VEHICLE COLOR IDENTIFICATION

USING CNN AND DARK CHANNEL DEHAZING

Jaideep Whabi A20403110

Problem Statement

- As a component of an Intelligent Traffic System, identify the color of an on-road vehicle.
- Based on images captured of moving or still vehicles on-road.

Uses

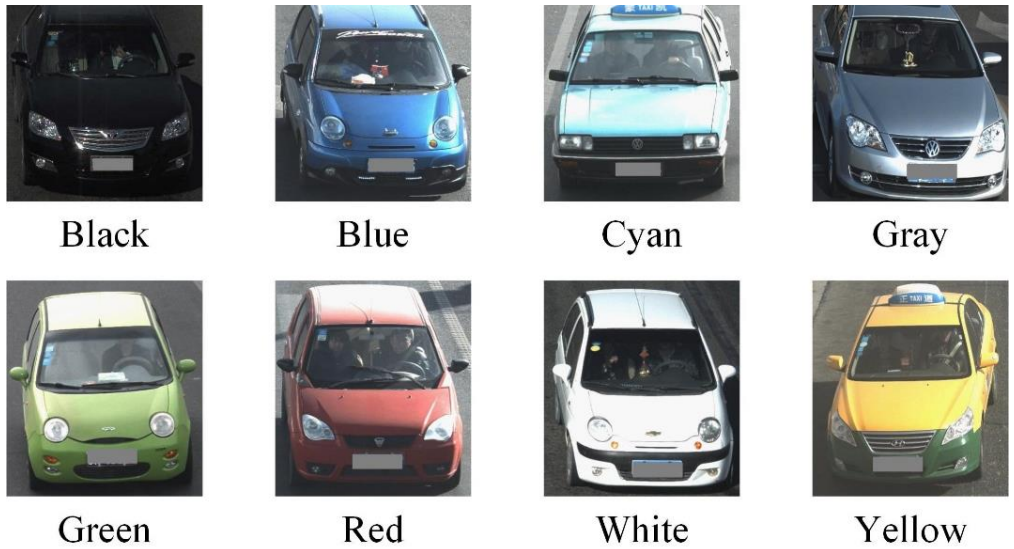
- An important component of Intelligent Traffic system for vehicle identification, alongside make & model and license plate identification.
- Important part of vehicle identification to provide visual cues for fast law enforcement

Challenges

- Quality of image captured can be affected by various issues like weather, capture device capabilities and strip combination of variables.
- Moreover, lower quality images , or images affected by weather are very noisy, thus making it harder for a model to distinguish colors.

DATASET USED

- The Vehicle Color Recognition Dataset contains 15601 vehicle images in eight colors, which are black, blue, cyan, gray, green, red, white and yellow.
- The collected data set is very challenging due to the noise caused by illumination variation, haze, and over exposure



IMPLEMENTATION DETAILS

- Dehazing of Images

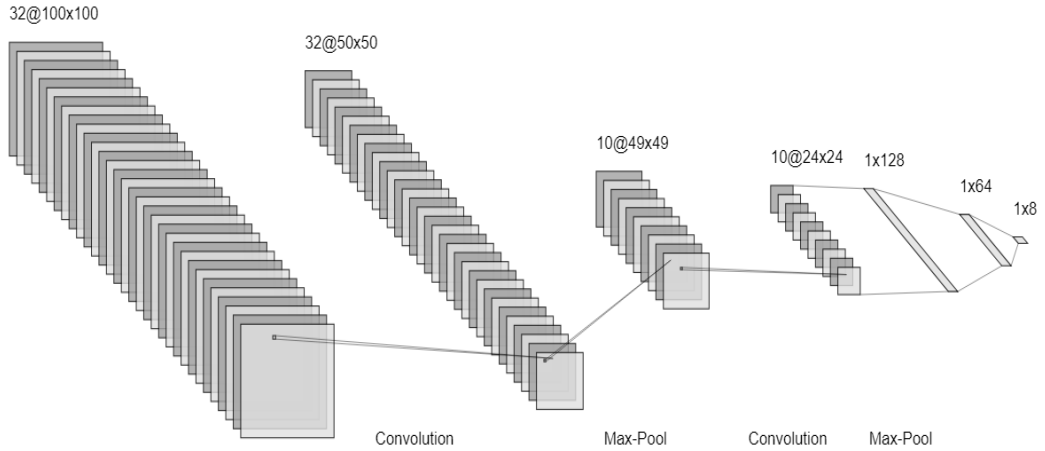
1. Detecting atmospheric light.
2. Detecting the dark channel for each pixel.
3. Creating a transmission map.
4. Combining the information to create a clear image which can be used for classification.

Input Image (with noise)

Output Image(Clear Image)

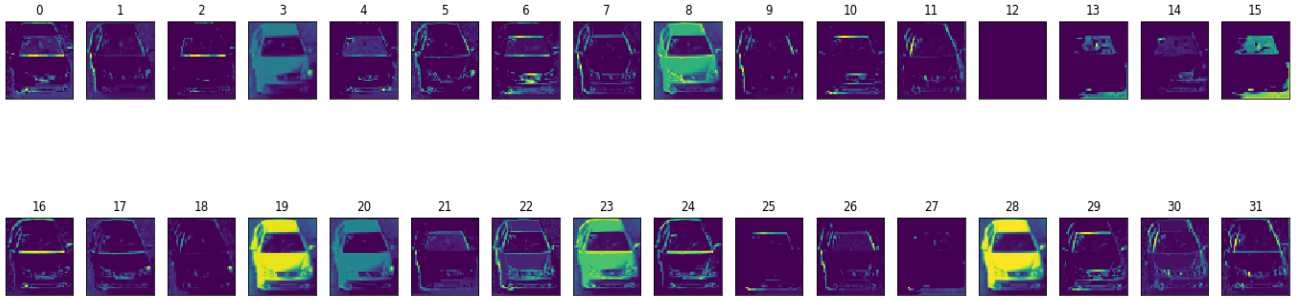
IMPLEMENTATION DETAILS

- Model Architecture:
  - Convolutional Network to learn features
  - Fully connected network to classify on features learnt.
  - Two CNN layers with max-pooling, 2 hidden layers and 1 output layer.
  - ReLU activation for all layers except for output layer which is softmax.
  - Loss function: categorical-crossentropy
  - Optimizer: RMSprop
  - Metrics: Accuracy

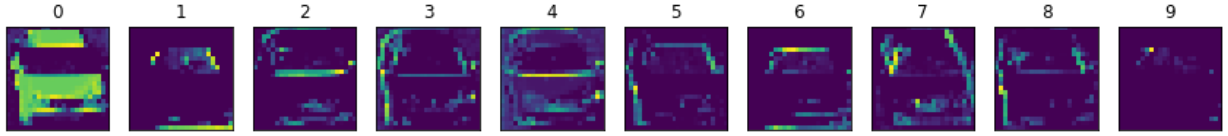


RESULTS

- Intermediate Outputs:
  - First layer identifies the general outline of the vehicle.
  - The second layer identifies more complex shapes like edges/regions of the car.
  - Features are more abstract in the second layer.
  - Nothing new is learnt if a third layer is added.



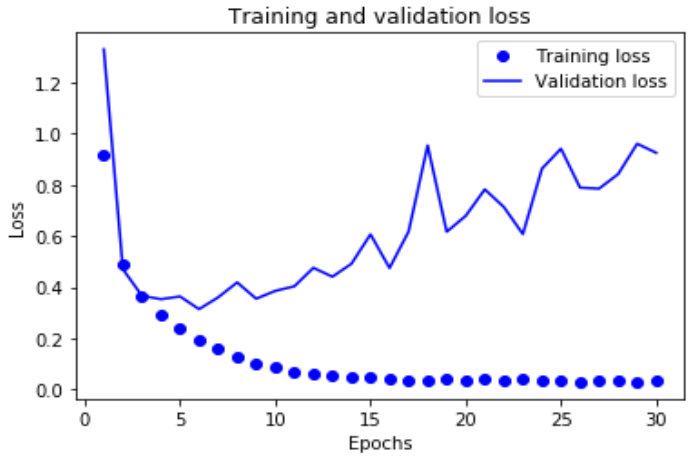
Output of first Convolution + MaxPool layer



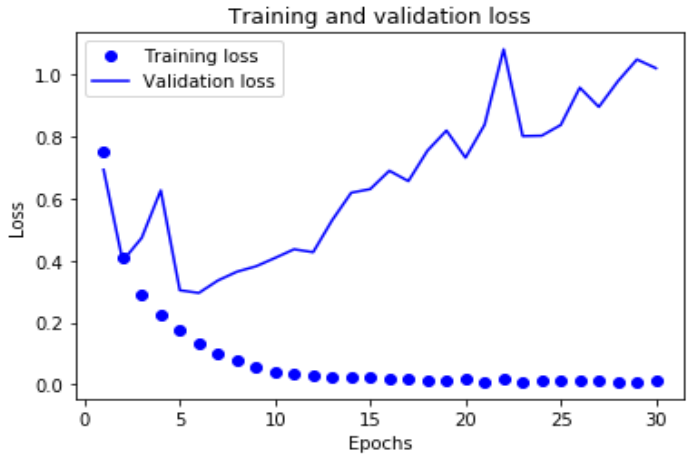
Output of second Convolution + MaxPool layer

RESULTS

- Performance on hazy images vs haze free images:
  - Training on haze free images leads to faster convergence and lower loss values.
  - Hazy images will need , comparatively, a larger network to reach the same level of performance.
  - Average accuracy per class after training on haze free images is 92%, whereas on hazy images it is 87%. Similar differences are observed in precision and recall.
  - Thus, prior treatment of images to remove haziness results in a simpler CNN network , able to classify with a higher accuracy.



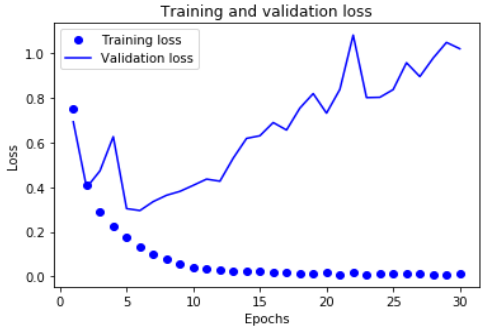
Training vs validation loss on hazy images



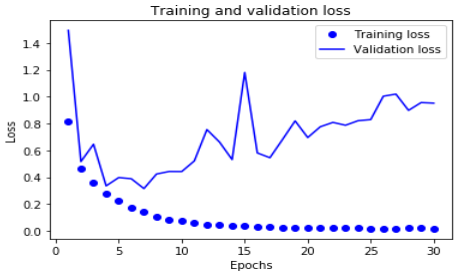
Training vs validation loss on haze free images

RESULTS

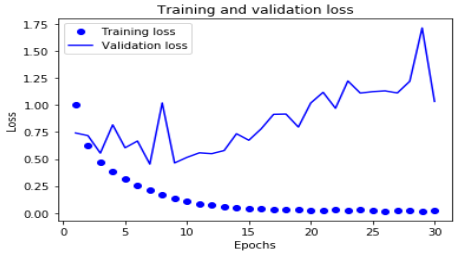
- Tests across different color spaces:
  - Training the same model architecture using data defined as a different color space leads to a difference in performance.
  - Model convergence is the best when the color space is RGB and worst when the color space is LAB.
  - The same goes for model performance. On test data, models trained using RGB color space have average accuracy per class as 92% whereas the other two classes yield 90%.



Training vs validation loss using RGB color space



Training vs validation loss using HSV color space



Training vs validation loss using LAB color space

CONCLUSION

Using hazefree(clean) images gives us simpler models that converge faster and give us 5% higher accuracy.

As seen from the training epochs and testing accuracy for the optimal model for each color space, the RGB color space looks to be the best way to define this dataset and leads to better model interpretation and performance. Precision and accuracy scores for each color space can also be calculated to confirm this.