Data

Qingfu Wan

strawberryfgalois@gmail.com

1. Intro

Before the burgenoning development of SSD, people typically relied on offline data generation. The downside of doing this is it is really inefficient. For instance, using Visual Studio C++ to generate HDF5 file for 100000 images takes about 2 hours. Python is even slower. The pioneering work of CPM first generates JSON, and then generates LMDB file (which takes about 1 hour or so). The generated LMDB for MPII is around 125 GB. **hmmmm** CPM Data Layer then fetches lmdb and reads meta data to parse image and other annotations. The thing is this preprocessing step can be totally obliterated as online data fetch or online data augmentation has become feasible with no overhead.

If you use python, it is convenient to load txt as numpy.

2. txt

In this repo, I use txt files to store all the annotation, each type in a separate folder. It is simple and naive. More importantly, it is super fast to locate information about a sample. Say, you want to get access to 3d ground truth, bbx of sample 32456. All you need to do is to visit gt 3d folder & bbx folder, and open 32456.txt. Basic information including frame_id, action_id, subject_id, are also stored for reference.

2.1. Train

The index range of training set is 0 to 1559571

2.2. Test

The index range of testing set is 1559572 to 2108570

3. 32 vs 16 joints

The whole h36m dataset provides 32 joints, of which we only care about 16 joints.
In h36m.h, see **enum**
1. joint_all_h36m
2. joint_part_h36m
3. bone_all_h36m
4. bone_part_h36m

and array
1. index_joint_in_all
2. index_joint_in_part
3. color_gt_joint_all_h36m
4. color_gt_joint_all_h36m
5. color_pred_bone_all_h36m
6. color_gt_bone_all_h36m
7. color_pred_bone_part_h36m
8. color_gt_bone_part_h36m

for more details.

4. Better data organization

As also pointed out in code.pdf, a more efficacious way of organizing data is to unify all annotation of the same sample into only one individual file. And then use a layer that fetches all the annotation for each sample. Caffe supports **MinTopBlobs**, and so it is easy to fetch partial annotation in cases where you have no interest in some annotations like gt 2d.