

# Bosch Age and Gender detection

---

IIT Ropar's Submission for the Bosch age and gender detection task as a part of the Inter IIT Tech Meet 2022

## Task

- Human detection in a low resolution video file
- Estimation of age and gender from the detected images in the objects

## Pipeline

- **Swin Transformer for Image Super Resolution of low quality video**

- MPII Human Pose Dataset used for training

<http://human-pose.mpi-inf.mpg.de/>

- Extract to the **dataset** directory. All images should be accessible at **dataset/mpii\_human\_pose\_v1/images**

- SwinIR implementation taken from the official implementation as described in the 2021 paper.

Paper Link :

[https://openaccess.thecvf.com/content/ICCV2021W/AIM/html/Liang\\_SwinIR\\_Image\\_Restoration\\_Using\\_Swin\\_Transformer\\_ICCVW\\_2021\\_paper.html](https://openaccess.thecvf.com/content/ICCV2021W/AIM/html/Liang_SwinIR_Image_Restoration_Using_Swin_Transformer_ICCVW_2021_paper.html)

Official Implementation : <https://github.com/JingyunLiang/SwinIR>

- **Face Recognition Library for Face extraction and bounding box prediction**

- Library used :

[https://github.com/ageitgey/face\\_recognition](https://github.com/ageitgey/face_recognition)

- **Vgg16 based model for age and gender prediction with deep ordinal regression to optimise grouping of bins for the age prediction task**

- Dataset used - UTK face

<https://susanqq.github.io/UTKFace/>

- Download the three parts and name them **utkface\_1.tar.gz**, **utkface\_2.tar.gz**, **utkface\_3.tar.gz** in the root directory
- Run **./utkface.sh** to extract and process the dataset

- Model used

<https://github.com/pytorch/vision/blob/main/torchvision/models/vgg.py>

- Deep Ordinal regression

<https://arxiv.org/abs/2006.15864>

## Training

### Image Super-Resolution

- To train the Transformer for image super-resolution, run the `train.py` file located in the root directory.
- The model has been implemented in the `swin_ir.py` and the dataloader for the same has been implemented in `data_loader.py` in the root directory. The model parameters and the data generators can be changed here.
- `predict_sr.py` can be used to test image super resolution on individual files.
- Output is 2x the resolution of original image.

### Face detection

- `face_detection.py` generates the bounding boxes in the top left bottom right CSS format

### Gender Detection

- `classifier/train_gender.py` trains the gender model

### Age Detection

- `classifier/train_age.py` trains the age model

#### Note :

- Training the models is an optional step as models trained by us have already been saved and provided along with the rest of the files.
- In case one does decided to train the model. It is recommended to have atleast 16GB of VRAM for quick and efficient training.

## Prediction

- To just predict the outputs and generate the `output.csv` file, use the `run_video.py` file. Run `python run_video.py -h` for a list and description of the arguments required
- The dataset does not need to be downloaded for prediction

## So, What's happening in the background?

- The predictor first takes in a video file, which is read using opencv, and each frame is processed, to generate a 3 channel RGB image.
- The image data is then fed to the image super-resolution model depending on the value passed to the arg parser. The image super-resolution is carried out using a swin transformer. The relevant links for reading more about this model have been given above.

- The above is performed for each frame. The processed image is then passed to the face detector, which detects the unique faces present in the frame and assigns them unique person ids. The person id for each person remains the same for further frames that they appear in, provided that the IOU (Intersection over Union) of their bounding box across consecutive images remains higher than 0.5. Do note that this depends on the person being detected in the previous. If for some reason, their face is obstructed, then the person id will be reset for them.
- The bounding box dimensions are then edited depending on whether super resolution was performed on the image, in order to match those of the original image to write in the output csv files.
- The faces are then taken from the bounding box and passed to the classifier models, which use VGG16 as their base layers.
- For the gender prediction task, simply, BCE (Binary CrossEntropy loss) is used.
- For the age prediction task, the ages are divided into multiple sets of age bins which map from ages 0-116. We use 10 such sets of bins with random division of bin widths. Then 10 separate outputs from the model are loaded to each of these bins and backpropagation performed for each of them. A custom loss function is used for this purpose defined in the Deep Ordinal Regression paper by Berg Et al. linked in the first section.
- The age output and bins are then predicted by taking weighted averages of each of the bins.
- The output is thus then saved into the output csv file.