Group Project 2

K. B. Ravi Kiran Reddy (IMT2017034)
Anurag Pendyala (IMT2017504)
SP 825 – Visual Recognition

Introduction

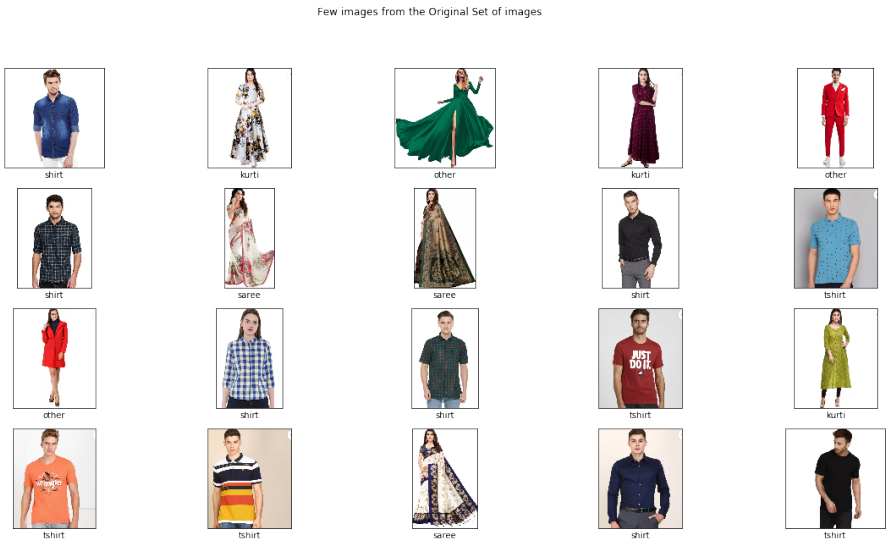
Assume a camera is placed at the entrance of a shop. The camera captures images of people walking by. Consider there is only one person at a time in the frame. The main goal is to classify whether the person in the frame is wearing an apparel belonging to one of the five following categories:

- Shirt
- T-Shirt
- Saree
- Kurti
- Other

Before doing this, we ran an Object Localization algorithm to detect the person and draw a bounding box around him/her. This procedure is explained in the section. The following section gives us some information on the collection of the dataset.

Dataset Collection

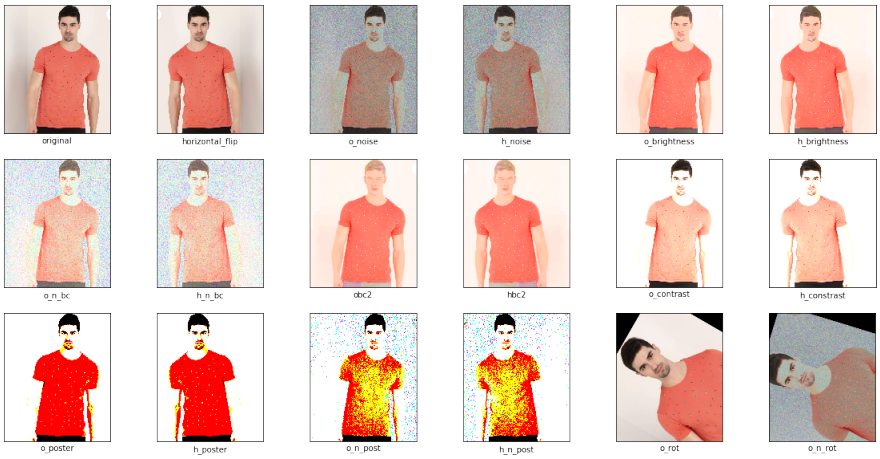
The dataset includes the images were procured from apparel purchase section from e-commerce websites like mazon, Flipkart and Myntra.The dataset is pretty varied across all colors and different poses. Since the camera is stationary, we have assumed it have similar lighting conditions in all the images. A small subset of the dataset collected is shown in the figure below.



Around 150 images from each class has been collected. The total number for the five classes turns out to be around 750. This is a very small number. To increase the size of the dataset, we have used some data augmentation techniques. This is explained further in the following subsection.

Data Augmentation

We have extracted 18 images from a single image. Each image is different in its own sense. All the images extracted from a single image is shown in the image below.



From the top left corner, the following are the 18 variations of the single image.

- Original image.
- Horizontal Flip of the original image.
- Adding Gaussian noise to the original image.
- Adding Gaussian noise to the horizontally flipped image.
- Increasing the brightness with one value in the Original image.
- Increasing the brightness with one value in the horizontally flipped image.
- Increasing the brightness with one value in the Original image with Gaussian noise.
- Increasing the brightness with one value in the horizontally flipped image with Gaussian noise.
- Increasing the brightness with other value in the Original image.
- Increasing the brightness with other value in the horizontally flipped image.
- Increasing the contrast in the original image.
- Increasing the contrast in the horizontally flipped image.

- Posterizing the original image.
- Posterizing the horizontally flipped image.
- Posterizing the original image with Gaussian Noise.
- Posterizing the horizontally flipped image with Gaussian Noise.
- Random rotation of the original image.
- Random rotation of the original image with Gaussian Noise.

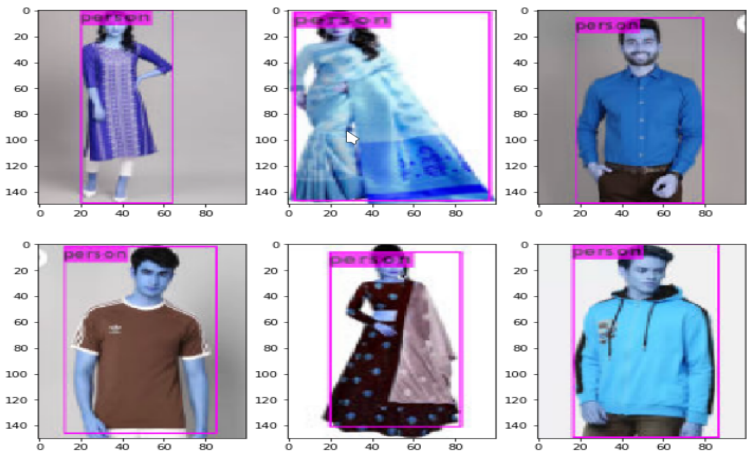
With all these augmentations, the total data set came out to be of the size 13680 images. Each class now has over 2500 images to train on.

Object Localization using YOLO

Once we have an image from the camera, we need to localize the person in the image before we can go on to classify the type of the dress put on by the person. YOLO is a state of the art object detection framework that is capable of detecting the objects of various classes it is trained to identify. In addition, it can also predict a bounding box enclosure for all of the identified objects in the image.

We have used YOLOv3 for our problem of person detection. It uses the darknet-53 architecture which is a deep CNN consisting of 106 layers. We used the pre-trained weights obtained by training it on MSCOCO (Common Objects in COntext) dataset which is a labelled dataset of 91 classes, with person being one of them.

We performed inference on each of the images in the dataset using the pre-trained model which gives as output the class confidence and the bounding box coordinates. Using this, we extract the person(s) from all of the images in our dataset and proceed to the next step of dress recognition. Below are some samples for person localization in an image.



The algorithm couldn't detect the person in a few images. This may be due to a lot of noise added after data augmentation or the person might even be cut out when the image was rotated. After using the algorithm on all the 13680 images, around 10,000 had a person detected in them. Moving forward, we cropped out the person out of the images and used them for apparel classification.

Apparel Classification using CNNs

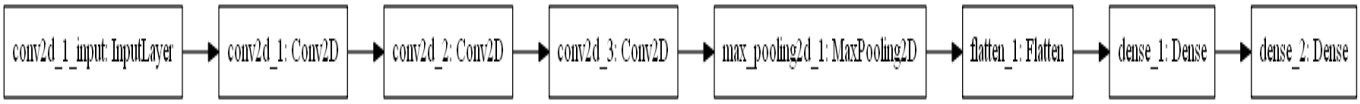
After getting around 10,000 images for apparel classification, we split the dataset into around 8,000 for training the model and the remaining 2,000 for testing. We have used various Convolutional Neural Network model for training and testing the dataset.

CNN 1

We used a basic CNN to extract features from the dataset. We later on used these extracted features and trained some other basic neural networks. The results weren't promising. The results are tabulated below.

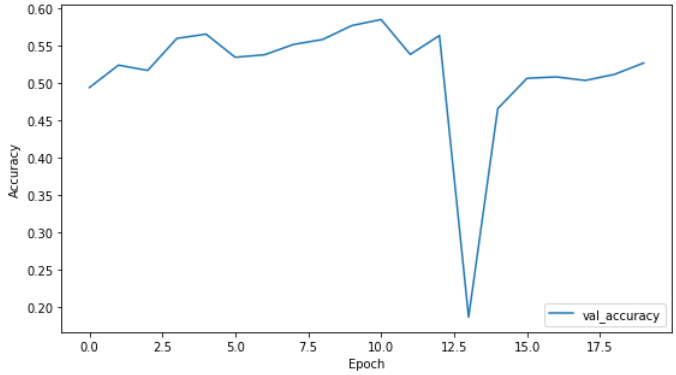
Model	Accuracy
Logistic Regression	21.28%
Neural Network [32, 64, 128]	25.21%
K Nearest Neighbors	19.81%

The model flow chart is shown below.



CNN 2

For the same above mentioned CNN we added a FCN. The results turned out be better than CNN 1. The accuracy came out to be 58.27%. The accuracy and loss graph is shown below.



AlexNet

We trained the dataset on Alex Net architecture. Instead of fine tuning the network, we trained the model from scratch. We tuned some hyperparameters in the network as well. The results are tabulated below.

Hyperparameter Update	Accuracy
Nothing	72.33%
Dropout = 0.5	74.12%
Dropout = 0.3 Activation function = LeakyReLU	76.52%

The final accuracy obtained is close to 77%.

VGG16

We have taken the weights of VGG16 from the ImageNet challenge and fine tuned the final fully connected layer. The result obtained was 65.29%. Here we think that training from scratch is better since the layers in the CNN are also trained according to the dataset. Therefore, we must've had better accuracy for AlexNet trained from scratch and lesser for Fine-tuned VGG16 network.

References

- Dataset from Amazon
- Dataset from Flipkart
- Dataset from Myntra
- YOLOv3 - Yolo-v4 and Yolo-v3/v2 for Windows and Linux - AlexeyAB
- MSCOCO dataset - cocodataset.org
- MSCOCO information - Microsoft COCO: Common Objects in Context
- An article on Fine-tuning with Keras and Deep Learning on pyimagesearch.
- Keras Documentaion
- The dataset collected can be viewed here: link
- The cropped set of images can be viewed here: link