Final Project Proposal

Author: Harshvardhan Bhatnagar

Project Overview

The project will be based on the dataset [Helmet Detection] (https://www.kaggle.com/andrewmvd/helmet-detection) from Kaggle. The dataset contains 764 images that have been labeled for the two categories:

- With Helmet
- Without Helmet

Working with this dataset will allow me to delve into computer vision and image classification. The goal of the project is to create a model that can detect whether a person is wearing a helmet or not. This can be used in a variety of applications some of which are described below.

1. Two-wheeler vehicles are commonplace in many developing countries and the number of accidents involving riders without helmets is very high. Countries like India struggle to enforce laws mandating the use of helmets. A model that can detect whether a person is wearing a helmet or not can be used to enforce the law and reduce the number of fatalities.
2. A very similar model can be trained to detect whether a person is wearing a helmet or not in a construction site. This can be used to enforce safety regulations and prevent accidents, improving safety in the workplace as well as reducing the number of lawsuits that construction firms face. We may even be able to use the model trained by us and apply transfer learning to train a model for this specific use case.

Some of the images in the dataset are shown below in Table 1.

Table 1: Images from the dataset

As seen above, the images clearly have varying backgrounds, lighting conditions, watermarks and some even have multiple people in them. This will make the task of classification more challenging.

However, the dataset also provides information about the bounding boxes for regions of interest in the images. This will allow us to crop and regularize the images before feeding them into the model.

Our primary focus will be on application of various methods for image classification and comparison of their performances. We will attempt to achieve the a decent balance between accuracy and computational requirements since the dataset is relatively small.

Potential Methods

We can explore several methods with varying levels of accuracy and feasibility. Some of these are described below.

Method 1: Basis Functions for Feature Extraction + SVM

We can use basis functions to crop and resize images, followed by applicaion of Kernels that help with feature extraction such as Gaussian Blurring and Sobel Edge Detection. The resulting images can then be fed into a Support Vector Machine (SVM) for classification.

Method 2: Convolutional Neural Networks

We can use a Convolutional Neural Network (CNN) which will be able to learn convolutional filters that on its own using backpropagation. This will allow us to avoid the need for feature extraction. While this may be more accurate, it will also be more computationally expensive. However, since we can use a GPU for training and parallelization, this may not be a major issue on a small dataset like this.

Method 3: Ensemble Learning

We can use an ensemble of classifiers to improve the accuracy of the model. This can be done by training multiple models using different methods and then using a voting classifier to combine the results of the individual models. This will allow us to leverage the strengths of different methods and improve the accuracy of the model. We may also be able to use a CNN along with another classifier to improve the accuracy of the model.

Method 4: Transfer Learning

We can use pre-trained models such as ResNet and VGG16 to perform transfer learning. These models already have their feature weights trained on large datasets and we may be able to produce better results by using these models as a starting point and then fine-tuning them for our dataset. This can help us achieve better results with less computational resources and possibly a smaller dataset.

Model Selection and Optimization Techniques

It will be interesting to compare the results of these methods on various accuracy metrics such as precision, recall and F1 score. as well as their computational requirements using tools that allow us to profile the code.

We can also explore how the performances of some of these methods vary with different preprocessing techniques such as cropping, resizing and normalization, especially for the SVM, which will be more sensitive to these changes.

We will also try to perform hyperparameter tuning using cross-validation techniques such as Grid Search and explore how the performance of the model varies with different hyperparameters and infer some things about how the model works.

Since this particular problem has potential real-world applications, we will try to incorporate some features that will make the model more useful in a real-world setting. For example, a false positive can be more dangerous than a false negative when classifying whether a person is wearing a helmet or not. We will try experimenting with our evaluation and model selection techniques to account for this.

Citations

```
@misc{make ml,
title={Bikes Helmets Dataset},
url={https://makeml.app/datasets/helmets},
journal={Make ML}}


License
Public Domain
```