

CELLCLASSIFICATION

[Github](#)

[Intro](#)

[Workflow](#)

[Results](#)

<https://github.com/Rachel-Ruixuan/CellClassification>

rl4789@nyu.edu

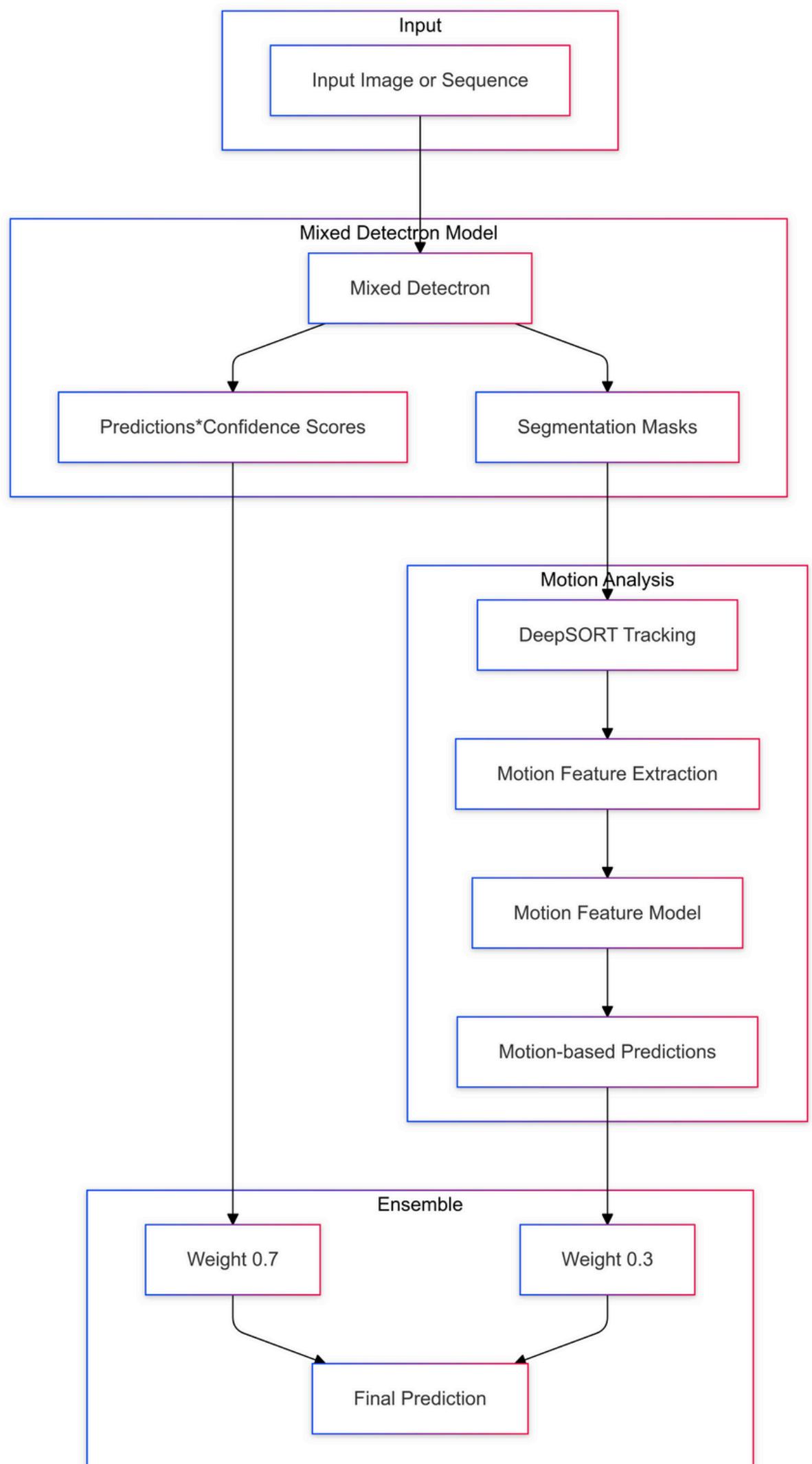
- Primary Goal:
 - Develop an automated system to identify and classify different cell types in microscopy videos
 - Replace time-consuming manual cell identification methods
- Key Challenges:
 - Distinguish between visually similar cell types
 - Capture unique behavioral patterns of different cells
 - Handle dynamic microscopic environments
- Solution Approach:
 - Combine spatial recognition (Detectron2) with motion analysis
 - Create an integrated pipeline for real-time cell tracking and classification
 - Achieve reliable automated classification through dual-model verification

Ensemble Strategy

Final Prediction = $0.7 * \text{Detectron_Prediction} + 0.3 * \text{Motion_Features_Prediction}$

Rationale

- **Detectron (70%)**
 - Strong shape feature extraction
 - Immediate classification capability
 - More reliable for static features
- **Motion Features (30%)**
 - Temporal pattern recognition
 - Behavioral characteristics
 - Complementary dynamic information



Inference Workflow

Phase 1: Frame Processing

- Process each frame to obtain segmentation and tracking data.
- Save all detections and tracking information.

Phase 2: Motion Feature Extraction

- After all frames are processed, extract motion features from the accumulated tracking data.
- Perform inference using the motion feature model.

Phase 3: Ensembling and Visualization

- Combine Detectron2 and motion model predictions.
- Annotate frames with the final ensemble predictions.
- Generate the final video.

1. Detectron-Only Performance

- Baseline accuracy
- Shape-based classification metrics

2. Motion-Features-Only Performance

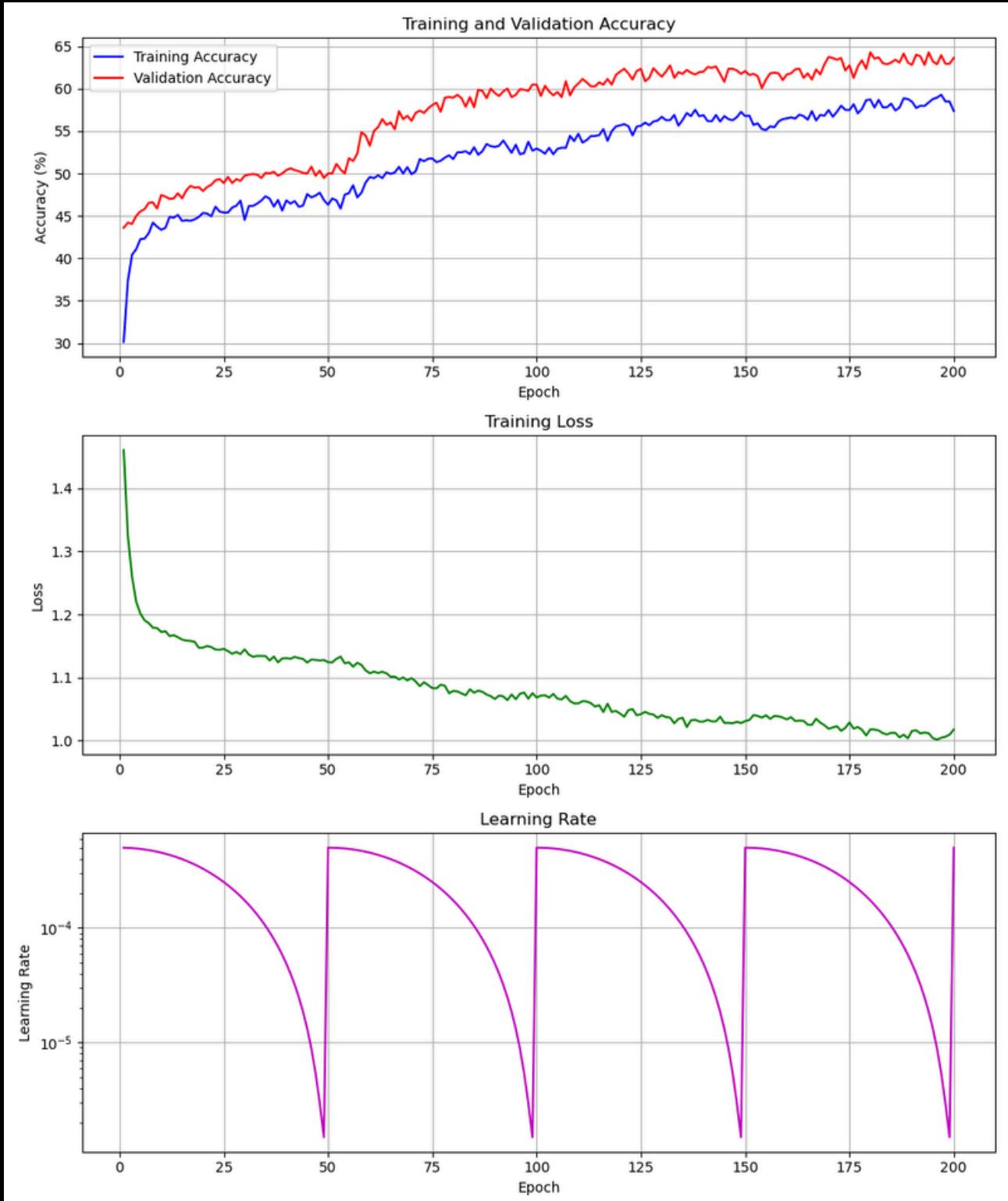
- Temporalpattern recognition
- Movement-based classification accuracy

3. Ensemble Model Performance

- Combined accuracy
- Feature importance analysis
- Confusion matrix comparison

Evaluation Framework Comparative Analysis

- Motion Feature Model :
- Measures classification accuracy (identifying cell types based on movement patterns)
 - Final validation accuracy: ~63-65%
 - Uses temporal features like speed, acceleration, direction changes



- Detectron Model has two types of metrics:
 - Bounding Box Detection (bbox):
 - Overall AP: 59.1%
 - Per-class AP: MDA (70.4%), FB (66.1%), M1 (50.8%), M2 (49.1%)
 - Segmentation (segm):
 - Overall AP: 45.3%
 - Per-class AP: MDA (69.3%), FB (11.6%), M1 (50.2%), M2 (50.3%)

```
[11/26 18:32:13 d2.evaluation.coco_evaluation]: Evaluation results for bbox:  
| AP | AP50 | AP75 | APs | APm | AP1 |  
|:---:|:---:|:---:|:---:|:---:|:---:|  
| 59.093 | 76.246 | 71.749 | 58.139 | 62.647 | 35.192 |  
[11/26 18:32:13 d2.evaluation.coco_evaluation]: Per-category bbox AP:  
| category | AP | category | AP | category | AP |  
|:-----|:-----|:-----|:-----|:-----|:-----|  
| MDA | 70.449 | FB | 66.066 | M1 | 50.798 |  
| M2 | 49.059 | | | | |
```

```
[11/26 18:32:22 d2.evaluation.coco_evaluation]: Evaluation results for segm:  
| AP | AP50 | AP75 | APs | APm | AP1 |  
|:---:|:---:|:---:|:---:|:---:|:---:|  
| 45.348 | 64.427 | 53.917 | 35.570 | 48.350 | 51.980 |  
[11/26 18:32:22 d2.evaluation.coco_evaluation]: Per-category segm AP:  
| category | AP | category | AP | category | AP |  
|:-----|:-----|:-----|:-----|:-----|:-----|  
| MDA | 69.286 | FB | 11.576 | M1 | 50.181 |  
| M2 | 50.347 | | | | |
```

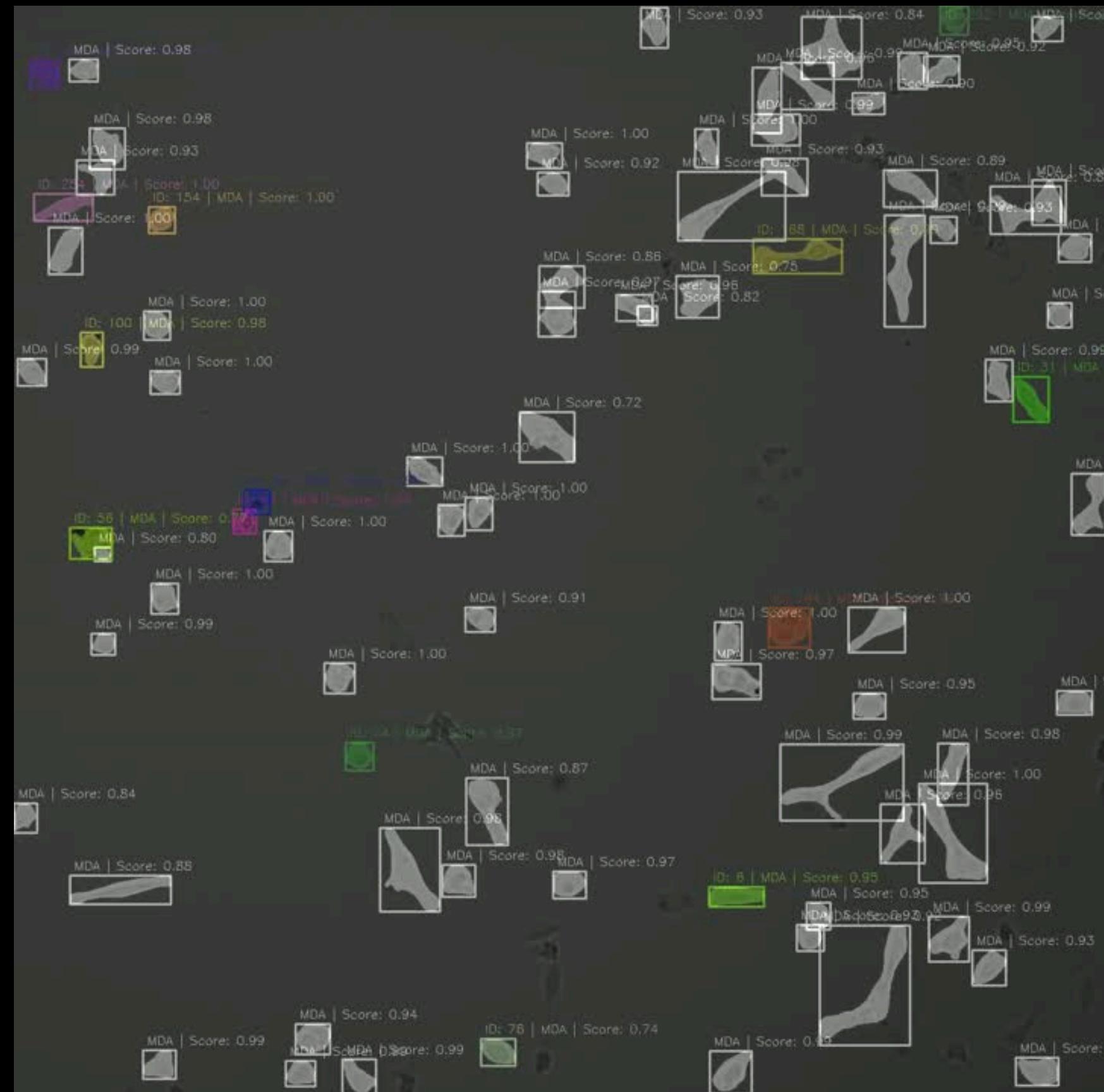
CELL CLASSIFICATION

Github

Intro

Workflow

Results



Next Steps

1. Mixed Detectron Model Evaluation

- Validate per-class accuracy using results from the mixed Detectron2 model.

2. Pipeline Integration

- Output the logits (last layer before classification) of detectron to facilitate ensembling.
- Experiment with motion feature ensembling to enhance classification accuracy.

3. Performance Analysis

- Compare individual vs ensemble results
- Conduct cross-validation studies to evaluate robustness across different datasets.
- Perform error analysis per cell type to identify and address misclassification patterns.