TOOL AUTOMATION

SCHOOL OF TECHNOLOGY

WOXSEN UNIVERSITY

22ND APRIL, 2023

PROBLEM STATEMENT

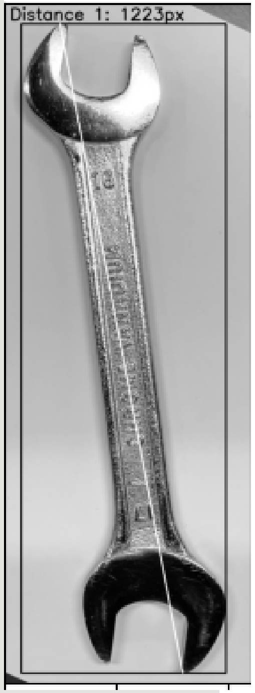
To be able to segregate and identify different industry tools based on their sizes and shapes

- Identifying the errors and variations among different tools.
- We should be able to predict whether a particular tool will be accepted or rejected based on whether the sizes and shapes match with the ideal tool.
- If the shape matches, then the distances of certain points in the tool will be calculated.
- Based on this, we can determine whether the tool is accepted or rejected.
- These being the issue when the input data is from a live video feed.

SOLUTION

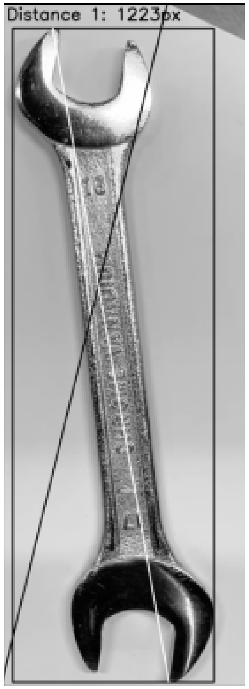
The following steps were taken to get the desired outcome:

- We perform certain preprocessing techniques and thresholding to first differentiate the tool from its environment.
- For detecting the shape of the tool, we use feature based approach, these features are compared to the features of ideal tool.
- If the tool is accepted, we will calculate the distances and compare the sizes of the ideal tool and the accepted tool.
- For the size based comparison, we use contours based approach.
- The distances taken in consideration are distance between the short sides of tooth and distance between bigger size of tooth.
- This gives us an good estimation of the size of tool and when the comparison is done the orientation of tool won't be making a difference.
- If the distances are ideal too, then the tool will be accepted, if not it will be rejected.

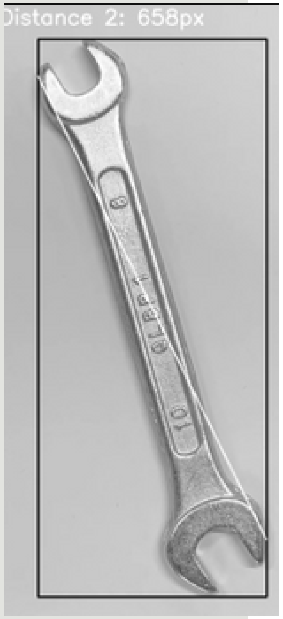


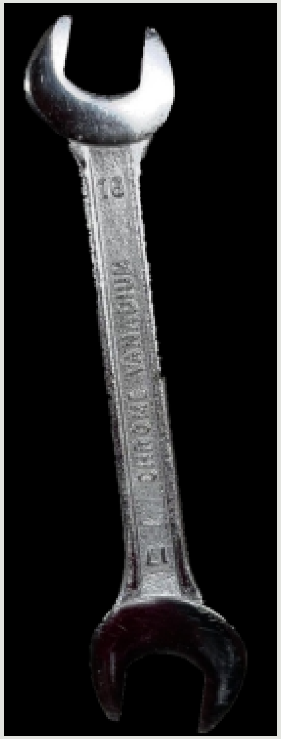
DIFFERENT APPROACHES

FOR CALCULATING DISTANCES



- We tried to calculate the distance from uppermost to lowermost tooth and second-uppermost to second-lowermost tooth of the tool simultaneously.
- Unfortunately it was calculating only the uppermsot to lowermsot tooth perfectly and not the other.
- To overcome this, we created two separate functions, one to calculate the distance between the uppermost tooth to lowermost tooth.
- The other was to calculate the distance between second-uppermost tooth to second-lowermost tooth of the tool.
- The average computing time here is around **0.2 secs.**

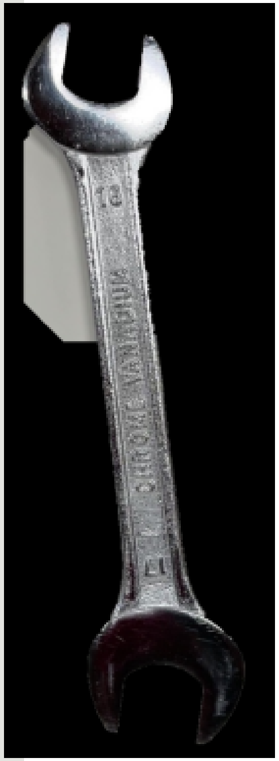




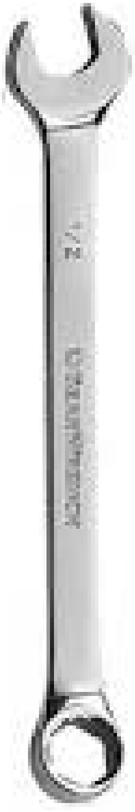
DIFFERENT APPROACHES

FOR MATCHING TOOL SHAPES

APPROACH - 1



- We initially used the grabcut algorithm to detect the shape of the tool and then compare the contours of tool shown aside(ideal tool) with the tool that is to be checked.
- Although the segmentation was done perfectly. The computing time for this algorithm was an average of **8.0 secs** only for 5-6 interations, which is not feasible.
- If we reduce the no. of iterations, the segmentation is compromised.
- The aside photo is with 3 iterations, it requires a minimum of 5 iterations to get the perfect shape.



DIFFERENT APPROACHES

FOR MATCHING TOOL SHAPES

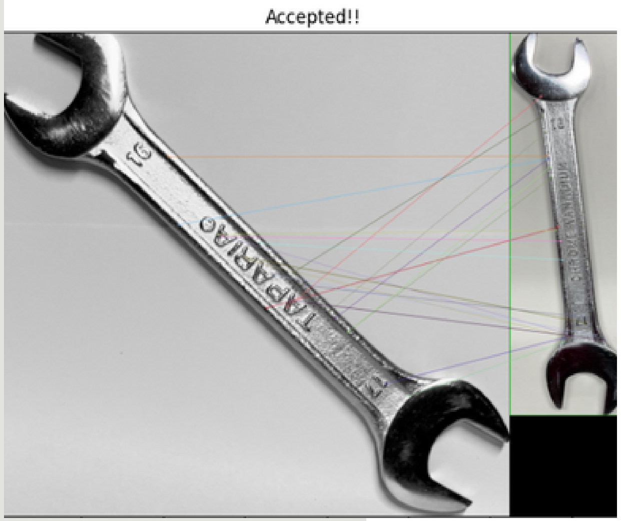
APPROACH - II

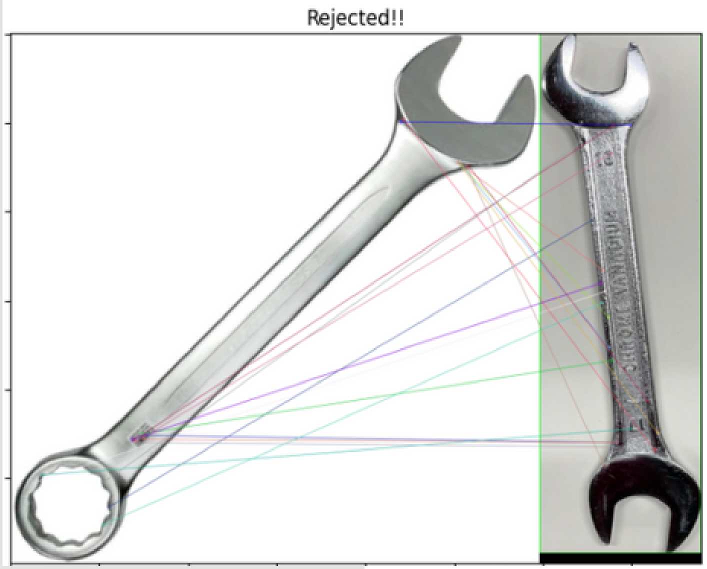
- Initially we used SIFT to extract features of the image.
- The next step we used Flann Based Matcher to match the features in both the images.
- If the score of the matches is greater than 10 then the image is considered to be the similar shape as the ground truth tool.
- Otherwise, the tool given for match is rejected.
- The drawback of this type of matching is it doesn't use any threshold which could be tweaked to get desired output.
- Many of the outputs given by this approach were incorrect.

DIFFERENT APPROACHES

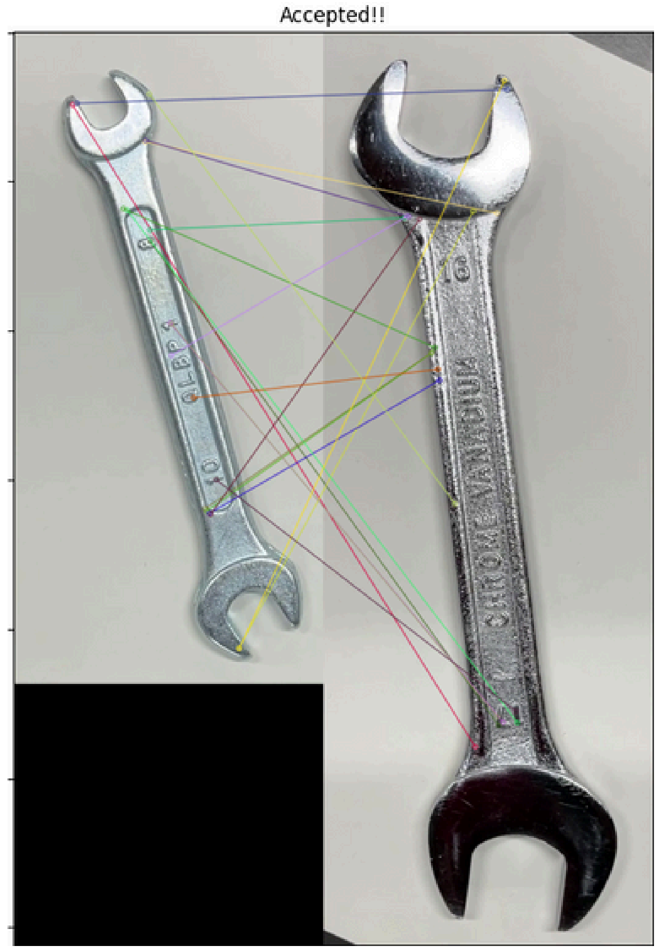
FOR MATCHING TOOL SHAPES

APPROACH - III

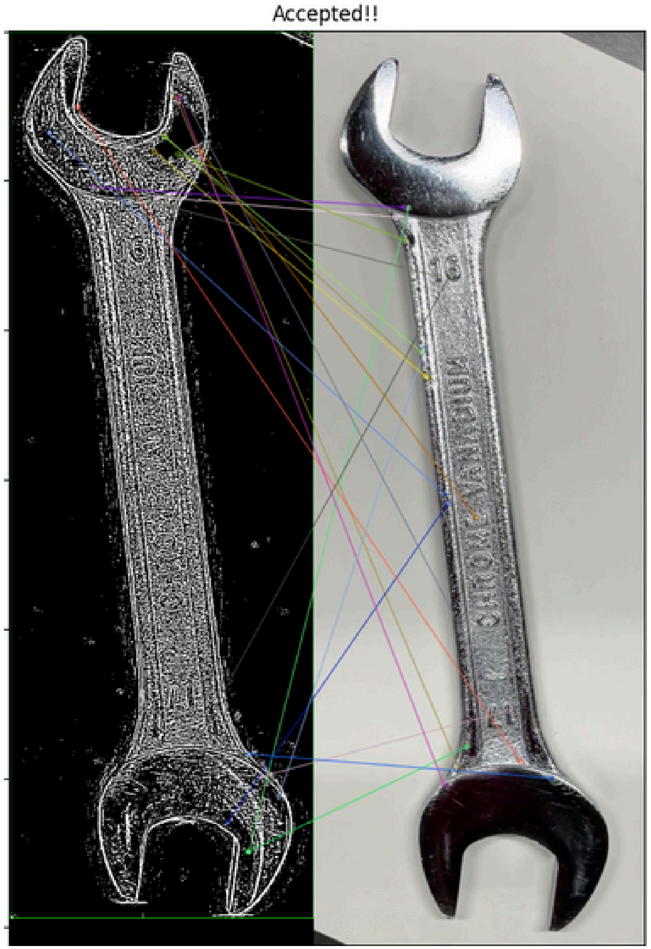




- In this approach, first, the features of the image are extracted using ORB operator.
- The 2 images are ground truth image which is on the right and second is tool to be compared which is on the left.
- Then these extracted features are matched using BF Matcher.
- Then the Matches are drawn on the images which can be seen on the left.
- If, the matches are greater than 135 then the tool is considered as a match.
- Otherwise, the image is rejected.
- Using this approach we got 8/10 outputs correct.

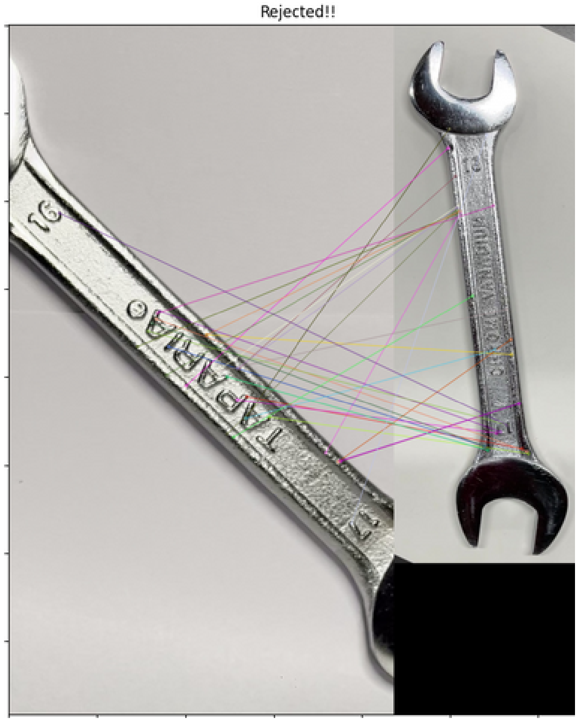


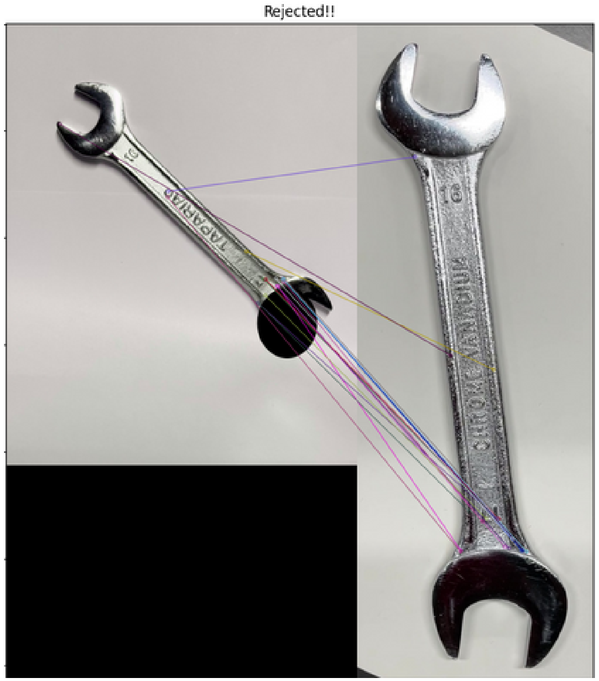
Accepted!!

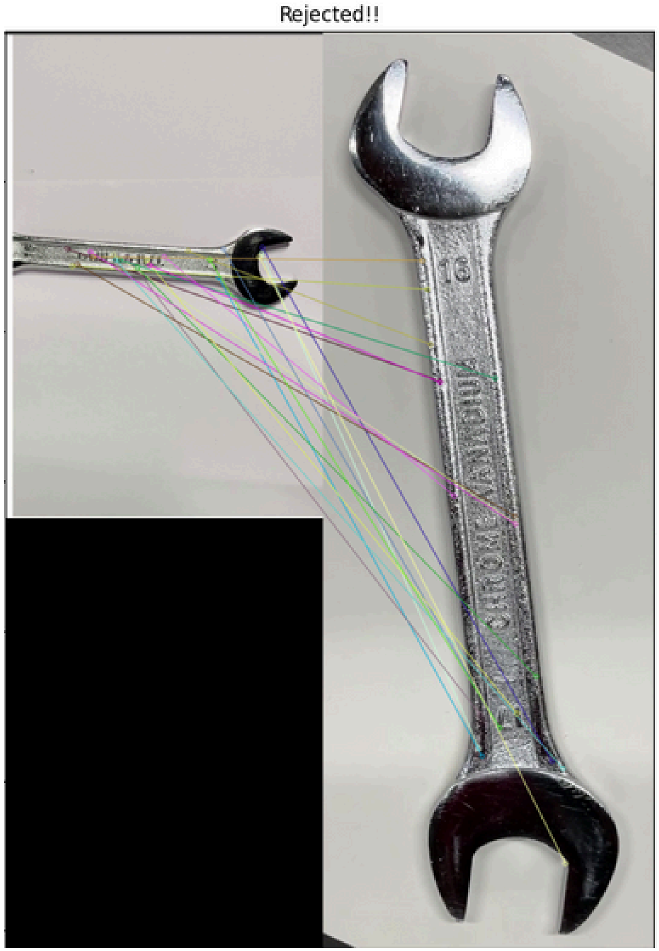


Accepted!!

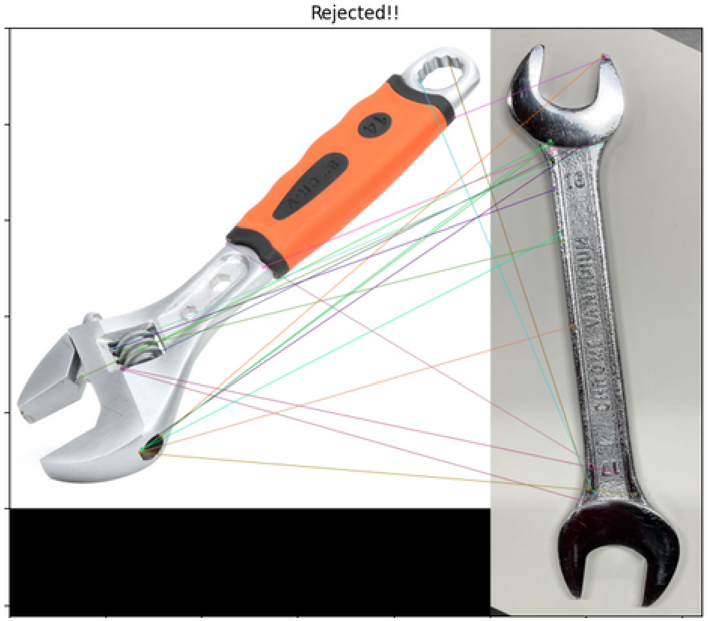
DIFFERENT TOOL IMAGES

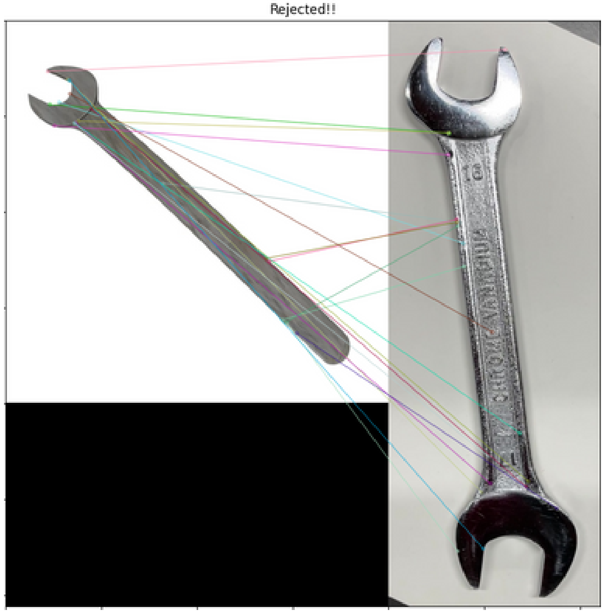


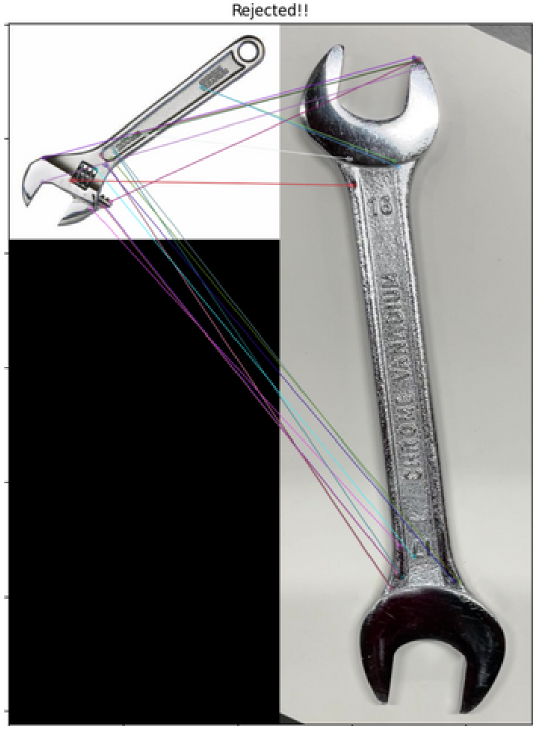




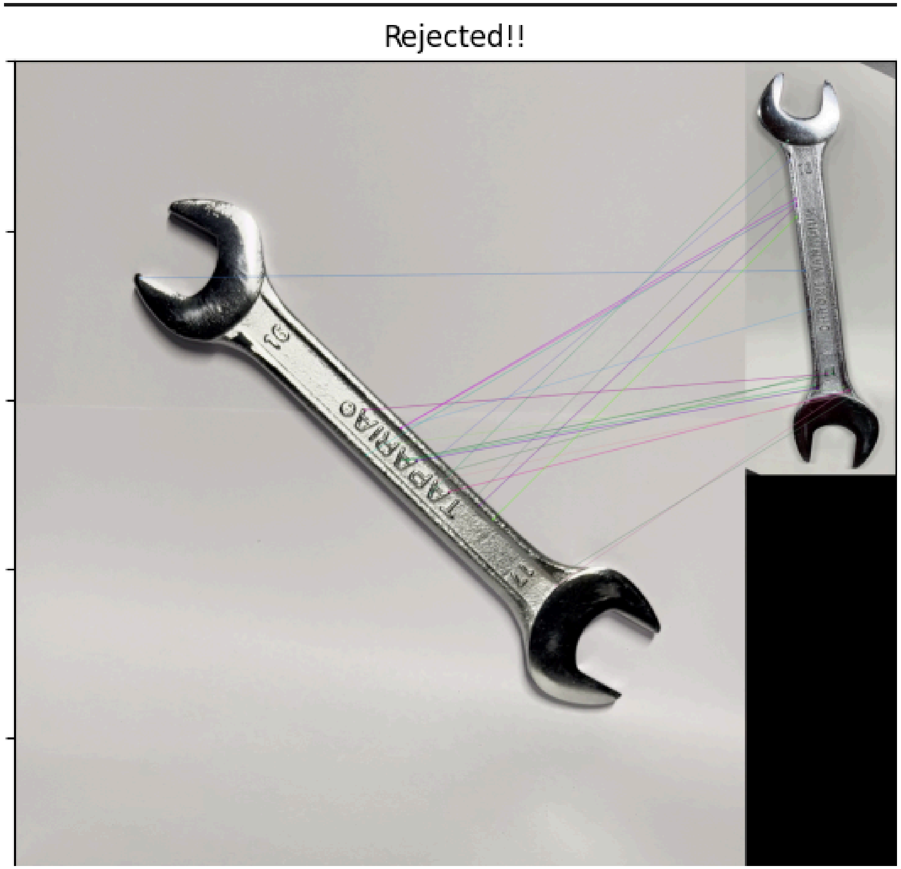
OCCLUDED IMAGES







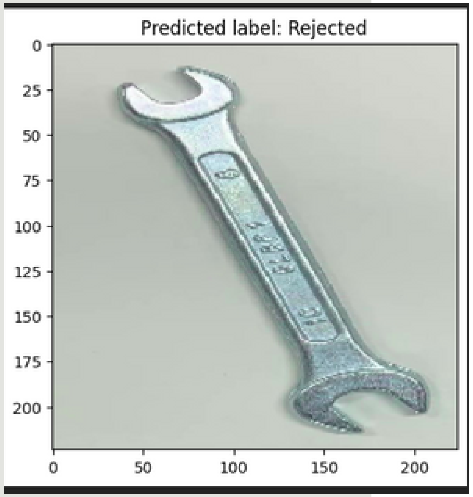
DIFFERENT TOOL IMAGES



It is assumed that this tool was rejected due to the distance between the camera and the tool.

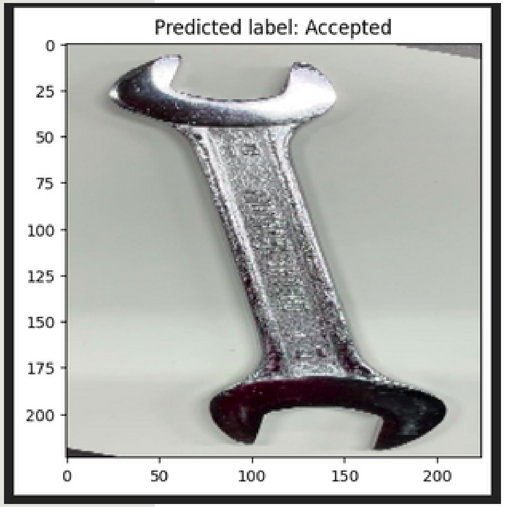
FAILED TOOL IMAGES

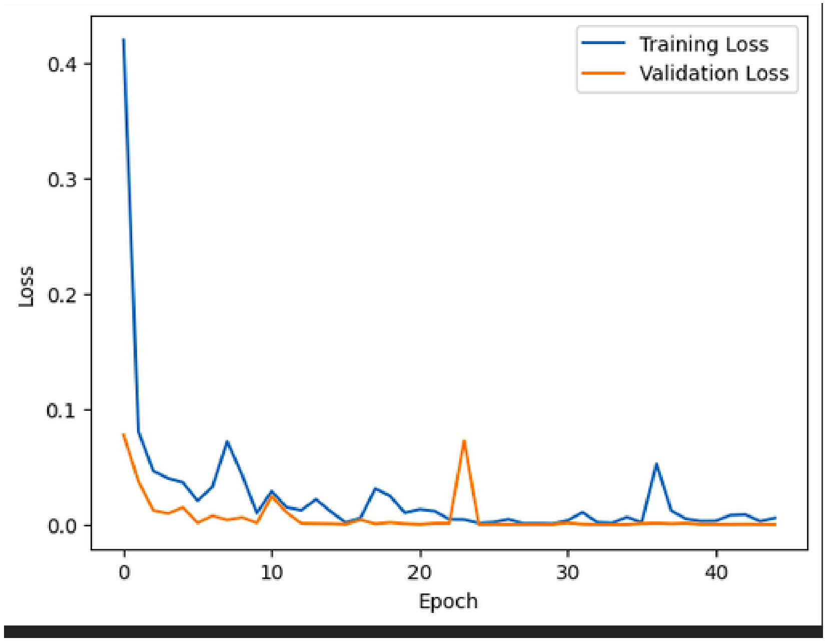
DIFFERENT APPROACHES

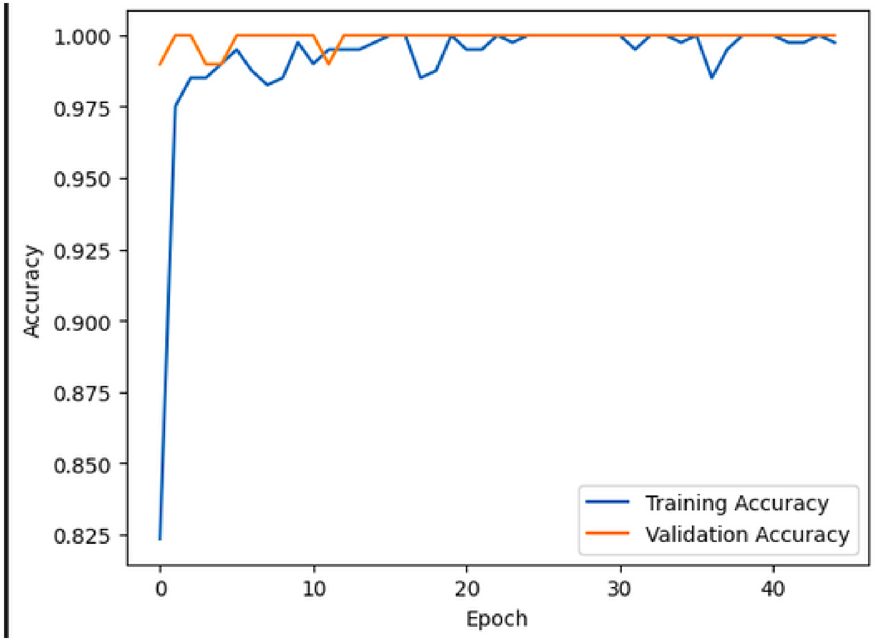


APPROACH - IV

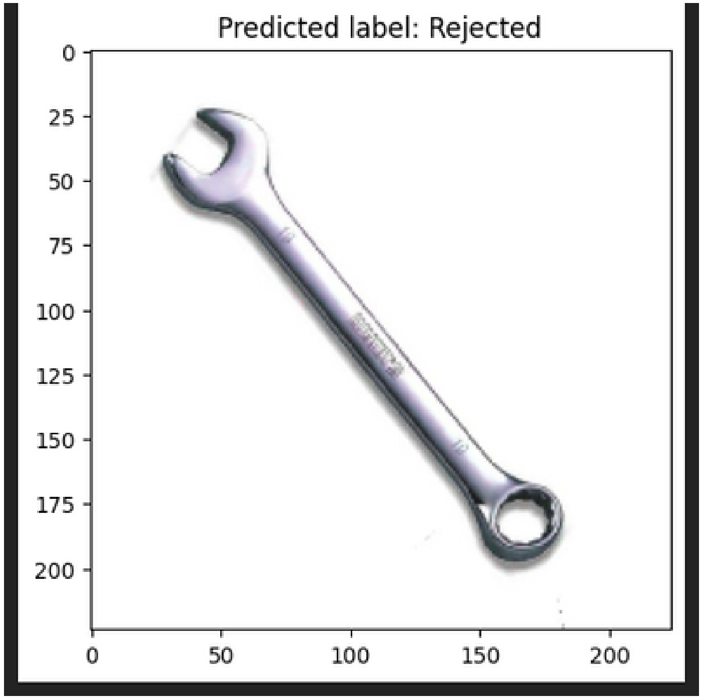
- Using 2D-CNN we have trained a model which had 402 images including accepted and rejected classes.
- After training the model we got the weight files of the model.
- Then using these weight files we have predicted outcomes of few images and the test results were pretty good.

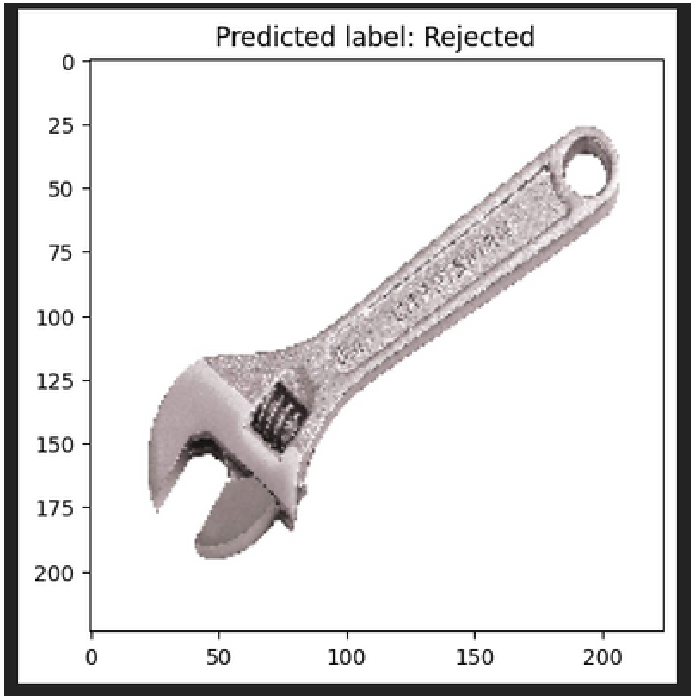




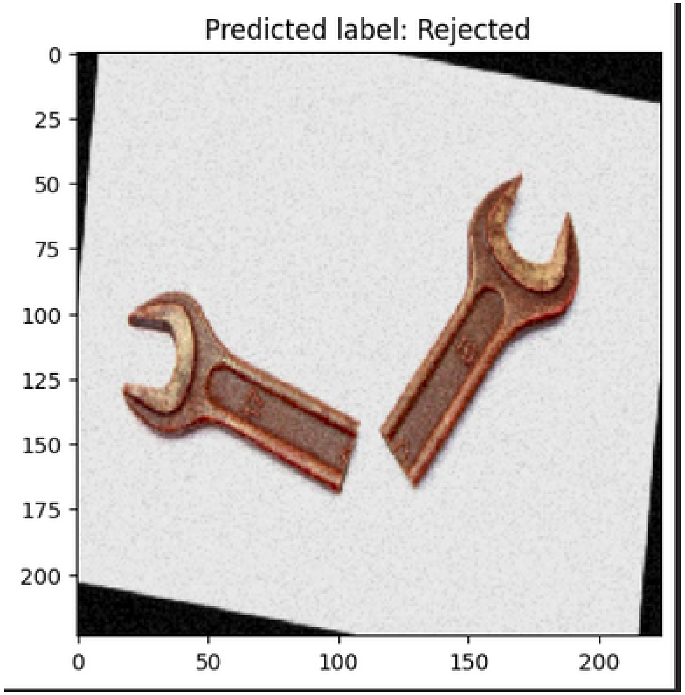


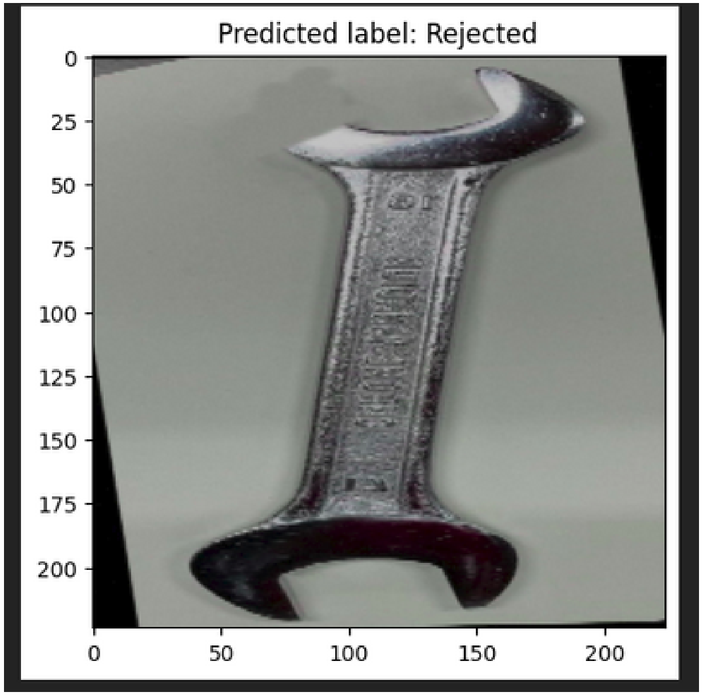
TRAINING AND VALIDATION GRAPHS



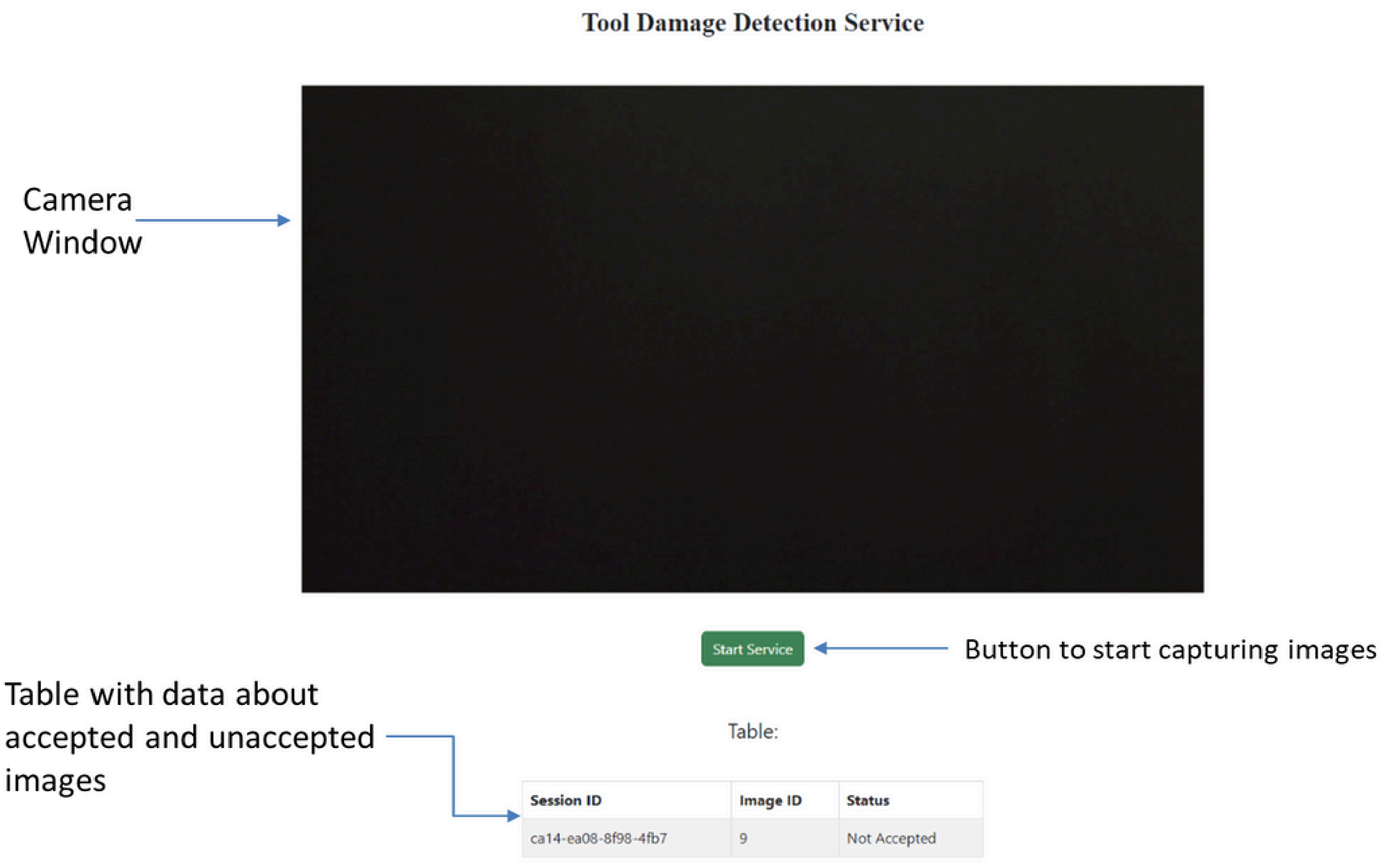


DIFFERENT TOOL IMAGES





OCCLUDED TOOL IMAGES



Tool Damage Detection Service

Camera Window

Button to start capturing images

Table with data about accepted and unaccepted images

Table:

Session ID	Image ID	Status
ca14-ea08-8f98-4fb7	9	Not Accepted

FRONT-END INTERFACE