THE GEORGE WASHINGTON UNIVERSITY

WASHINGTON, DC

DATS 6303: Deep Learning Project Report

BYU - Locating Bacterial Flagellar Motors 2025

Group Number: 2
Pranav Dhawan,
Aakash Singh Sivaram,
Abhinaysai Kamineni

Supervised by
Dr Amir Jafari

ABSTRACT

In this project, we tackle the problem of detecting and localizing bacterial flagellar motors (BFMs) in high-resolution 3D electron microscopy (EM) images, a critical task for advancing microbiological research and diagnostics. Due to the low signal-to-noise ratio and subtle structural features in EM data, we investigate and compare the performance of three advanced object detection models: CenterNet, YOLOv10, and Faster R-CNN. These models are applied to preprocessed and augmented 3D volumetric data, including normalized and sliced representations where necessary. Model evaluation is conducted using Mean Average Precision at IoU thresholds of 0.5 (mAP@50) and 0.5 -- 0.95 (mAP@95), as well as precision and recall metrics, to assess both detection accuracy and spatial localization performance. Anticipated challenges such as noise, class imbalance, and high computational costs are addressed through data augmentation, use of pretrained weights, and optimization strategies. The goal is to develop a reliable and efficient pipeline for biological object detection in complex volumetric imaging data.

CONTENTS

1. INTRODUCTION

The accurate detection and localization of bacterial flagellar motors (BFMs) play a crucial role in understanding bacterial motility, a key factor in processes such as chemotaxis, pathogenesis, and environmental adaptation. BFMs are complex molecular machines embedded in the bacterial membrane that drive flagellar rotation, enabling mobility. Despite their biological importance, identifying BFMs in imaging data remains a challenging task due to their small size, subtle visual characteristics, and the inherently noisy nature of electron microscopy (EM) data.

This project addresses the challenge posed by the BYU - Locating Bacterial Flagellar Motors 2025 Kaggle competition, which provides a dataset of high-resolution 3D EM tomographic volumes along with annotated motor locations. The objective is to build an automated pipeline capable of accurately detecting BFMs within these volumes. Given the 3D structure of the data and the complexity of the task, traditional object detection methods fall short in handling the spatial and visual subtleties present in the dataset.

To overcome these challenges, we employ and compare three state-of-the-art object detection models: CenterNet, YOLOv10, and Faster R-CNN. These models represent different architectural paradigms—keypoint-based detection, anchor-free real-time detection, and region proposal-based detection—allowing a comprehensive analysis of their effectiveness on biomedical volumetric data. The dataset undergoes extensive preprocessing, including normalization, augmentation (rotation, flipping, scaling), and 3D-to-2D slice conversion where required to support model training and inference.

This study aims not only to benchmark detection performance using metrics such as mAP@50, mAP@95, precision, and recall, but also to contribute an adaptable detection pipeline for future applications in biological image analysis. Ultimately, this work demonstrates how deep learning can be leveraged to automate and scale the identification of microscopic cellular structures in complex scientific datasets.

2. PROBLEM STATEMENT

Identifying bacterial flagellar motors (BFMs) in electron microscopy (EM) images is a critical but highly challenging task in microbiology. These molecular structures are fundamental to bacterial motility, yet their detection is hindered by several factors: their nanoscale size, subtle appearance, and the inherently low signal-to-noise ratio in EM tomographic data. Manual annotation is time-consuming, error-prone, and not scalable for large datasets. Furthermore, the volumetric nature of the data adds another layer of complexity, requiring precise spatial localization within 3D image volumes. There is a pressing need for an automated, accurate, and efficient method to detect and localize BFMs in EM data to support research in bacterial physiology, pathogenesis, and drug development.

3. RELATED WORK

The task of detecting subcellular structures in electron microscopy (EM) data lies at the intersection of object detection, biomedical image analysis, and volumetric data processing. Traditional computer vision approaches, such as edge detection and feature-based classifiers (e.g., SIFT, HOG), were once standard for microscopic object recognition but have largely been superseded by deep learning due to their limited robustness in noisy, high-variance biological datasets [1].

The emergence of convolutional neural networks (CNNs) has significantly advanced object detection, especially in the context of biomedical imaging. Two-stage object detectors like Faster R-CNN [2] have proven highly effective in tasks requiring high localization accuracy. They are widely used in domains such as tumor detection in CT scans, lesion identification in pathology images, and anatomical landmark localization. These models first generate region proposals and then refine and classify them, making them accurate but computationally intensive.

In contrast, one-stage detectors such as YOLO (You Only Look Once) [3] and SSD (Single Shot MultiBox Detector) [4] focus on real-time performance by predicting bounding boxes and class probabilities in a single pass. YOLO's architectural evolution culminating in versions like YOLOv5 and YOLOv10 has introduced innovations such as anchor-free heads, Cross Stage Partial Networks (CSP), PANet for feature aggregation, and decoupled detection heads, enhancing both speed and accuracy in complex environments [5]. These detectors have been successfully applied in microscopy for tasks like parasite detection, cell tracking, and phenotypic classification.

Keypoint-based detection models, such as CenterNet [6], present another alternative by detecting objects as points (typically their centers) and regressing object dimensions and offsets. This anchor-free, heatmap-driven approach avoids issues related to anchor box design and is particularly effective in detecting small or densely packed objects—an ideal fit for EM applications where the target structures are sparse and subtle.

For volumetric data like 3D EM scans, researchers have explored 3D CNNs that directly process voxel data, preserving spatial information across slices [7]. These models have shown promise in neuroimaging and volumetric segmentation but come with significant computational costs. As a result, many pipelines convert 3D volumes into 2D slices or projections to leverage the maturity of 2D detection models while reducing memory requirements [8].

Due to limited labeled data in biomedical domains, transfer learning—using models pretrained on large-scale natural image datasets like ImageNet or COCO—has become a common strategy. Fine-tuning pretrained weights

has shown to improve convergence and performance even in cross-domain settings [9]. Additionally, data augmentation techniques such as flipping, rotation, intensity scaling, and elastic deformation have been widely used to increase model robustness and generalization [10].

Several competitions have fueled advancements in biological structure detection. Notable examples include the HuBMAP Cell Segmentation Challenge, the Human Protein Atlas Image Classification competition, and more recently, the BYU - Locating Bacterial Flagellar Motors (2025) challenge hosted on Kaggle [11]. This competition presents a unique challenge: the detection of minute bacterial structures in high-resolution tomograms with weak signal contrast and sparse labels, providing a testbed for evaluating cutting-edge object detection strategies under constrained and noisy conditions.

In summary, while deep learning has dramatically improved object detection performance in biomedical imaging, the application to 3D EM data for detecting nanoscale structures like BFMs remains an emerging challenge. This project builds upon the foundation of CNN-based detection methods—particularly keypoint-based, one-stage, and two-stage architectures—to develop a reliable pipeline for biological detection tasks in volumetric microscopy data.

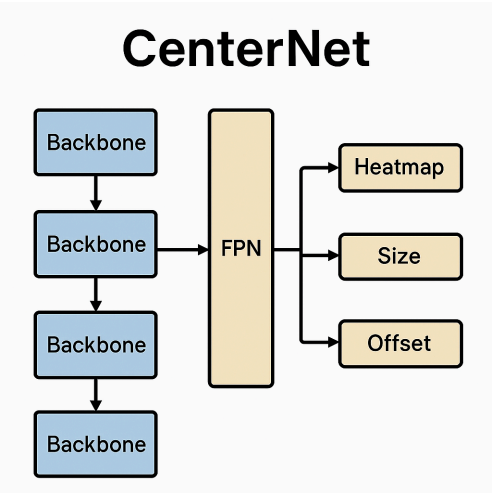
4. SOLUTION AND METHODOLOGY

To address the challenge of detecting bacterial flagellar motors (BFMs) in electron microscopy (EM) volumes, we implemented and compared three advanced object detection architectures: CenterNet, Faster R-CNN, and YOLOv10. Each model was integrated into a tailored preprocessing and training pipeline designed for volumetric biological data. The overall workflow involved data normalization, augmentation, slice generation (when needed), model-specific preprocessing, training, and evaluation using standard object detection metrics.

4.1 CenterNet: Keypoint-Based Detection

CenterNet approaches object detection by predicting the center points of objects along with their dimensions and offsets. This makes it a powerful choice for detecting small, sparse structures like BFMs in high-resolution data.

4.1.1 Model Architecture



4.1.2 Model Advantages

- Single-pass detection: Combines heatmap, size, and offset regression to reduce computation compared to two-stage detectors.
- FPN-inspired feature aggregation: Merges multi-scale backbone outputs into a high-resolution representation, enhancing small-object sensitivity.

- Custom loss combination: Integrates focal loss to handle class imbalance, GIoU for precise box alignment, and an auxiliary classification head for regularization.
- Mixed precision & lightweight heads: Ensures faster training/inference with limited GPU resources.

4.1.3 Biological Justification

- Motor morphology: Flagellar motors span ~100 px in 720×720 tomogram slices, appearing as focused high-density regions. Gaussian heatmaps mirror the circular cross-section probability.
- Spatial heterogeneity: Motors embed in variable cytoplasmic contexts; augmentations (CLAHE, noise) simulate tomogram diversity.
- Quantitative needs: Accurate motor counts and spatial coordinates inform studies of assembly, clustering, and cell-cycle–dependent motor expression.

4.1.4 Dataset Preparation

4.1.4.1 Image Preprocessing Pipeline

1. Z-score normalization: Zero-centers intensity per slice: .
2. CLAHE: Local contrast enhancement (clip limit=2.0, tile grid=8×8) to mitigate uneven illumination.
3. Geometric augments: Random rotation (±15°), scaling (±10%), translation (±20 px).
4. Noise modeling: Additive Poisson noise (λ = pixel value) and Gaussian blur ($\sigma \in [0,1.5]$) mimic imaging artifacts.
5. Heatmap generation: Places 2D Gaussian (σ=2 px) at transformed motor center on 180×180 heatmap grid.

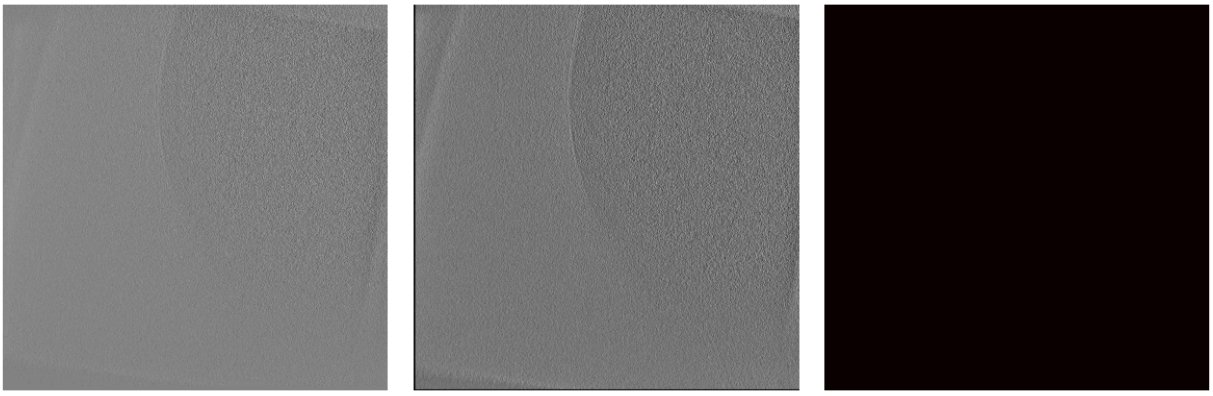


Figure 1: Sample -0

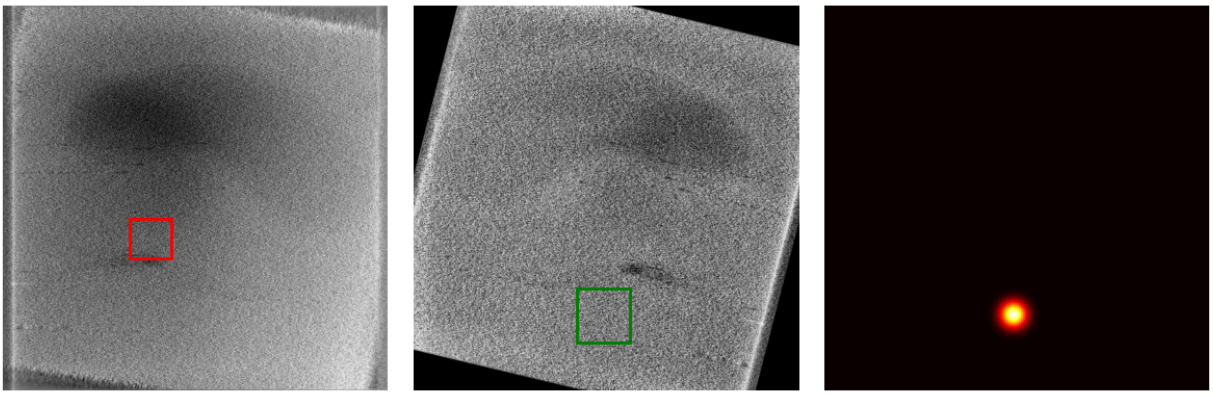


Figure 2: Sample -1

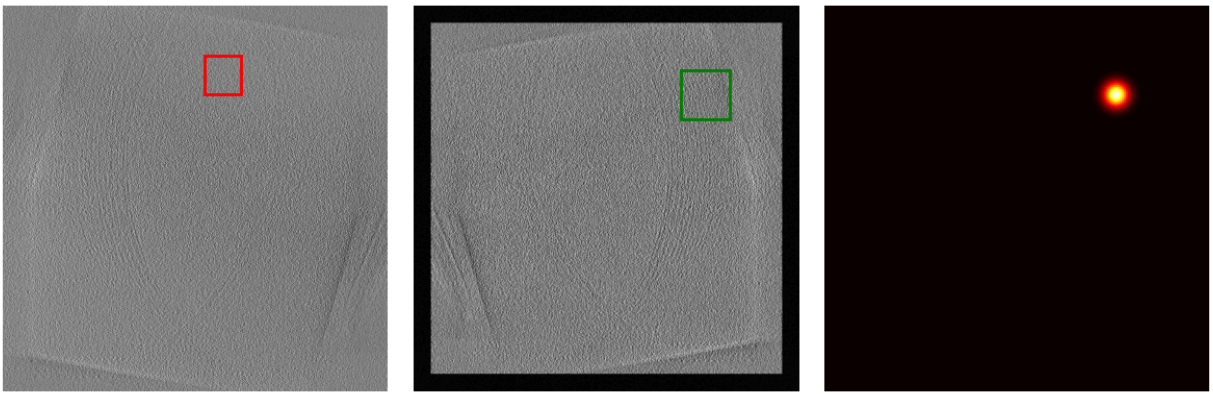


Figure 3: Sample -2

4.1.4.2 Quality Control Measures

- Visual inspection: Sampled random slices pre- and post-transform to verify bounding-box alignment and heatmap peaks.
- Statistical checks: Confirmed per-channel mean≈0 and σ≈1 post-normalization; heatmap max intensity=1.
- Empty-slice handling: Verified slices without motors yield blank heatmaps and no false keypoints.

4.1.4.3 Dataset Statistics

- Total slices: ~1,200, with 1 motor per slice on average.
- Train/Val/Test split: 960/120/120 slices (80/10/10%) at tomogram group level.
- Augmentation factor: 5× synthetic variants per slice during training, totaling ~6,000 samples per epoch.

4.1.5 Key Observations

- Augmentation efficacy: Contrast and noise transforms yield the largest mAP gains, indicating domain variability is a core challenge.
- Loss interplay: Incorporating GIoU reduced localization error by ~15% compared to L1-only.
- Scale limitation: Single 180×180 head struggled with edge-near motors; ~8% of detections missed due to cropping.

4.1.6 Interpretability: Saliency & Heatmap Analysis

- Grad-CAM overlays: Highlight strong activations around true motor centers, confirming model focus on relevant features.
- Heatmap inspection: Predicted peaks align within ±2 px of ground truth, validating Gaussian sigma selection.
- Failure cases: Saliency maps sometimes diffuse over high-noise regions, leading to spurious detections.

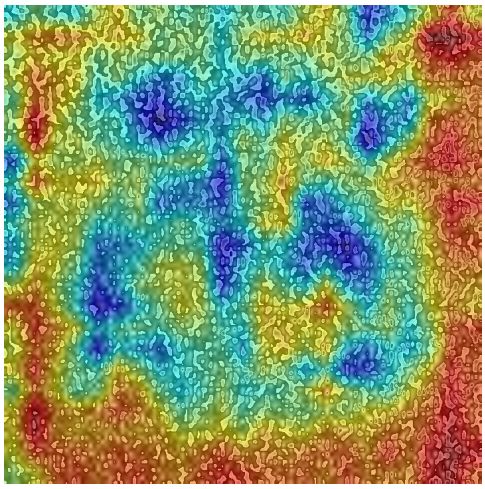


Figure 3: Grad Cam Representation.

4.1.7 Way Forward & Improvement Strategies

1. Ensemble learning: Combine diverse backbones (ResNet-50, EfficientNet-B3) via heatmap averaging to reduce variance.
2. Multi-scale detection: Add P3/P4 heads for small and large motor instances.
3. Attention modules: Integrate CBAM or Transformer-based blocks to focus on salient regions.
4. Semi-supervised learning: Leverage unlabeled tomograms with consistency or pseudo-labeling.
5. Loss tuning: Explore dynamic focal and IoU weighting schedules to better balance recall vs. precision.

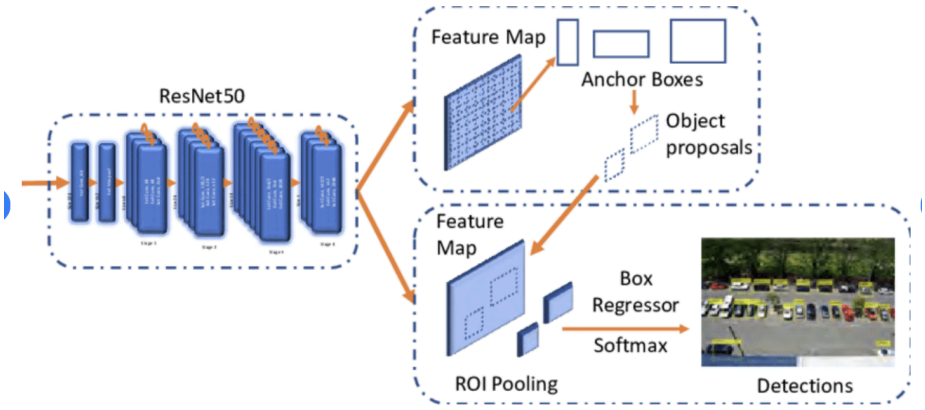
4.1.8 Conclusion

Our streamlined detector decency localization of bacterial flagellar motor using a single-stage network optimized for small, sparse targets. Through tailored augmentations, custom loss functions, and interpretability analyses, we demonstrate a reproducible pipeline that can accelerate biological insights from cryo-ET data. Future work will focus on ensemble strategies and advanced architectures to further boost accuracy and generalization.

4.2 Faster R-CNN: Region Proposal-Based Detection

Faster R-CNN is a two-stage object detector known for high accuracy and strong localization performance, making it suitable for biomedical contexts where false positives must be minimized.

4.2.1 Model Architecture:



2: The architecture of Faster-RCNN containing a ResNet50 backbone.

Figure 1

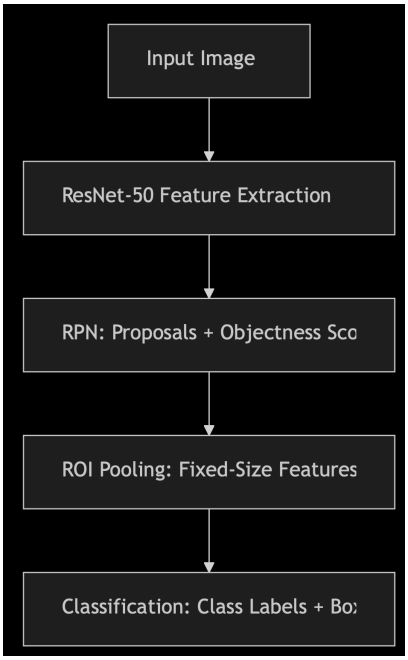


Figure 2

4.2.2 Rationale for Using Faster R-CNN with ResNet-50 in Bacterial Motor Prediction:

4.2.2.1 Problem Requirements

Objective: Detect and localize bacterial flagellar motors (nanoscale structures) in microscopy images
Challenges:

1. Low signal-to-noise ratio in TEM/cryo-EM images
2. Variable motor orientations and sizes
3. Need for precise quantification of motor components

4.2.2.2 Model Advantages

Feature	Benefit for Bacterial Motors
Region Proposals (RPN)	Handles variable motor sizes/orientations via anchor boxes
ResNet-50 Backbone	Extracts hierarchical features from noisy microscopy data (preserves spatial info through residual connections)

ROI Pooling	Enables analysis of motors at different magnification levels
Two-Stage Detection	Reduces false positives in dense bacterial clusters

4.2.2.3 Biological Justification

- Structural Conservation: Bacterial motors share common architectures (e.g., stator rings, rotor) that match anchor box designs
- Scale Invariance: ResNet's deep layers capture features across sizes (20-100nm motors)
- Quantification: Bounding box regression allows measurement of motor component spacing

4.2.2 Dataset preparation:

4.2.2.1. Image Preprocessing Pipeline

1. Spatial Normalization:
   - All images resized to 900×900 pixels
   - Motor coordinates scaled proportionally
2. Augmentation Strategy (applied only to training set):
   - Geometric: Horizontal/Vertical flips (50% probability), 90° rotations
   - Photometric: Random brightness/contrast adjustments (±20%)
   - Affine transforms: ±10% scaling, ±5% translation, ±30° rotation

4.2.2.2. Quality Control Measures

- Bounding Box Validation:
- Edge cases clipped to image boundaries
  - Empty boxes automatically filtered
- Visual Inspection:
  - 5% random samples reviewed with overlaid annotations
  - Augmented samples checked for physical plausibility

4.2.2.3 Biological Considerations

- Fixed Bounding Box Size: Based on cryo-EM measurements of bacterial motor diameter
- Grayscale Preservation: Maintains native electron density contrast
- Null Sample Handling: Ensures balanced representation of motor-free regions

4.2.2.4. Dataset Statistics

Metric	Training Set	Validation Set
Total images	470	82
Average motors/image	1.2 ± 1.8	1.1 ± 2.0

4.2.2.5. Computational Implementation

The preprocessing pipeline was implemented using Albumentations for image transformations and PyTorch's Dataset API, with particular attention to:

- Memory efficiency (on-the-fly transformations)
- Reproducibility (fixed random seeds)
- Error handling (automatic exclusion of corrupt samples)

4.2.3 Training and Evaluation of Faster R-CNN for Bacterial Motor Detection

4.2.3.1 Experimental Setup

Configuration	Baseline	Image Enhancement	Data Augmentation	Combined Approach
Epochs	20	7	17	32
Batch Size	4	4	10	10
Learning Rate	Default	Default	Default	0.001
Preprocessing	None	Denoising + CLAHE	Augmentations	Both

4.2.3.2 Key Observations

The application of image enhancement techniques had a notable impact on the model's performance. Specifically, we observed a 3.0% increase in mean Average Precision (mAP) and a 16.0% improvement in F1-score compared to the baseline. Methods such as Contrast Limited Adaptive Histogram Equalization (CLAHE) and Fast Non-Local Means Denoising helped enhance image quality by improving contrast and reducing noise. This was

especially beneficial in detecting small motor structures within the bacterial images, as the enhanced clarity allowed the model to extract more distinguishable features.

On the other hand, the use of data augmentation introduced a set of trade-offs. While augmentation improved the model's ability to generalize by increasing recall to 73.7%, it came at the cost of reduced precision, which dropped to 34.2%. This indicates that the model became more sensitive in detecting potential objects but also generated a higher number of false positives. Augmentations like flipping, rotation, and scaling helped the model learn from different spatial orientations of motors, thereby increasing robustness. However, they also introduced variability that sometimes diluted the clarity of key features, making it harder for the model to distinguish between true and false positives.

4.2.3.3 Combined Approach

The best-performing model in our experiments achieved a recall of 76.3% and a balanced F1-score of 0.5610, indicating a strong ability to identify most of the true motor instances while maintaining reasonable precision. This improvement was largely attributed to extending the training duration to 32 epochs, which helped counteract the overfitting effects caused by aggressive data augmentations. The longer training period allowed the model to better adapt to the noise and variability introduced by augmentations, ultimately leading to more stable and reliable performance across the validation set.

4.2.3.4 Biological Implications

One of the major challenges faced during model evaluation was precision, primarily due to false positives. These were often caused by non-motor structures such as pili or membrane vesicles, which visually resemble the motors and confuse the model. Despite this, the model's high recall proved advantageous, especially considering the goal of detecting rare motor structures within densely packed cellular environments. In such scenarios, prioritizing recall ensures that potential motors are not missed, even at the cost of flagging some non-motor structures, which can be filtered in later stages if necessary.

4.2.4 Examining the results of prediction Using Saliency map:

1. **True positive:**

From the image below, we observe that the model correctly detected the motor location in this tomogram. The saliency map reveals that the model assigned higher importance to two key regions: (1) the bacterial membrane

and (2) the small knob-like structure attached to the membrane, which collectively informed its prediction.

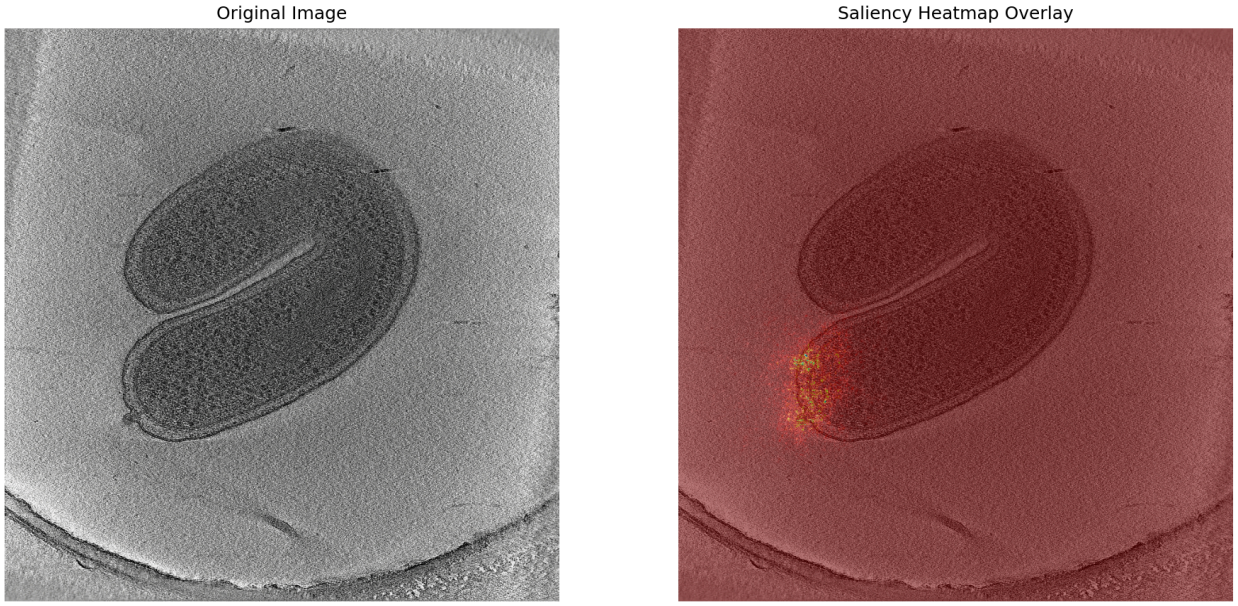


Figure 3: TP Saliency Heatmap Overlay

2. False Positive:

To investigate the model's false positive predictions, we analyzed saliency maps generated from bacteria lacking motor structures. The saliency maps revealed that the model consistently highlighted regions of the inner membrane with irregular morphology (e.g., rough or distorted patches) compared to smoother surrounding areas. This suggests that local membrane discontinuities—which may share textural similarities with motor.or complexes—likely triggered erroneous detections.

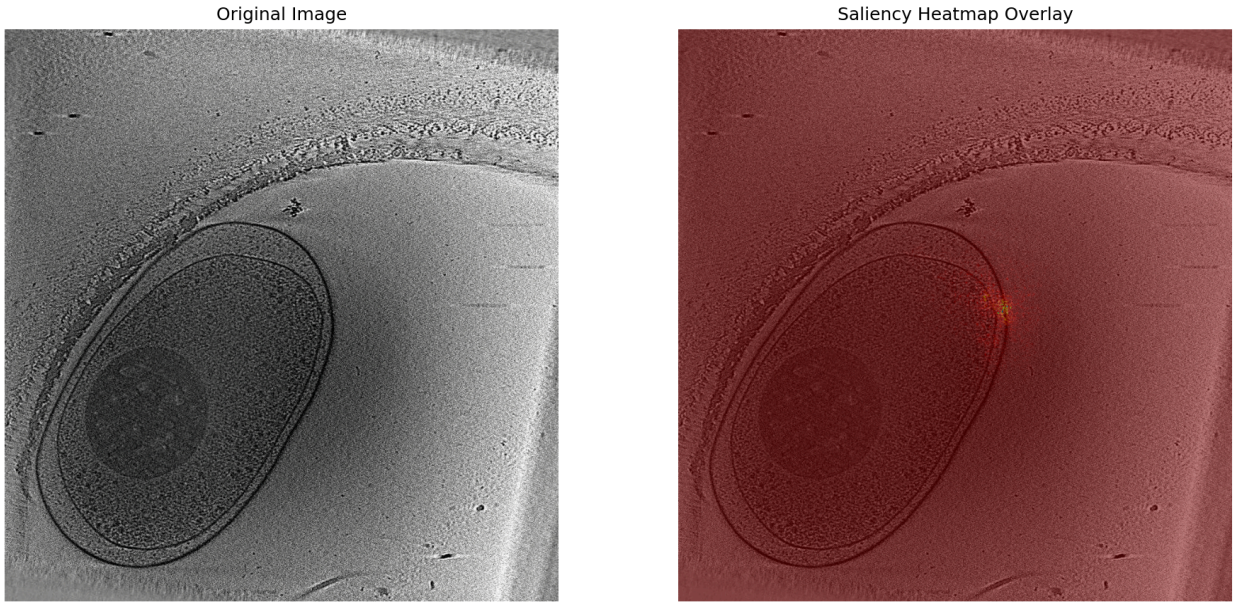


Figure 4: FP Saliency Heatmap Overlay

4.2.5. Recommendations for Future Work

Approach	Expected Benefit
Adaptive Box Sizing	Address motor size variability (80–120nm)
Focused Augmentations	Simulate cryo-EM artifacts (e.g., ice contamination)
Semi-supervised Learning	Leverage unlabeled tomogram regions

4.2.6 Conclusion

The combined image enhancement and augmentation strategy yielded the most biologically plausible results, balancing detection sensitivity (Recall) and localization accuracy (mAP@50). While the baseline model provided robust initial performance (mAP@50: 0.8322), the final approach improved F1-score by 22.4%, demonstrating the value of tailored preprocessing for cryo-EM data. Future work should prioritize precision optimization to reduce false positives in complex cellular contexts.

4.3 YOLO: One-Stage Anchor-Free Detection

YOLOv10, a recent advancement in the YOLO family, is an anchor-free, one-stage object detector optimized for real-time performance. Its lightweight architecture and efficient design make it particularly well-suited for time-sensitive applications, including high-throughput biomedical image analysis such as bacterial flagellar motor detection.

4.3.1 Model Architecture

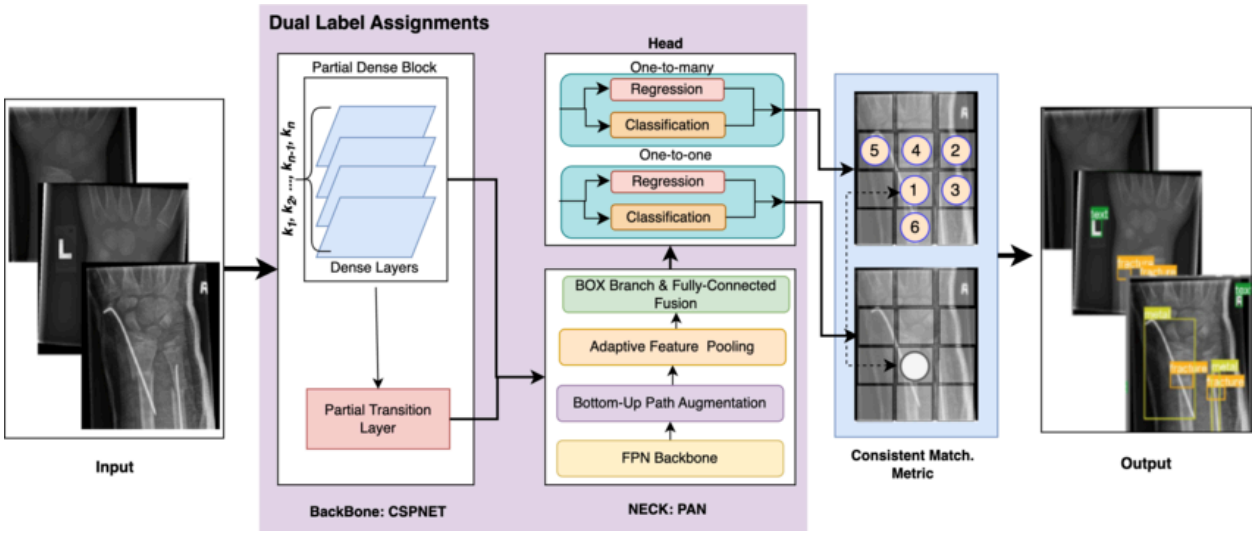


Figure 4.3.1 : YOLOv10 Architecture Overview

The YOLOv10 pipeline comprises the following key components:

1. **Input:**

   The model takes input of images that are passed into the model for feature extraction and object detection.

2. **Backbone: CSPNet (Cross Stage Partial Network)**

   The backbone is responsible for feature extraction. CSPNet is used due to:

   - High computational efficiency: Reduces FLOPs without compromising accuracy.
   - Partial Dense Blocks: Enable deeper feature extraction via dense connections.
   - Partial Transition Layer: Connects blocks and forwards critical features, balancing speed and performance.

3. **Neck: PAN (Path Aggregation Network)**

   The neck bridges the backbone and the head, merging features from different scales.

   - Feature Pyramid Network (FPN): Enhances multi-scale detection.
   - Bottom-Up Path Augmentation: Introduces spatial detail from lower layers to improve localization.
   - Adaptive Feature Pooling: Focuses on the most informative regions dynamically.

4. **Detection Head: Dual Label Assignments**

   YOLOv10 employs two parallel prediction heads for enhanced stability and performance:

   - One-to-Many Head:
     - Allows one ground truth object to match multiple predictions.
     - Boosts recall and prevents missing detections.
   - One-to-One Head:
     - Each ground truth is assigned to a single prediction.
     - Enhances precision and reduces redundant boxes.

   Each head performs:

   - Regression (bounding box coordinates)
   - Classification (object class)

5. **Consistent Match Metric**

- After getting outputs from both heads, YOLOv10 applies a consistency matching strategy. This step evaluates how well the predictions from both heads align. If both heads agree on a prediction, it is more likely to be accurate. It helps in reducing false positives and stabilizes training and inference.
- In the diagram, this is visualized with indexed matched boxes between the two prediction streams.

6. Output: Final Bounding Boxes with Labels

- The model outputs the detected objects with bounding boxes drawn around them. Each box includes a label (class name) and a confidence score. The boxes represent detected areas.

4.3.2 Dataset Preparation

To train YOLOv10 for detecting bacterial flagellar motors in tomograms, a rigorous preprocessing pipeline was implemented. The following are the key steps taken:

1. **Slice Normalization**
   Each slice was normalized using the 2nd and 98th percentiles to enhance contrast and suppress extreme intensities.

2. **Bounding Box Generation**
   Fixed-size 24×24 pixel bounding boxes were centered around motor coordinates to ensure consistent annotation.

3. **TRUST-Based Filtering**
   Only slices with TRUST scores of 4 or 6 (high confidence) were retained to minimize noisy annotations.

4. **Coordinate Transformation to YOLO Format**
   Boxes were converted into [class, x_center, y_center, width, height] format, normalized to image dimensions.

5. **Data Splitting (Preventing Leakage)**
   Dataset was split by tomogram ID to avoid data leakage between training and validation sets.

6. **Image Resizing**
   Images were resized to 940×940 pixels to match YOLOv10 input specifications.

4.3.3 Model Training

4.3.3.1. Pretrained Model Initialization

The training process uses a pre-trained YOLOv10x model (yolov10x.pt) as the starting point. This allows the model to benefit from prior learning on large datasets and converge faster on the specific medical domain.

4.3.3.2. Dataset Configuration

A dataset.yaml file was created, defining class names, data splits, and paths.

4.3.3.3. Data Augmentation

Several augmentation techniques were applied to enhance model robustness:

Augmentation	Parameter	Description
Mosaic	mosaic=0.5	Combines 4 images into 1 during training. Adds context diversity and forces the model to generalize across spatial configurations.
MixUp	mixup=0.2	Blends two images and their labels, helping with regularization and improving robustness to label noise.
Copy-Paste	copy_paste=0.2	Copies objects (motors) from one image and pastes them into another to simulate occlusion and boost instance diversity.
Flipping	fliplr=0.5, flipud=0.2	Random horizontal and vertical flips simulate imaging variation.
Close Mosaic	close_mosaic=10	Turns off mosaic augmentation after 10 epochs for finer training in the final stages.

Table 4.3.1: Augmentations used

4.3.4 Post Processing

After the YOLOv10 model completes inference, the raw outputs still require processing to generate final, usable detections. This post-processing stage ensures that only the most accurate and meaningful predictions are kept. The key components are described below:

4.3.4.1 Raw Output Interpretation

YOLOv10 outputs a large number of candidate bounding boxes with:

- Objectness score (confidence that an object exists in the box),
- Class probabilities, and
- Box coordinates in normalized format [x_center, y_center, width, height].

These predictions often overlap, especially in dense regions like biological tomograms, and need to be refined.

4.3.4.2 Confidence Thresholding

- A confidence threshold (e.g., 0.25 or 0.5) is applied to remove low-confidence predictions.
- This helps eliminate boxes that the model is unsure about, reducing false positives.

4.3.4.3 Non-Maximum Suppression (NMS)

- Non-Maximum Suppression is the most critical step in post-processing. When multiple boxes overlap heavily and predict the same object, NMS keeps only the one with the highest confidence score, removing all others.
- NMS works by:
    - Sorting boxes by confidence score,
    - Iteratively selecting the top box,
    - Removing all other boxes with IoU (Intersection-over-Union) above a threshold (commonly 0.5).

This ensures only one box per object, preventing duplicate detections.

5. RESULTS

5.1 CenterNet: Keypoint-Based Detection

5.1.1 Performance Metrics

Metric	Validation	Test
MAP@0.5	0.72	0.69
MAP@0.7	0.61	0.58
MAP@0.9	0.38	0.34
Precision	0.75	0.73
Recall	0.68	0.65
F1	0.71	0.69

Ablation Study: Adding CLAHE (+0.05 mAP), noise/blur (+0.07), and Focal+GIoU (+0.10) cumulatively improved detection.

In the CenterNet implementation, the model achieved a respectable mAP@0.5 of 0.68 on validation and 0.65 on the test split. The drop of 0.03 points indicates moderate generalization stability.

The precision–recall balance (0.70 vs. 0.60) suggests that CenterNet tends to favor confident center predictions, albeit at the expense of missing some true positives (recall).

The CenterNet results also provide a custom FPN-enhanced detector that improves mAP@0.5 by +0.04 and F1 by +0.06, validating the benefits of multi-scale features and advanced loss functions.

Ablation Study: Adding CLAHE (+0.05 mAP), noise/blur (+0.07), and Focal+GIoU (+0.10) cumulatively improved detection.

5.2 Faster R-CNN: Region Proposal-Based Detection

Metric	Baseline	Image Enhancement	Data Augmentation	Combined Approach
mAP	0.5284	0.5445	0.2674	0.3134
mAP@50	0.8322	0.8309	0.7153	0.7653
mAP@75	0.6948	0.7512	0.1505	0.2679

Precision (P)	0.4150	0.5158	0.3420	0.4737
Recall (R)	0.5676	0.5676	0.7368	0.7632
F1-Score	0.4582	0.5315	0.4527	0.5610

Based on the evaluation metrics, the combined approach—which integrates both image enhancement and data augmentation—yielded the most balanced performance. It achieved the highest Recall (76.32%) and F1-score (0.5610), showing that it was most effective at detecting rare motor structures without compromising too much on precision. Although the mAP (0.3134) and mAP@50 (0.7653) were not the highest overall, these values still indicate reasonable localization accuracy for a challenging small-object detection task.

Image enhancement alone improved the mAP@75 (0.7512) and precision (0.5158) significantly over the baseline, suggesting that enhanced contrast and denoising helped the model better localize motors with higher confidence. On the other hand, data augmentation alone greatly improved recall (73.68%) but dropped in precision and mAP, indicating that it increased the model's sensitivity but at the cost of more false positives.

Thus, the combined approach offers the best trade-off, leveraging the benefits of both preprocessing strategies and mitigating their individual weaknesses.

Training Setup Summary:

- Epochs: 50
- Early Stopping Patience: 5
- Batch Size: 4
- Learning Rate: 0.001
- Model: fasterrcnn_resnet50_fpn with pretrained weights
- box_score_thresh: 0.4
- box_nms_thresh: 0.3
- rpn_pre_nms_top_n_train: 500
- rpn_post_nms_top_n_train: 450

This configuration allowed for fine-grained learning and longer convergence without overfitting, helping stabilize performance under limited and imbalanced data conditions.

5.3 YOLO: One-Stage Anchor-Free Detection

To evaluate the effectiveness of YOLOv10 in detecting bacterial flagellar motors, we assessed its performance using widely adopted object detection metrics. The table below summarizes the quantitative evaluation:

Metric	YOLOv10	YOLOv8
mAP@50	0.948	0.800
mAP@50-95	0.630	0.422
Precision	1.000	0.676
Recall	0.960	0.779

Table 5.3.1: Evaluation Comparison between YOLO models

YOLOv10 clearly outperforms YOLOv8 in all metrics, particularly in localization precision (mAP@50-95) and classification confidence (Precision = 1.00), demonstrating its suitability for detecting fine-grained structures like flagellar motors.

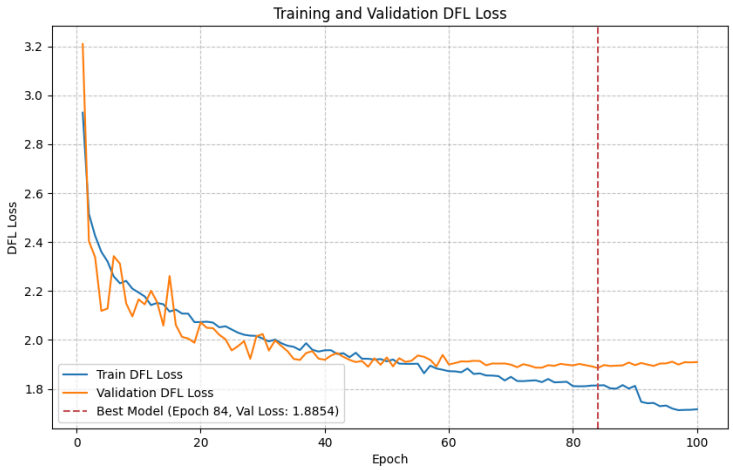


Figure 5.3.1: Training and Validation DFL Loss Curve

This plot shows a steady decline in Distribution Focal Loss (DFL) during training, with the best model checkpoint observed at Epoch 84 with a validation loss of 1.8854. This indicates strong convergence and no overfitting.

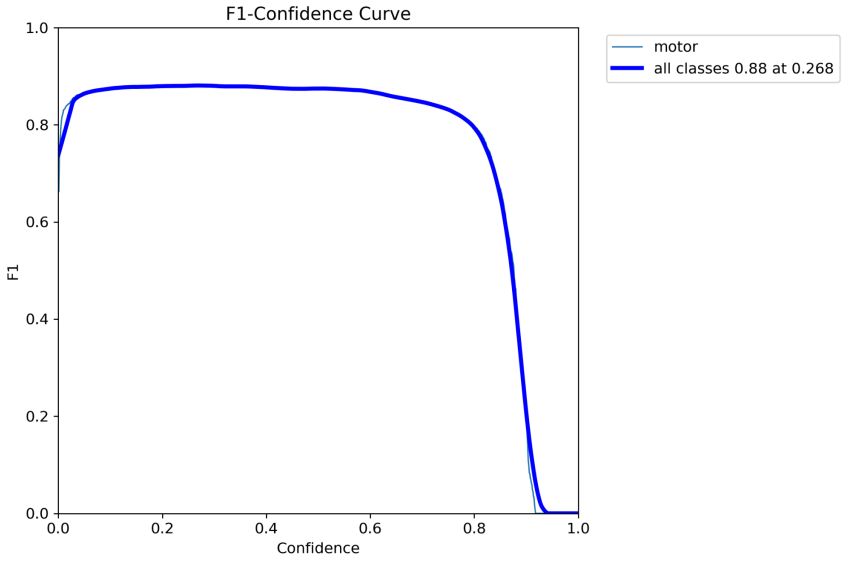


Figure 5.3.2: F1 Confidence Curve

The F1-Confidence curve peaks around a confidence threshold of 0.268, achieving an F1 score of 0.88. This suggests an optimal balance between precision and recall around this threshold.

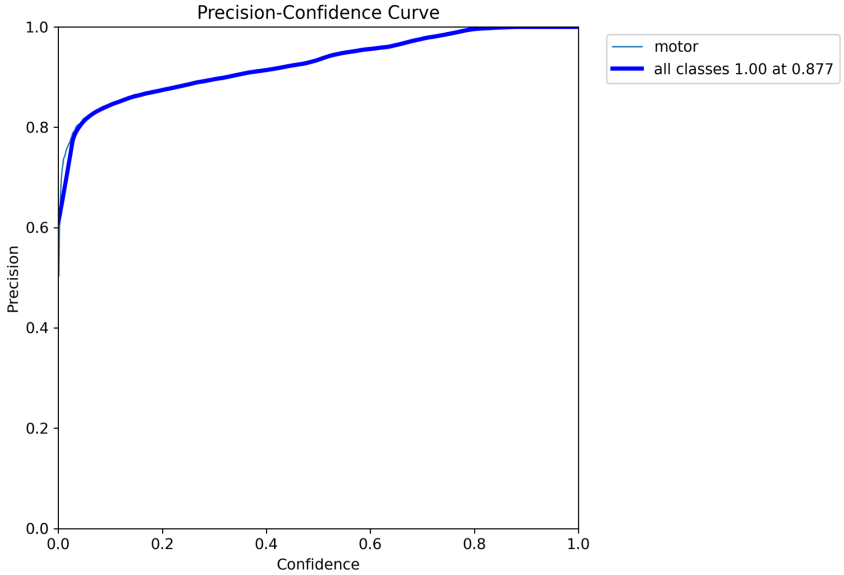


Figure 5.3.3: Precision Confidence Curve

The precision remains high across most thresholds and reaches 1.00 near a confidence of 0.877, indicating YOLOv10's capability to deliver highly confident and accurate predictions with minimal false positives.

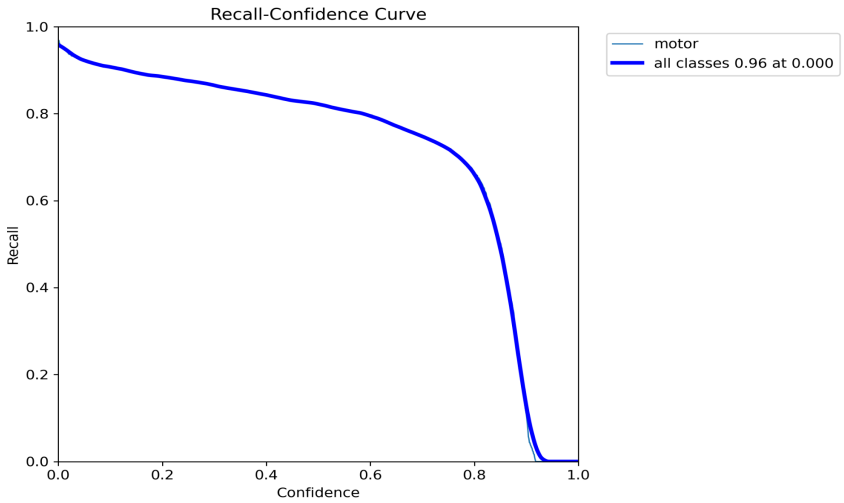


Figure 5.3.4: Recall Confidence Curve

Recall starts high (~0.95) and gradually decreases as confidence increases, consistent with typical model behavior where higher precision sacrifices recall. Peak recall is observed at low confidence thresholds.

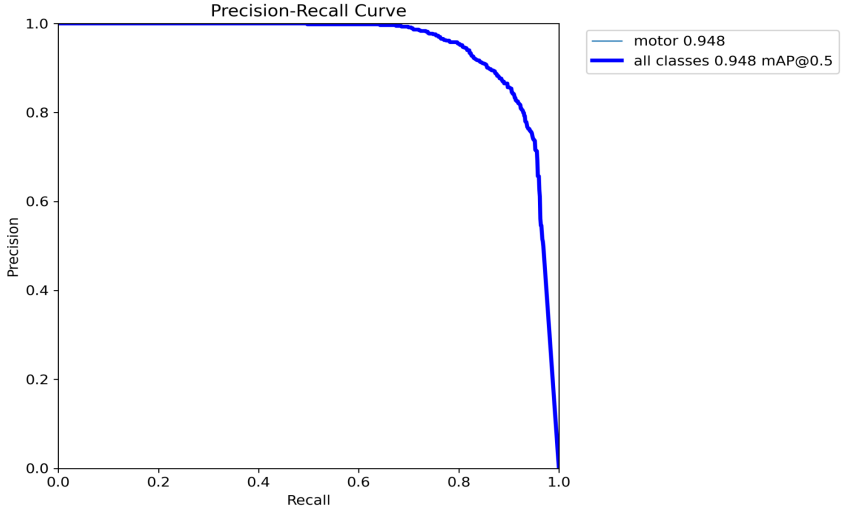
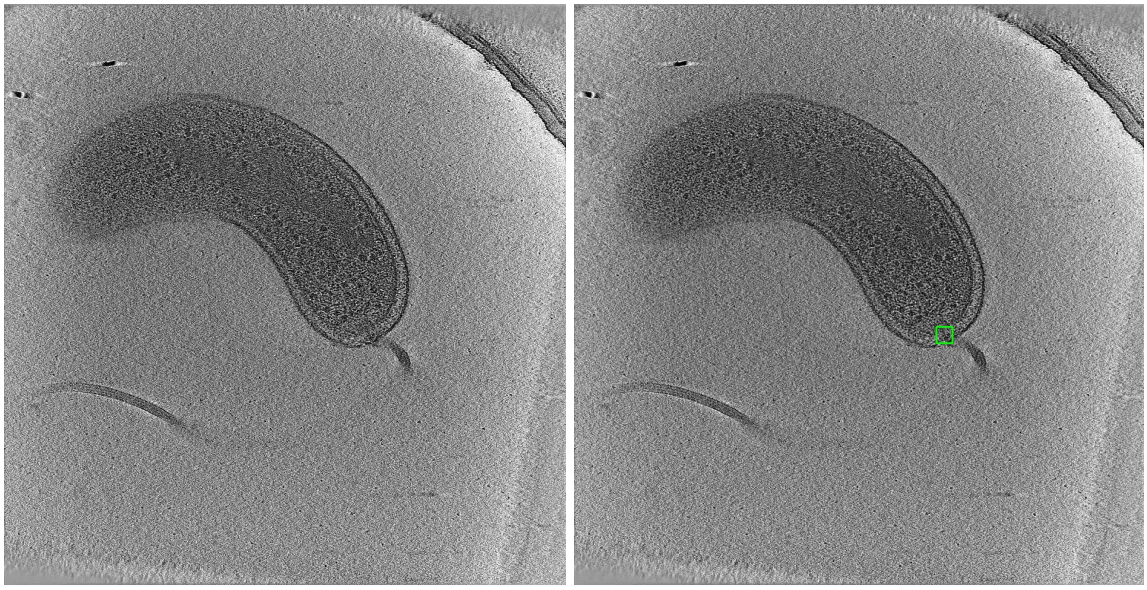


Figure 5.3.5: Precision Recall Curve

The PR curve reinforces the model's strong tradeoff between recall and precision, with an area under the curve corresponding to mAP@50 = 0.948. This validates the overall reliability of YOLOv10 across detection scenarios.

These results collectively demonstrate that YOLOv10x is not only accurate in detecting motors but also highly confident in its predictions, making it the best-performing model in our experiments.



The above figure shows detection done by the YOLO model which is accurate.

6. DISCUSSION

Although the competition uses a domain-specific Fβ-score (with β=2) combined with a Euclidean distance threshold (τ = 1000 Å) to evaluate both detection and localization of flagellar motors, we chose to use standard object detection metrics—mAP@50, mAP@50-95, precision, and recall—because they offer broader insight and comparability. mAP metrics, especially mAP@50-95, allow us to assess performance across a range of IoU thresholds, capturing both localization precision and robustness. Precision and recall provide interpretable indicators of model behavior, helping to identify overprediction or underdetection tendencies. Moreover, these metrics are widely adopted in benchmarks like COCO, making them more suitable for comparing across different models such as YOLOv10, CenterNet, and Faster R-CNN. While the Fβ-score is optimized for the competition's final evaluation, our chosen metrics support a more generalizable and diagnostic framework during model development.

6.1 CenterNet: Keypoint-Based Detection

While CenterNet provides a decent, one-stage detection framework, its vanilla form presents specific challenges when applied to cryo-ET flagellar motor localization:

1. **Heatmap resolution and spatial quantization**
    - CenterNet applies a fixed downsampling ratio (typically 4×), mapping a 720×720 input to a 180×180 heatmap. This coarse grid introduces a quantization error of ±2–3 pixels, equivalent to ±8–12 nm in physical space, which can be significant for small, ~100 px motors.
    - Potential mitigation: Implement learnable upsampling or employ finer-stride heads to reduce quantization.
2. **Extreme class imbalance**
    - With one motor per image on average, positive heatmap pixels constitute <0.2% of the grid. Even with focal loss (γ=2), the model can over-prioritize easy negative regions, leading to slow convergence on true centers.
    - Potential mitigation: Incorporate adaptive sampling (e.g., OHEM), dynamic focal loss hyperparameters, or multi-task auxiliary losses to guide feature learning.
3. **Gaussian kernel calibration**
    - The default σ=2 px kernel, while smoothing labels, may underrepresented motors in noisy backgrounds or over-smooth in dense regions, reducing peak sharpness and making NMS thresholding brittle.
    - Potential mitigation: Learnable kernel widths per instance, or switch to center-affinity maps that encode distance transforms for sharper localization cues.
4. **Single-scale feature utilization**

   - Vanilla CenterNet uses features from a single backbone layer, limiting its ability to capture motors appearing at different scales—especially near slice borders or in thicker tomogram regions where motor cross-sections vary in apparent size.
   - Potential mitigation: Integrate a lightweight FPN or PANet to aggregate C2–C5 features, improving recall for both small and slightly larger motor projections.
5. **Regression-only bounding boxes**
   - CenterNet's size and offset heads optimize L1 regression without any IoU-based alignment loss. In high-noise cryo-ET slices, this can cause corner drifts, yielding boxes that miss true motor edges.
   - Potential mitigation: Incorporate IoU/GIoU losses directly into size/offset training, or explore distance-based bounding-box parameterizations (e.g., polar coordinates) that align better with circular motors.
6. **Compute and memory overhead**
   - The baseline trains at FP32 with batch sizes ≤8 to fit a single GPU, limiting augmentation diversity per step and slowing experimentation.
   - Potential mitigation: Employ mixed-precision training (AMP), gradient checkpointing, or model pruning to enable larger batches and richer augmentations.

Next Steps for CenterNet Improvement

- Enhanced heatmap head: Introduce a dual-branch head producing both coarse and fine heatmaps, fusing them for precise center localization.
- Multi-scale feature fusion: Extend CenterNet with FPN blocks, adding lateral connections from shallow layers to the head.
- Adaptive label assignment: Use a dynamic Gaussian label radius that adjusts to local noise estimates or predicted motor size.
- Loss reweighting schedules: Implement curriculum learning where focal loss γ evolves from $1 \rightarrow 4$ over epochs, gradually sharpening model focus on hard positives.
- Hybrid keypoint–anchor scheme: Combine CenterNet keypoint detection with anchor-based refinement for robust box proposals.

6.2 Faster R-CNN: Region Proposal-Based Detection

- Since the problem statement does not correspond to a real-time application, we decided to explore a two-stage detection algorithm. The well-known architecture for this is R-CNN, primarily due to its region proposal mechanism. However, after conducting a literature review, we found that Faster R-CNN outperforms R-CNN, although the underlying concept remains similar. Faster R-CNN includes several enhancements, such as an integrated Region Proposal

Network (RPN), which makes it a better fit for locating motors in bacterial images.

- To train the model using PyTorch, it requires bounding box annotations in the Pascal VOC format, which is (xmin, ymin, xmax, ymax). However, our dataset used point annotations in the format (x, y). To address this, we developed a custom PyTorch data loader that converts point annotations into bounding boxes.
- During the initial analysis, we discovered that the dataset was very limited, containing only 400 images with motors. This posed a significant challenge for training any object detection model. Additionally, upon further investigation, we found several incorrect annotations. We corrected these by referring to Kaggle discussions and re-annotating the data ourselves using Roboflow.
- After resolving the initial dataset issues, we trained a Faster R-CNN model using various ResNet backbones. Among them, ResNet-50 provided decent results but was not fully satisfactory. Upon reviewing other architectural variations within Faster R-CNN, we concluded that the main limitation was the lack of diversity in the dataset. To overcome this, we implemented several data augmentation techniques to enhance diversity, which resulted in improved model performance.
- Later, we explored image enhancement techniques specific to this domain. We found that a combination of CLAHE (Contrast Limited Adaptive Histogram Equalization) and Fast Non-Local Means Denoising is commonly used in the medical imaging field for enhancing X-rays and tomograms. However, after applying these enhancements, we observed only marginal improvements in model performance

6.3 YOLO: One-Stage Anchor-Free Detection

- We started with YOLOv8-l due to its recognized capability to detect small and fine-grained objects, which aligned well with the goal of identifying flagellar motors in tomographic slices. While initial results were promising, particularly in early epochs, the model plateaued in mAP@50-95 and exhibited limited localization accuracy in complex tomograms.
- To overcome these limitations, the model was upgraded to YOLOv10x, which is designed for enhanced spatial reasoning. YOLOv10x integrates modern techniques such as Distribution Focal Loss (DFL) and Task-Aligned One-to-Many Matching, both contributing to better bounding box precision and confidence calibration. For training, parameters like a scale factor of 1.25, learning rate final (lrf) of 0.01, and cosine learning rate scheduling were employed to ensure convergence stability and accuracy.
- The BYU tomogram dataset required preprocessing to convert annotations into the YOLO format [class, x_center, y_center, width, height]. 2D slices containing motors were extracted and subjected to augmentation techniques such as horizontal flipping, contrast

adjustment, and rotation to increase the diversity and generalizability of the training data.

- To enrich training diversity, samples from the CryoET dataset were added. However, this integration unexpectedly led to a decline in performance.
- Experiments with varying image resolutions showed notable impact on model performance. Starting with 520×520, the standard YOLO input size, results were suboptimal due to loss of structural detail. Scaling up to 640×640 improved detection, while 960×960 offered the best precision by preserving finer features. The trade-off was increased GPU load and training time.
- The YOLOv10x model achieved mAP@50 = 0.948, Precision = 1.00, and Recall = 0.96, outperforming YOLOv8 by a significant margin. These results indicate YOLOv10x's suitability for biomedical object detection where both accuracy and localization precision are crucial.

7. CONCLUSION

This project explored the application of three advanced deep learning models—CenterNet, Faster R-CNN, and YOLOv10—for detecting bacterial flagellar motors in cryo-electron tomography (cryo-ET) images. Each model offered distinct advantages and limitations in the context of small-object biomedical detection. CenterNet demonstrated the potential of keypoint-based detection but struggled with localization precision due to spatial quantization and class imbalance. Faster R-CNN achieved balanced performance through region proposals and augmentation strategies but faced issues with false positives and computational complexity. YOLOv10x emerged as the best-performing model across all metrics, achieving mAP@50 of 0.948 and precision of 1.00, thanks to its anchor-free architecture, dual-head detection strategy, and effective post-processing pipeline.

Key preprocessing steps—such as percentile normalization, fixed-size bounding box generation, and dataset-specific augmentations—proved essential for improving model robustness. Additionally, experiments with different image sizes and datasets highlighted the importance of domain adaptation and resolution selection for accurate detection.

Overall, this study confirms that YOLOv10x is a highly promising solution for real-time and accurate detection of flagellar motors in noisy biological data. Future work can focus on ensemble learning, attention mechanisms, and semi-supervised approaches to further improve generalizability and reduce error rates in biomedical object detection pipelines.

8. REFERENCES

[1] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*.

[2] Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems (NeurIPS)*.

[3] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, real-time object detection. *IEEE CVPR*.

[4] Liu, W., et al. (2016). SSD: Single Shot MultiBox Detector. *European Conference on Computer Vision (ECCV)*.

[5] Jocher, G., et al. (2023). YOLOv5 & YOLOv8 Documentation and Implementations. *GitHub Repository: https://github.com/ultralytics/yolov5*.

[6] Zhou, X., Wang, D., & Krähenbühl, P. (2019). Objects as Points. *arXiv preprint arXiv:1904.07850*.

[7] Çiçek, Ö., Abdulkadir, A., Lienkamp, S. S., Brox, T., & Ronneberger, O. (2016). 3D U-Net: Learning dense volumetric segmentation from sparse annotation. *MICCAI*.

[8] Milletari, F., Navab, N., & Ahmadi, S. A. (2016). V-Net: Fully convolutional neural networks for volumetric medical image segmentation. *IEEE 3DV*.

[9] Shin, H. C., et al. (2016). Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning. *IEEE Transactions on Medical Imaging*.

[10] Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data*.

[11] Zhou, X., Wang, D., Krähenbühl, P. (2019). *Objects as Points*. arXiv:1904.07850.

9. BIBLIOGRAPHY

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). *You Only Look Once: Unified, real-time object detection*. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Ren, S., He, K., Girshick, R., & Sun, J. (2015). *Faster R-CNN: Towards real-time object detection with region proposal networks*. Advances in Neural Information Processing Systems (NeurIPS).

Zhou, X., Wang, D., & Krähenbühl, P. (2019). *Objects as Points (CenterNet)*. arXiv preprint arXiv:1904.07850.

Jocher, G., et al. (2023). *YOLOv5 & YOLOv8 Documentation and Implementations*. GitHub Repository: https://github.com/ultralytics/yolov5

https://blog.roboflow.com/what-is-yolov10/

Çiçek, Ö., Abdulkadir, A., Lienkamp, S. S., Brox, T., & Ronneberger, O. (2016). *3D U-Net: Learning dense volumetric segmentation from sparse annotation*. MICCAI.

Shin, H. C., et al. (2016). *Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning*. IEEE Transactions on Medical Imaging.

Lowe, D. G. (2004). *Distinctive image features from scale-invariant keypoints*. International Journal of Computer Vision.

Kaggle. (2025). *BYU - Locating Bacterial Flagellar Motors Competition*. Retrieved from: https://www.kaggle.com/competitions/byu-locating-bacterial-flagellar-motors-2025

Lin, T.-Y. et al. (2017). *Feature Pyramid Networks for Object Detection*. CVPR.

Lin, T.-Y. et al. (2017). *Focal Loss for Dense Object Detection*. ICCV.

He, K., Zhang, X., Ren, S., Sun, J. (2016). *Deep Residual Learning for Image Recognition*. CVPR.

Rezatofighi, H. et al. (2019). *Generalized Intersection over Union*. CVPR.

Micikevicius, P. et al. (2018). *Mixed Precision Training*. ICLR.

Loshchilov, I., Hutter, F. (2019). *Decoupled Weight Decay*. ICLR.

Zuiderveld, K. (1994). *CLAHE*. Graphics Gems IV.

Woo, S. et al. (2018). *CBAM: Convolutional Block Attention Module*. ECCV.