Name: Aisun Zarghami

Deep Learning Model for Color Detection in Images

This report outlines the implementation of a convolutional neural network (CNN) for color detectio using TensorFlow and Keras. The primary objective is to classify images into various categories based on a dataset located at a specified directory.

Image Preprocessing

Images are resized to a dimension of 128x128 pixels.

Augmentation and Normalization

Data augmentation is applied through the following techniques:

- Rescaling pixel values between 0 and 1.
- Shear transformations.
- Zoom transformations.
- Horizontal flipping.

The dataset is split into training and validation sets with a ratio of 80% training and 20% validation.

Model Architecture

The CNN model is designed using a sequential approach and consists of the following layers:

1. **Convolutional Layer**: 32 filters, kernel size of 3x3, ReLU activation.
2. **Max Pooling Layer**: Pool size of 2x2.
3. **Convolutional Layer**: 64 filters, kernel size of 3x3, ReLU activation.
4. **Max Pooling Layer**: Pool size of 2x2.
5. **Convolutional Layer**: 128 filters, kernel size of 3x3, ReLU activation.
6. **Max Pooling Layer**: Pool size of 2x2.
7. **Flatten Layer**: Converts the 2D matrix to a 1D vector.
8. **Dense Layer**: 512 neurons, ReLU activation.
9. **Dropout Layer**: To prevent overfitting (dropout rate of 0.5).
10. **Output Layer**: Neurons equal to the number of classes, softmax activation.

Model Compilation

The model is compiled using:

- **Optimizer**: Adam with a learning rate of 0.001.
- **Loss Function**: Categorical crossentropy is used for multi-class classification.
- **Metrics**: Accuracy is tracked.

Training

The model is trained for 10 epochs. The training process utilizes the training data, monitors validation accuracy, and evaluates performance on the validation dataset.

Training and Validation Accuracy

During the training process, the accuracy for both training and validation sets is plotted to visualize performance over epochs.

Evaluation

The model is evaluated using the validation set, and the validation accuracy is printed as a percentage.

Prediction Function

A prediction function is defined to classify new images. It involves resizing the input image, normalizing pixel values, and using the trained model to make predictions.

Example Usage

An example prediction is made for a specific image .

The predicted class for the image is printed.

The predicted color is: yellow

Conclusion

The implemented CNN model successfully trains on the specified dataset and provides predictions for new images. The data augmentation and normalization steps enhance model generalization when classifying images across various categories.

Visualizations

The model’s accuracy over epochs can be visualized using the generated plots, which aid in understanding the model's performance.

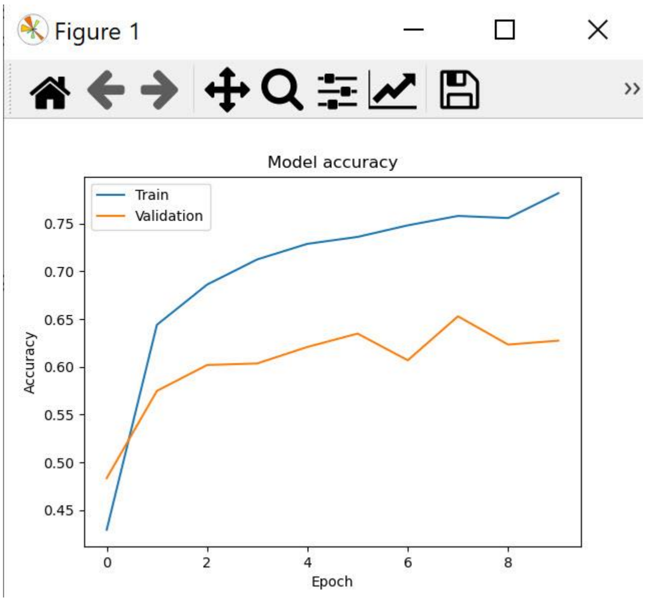
Figure 1
Model accuracy
Train
Validation
Accuracy
Epoch