Basic Quality inspection using image processing

Arunkumar B

Balachandra P Shetty

OBJECTIVE

Build a Quality system which is capable of doing some of the quality inspection using image process techniques which are currently done manually by human

INTRODUCTION

A machine tool is a machine for machining metal or other rigid materials, usually by cutting, boring, grinding, shearing, or other forms of deformations.

Alligator shear

Abrasive saw

Milling machines

Drill press machine

Bandsaw

Lathe machine

Diamond saw

MACHINING PROCESS

Machining is process where metal will be removed from the input material to get desired shape

Quality PROCESS

Once machining process complete next is Quality process ,Quality ensure the correct finished product reaches the customer,If Quality fails component needs to be send for for rework

- Examining
- Measuring
- Testing

VISUAL INSPECTION

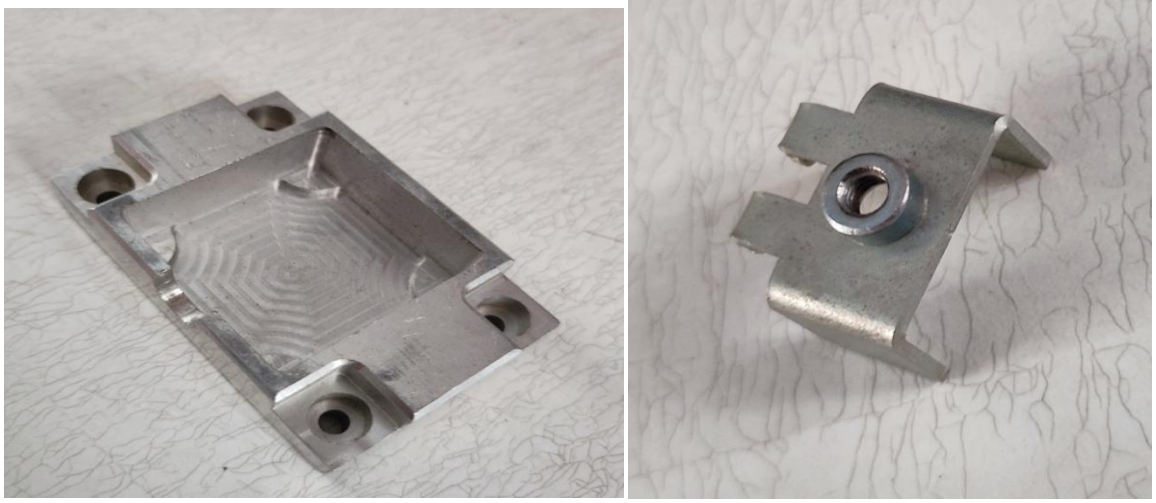
1. Identify if all machining operation competed
2. Checking for any physical damage to the component
3. Checking for improper machining
4. Checking for any corrosion /rust
5. Checking for cutter mark

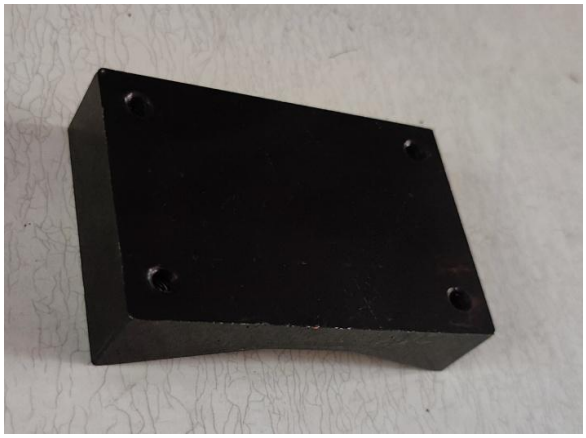
Why AI based visual inspection

https://www.ibm.com/in-en/topics/visual-inspection

DATA COLLECTION

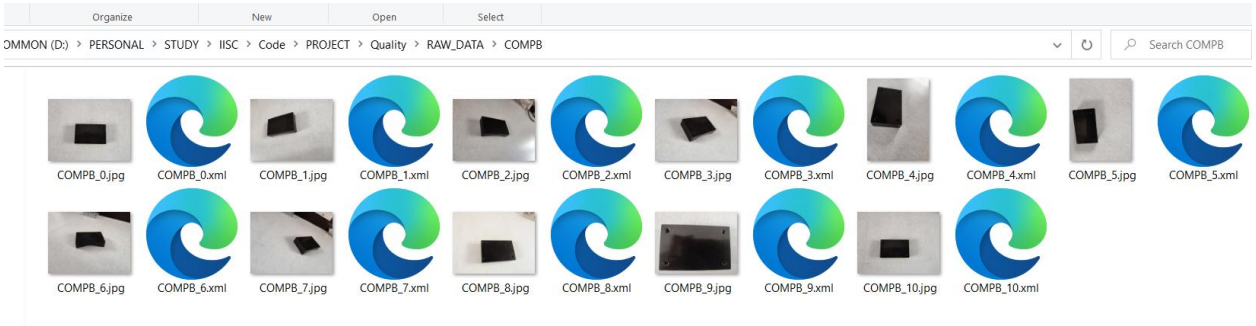
- USB based camera is connected to Laptop which can capture
- Python code is made which request user to select location, number of images to acquire and select name for each class
- Based on the input data will be saved in the location




L


LABELLING

- Downloaded open source labelling tool “LABELIMG”
- Compiled and used for Labelling all the images
- Its created xml which have selected box in it



- Totally 11 images collected for each segment

Named under four classes

- COMPA
- COMPB
- CLAMP
- SCREW

Created label file also with same name which will be used in further work

labels = [{'name':'CLAMP', 'id':1}, {'name':'COMPA', 'id':2}, {'name':'COMPB', 'id':3}, {'name':'SCREW', 'id':4}]

SPLITTING DATA

- Made a small python code which ask for location where data will be available, Number of data required for training and number of data required for testing
- Based on the data it will randomly split data for testing and training

ENVIRONMENT

- Following library/models is installed
- Tensor flow
- Object detection
- Xml
- Matplotlib
- (Procedure for installation is added in python script)
- SSD MobileNet V2 FPNLite model is selected for this application
- There will be model_builder_tf2_test.py which will help to check all the needed dependency is available or not

MODEL SELECTION

- Got Numerous number of Model form https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/tf2_detection_zoo.md
- I have Selected SSD MobileNet V2
- SSD Stand for Single shot detection, Single image is required to detect image
- This model is very lite which can be deployed even in the low compute devices

TRAINING MODEL

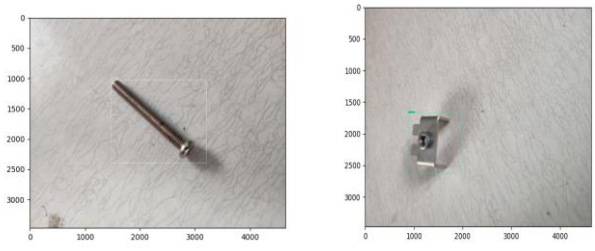
- Training image location, Label map file and converted TF record file is provided as input for training
- Pipe line config file is edited for all the configuration and path related
- Total time of 55 min taken for training
- Training has been done using only CPU

TESTING MODEL

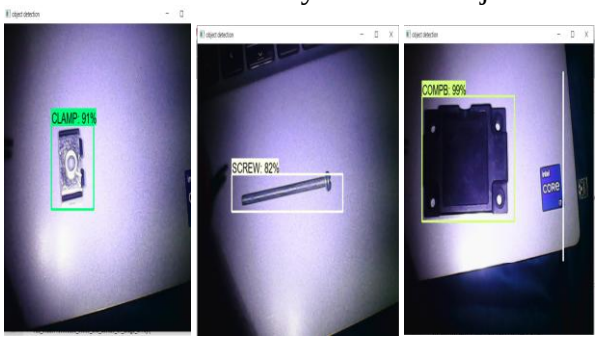
Test data folder, Label map is provided as input

EVALUATING MODEL

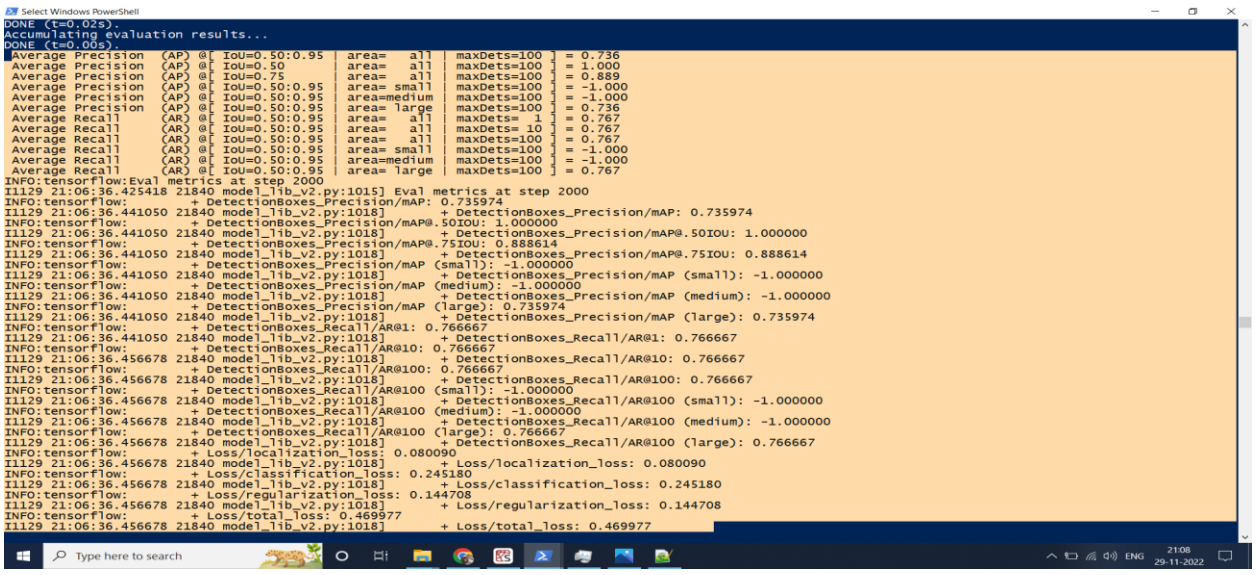
- Small python code is made to manually pass image to trained model
- Object in the model has been correctly identified got boxes and label around it



- Small test python code is made which is acquire image from the camera and pass to model to identify if trained object is detected



- Image will Multiple variety of component is passed and tested
- Image will multiple number of component is passed and tested



CONCLUSION

- If the required component is available

- How many number of components available
- Various variety of component available

FUTURE

- Check if machining process completed or not
- Check if there is physical damage in the component

REFERANCE

- https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/tf2_detection_zoo.md
- https://www.tensorflow.org/guide/model_garden
- https://www.youtube.com/watch?v=yqkISICHH-U
- https://vidishmehta204.medium.com/object-detection-using-ssd-mobilenet-v2-7ff3543d738d