

---

## DeepDeblur\_release

Single image deblurring with deep learning.

This is a project page for our research. Please refer to our CVPR 2017 paper for details:

Deep Multi-scale Convolutional Neural Network for Dynamic Scene Deblurring [paper] [supplementary] [slide]

If you find our work useful in your research or publication, please cite our work:

```
1 @InProceedings{Nah_2017_CVPR,  
2   author = {Nah, Seungjun and Kim, Tae Hyun and Lee, Kyoung Mu},  
3   title = {Deep Multi-Scale Convolutional Neural Network for Dynamic  
4     Scene Deblurring},  
5   booktitle = {The IEEE Conference on Computer Vision and Pattern  
6     Recognition (CVPR)},  
7   month = {July},  
8   year = {2017}  
9 }
```

## PyTorch version

PyTorch version is now available: <https://github.com/SeungjunNah/DeepDeblur-PyTorch>

## New dataset released!

Check out our new **REDS** dataset! In CVPR 2019, I co-organized the 4th NTIRE workshop and the corresponding video restoration challenges. We released the **REDS** dataset for challenge participants to train and evaluate video deblurring / super-resolution methods. Special thanks go to my colleagues, Sungyong Baik, Seokil Hong, Gyeongsik Moon, Sanghyun Son, Radu Timofte and Kyoung Mu Lee for collecting, processing, and releasing the dataset together.

## Updates

Downloads are now available for training, validation, and test input data. A public leaderboard site is under construction. Download page: <https://seungjunnah.github.io/Datasets/reds>

## Dependencies

- torch7

- 
- torchx

```
1  luarocks install torchx
```

- cudnn

```
1  cd ~/torch/extra/cudnn
2  git checkout R7 # R7 is for cudnn v7
3  luarocks make
```

## Code

To run demo, download and extract the trained models into “experiment” folder.

- models

Type following command in “code” folder.

```
1  qlua -i demo.lua -load -save release_scale3_adv_gamma -blur_type gamma2
   .2 -type cudaHalf
2  qlua -i demo.lua -load -save release_scale3_adv_lin -blur_type linear -
   type cudaHalf
```

To train a model, clone this repository and download below dataset in “dataset” directory.

The data structure should look like “dataset/GOPRO\_Large/train/GOPRxxxx\_xx\_xx/blur/xxxxxx.png”

Then run main.lua in “code” directory with optional parameters.

```
1  th main.lua -nEpochs 450 -save scale3 # Train for 450 epochs, save in '
   experiment/scale3'
2  th main.lua -load -save scale3 # Load saved model
3  > blur_dir, output_dir = ...
4  > deblur_dir(blur_dir, output_dir)
```

Optional parameters are listed in opts.lua

ex) -type: Operation type option. Supports cuda and cudaHalf. Half precision CNN has similar accuracy as single precision in evaluation mode. However, fp16 training is not meant to be supported in this code. ADAM optimizer is hard to use with fp16.

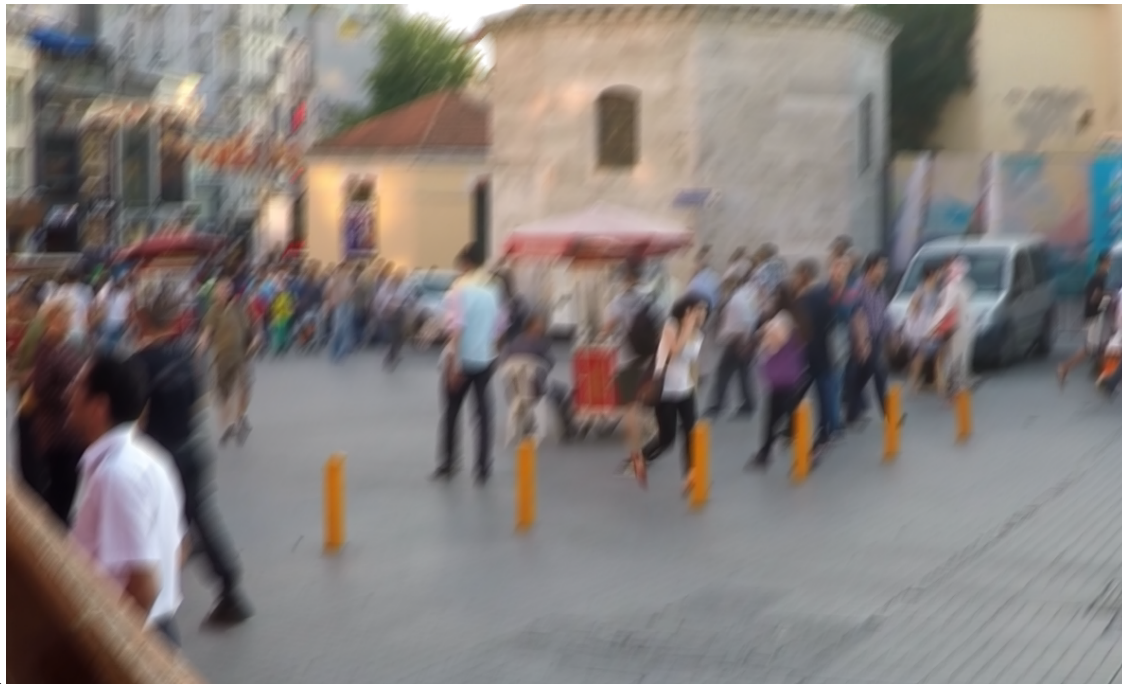
## Dataset

In this work, we proposed a new dataset of realistic blurry and sharp image pairs using a high-speed camera. However, we do not provide blur kernels as they are unknown.

- 
- Downloads available [here](#)

Statistics	Training	Test	Total
sequences	22	11	33
image pairs	2103	1111	3214

Here are some example images.



Blurry image example 1



Sharp image example 1



Blurry image example 2





Sharp image example 2

### **Acknowledgment**

This project is partially funded by Microsoft Research Asia