DetectorPLS

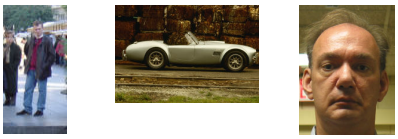
version 0.1.1

(project webpage)

Author: William Robson Schwartz (schwartz@cs.umd.edu)

Capabilities

- Perform object detection (studied for humans and faces).
- Learn new PLS models for classes of objects.
- Good for:
  - Objects that could have templates designed for.



- Bad for:
  - Objects with high intra-class variability.

Detection

- Input data:
  - Single Image – user may specify multiple regions of interest;
  - Directory with images;
  - Stored video;
  - Stream from a camera;
- Output:
  - Overlay bounding boxes (rectangle, ellipse) and write in a text file;
  - Non-maximum suppression is an option;
- Interface to the user:
  - Config file;
  - Arguments in the command line;

Detection

- Config file:
  - Specify training the set to be used in each stage;
  - Specify which detectors will be used;
  - Specify the scale for each object;

```
# comment

# body detector
model <Body, 0, 0.2, learnedFullBody64x128FastHOG.ret02.yml>
model <Body, 1, 0.999999, All.64x128Learning.ret03.yml>
scale <Body, 65, 400, 0.15, 0.05, 1.1>

# face detector
model <Face, 0, 0.9999, learnedFaceInternet.ret03.yml>
scale <Face, 30, 140, 0.15, 0.05, 1.1>
```

```
scale <object name, min size, max size, delta_X, delta_Y, scale>
```

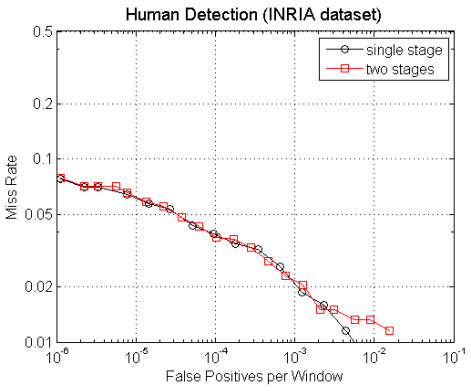
Detection

- Command line options:
  - input, output, file mask;
  - Overlay or not bounding boxes;
  - Perform non-maximum suppresion;
  - Specify first, last frame of a video and how many to skip;

```
./detectorPLS -d -c ParamsExecutionNew64x128.txt -i wm2.ppm \
    -o output -s -b
```

Human Detection

- Human detection training models are provided.
  - Detection window size 64x128.



  - It takes approximately 100 seconds per frame (640x480) considering 16 scales.

Learning New PLS Models

- User provides directories with positive and negative samples.
- Options:
  - Cross-validation;
  - Retraining – add samples that were misclassified;
  - Choice of features, block sizes and strides;
  - Use of QDA or PLS regression;
  - Specify the detection window and object size;
  - Specify regions inside the detection window (part-based approaches);
- Reduction in memory consumption:
  - Option to use block-based PLS (reduces about 8x memory consuption).

```
# internal identifier (ignore this)
ModelID <FaceInternet42x48AllFeat>

# feature methods used
method <HOG,HOG,0,HOGPARAM>
method <COOCH,COOC,1,COOCPARAM>
method <COOCS,COOC,1,COOCPARAM>
method <COOCV,COOC,1,COOCPARAM>

# feature parameters
params <COOCPARAM,bins,16>
params <COOCPARAM,distance,1>
params <COOCPARAM,cache,1>

params <HOGPARAM,UseColorFrequency,1>
params <HOGPARAM,UseHOG,1>
params <HOGPARAM,cache,0>

# block sizes and strides for each feature
block <HOG,16,16,4,4,5>
block <HOG,32,32,8,8,5>

block <COOC,16,16,4,4,5>
block <COOC,32,32,8,8,5>

# classifier to be used and number of factors
classifier <qda>
factors <20>

# detection window and object size
window <42,48>
object <5,6,30,34>

# region inside the detection window to be considered
region <0,0,41,47>
```

Future Versions

- Incorporate temporal feature caching for videos.
- Allow features to be loaded as plugins.