📖 **README.md**

hide-and-seek-tensorflow

Tensorflow implementation of [Hide-and-Seek: Forcing a Network to be Meticulous for Weakly-supervised Object and Action Localization, ICCV 2017](#)

Description

Implement CAM and HAS in Tensorflow. Unlike the original paper, it is only tested on [Tiny ImageNet](#).

Prerequisites

All codes are tested at Linux environment only.

- Python 2.7
- NumPy
- Scipy
- Scipy.misc
- Matplotlib
- Tensorflow 1.2 (or 1.3)
- [imgaug](#)

Installing

First clone the repository.

```
git clone https://github.com/seokjunS/hide-and-seek-tensorflow
cd hide-and-seek-tensorflow
```

Prepare `data` directory.

```
mkdir data
```

Downlaod Tiny ImageNet dataset at `data` and unzip it.

```
cd data
wget http://cs231n.stanford.edu/tiny-imagenet-200.zip
unzip tiny-imagenet-200.zip
cd ..
```

Generate data. (labels.txt, train.tfrecord, valid.tfrecord)

```
python src/tiny_imagenet.py gen_data
ls -l data
```

Running the models

Explain how to run various models. Basic models used in paper is AlexnetGAP (at `src/alexnet_gap`) and GooglenetGAP (at `src/googlenet_gap`). For intensive experiment, I implemented variation of models including,

- SmallAlexnetGAP - `src/small_alexnet_gap.py` : Resized Alexnet in order to feed 64x64 images.
- AlexnetGMP - `src/alexnet_gmp.py` : Replace GAP to GMP.
- DropAlexnetGAP - `src/drop_alexnet_gap.py` : Add dropout after mean normalization.
- Drop2AlexnetGAP - `src/drop2_alexnet_gap.py` : Add dropout before mean normalization.

- CustomnetGAP - `src/customnet_gap` : Introducing fire module of SqueezeNet.

Training

Following command is an example of training AlexnetGAP with hiding 16 patches.

```
python src/train.py --method AlexnetGAP \
                    --train_dir models/alexnet_gap \
                    --max_epoch 200 \
                    --batch_size 128 \
                    --do_hide 16
```

To feed 64x64 image to original model (without resize) set `--without_resize True` option.

Two types of augmentation is implemented and can be set as '--do_augmentation 1' or '--do_augmentation 2' (only for AlexnetGAP).

- Augmentation 1

```
data = iaa.Sequential([
        iaa.Fliplr(0.25),
        iaa.Flipud(0.25),
        iaa.Sometimes(0.25, iaa.Affine(
          rotate=(-180, 180)
        )),
        iaa.Sometimes(0.2, iaa.Affine(
          translate_percent={'x': (-0.15, 0.15), 'y': (-0.15, 0.15)}
        ))
      ]).augment_images(data)
```

- Augmentation 2

```
data = iaa.Sequential([
        iaa.Fliplr(0.25),
        iaa.Flipud(0.25),
        iaa.Sometimes(0.25, iaa.Affine(
          rotate=(-180, 180)
        )),
        iaa.Sometimes(0.2, iaa.Affine(
          translate_percent={'x': (-0.1, 0.1), 'y': (-0.1, 0.1)}
        )),
        iaa.Sometimes(0.2, iaa.OneOf([
          iaa.CoarseDropout(0.2, size_percent=(0.05, 0.1)),
          iaa.CoarseSalt(0.2, size_percent=(0.05, 0.1)),
          iaa.CoarsePepper(0.2, size_percent=(0.05, 0.1)),
          iaa.CoarseSaltAndPepper(0.2, size_percent=(0.05, 0.1))
        ]))
      ]).augment_images(data)
```

Please use `--help` command for options and descriptions.

Evaluation

After training, to evaluate and visualize, please use `src/test.py` . Test is done in various localization thresholds {0.2, 0.3, 0.4, 0.5}

Following command is an example of evaluating trained AlexnetGAP.

```
python src/test.py --method AlexnetGAP \
                   --checkpoint models/alexnet_gap/ckpt/model-200
```

To visualize several well localized object, please set `--do_vis Ture` option. To do a multi-crop test (10 crops: 4 corners + center, also for flipped one), please set `--do_multi_crop` option.
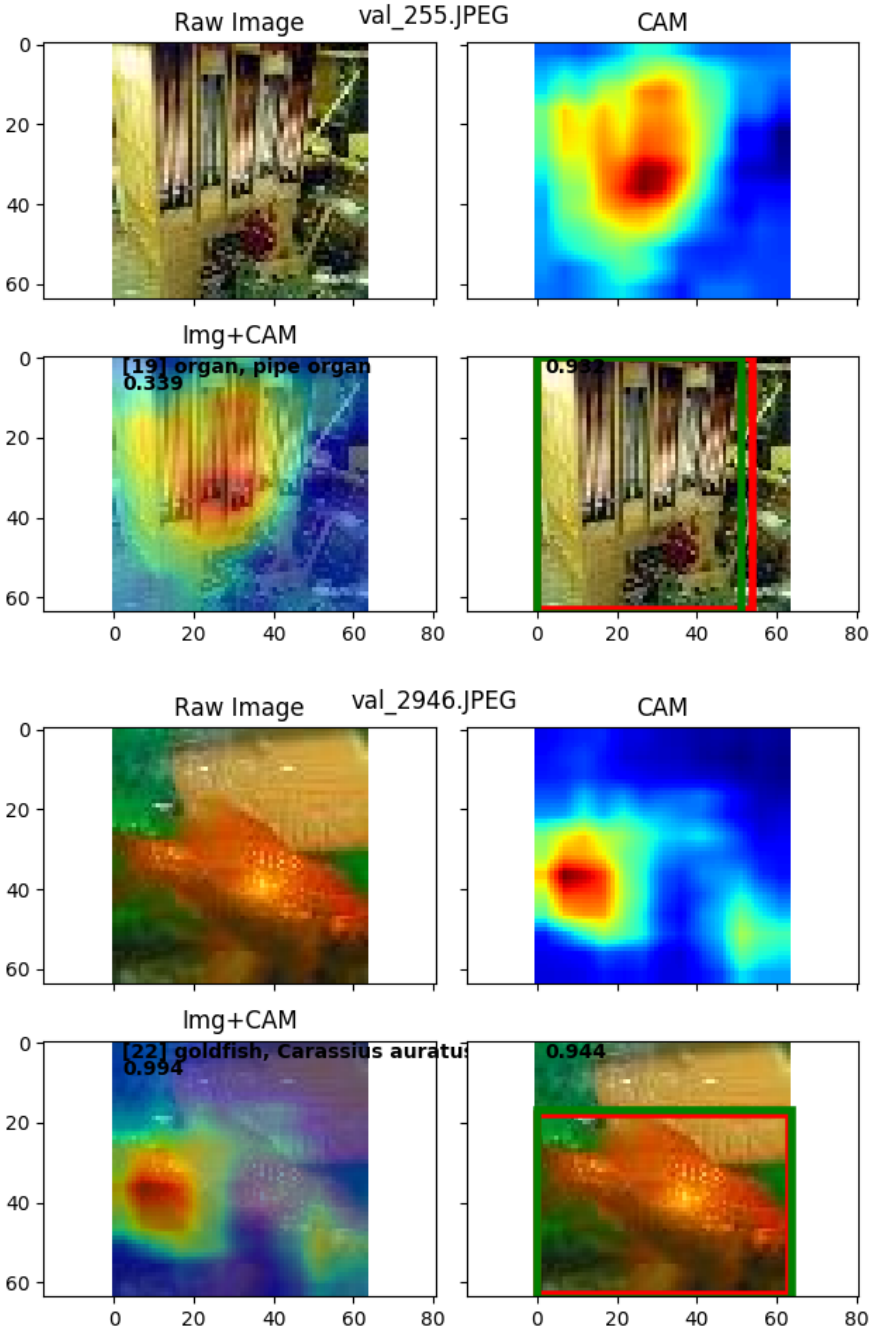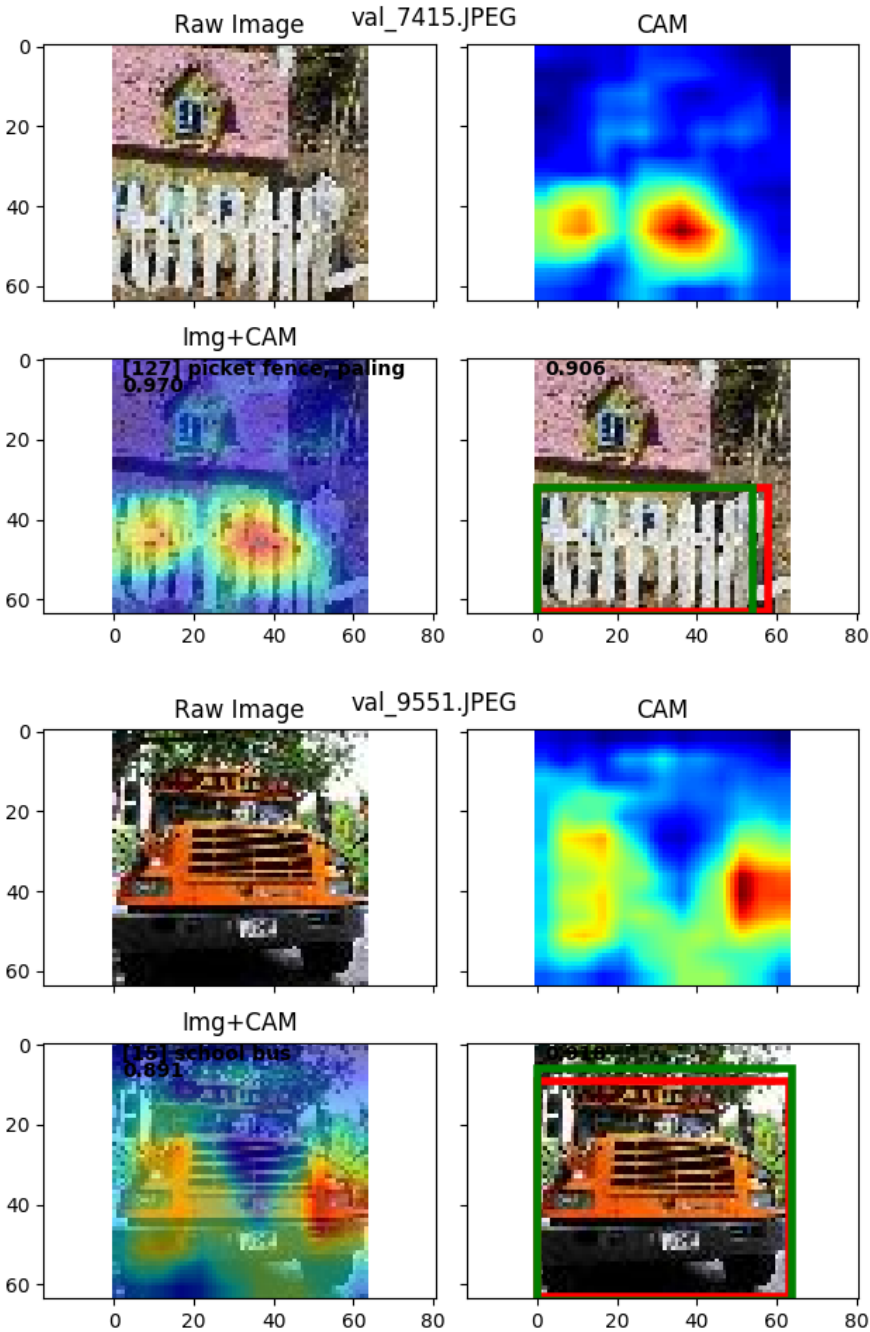
Sample Results

Result Table

Tests are done without multi-crop test and reported values are the highest one among four localization thresholds.

Methods	Best			Methods	Best		
	GT-known Loc	Top-1 Loc	Top-1 Clas		GT-known Loc	Top-1 Loc	Top-1 Clas
AlexNet-GAP	54.31	24.53	39.27	Drop-Without-Norm	50.70	0.89	1.16
AlexNet-HAS-16	54.70	26.31	42.23	Drop-With-Norm	52.85	16.15	26.90
AlexNet-HAS-36	54.86	26.22	42.24	AlexNet-GMP	50.72	23.46	42.49
AlexNet-HAS-64	55.22	27.74	44.24	AlexNet-GMP-HAS-16	51.07	24.25	43.28
AlexNet-HAS-Mix	55.04	28.47	45.21	AlexNet-w/o-Resize	48.37	14.30	26.21
GoogleNet-GAP	55.47	30.22	47.63	AlexNet-w/o-Resize-HAS-16	48.35	15.97	28.96
GoogleNet-HAS-16	55.75	**32.24**	**51.38**	AlexNet-Small	53.91	23.87	38.26
CusNet-GAP	53.35	23.29	37.89	AlexNet-Small-HAS-16	55.68	26.21	40.16
CusNet-HAS-16	55.06	28.74	45.80	AlexNet-AUG-1	56.65	29.84	46.59
CusNet-HAS-36	55.41	29.68	47.80	AlexNet-AUG-2	**56.62**	**31.06**	**48.09**
CusNet-HAS-64	54.95	29.18	47.01	AlexNet-AUG-1-HAS-16	**56.92**	29.50	46.01
CusNet-HAS-Mix	55.68	**31.20**	**49.70**	AlexNet-AUG-2-HAS-16	**56.70**	29.87	45.98

Visualizations

val_7415.JPEG

val_9551.JPEG

Contact

- **Seokjun Seo** - seokjunS