CV Project -Face Mask Segmentation

Project Description:

In this hands-on project, the goal is to build a Face Mask Segmentation model which includes building a face detector to locate the position of a face in an image.

Data Description:

WIDER Face Dataset

WIDER FACE dataset is a Face Mask Segmentation benchmark dataset, of which images are selected from the publicly available WIDER dataset.
This data have 32,203 images and 393,703 faces are labeled with a high degree of variability in scale, pose and occlusion as depicted in the sample images.
In this project, we are using 409 images and around 1000 faces for ease of computation.

We will be using transfer learning on an already trained model to build our segmenter. We will perform transfer learning on the MobileNet model which is already trained to perform image segmentation. We will need to train the last 6-7 layers and freeze the remaining layers to train the model for face mask segmentation. To be able to train the MobileNet model for face mask segmentation, we will be using the WIDER FACE dataset for various images with a single face and multiple faces. The output of the model is the face mask segmented data which masks the face in an image. We learn to build a face mask segmentation model using Keras supported by Tensorflow.

Reference

Acknowledgment for the datasets.
http://mmlab.ie.cuhk.edu.hk/projects/WIDERFace/
Mobile Net paper: https://arxiv.org/pdf/1704.04861.pdf

Objective:

In this problem, we use "Transfer Learning" of an Image Segmentation model to detect any object according to the problem in hand.

Here, we are particularly interested in segmenting faces in a given image

Steps and Tasks:

- Load the dataset given in form .npy format.
  - We have already extracted the images from wider face-dataset and added it in the file images.npy. You can directly use this file for this project.
  - "images.npy" contains details about the image and it's masks, there is no separate CSV file for that
  - There is no separate train and test data given
- Create Features(images) and labels(mask) using that data.
- Load the pre-trained model and weights.
- Create a model using the above model.
- Define the Dice Coefficient and Loss function.
- Compile and fit the model.
- Evaluate the model.

** Note: Detailed Instructions for all the above steps are given in the notebook.

Project Support

You can clarify your queries related to the project by dropping a mail on Olympus

Happy Learning!