REBIKE

Case Study- Data Points Extraction from Bike Images

-Anirudh Balaraman

Case Study- Data Points Extraction

Idea:

Using an ensemble of models that are better suited for each task.

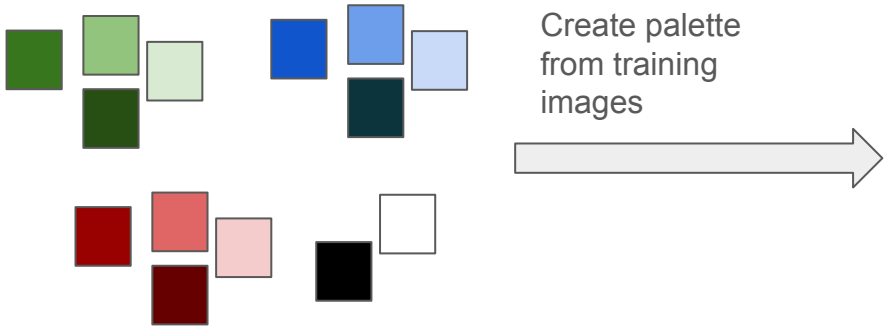
Leverage the structure of data.

Case Study- Data Points Extraction

Bike Color Prediction

- Classification not feasible.
  - When number of classes changes, retraining required.
  - Difficult to debug.

- Clustering and KNN used.
  - Easy to retrain and add new labels.
  - No class imbalance issue.



Create palette from training images

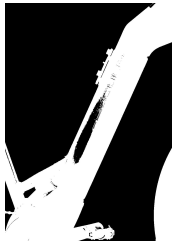
```
204,22,0,red
220,0,3,red
128,24,24,red
209,23,23,red
139,0,0,red
254,0,0,red
206,0,25,red
174,32,26,red
61,13,3,red
254,0,2,red
249,217,94,yellow
252,234,4,yellow
255,183,9,yellow
254,242,0,yellow
247,224,23,yellow
255,166,0,yellow
255,242,39,yellow
255,215,12,yellow
246,191,39,yellow
255,255,0,yellow
0,255,0,green
64,189,85,green
37,202,38,green
0,166,82,green
125,194,75,green
125,232,88,green
123,252,1,green
35,67,17,green
159,217,140,green
255,128,0,orange
255,153,0,orange
255,103,0,orange
254,101,33,orange
255,127,0,orange
252,79,19,orange
255,102,0,orange
255,122,1,orange
255,103,0,orange
255,237,231,white
248,249,254,white
242,233,228,white
253,251,251,white
```

Case Study- Data Points Extraction

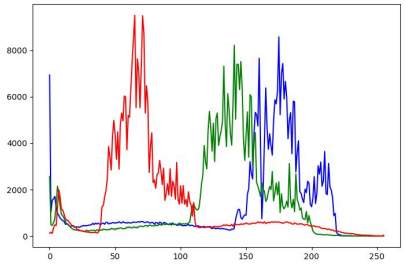
Bike Color Prediction

Input image

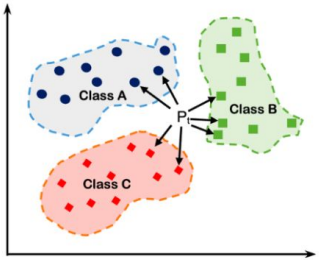
Cropped image to focus on the chassis



Masking to create histogram. Since dominant colour from the histogram is chosen as the color of the bike, the mask filters the white background



Palette of bike color:
75, 145, 175



Assigned a color label from the training labels

Case Study- Data Points Extraction

Bike Color Prediction

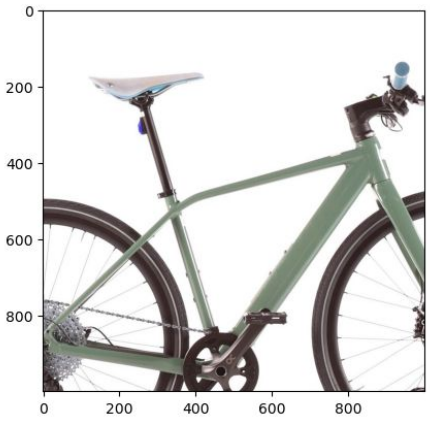
Performance: Accuracy- 76%

Color is black

<matplotlib.image.AxesImage at 0x7fe7dc561060>

Color is grey

<matplotlib.image.AxesImage at 0x7fe7dc19f4c0>



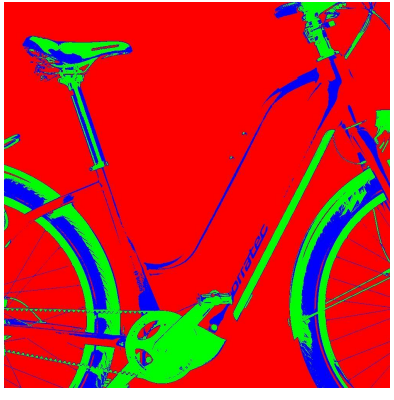
Next Steps:

- Improve masking.
- Tweaking the palette has drastic changes in performance.
- Analyse the embedding space and clusters and regularise them.

Brand Prediction

- Text detection and recognition using pre-trained keras OCR pipeline. Again, easy to add new labels, finetuning etc.
- Pre-processing: Contrast applied to images to highlight the brand label.
- Predicted texts are sequence compared against brand names.

habihe-> Haibike



Case Study- Data Points Extraction

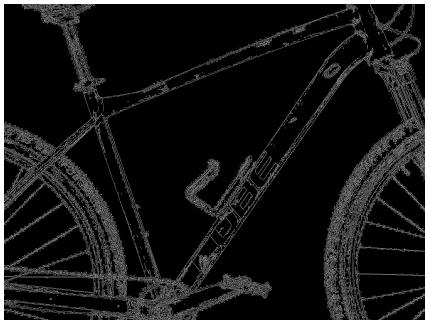
Brand Prediction

Performance: Accuracy- 68%

- Failed to detect texts that are visually not discernible from their background
- Failed to detect symbol like fonts. Eg-KTM

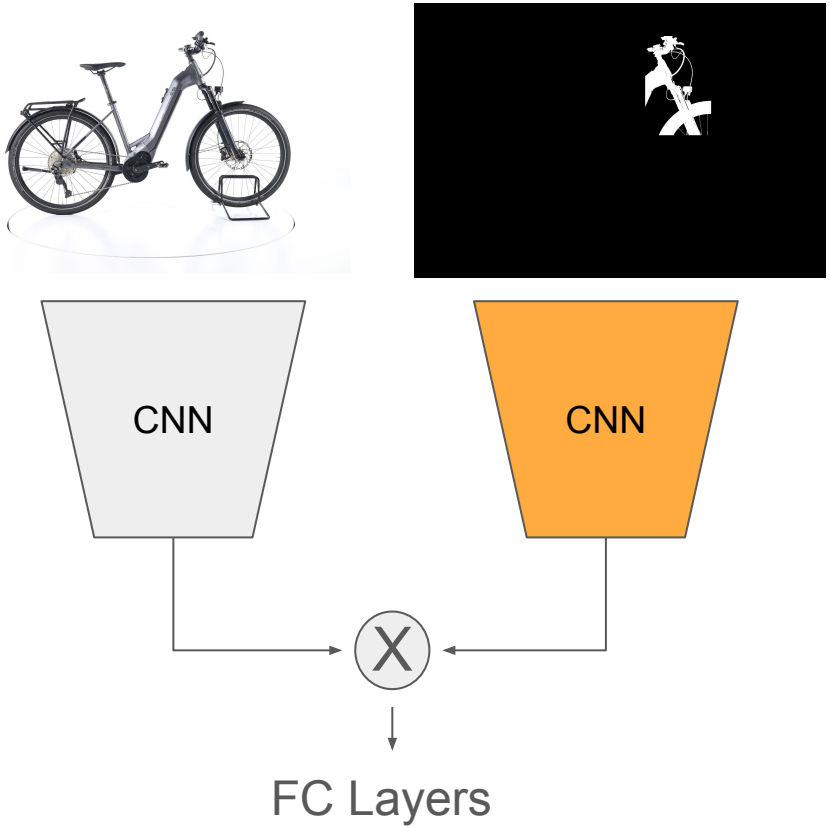
Next Steps:

- Fine tune pipeline with symbol like names like KTM
- Add the brand names to vocabulary.
- Fine tune detection Models, YOLO
- Use segmentation instead of detection.



Classification

- Check if bikes have a front lamp
- Treated as a classification task using CNN
- Spatial attention using ROI masking
- GradCAM to analyse performance



Case Study- Data Points Extraction

Classification

- Can be used to perform any classification
- Other features that can be extracted similarly- Male or female, front gear, seat suspension, type of brakes, type of bike, mud guards type

Summary of Folder Structure

color_recognition- knn and palette creating functions.

dataset- Bike images downloaded and processed.

training_dataset_color- Colored images to create the palette.

utils.py- Contains necessary functions

training.data & test.data- Stores palette

THANK YOU!