Sunglasses Detection/Removal

Chan, Natural

19 April, 2022

Abstract

The goal of this project is to train a neural network to distinguish between individuals wearing eyeglasses and sunglasses. We will primarily use computer generated images of faces alongside the FFHQ (Flickr Faces HQ) dataset. From there, we will hand label each image into three categories - no glasses, eyeglasses, and sunglasses. Using this data, we will perform data augmentation before using variations of the VGG model architecture to predict on a held out set of images.

1 Design

Sunglasses are a very common fashion accessory, but they often obscure the eyes of an individual, resulting in problems with facial recognition. In an age where security footage has become the norm, sunglasses combined with face masks make it very easy to avoid being identified. In order to address this problem, we want to know: is it possible to remove sunglasses from an image and approximate what a person may look like?

For this project, we will start small by simply identifying which individuals are wearing sunglasses. We will train a model on many images of faces - some without glasses, some with eyeglasses, and some with sunglasses which will allow us to categorize new images into these respective categories.

2 Data

There are many datasets containing images for facial recognition. However, many of these datasets are pre-labeled; one dataset on Kaggle has computer generated images that are labeled by no glasses and glasses. Although not ideal, this allows us to only go through the images with glasses and relabel any that contained sunglasses. However, the dataset did not contain enough images with sunglasses so we had to supplement this with additional images from any appropriate dataset. For this project, we use the FFHQ (Flicker Faces HQ) dataset and manually gathered almost every image with sunglasses. Also, we randomly selected 150 images from this dataset (50 of each class) for testing later on.

3 Algorithms

3.1 Data Cleaning/Preprocessing

The first step was to label/relabel applicable images by creating and using a quick script to manually go through and move files to different folders based on each image's class. Many images from the computer generated dataset appeared to be mislabeled, regardless if they were in the no glasses or the glasses folder so they had to be relabeled as well. Our dataset came out to be quite large and hand labeling every image turned out to be a very daunting class. To save time, we randomly selected 1000 images from the no glasses and eyeglasses classes and 500 images from the sunglasses class for our training set.

3.2 Modeling

Our primary form of modeling is through VGG as the architecture is simple and allows for quick adjusting by implementing it from scratch using blocks - each block consisting of a number of convolutional layers followed by a max pool layer. Before modeling, we took 20% of the dataset and used it as a validation set for our model. Each model will be implemented with an early stopping condition using validation accuracy over 10 epochs.

We begin with only 1 block with 32 filters in each layer and 3 by 3 kernels. On the validation set, this managed a 99.37% accuracy which, at first glance, seems very successful. On the 150 images in the test set, it achieved an 86.33% accuracy which is great for a starting model, but there is a potential that it may be learning the wrong information. To test this theory, we implemented data augmentation and found a 76.67% accuracy on the test set, proving that the image may not be detecting sunglasses, but learning something similar that allowed it to achieve fairly good results.

We increased the complexity of the model by switching to 3 blocks, again starting with 32 filters in the first block and doubling this number in each following block. Without data augmentation, this performed worse than 1 block, achieving an accuracy of 81.33% which may again be a result of learning the wrong information and not specifically detecting the sunglasses. Once data augmentation was implemented, this achieved an accuracy of 87.33% which is a fairly significant improvement and means that the model may be appropriately identifying sunglasses.

Unfortunately, we ran into memory issues when further increasing the complexity of this model. Instead, we move onto transfer learning using VGG16 as a base with various weights from facial recognition models. Unfortunately, our model failed to converge on the validation set, achieving only a 64% accuracy on the test set.

4 Conclusion

Our model was fairly successful at predicting which images contained sunglasses. However, the images that were misclassified were often classified as having sunglasses. Although the reason is not explicitly clear, it may be due to the limitations of our datasets or potential mislabeling. To address this in the future, we will want to choose more consistent data with more diverse images rather than segmenting two datasets that may vary too much. Once we address these two problems, we can attempt to train more complex models or even try transfer learning again. Since we achieved a 100% recall on the sunglasses images, we can move forward and implement GANs to remove sunglasses and fill in the space in order to approximate what an individual may look like. However, this will require some relabeling as some images contained sunglasses, but were not covering an individuals eyes.

5 Tools

- Pandas for data processing and cleaning
- Numpy for image processing
- Keras for labeling and modeling
- Sci-kit learn for creation of validation sets
- Matplotlib to generate plots

6 Communication

Presentation slides and visualizations are located in my github repository.