Report: Progress until now

Instead of directly jumping into DL methods, I first tried to apply known c.v methods and create a pipeline to solve the given problem statement. (since they are more faster at execution, and more intuitive to debug)

Problem Statement:

To find the actions of the JCB arm, i.e; (dig, swing, dump)

My Approach:

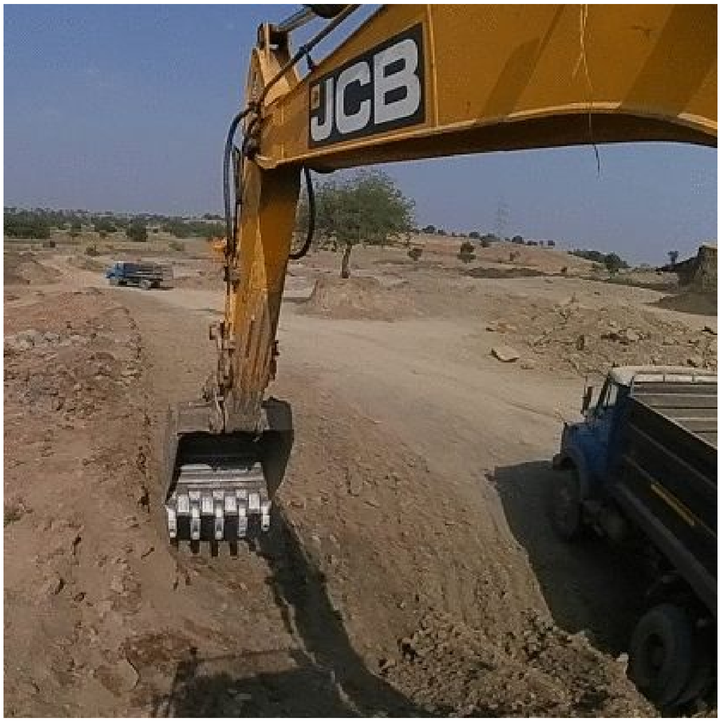
1. Localize the lower part of the JCB arm
2. Compute its optical flow thereby finding the direction of movement of the arm (i.e up, left, etc).

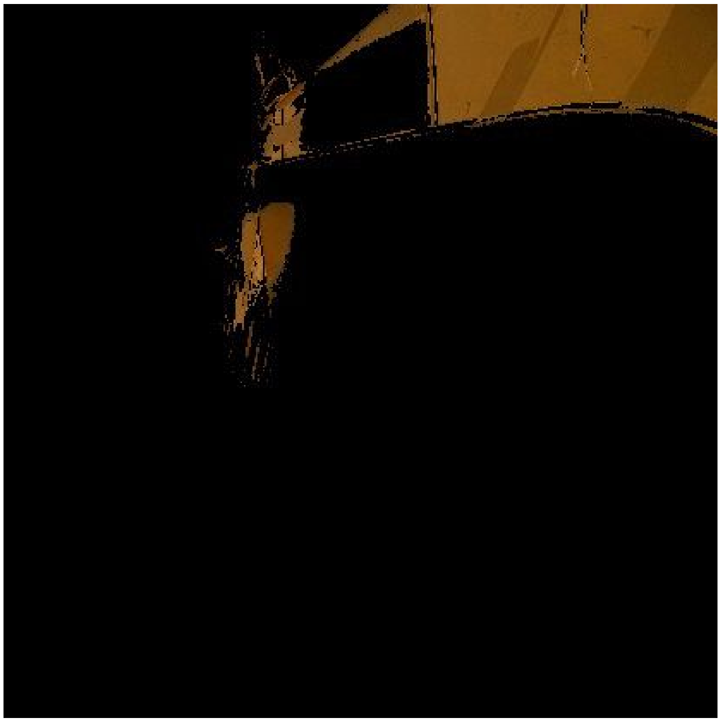
Work Progress:

Iteration 1:

Workflow:

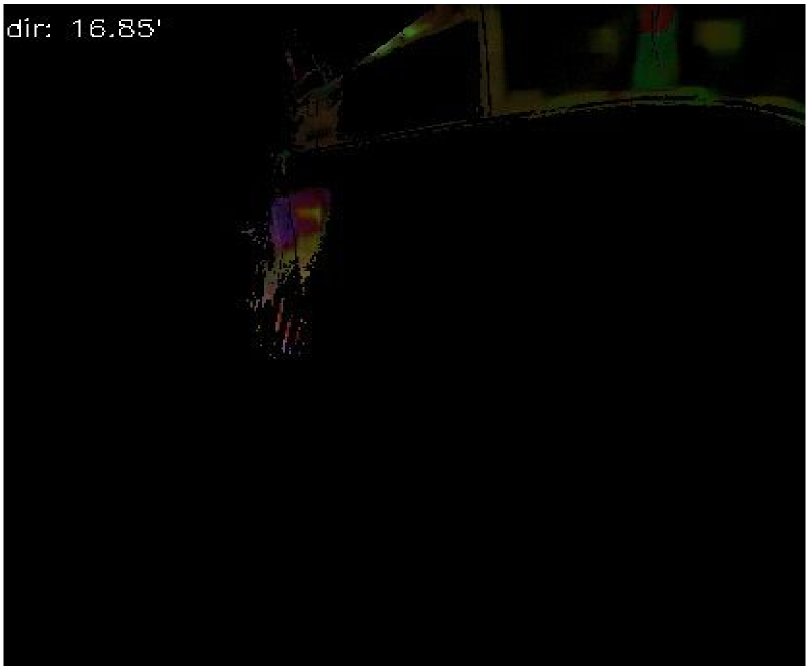
1. Segment the arm (based on colour segmentation / clustering). This localizes the arm.
2. Compute optical flow in the frame.
3. Compute the optical flow of the arm by combining the first two outputs.
4. Use this to find the global direction of the arm by taking weighted mean of optical flow of the arm.





Original Frames

Segmented JCB arm (k-means, color-seg)



Optical flow of JCB arm (Farneback method)

Challenges Faced:

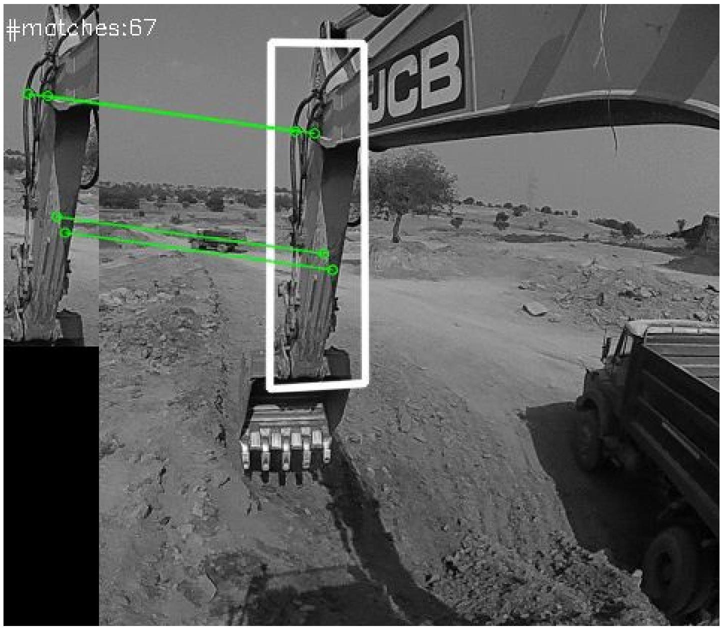
1. Not able to entirely segment the arm due to illumination(shadows etc)/occlusions(dirt etc).
2. Optical flow not computed properly due to discontinuous frames in the dataset.

Iteration 2:

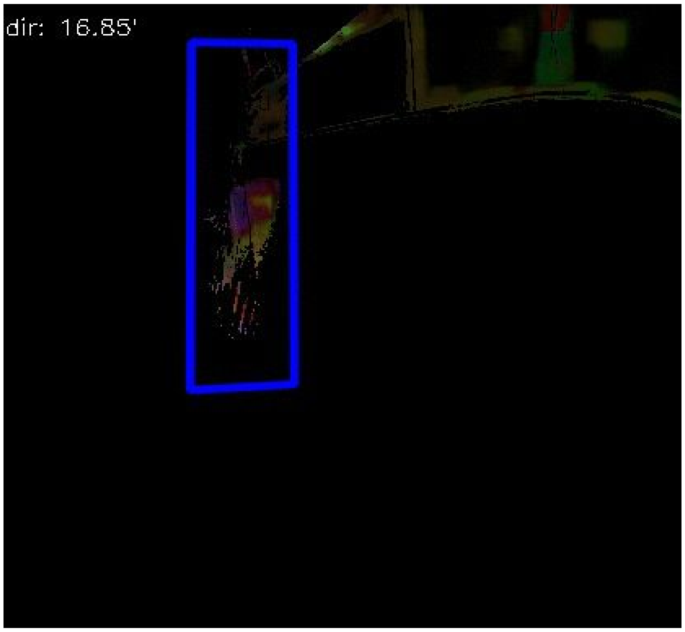
Workflow:

1. Used feature detectors like (ORB, FAST) to improve the localization of the arm.

Output:



JCB arm localization (SURF features)



Segmented Optical Flow

Challenges Faced:

1. Requires more template images which might not be good for scalability.
2. Though we get the approximate angular movement of the arm, it is still not enough to accurately measure the overall direction of the arm.

Iteration 3: (In Progress)

1. Train Faster - RCNN's to localize the arm.
2. Use RNN’s for improving the action detection.

Additional requirements:

1. Full video without any frame-cuts.
2. probably more time.