Motivation

Given a particular sensor resolution and face recognition block, there is a trade-off between the camera’s field of view and the working distance. In this project, we explore a deep learning-based approach such that given a particular face recognition block, e.g. FaceNet, we can significantly reduce the input image size (e.g. from NxN to N/8xN/8) while matching as much as possible the original accuracy.

Dataset

Labeled face in the wild (LFW):
- 13,233 RGB images with a size of (250, 250)
- Training set: 11,910 images – 90%
- Development set: 972 images – 7.35%
- Test set: 351 images - 2.65%

Data preparation:
- Crop face bounding boxes out of images with an open-source face detector – MTCNN. All boxes are resized to (160, 160) with PIL.image.resize()
- Prepare training, dev/test datasets:
  - X, low-resolution face images, by down-sampling the bounding boxes to (20, 20) with PIL.image.resize()
  - Y, 128-dim embedding vectors, by feeding high-resolution bounding boxes (160, 160) to an open-source FaceNet
- Prepare data for performance comparison:
  - high-resolution images X’ with interpolation-based methods – nearest, bilinear and bicubic, from X
  - embedding vectors for X’

Model architecture:

Performance:

# of samples that our model performs better or worse than nearest, bilinear and bicubic interpolations

Future Work

To further improve the model performance, need to try:
- Bigger filters (i.e., 5x5) since the bicubic interpolation explores information from 4x4 neighboring pixels
- Different architectures, such as GAN framework or DenseNet
- Different loss functions, such as L1