**Fine-grained Person Re-identification**

**Release 2 Documentation of task2**

**Centre for Smart Systems, SUTD**

**FEB 2021**

**Learning local sensitive feature instruction**

1. **Configure the environment**.
2. Install anaconda. <https://www.anaconda.com/products/individual>
3. Set up the environment for our project by the following commands.

* cd Human-Parsing
* conda env create -f reid.yaml

The conda environment named reid will be set up.

1. Then activate reid.

* conda activate reid

1. **Prepare the dataset to train the reid model.**

We will use the task 1 human parsing model to get the upper body and lower body of the person, which can be used to train the reid model together with the original image. We will put 3 kinds of images together in a directory for the dataloader to easily load the data for the training process. You can choose to use the prepared dataset directly(Option1) or build it by yourself(Option2).

**Option 1.**

We have uploaded the prepared dataset (both Market1501 and our fine-grained dataset) to Google drive, and you can simply download them from .................

We recommend you directly use them to train the reid model for convenience.

**Option 2.**

You can also prepare the data by the following steps:

Take Market1501 as an example, and you can also follow these steps to prepare our fine-grained dataset.

1. Download Market1501 and put it under ./Human-Parsing.
2. Use the human parsing model to get the binary mask of upper body and lower body respectively.

The next few steps are to get the masks of the upper body.

* python simple\_extractor.py

--get\_upper

--input-dir market/bounding\_box\_test

--output-dir market\_upper\_mask/bounding\_box\_test

* python simple\_extractor.py

--get\_upper

--input-dir market/bounding\_box\_train

--output-dir market\_upper\_mask/bounding\_box\_train

* python simple\_extractor.py

--get\_upper

--input-dir market/query

--output-dir market\_upper\_mask/query

The next few steps are to get the masks of the lower body.

* python simple\_extractor.py

--get\_lower

--input-dir market/bounding\_box\_test

--output-dir market\_lower\_mask/bounding\_box\_test

* python simple\_extractor.py

--get\_lower

--input-dir market/bounding\_box\_train

--output-dir market\_lower\_mask/bounding\_box\_train

* python simple\_extractor.py

--get\_lower

--input-dir market/query

--output-dir market\_lower\_mask/query

c) Run prepare.py to prepare market, market\_upper\_mask and market\_lower\_mask for training.

* python ./reid-baseline-code/prepare.py

--download\_path market

* python ./reid-baseline-code/prepare.py

--download\_path market\_upper\_mask

* python ./reid-baseline-code/prepare.py

--download\_path market\_lower\_mask

d) Rename the upper body and lower body images. Append ‘\_upper’ or ‘\_lower’ to the original file name. The results will be 0001\_c1s1\_001051\_03.jpg 🡪 0001\_c1s1\_001051\_03\_upper.jpg 0001\_c1s1\_001051\_03\_lower.jpg

* python ./reid-baseline-code/rename.py

--dir market\_upper\_mask/pytorch

--rename\_upper

* python ./reid-baseline-code/rename.py

--dir market\_lower\_mask/pytorch

--rename\_lower

1. Mix the original images and their corresponding upper body and lower body masks.

* cp -r market\_upper\_mask/pytorch market
* cp -r market\_lower\_mask/pytorch market

表格

描述已自动生成Finally, the data will be this format.

1. **Train the reid model.**

* cd ./reid-baseline-code
* python ./train\_IDE\_original.py

--name directory to save the trained model

--data\_dir ../market/pytorch

-- use\_three\_stream\_resnet

1. **Extract feature.**

* python test\_original.py

--name directory to load the trained model

--test\_dir ../market/pytorch

--cross the name of feature

--training\_set\_classes number of classes of training set (Market1501’s training set has 751 classes)

-- use\_three\_stream\_resnet

1. **Get the final results.**

* python evaluate.py

--name same as 3 and 4

--cross the name of feature

--logs\_dir directory to save the log