EE838A

Special Topic on Image Engineering: Advanced Image Restoration and Quality Enhancement

Homework Assignment 3

Implementation and Verification of deep learning based image deblurring networks

The third home assignment is to implement a given deep neural network for single image deblurring and to train it properly. After training, the performance verification is required for test images by providing the performance measures in terms of PSNR, SSIM and MS-SSIM (Multi-scale SSIM).

The data set for experiments is provided via the below Dropbox link where the three sets of training and validation images. Note that the test set is not provided. Instead, the submitted trained networks by the class students will be tested for evaluation with some test images that are not used for training and validation periods. The evaluation of the submitted trained networks for Homework Assignment 3 will be performed based on the correct implementation and performance of the given network.

Dataset

- Training samples: GOPRO dataset (2,103 pairs)
- Validation samples: GOPRO dataset (1,111 pairs)
- Download links:
  - Training/Validation data: https://github.com/SeungjunNah/DeepDeblur_release

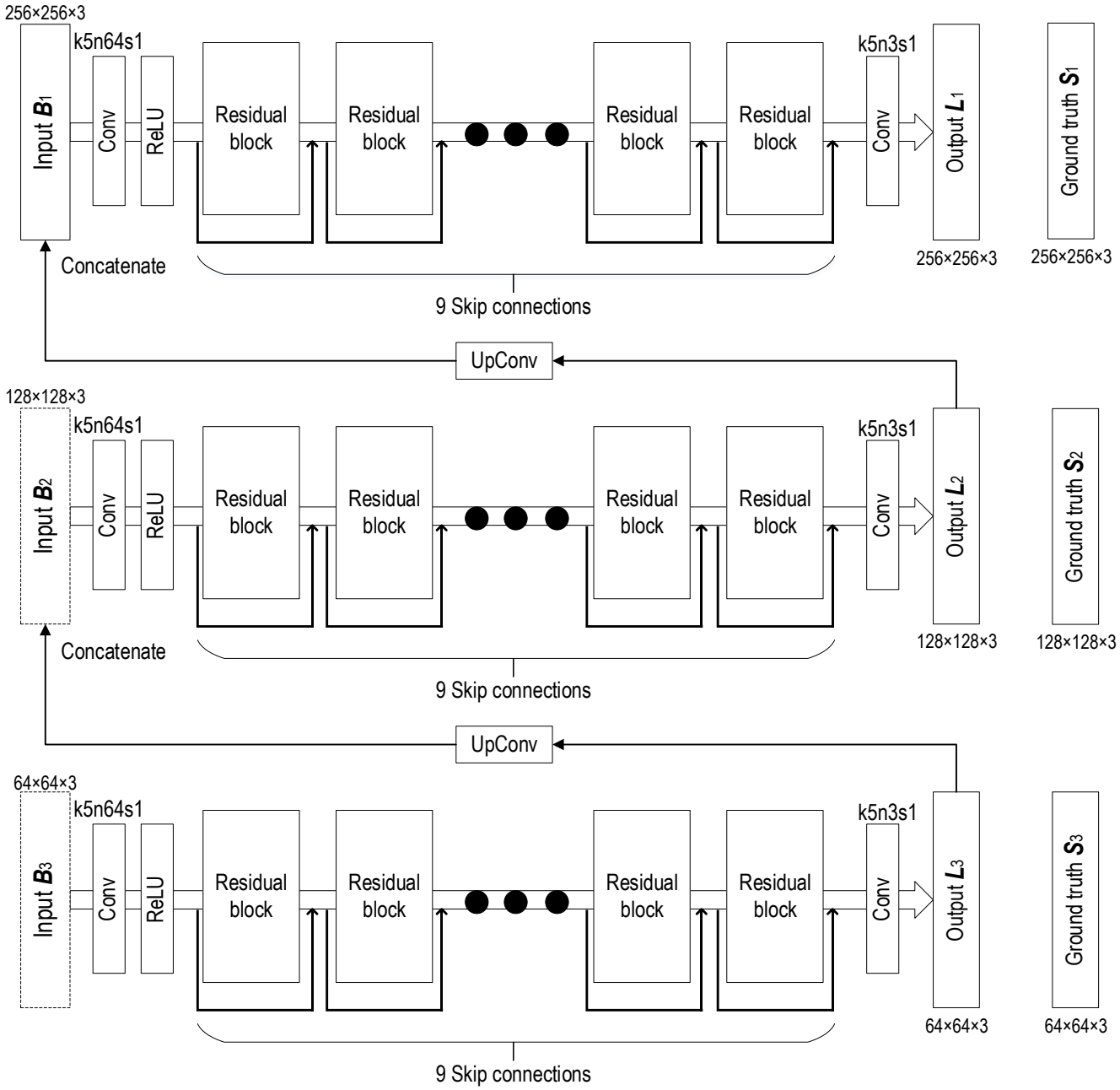
Implementation

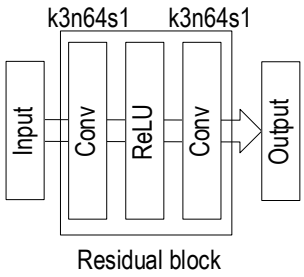
- Adam Optimizer
- Multi-scale content loss function

$$L = \frac{1}{2K} \sum_{k=1}^{K} \frac{1}{c_k w_k h_k} \left\| L_k - S_k \right\|^2$$

  where k is the pyramid level index with $K = 3$, $c_k = 3$ is the number of the output channels at the at the k-th pyramid level, and w_k (h_k) is the width (height) of the outputs at the k-th pyramid level.

- Training patch size : 256×256×3 RGB images
- Batch size : any number (defined by users)
- You can use several data augmentation techniques such as randomly flipping horizontally and vertically, and rotating by 90 degrees etc.
- Other hyper-parameters for training can be freely selectable.
- The multi-scale network architecture for image deblurring is supposed to implement:





- k5: kernel size of 5x5
- n3, n64: 3, 64 output channels
- s1: stride 1 for convolution
- UpConv: any transposed convolution filter
- All the skip connections indicate the element-wise summations to the output feature maps of the convolution layers
- Input $\boldsymbol{B}_2$ and $\boldsymbol{B}_3$ are produced by Bicubic downsampling of original input $\boldsymbol{B}_1$.

The following deliverables must be submitted:

- Both training and test codes (Tensorflow or Pytorch)
- Readme.txt (A simple description of how to execute your codes)
- Report (including the experimental conditions, the deblurred images produced by your test code for the three validation images (GOPR0384_11_00/blur/000001.png, GOPR0384_11_05/blur/004001.png, GOPR0385_11_01/blur/003011.png), the analysis of your results and a simple code description for each

component of the neural network)

Submission

- Due date: 2018-11-15 (Thu.) 23:59
- Submission should go to the class TAs at: shki@kaist.ac.kr and pys5309@kaist.ac.kr
- Submission format
  - Your report must include your name, student ID, phone number and e-mail
  - Your report must be in ZIP format with following directories:
    - ✓ `source` where readme.txt, training code and test code must reside
    - ✓ `report` where your report is put
  - The file name of your submission should be "HW3_studentID_YourName.zip".