FinalAssignment

April 26, 2020

1 Keypoint Detection and Image Generation

Your Name (s123456789)

- If there is any questions, please email liuzishun@gmail.com.
- Presentation: 6:30pm, May 4. Each student has 2-3 minutes.

2 Data

```
In [1]: from google.colab import drive
        drive.mount('/content/gdrive/')
        ROOT_FOLDER = './gdrive/My Drive/Colab Notebooks/MAEG5735-2020-Assignment4/'

        import glob
        print('\nContents in the data folder:')
        for x in glob.glob(ROOT_FOLDER+'data/*'):
            print(x)
```

```
Go to this URL in a browser: https://accounts.google.com/o/oauth2/auth?client_id=947318989803-

Enter your authorization code:
ůůůůůůůůůů
Mounted at /content/gdrive/

Contents in the data folder:
./gdrive/My Drive/Colab Notebooks/MAEG5735-2020-Assignment4/data/imgs1.npy
./gdrive/My Drive/Colab Notebooks/MAEG5735-2020-Assignment4/data/kpts1.npy
./gdrive/My Drive/Colab Notebooks/MAEG5735-2020-Assignment4/data/imgs2.npy
./gdrive/My Drive/Colab Notebooks/MAEG5735-2020-Assignment4/data/kpts3.npy
./gdrive/My Drive/Colab Notebooks/MAEG5735-2020-Assignment4/data/README.txt
```

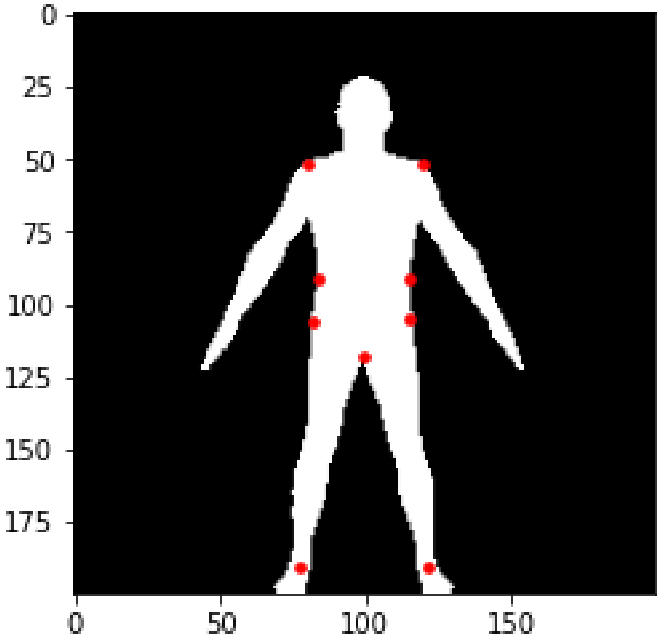
There should be 4 files: * `imgs1.npy` * `kpts1.npy` * `imgs2.npy` * `kpts3.npy`

```
In [5]: import matplotlib.pyplot as plt
        import numpy as np
```

```
def draw_points(image, kpts):
    plt.figure()
    plt.imshow(image, cmap='gray')
    keypoints = (kpts+0.5)*IMG_SIZE
    plt.scatter(keypoints[:, 0], keypoints[:, 1], s=50, marker='.', c='r')

# load side view data
IMG_SIZE = 200
IMG_TRAIN = np.load(ROOT_FOLDER+'data/imgs1.npy')
IMG_TRAIN = np.unpackbits(IMG_TRAIN).reshape((-1,IMG_SIZE,IMG_SIZE))
KPT_TRAIN = np.load(ROOT_FOLDER+'data/kpts1.npy')/IMG_SIZE - 0.5

# show one
idx = 10
draw_points(IMG_TRAIN[idx,:,:], KPT_TRAIN[idx,:,:])
```



3 Task 1: Keypoint Detection

- Train a keypoint detection model.
    - The same as Assignment #2. Any models besides neural networks can be used.
- Training data: `imgs1.npy` and `kpts1.npy`.
- Predict keypoints for images in `imgs2.npy`.

4 Task 2: Image Generation

In the previous task, we predict low dimensional data (keypoints `9x2`) from high dimensional input (images `200x200`). Let's consider a different case now.

- Train a image generation model to predict silhouette images from keypoints.
  - Input: `9x2` keypoints coordinates
  - Output: corresponding sihouette images
- All of the available data can be used, including the keypoints you predicted on `imgs2.npy` in the previous task.
- Predict silhouett images for keypoints in `kpts3.npy`.
  - Visualize the generated image together with input keypoints.