. (**Left**) UNet3 predictions are blurry and frequently misclassify head/tail regions due to noise and clutter. (**Right**) ResUNet18 predictions are sharper and better capture head and tail shapes, even in the presence of background bees and occlusions.

overall distributions are otherwise similar, with both models displaying a noticeable secondary peak at very high errors (approximately 180°), which is likely caused by head–tail confusion or annotation errors. We investigate this phenomenon further in section 4.5.

We also examined the signed orientation errors to check for systematic directional bias in the predictions (see Figures A.6 and A.7). No substantial bias was observed, with the errors being approximately symmetrically distributed around zero.

Table 3: Absolute orientation error against ground-truth CSV angles.

Model	**Mean ± SD**	**Median**	**75%ile**	**90%ile**	**95%ile**	**99%ile**
UNet3	14.79° ± 31.57°	6.49°	12.16°	21.91°	50.57°	174.60°
ResUNet18	13.23° ± 31.12°	5.73°	10.59°	17.98°	30.25°	175.74°

4.4. IoU vs. Orientation Error Relationship

We next investigate the relationship between segmentation quality and orientation accuracy. Specifically, we analyze whether higher foreground mIoU correlates with lower orientation errors.

Figure 4 shows hexbin plots of foreground mIoU versus absolute orientation error for UNet3 and ResUNet18 on the test set. A clear negative correlation is evident for both models, as confirmed

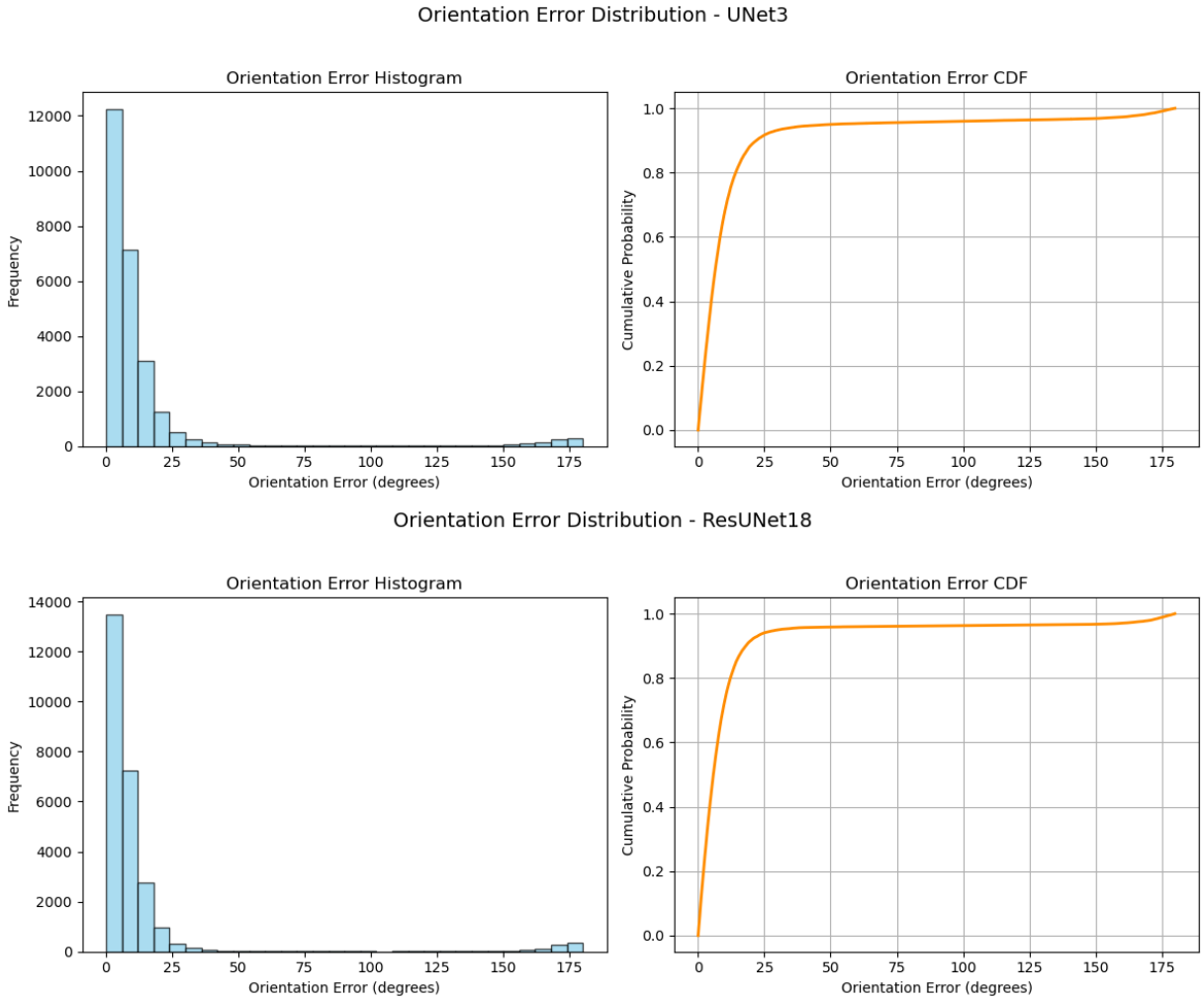

Figure 3: Orientation error distribution on the test set for UNet3 (**top**) and ResUNet18 (**bottom**). For each model, the left panel shows a histogram of absolute orientation errors; the right panel shows the cumulative distribution function (CDF). Both models produce similar distributions with most errors below 20° and a small bump at very high errors (≈180°).

by Spearman's rank correlation ($\rho = -0.40$, $p < 0.001$ for UNet3 and -0.73, $p < 0.001$ for ResUNet18) [10]: as mIoU increases, orientation error decreases.

However, ResUNet18 predictions are concentrated more tightly in the top left of the plot (high mIoU and low error), while UNet3 predictions are more scattered. UNet3 produces many examples with low mIoU even when the orientation error is moderate ($\leq 50°$), whereas ResUNet18's predictions are more consistent, with higher mIoU and less variation.

This suggests that better segmentation masks (higher mIoU) generally enable more accurate orientation estimation and that ResUNet18 produces more consistent and reliable segmentations.

We also examined the signed orientation error as a function of the ground-truth angle (see Figures A.6 and A.7). No clear dependency was observed; the error distribution appears uniform across the range of ground-truth angles.

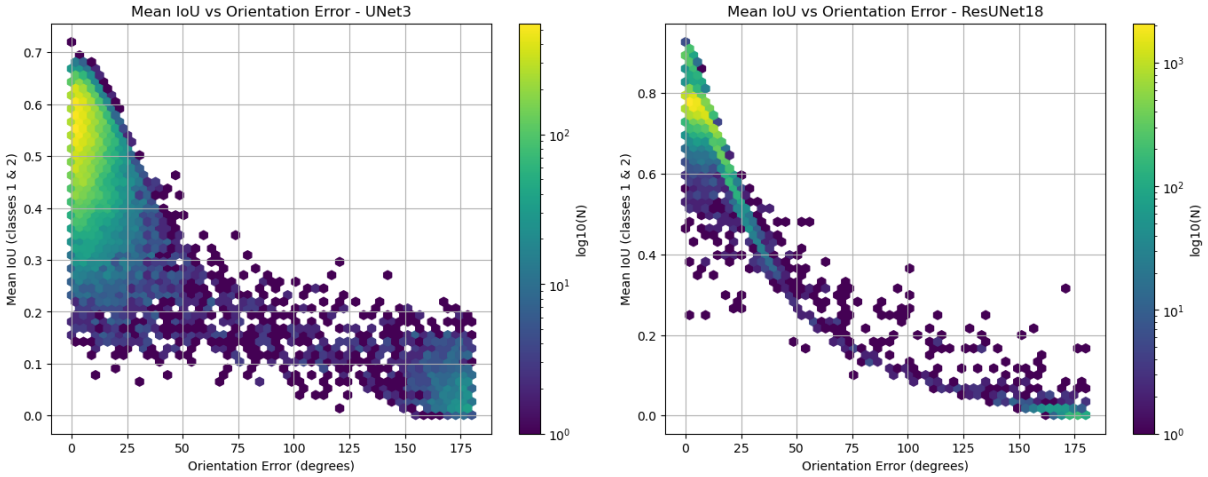

Figure 4: Hexbin plots of foreground mIoU versus absolute orientation error for (**left**) UNet3 and (**right**) ResUNet18. Both models exhibit a negative correlation (Spearman's $\rho = -0.40$ for UNet3, $\rho = -0.73$ for ResUNet18), but ResUNet18 predictions are more concentrated in the upper-left corner (high mIoU, low error). In contrast, UNet3 predictions are dispersed with a broader spread of mIoU for any error.

4.5. Qualitative Analysis

To gain a better understanding of the areas in which the models are struggling, we conducted a qualitative analysis of failure cases from the test set. We focused on examples with significant orientation errors, particularly those around 90° and 180°, as these are indicative of challenging scenarios or annotation issues.

Representative examples are shown in the appendix (Figure A.8), grouped by error type. Each panel displays the input image, the ground-truth mask, the predicted mask, and the measured orientation error.

For 180° errors, the majority appear to stem from annotation errors rather than model mistakes: in all the examples shown, the annotated head and tail regions are reversed compared to the

visible anatomy, while the model predictions point in the correct direction despite the imperfect masks. For instance, the UNet3 prediction in the first image provides a blurry yet correctly oriented segmentation, whereas the annotation erroneously labels the tail as the head. Similarly, the ResUNet18 predictions align well with the actual bee orientation, even when the annotation is flipped.

Both models struggle with 90° errors in highly cluttered images where no bee is fully visible. In the UNet3 examples, the predicted masks are diffuse and are often confused by surrounding bees. The annotations themselves are ambiguous or clearly do not correspond to any individual bee. ResUNet18 performs better, often correctly segmenting one plausible bee even when the annotation does not correspond to it. In some cases, the image quality or visibility is so poor that neither the model nor the annotation appears reliable.

These examples highlight the limitations of the models, particularly in cluttered and low-quality regions, as well as the inconsistencies in the ground-truth annotations, which may artificially inflate measured errors.

5. Discussion

We demonstrated that head/tail segmentation could be employed to estimate bee orientation to a certain degree. However, we did not achieve the level of orientation accuracy reported by Bozek et al. [1].

ResUNet18 produced slightly lower orientation errors and better segmentation than UNet3. However, both models fell far short of the ground-truth mask baseline error of approximately 0.25°, suggesting that segmentation errors dominate. Furthermore, our data suggests a correlation between segmentation quality and orientation accuracy. Nevertheless, even low-mIoU predictions occasionally yielded acceptable orientations. Many extreme errors appear to stem from mislabelled annotations, suggesting that the true orientation performance may be better than the measurements indicate.

While ResUNet18's stronger encoder helps in cluttered conditions, it also overfits quickly. Both models struggle in extreme clutter or when the bee is barely visible. Potential improvements include better annotations, regularization, and post-processing of predictions to resolve ambiguities. Finally, leveraging temporal information or multi-frame context, as in the work of Bozek et al. [1], [2], could enhance robustness and orientation estimation further.

References

[1] K. Bozek, L. Hebert, Y. Portugal, A. S. Mikheyev, and G. J. Stephens, "Markerless tracking of an entire honey bee colony," *Nature communications*, vol. 12, no. 1, p. 1733, 2021.

[2] K. Bozek, L. Hebert, A. S. Mikheyev, and G. J. Stephens, "Pixel personality for dense object tracking in a 2d honeybee hive," *arXiv preprint arXiv:1812.11797*, 2018.

[3] A. Mukasheva, D. Koishiyeva, G. Sergazin, M. Sydybayeva, D. Mukhammejanova, and S. Seidazimov, "Modification of u-net with pre-trained resnet-50 and atrous block for polyp segmentation: Model taspp-unet," *Engineering Proceedings*, vol. 70, no. 1, p. 16, 2024.

[4] P. Iakubovskii, *Segmentation models pytorch*, `https://github.com/qubvel/segmentation_models.pytorch`, 2019.

[5] T.-Y. Lin, M. Maire, S. Belongie, *et al.*, "Microsoft coco: Common objects in context," in *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part v 13*, Springer, 2014, pp. 740–755.

[6] A. Kirillov, E. Mintun, N. Ravi, *et al.*, "Segment anything," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 4015–4026.

[7] D. Jha, M. A. Riegler, D. Johansen, P. Halvorsen, and H. D. Johansen, "Doubleu-net: A deep convolutional neural network for medical image segmentation," in *2020 IEEE 33rd International symposium on computer-based medical systems (CBMS)*, IEEE, 2020, pp. 558–564.

[8] 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: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, Springer, 2015, pp. 234–241.

[9] "Math — mathematical functions — python 3.13.5 documentation." (), [Online]. Available: `https://docs.python.org/3/library/math.html#math.atan2` (visited on 07/11/2025).

[10] "Spearman's rank correlation coefficient - wikipedia." (), [Online]. Available: `https://en.wikipedia.org/wiki/Spearman%27s_rank_correlation_coefficient` (visited on 07/11/2025).

A. Additional Figures

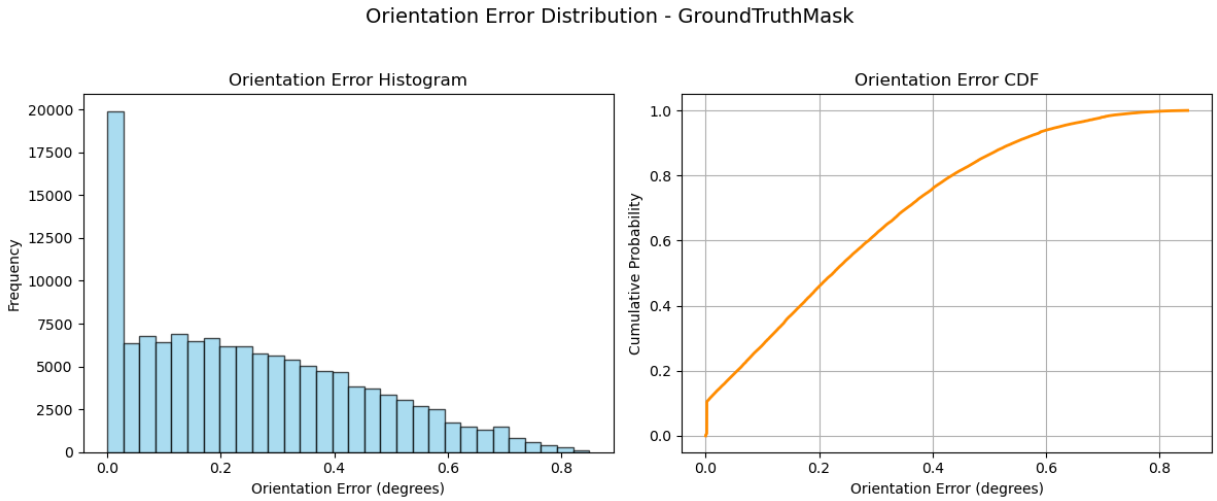

Figure A.5: Baseline orientation error between ground-truth (GT) angles from the CSV annotations and those computed from the GT segmentation masks. (**Left**) Histogram of the error in degrees. (**Right**) Cumulative distribution function (CDF) of the error. The error originates from discretizing continuous ellipses into pixel masks and represents the minimal possible error achievable.

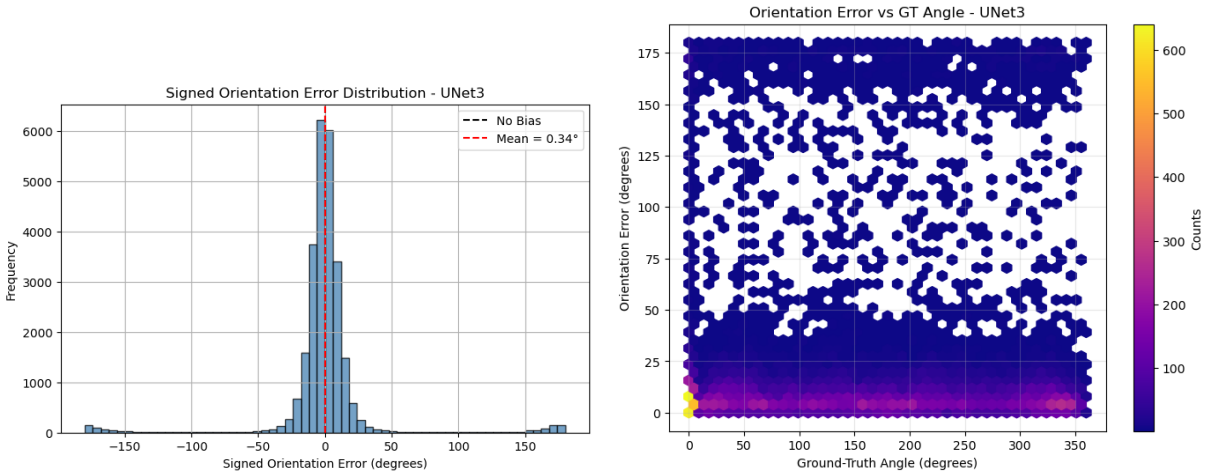

Figure A.6: Analysis of orientation errors for UNet3. (**Left**) Distribution of signed orientation errors showing no substantial bias, with errors approximately symmetrical around zero. (**Right**) A hexbin plot showing the distribution of absolute orientation error versus the ground-truth angle. Errors appear to be uniformly distributed across the range of ground truth angles. The concentration of points at 0° reflects the overrepresentation of 0° GT angles in the dataset, but the error pattern remains similar at other angles.

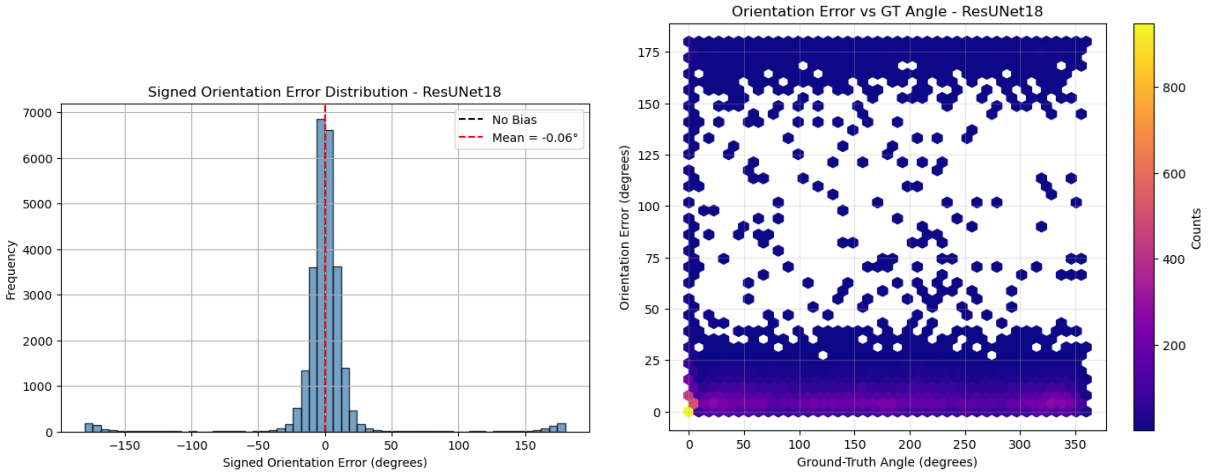

Figure A.7: Analysis of orientation errors for ResUNet18. (**Left**) Distribution of signed orientation errors, again showing no systematic bias and symmetric distribution around zero. (**Right**) Hexbin plot of absolute orientation error versus ground-truth angle, indicating no particular GT angle is associated with higher or lower errors. The dense region at 0° reflects dataset imbalance rather than a modeling issue.

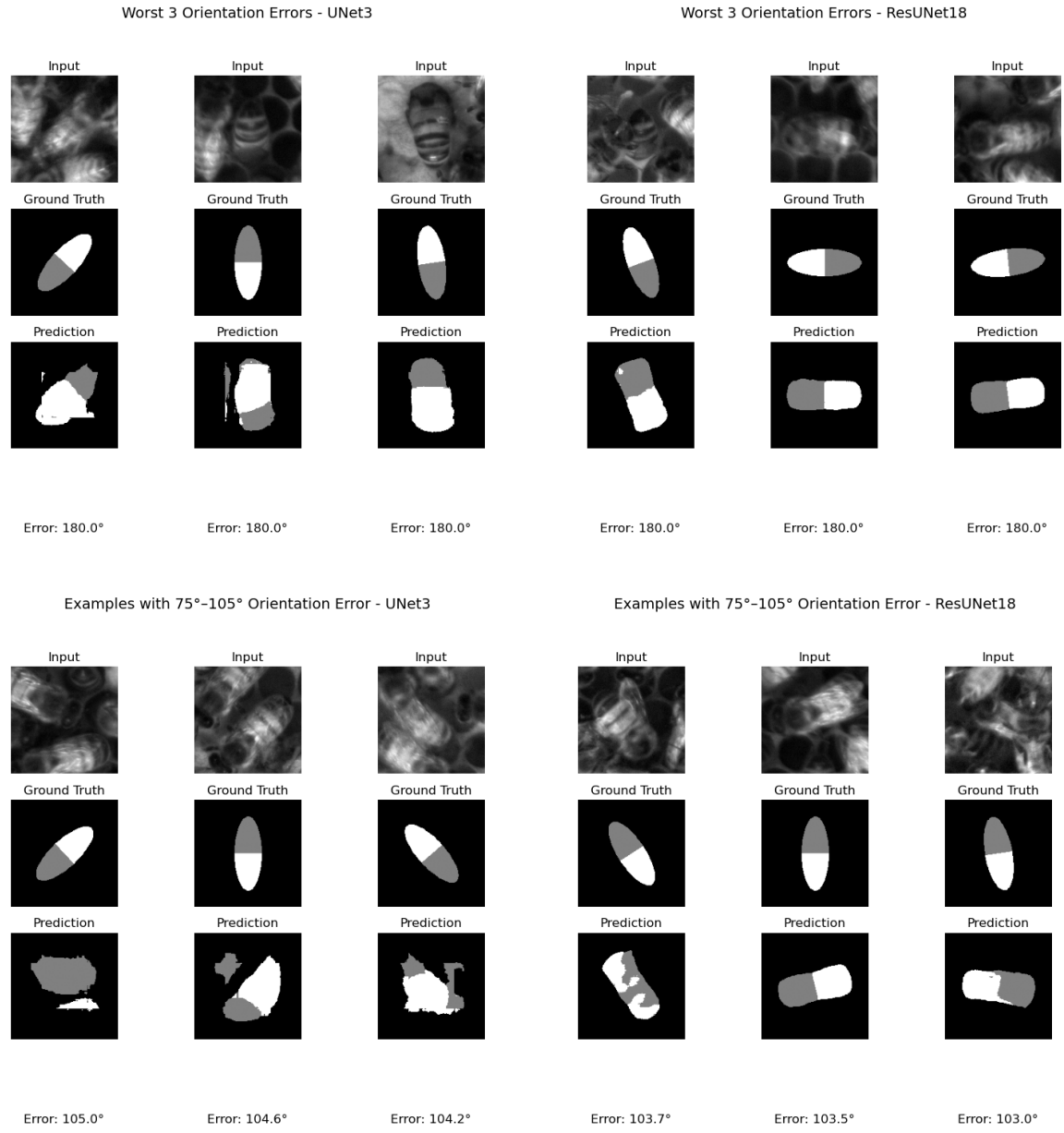

Figure A.8: Qualitative failure cases for UNet3 (**left**) and ResUNet18 (**right**), with the worst absolute orientation errors (**top row**) and examples with errors around 90° (**bottom row**). Each panel shows the input image, ground-truth mask, predicted mask, and measured error. In the worst (180°) cases, annotations are often flipped, while model predictions remain plausible. Around 90°, both models struggle in highly cluttered or ambiguous scenes, but ResUNet18 still produces more coherent segmentations than UNet3.