IMPROVING EXTREME LOW-LIGHT IMAGE DENOISING VIA RESIDUAL LEARNING

Paras Maharjan, Li Li, Ning Xu, Chongyang Ma, Yu Li
2016

By: Ofir Yaish & Maor Asif

Outline

- Low light imaging
- Classical Algorithms
- Previous Deep learning models
- Paper innovation
- Results
- Project
- Questions

Low light imaging

- Taking a satisfactory picture in a low-light environment remains a challenging problem

- Low-light imaging mainly suffers from noise due to the low signal-to-noise ratio, and low photon count.

Low light imaging

- Low light imaging can arise from 2 main factors:
  - *Low light environment*
  - *Short exposure time of the camera*

Low light imaging

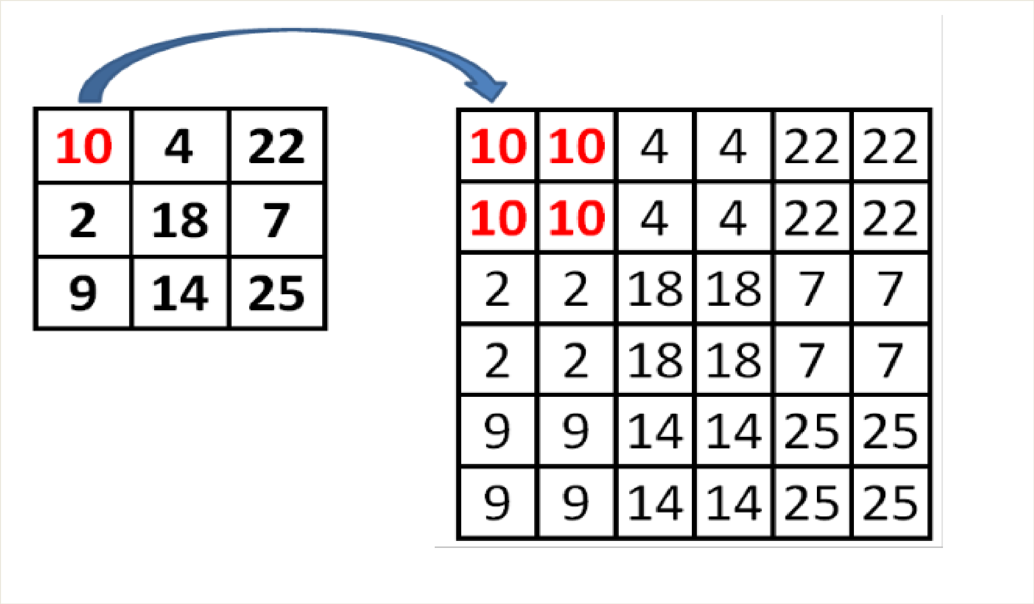
- Technology is constantly improving
- Low light imaging can be in fields such:
  - *Defense industry*
  - *Videos (short light exposure)*
  - *Smartphone*
  - *Agriculture*

Super Resolution

- Super Resolution is the process of recovering a High Resolution (HR) image from a given Low Resolution (LR) image
- An image may have a “lower resolution” due to a smaller spatial resolution (i.e. size) or due to a result of degradation (such as blurring)

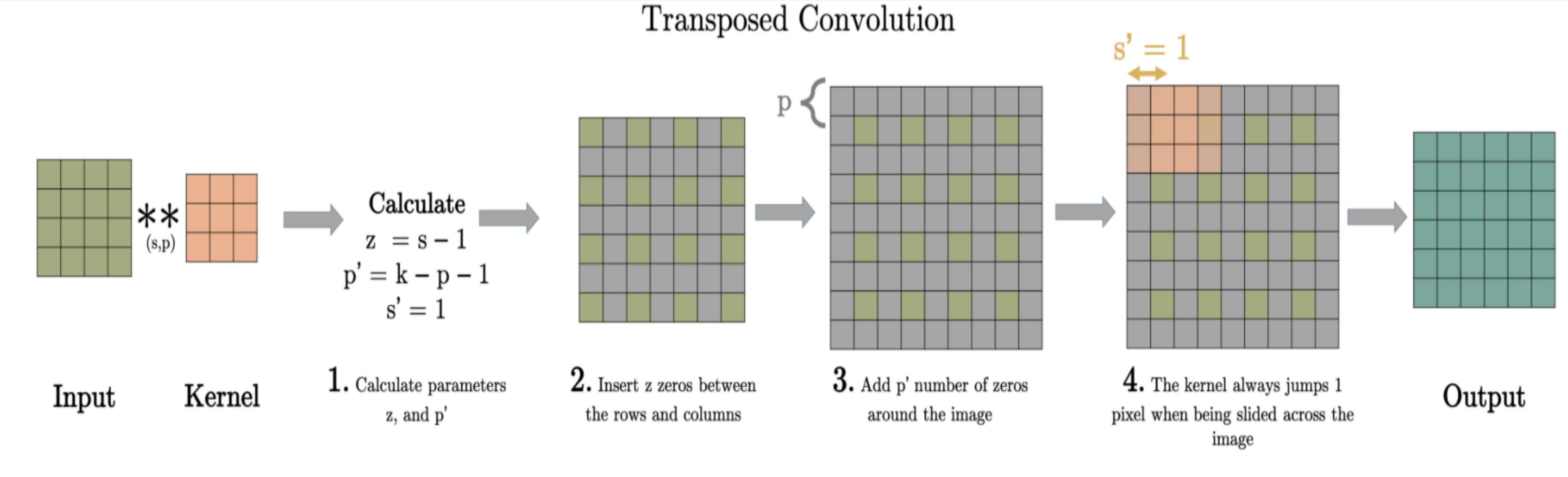
Up-sampling unit
Nearest Neighbor interpolation

- A simple method of multivariate interpolation in one or more dimensions
- The nearest neighbor algorithm selects the value of the nearest point and does not consider the values of neighboring points at all, yielding a piecewise-constant interpolant



Up-sampling unit
Transposed Convolution Layer

- Generate an output feature map that has a spatial dimension greater than that of the input feature map
- Pixels are padded and then convoluted which enables higher resolution



Classical Algorithms

- Until recent years Deep Learning algorithms were not commonly used (if any)

- Classical methods used filters and transformations to enhance the image readability

- Image Denoising by Sparse 3-D (2007)
  Dimensionality transformation. Still commonly used as a benchmark

- Image Denoising via Sparse and Redundant Representations Over Learned Dictionaries

  K-SVD algorithm. Two Israeli researchers

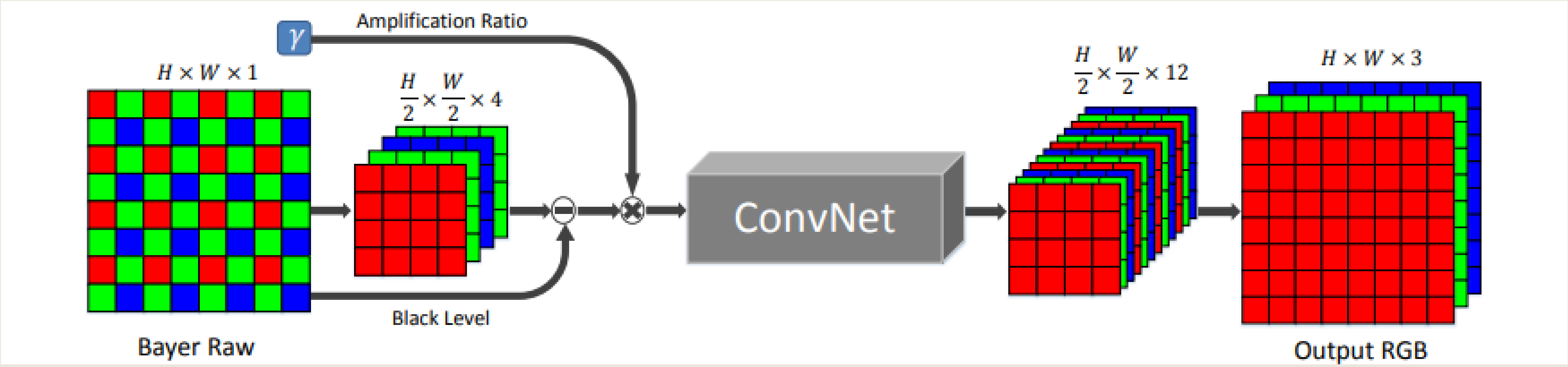
Deep Learning methods

- In the recent decay Deep Learning method had overcome classical methods for image processing tasks
- Neural network can adjust non-linear functions
- Since defining a problem is difficult when it comes to images neural network suits well
- Convolutional neural networks are the preferred choice for image processing (although in the last two years several other architectures managed to equalize their performance)

Learning to See in the Dark

- Deep Learning algorithm to approach extreme low light imaging, 2018

- The neural network is U-Net based



Learning to See in the Dark - dataset

- Independent dataset collected from two cameras:
  - *Sony α7S II*
    full-frame Bayer sensor
    Resolution: 4240x2832

  - *Fujifilm X-T2*
    APS-C X-Trans sensor
    Resolution: 6000x4000

Learning to See in the Dark - dataset

- The cameras were mounted on sturdy tripods
- Long exposure image was taken first – 424 images
- Afterwards using the same location, a short exposure image was taken – 5094 images

Improving Extreme Low-light Image Denoising via Residual Learning

- New Deep Learning algorithm published 2019

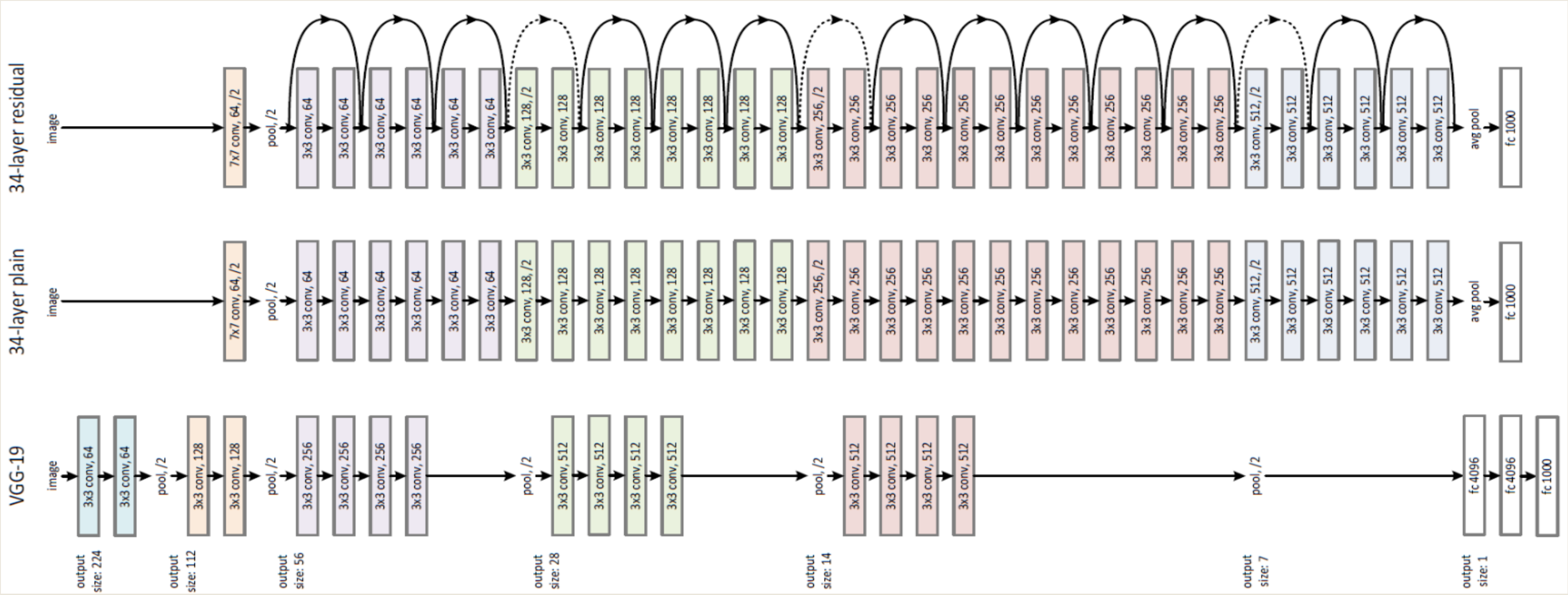
- The data is comprised only from Sony α7S II images

- The paper aim is to design a fast inference network to improve extreme low light imaging

Architecture

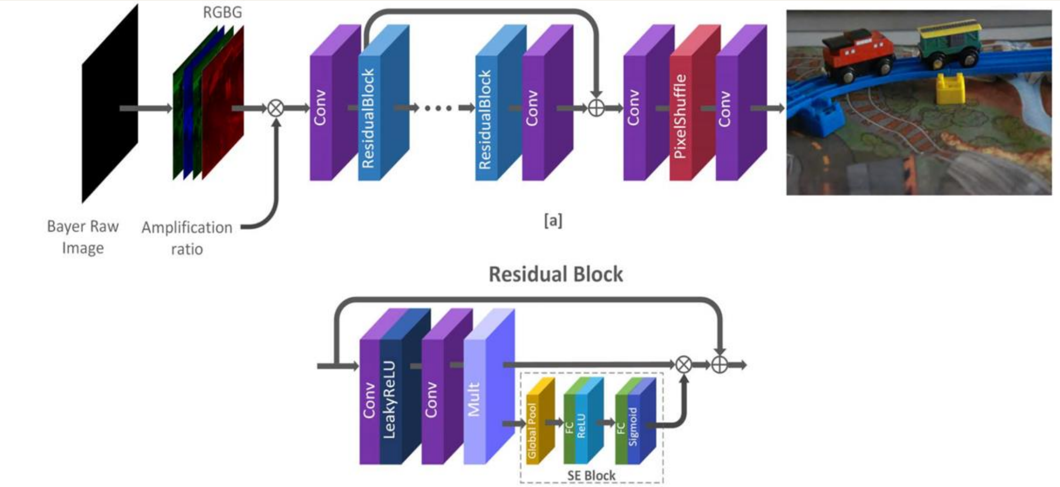
- The chosen architecture was based on residual neural network (ResNet)

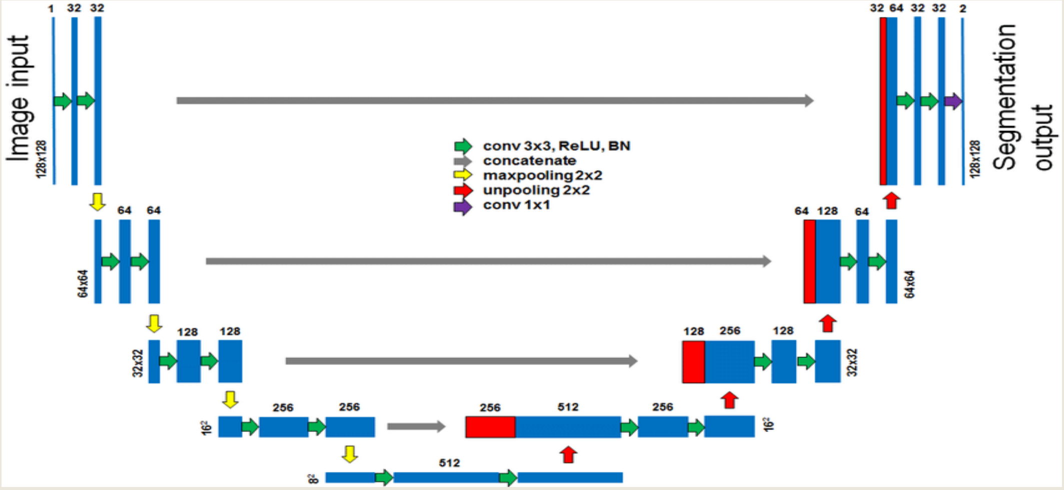
- ResNet is a convolutional neural network architecture that based on skip connections.



Architecture

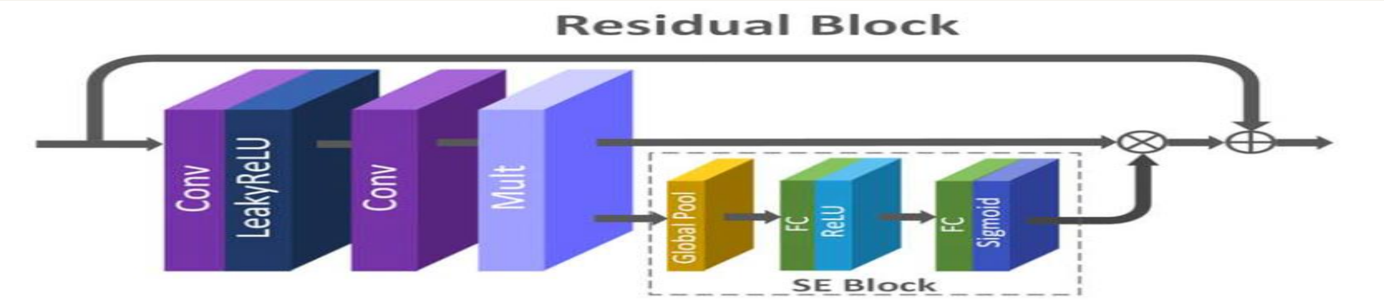
- U-Net graph requires calculation from all phases of the net – slow learning procedure
- ResNet graph is straightforward
- Therefore, ResNet is superior for fast inference application such as video
- The current U-Net architecture based on max polling layer for feature down-sampling which causes to lose of image details and generates output with blurry edges





Architecture

- Each residue block contains:
  1. *A first 3x3 convolution layer*
  2. *Followed by a Leaky ReLU layer*
  3. *A second 3x3 convolution layer*
  4. *A constant linear scaling unit*
  5. *Finally the output layer which is re-calibrated by an Squeeze and Excitation block*
- Leaky ReLU layer instead of ReLU to preserve important information



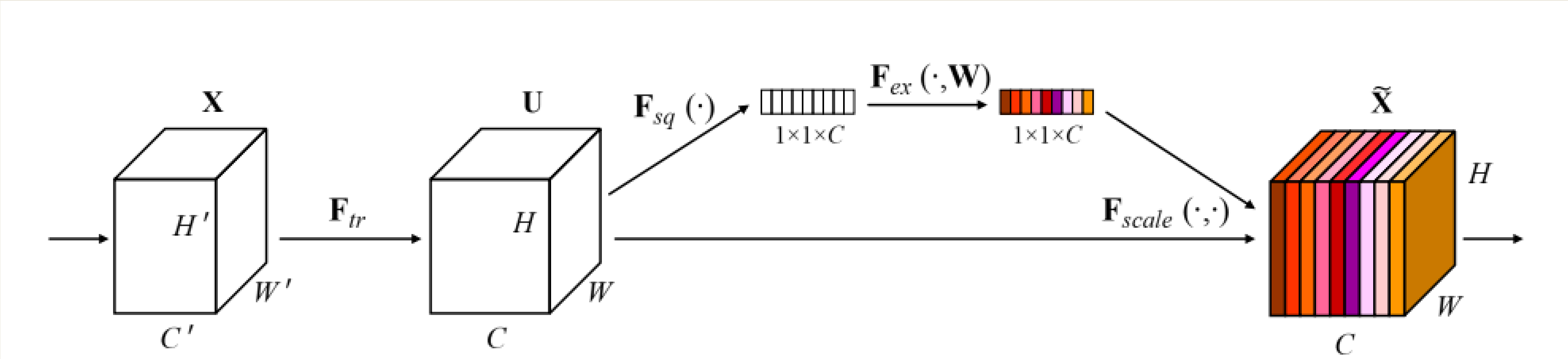
- The Output is up Sampled x2 using convolution layers with pixel shuffling

Squeeze and Excitation

- Deep Learning block comprised from two phases

- Published in 2017

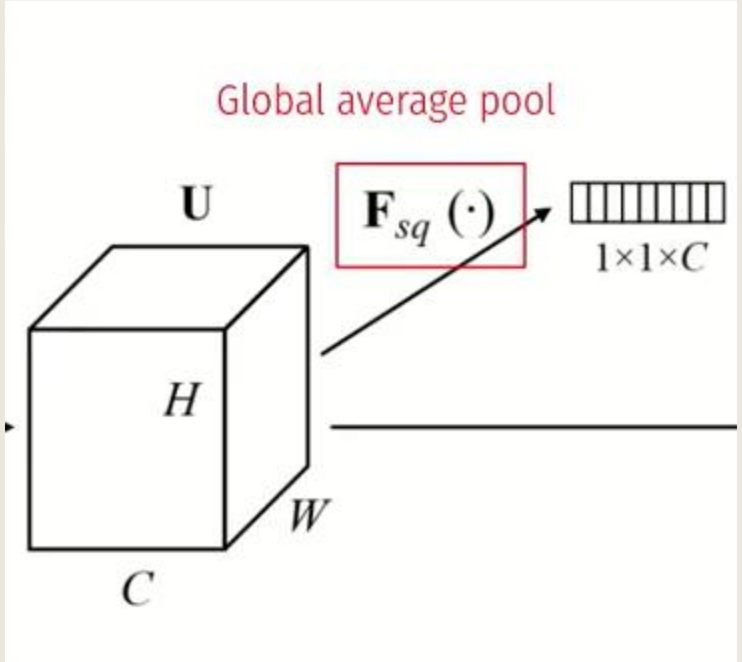
- Squeeze and Excitation block improves the feature representation of network by using the channel wise feature scaling



Squeeze

- Produce a channel descriptor by aggregating features maps across their spatial dimension

- This is just a global average pooling along the convolution channels



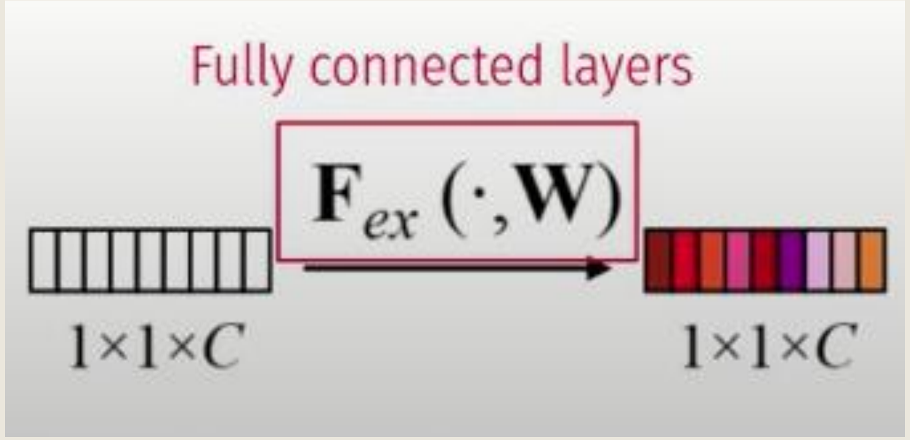
$$z_c = \mathbf{F}_{sq}(\mathbf{u}_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i, j)$$

Excitation

- Produce a collection of per-channel modulation weights

- Composed from 2 fully connected layers

- It can be viewed as channel “excitation”

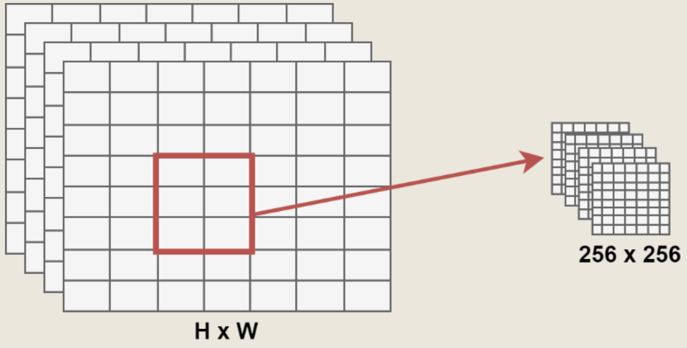


δ : $ReLU$
σ: $Sigmoid$
W_1: $First\ layer\ weights$
W_2: $Second\ layer\ weights$

$$\mathbf{s} = \mathbf{F}_{ex}(\mathbf{z}, \mathbf{W}) = \sigma(g(\mathbf{z}, \mathbf{W})) = \sigma(\mathbf{W}_2 \delta(\mathbf{W}_1 \mathbf{z}))$$

Training Phase

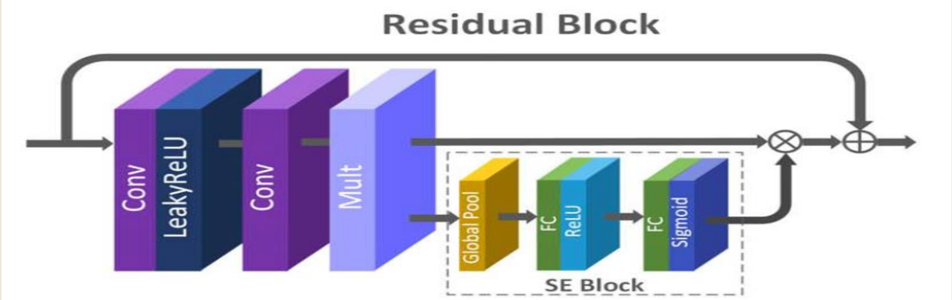
- During training the input is 4 channel of size 256x256 randomly cropped from the input image



- For data augmentation, the input is flipped and rotated randomly

- The output is 3 channel 512x512 sRGB image

- Trained with 16 or 32 residual blocks



- L1 Loss and Adam optimizer

- Trained with 6000 epochs with initial learning rate of 10^{-4} which reduced by a factor of 10 after every 2000 epochs

PSNR - Peak Signal to Noise Ratio

- Given two images I, K MSE is calculated
- Afterwards the maximal pixel value of the original image is taken
- Take the log of the ration and multiply it by 10

SSIM – Structural Similarity

- Calculates the proximity between two image windows
- Structural similarity index (SSIM) is proposed for measuring the structural similarity between images, based on three relatively independent comparisons namely luminance, contrast, and structure

$$\mu_x = average\ of\ x\ window, \qquad \mu_y = average\ of\ y\ window$$
$$\sigma_x = variance\ of\ x\ window, \qquad \sigma_y = variance\ of\ y\ window$$
$$\sigma_{xy} = covariance\ of\ x\ and\ y\ window$$
$$c_i = (k_i L)^2$$
$$L = Dynamic\ range\ of\ pixel\ values, 2^{bits\ per\ pixel} - 1$$
$$k_1 = 0.01, k_2 = 0.03$$

Results

- SID: “See in the Dark”

- Authors model in different ResNet architectures

Experiments	PSNR	SSIM
SID	28.97	**0.8857**
Ours - No SE Block	28.49	0.8817
Ours - 16 Residual Blocks	29.15	0.8829
Ours - 32 Residual Blocks	**29.16**	0.8856

Results

- BM3D: Classical method
- SID: “See in the Dark”
- Authors model in different ResNet architectures

Experiments	# of parameters	Time(sec)
BM3D	-	385.90
SID	7.76M	0.235
Ours - 16 Residual Blocks	1.38M	0.008
Ours - 32 Residual Blocks	2.5M	0.011

Results

- Produce a collection of per-channel modulation weights

Project

- For our project expanded the research several directions:

  1. Train the ResNet model on the Sony α7S II dataset and revaluate the performance as it was done in the original paper

  2. Adjust the current ResNet architecture to fit to Fujifilm X-T2 dataset. This is an innovation since it was not done in the original work

  3. Compare up-sampling units on the output layer for super-resolution

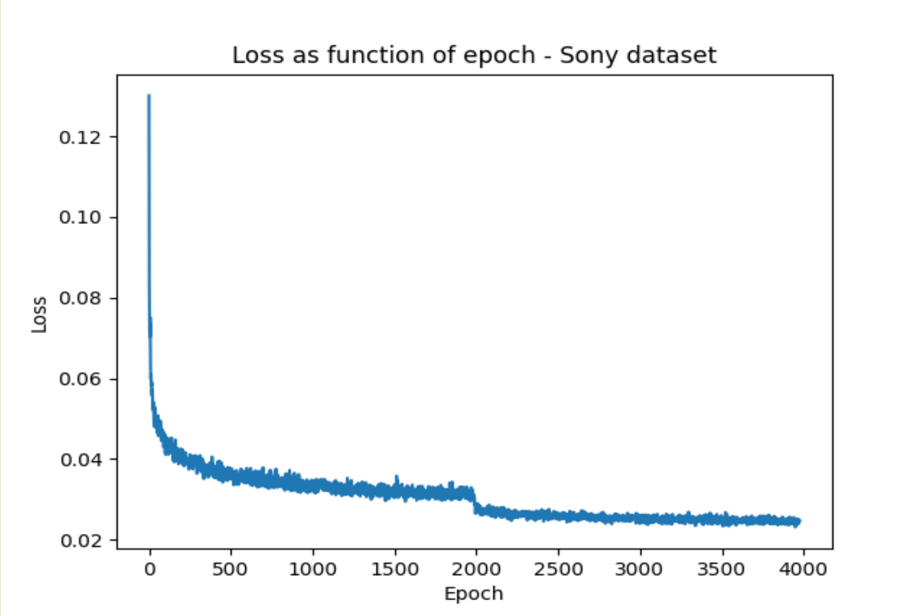
Revaluating the results on the Sony dataset

- To make sure we had done things right we had to reconstruct the original results

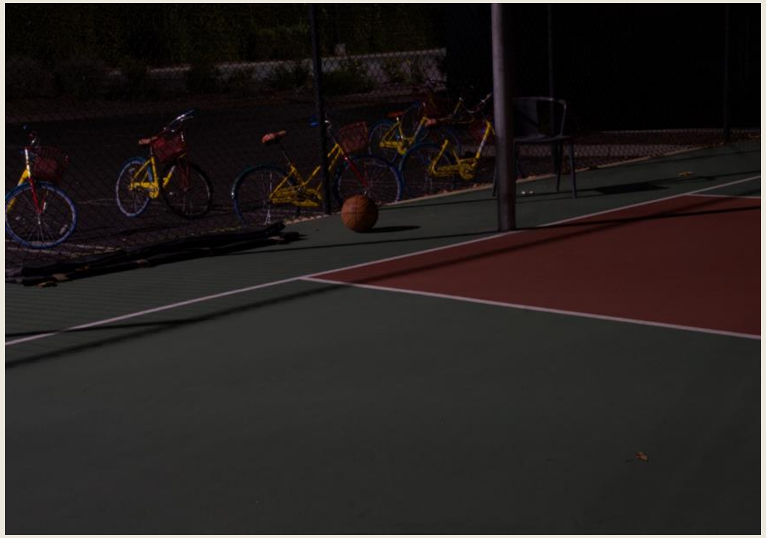
- Therefore, we retrained the ResNet model on the Sony α7S II dataset

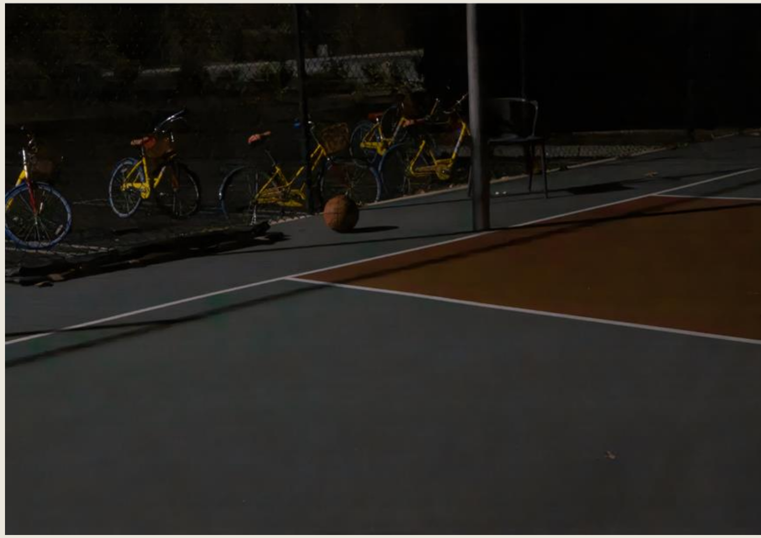
Our results (4000 epoch):
   mean PSNR: 28.19
   mean SSIM: 0.88

Original results (6000 epoch):
   mean PSNR: 29.16
   mean SSIM: 0.88



Revaluating the results on the Sony dataset





Short exposure

Long exposure (ground truth)

Reconstructed image

ResNet model – Fujifilm dataset

- The authors of the paper did not train the network on the Fujifilm dataset

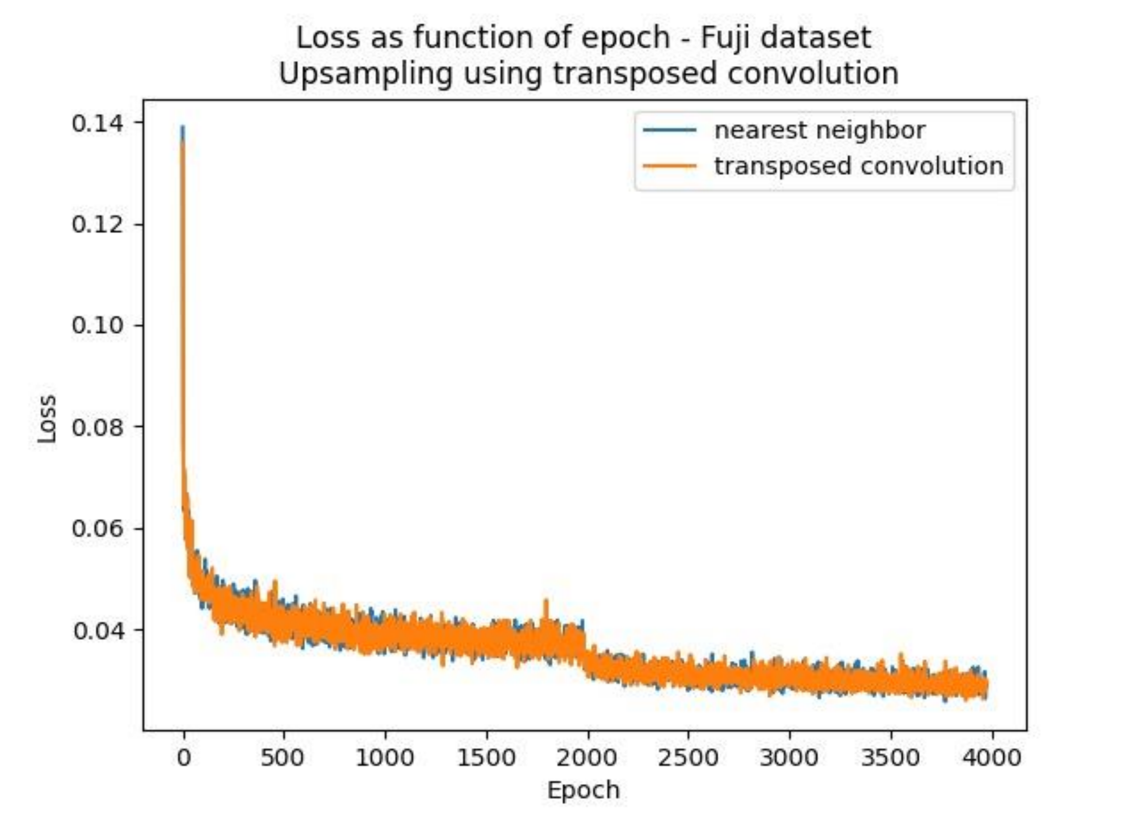
- To integrate the model architecture with this dataset we had to take adjust two units:
  1. The input unit:
     Since the Fujifilm sensor did not match Sony sensor, the raw image structure is massively different. Therefore, we used the code from “Learning to See in the Dark” work which pack the raw image array to 9 channels.
  2. Up-sampling unit:
     In order to match the ground truth resolution to the reconstructed image resolution we had to modify the up-sampling unit.

Fujifilm dataset - results

- We train the model with each up-sampling unit:
  1. Transposed convolution layer
  2. Nearest neighbor interpolation

Transposed convolution layer:
mean PSNR: 26.50
mean SSIM: 0.824

Nearest neighbor interpolation:
mean PSNR: 26.57
mean SSIM: 0.826



Fujifilm dataset - results

- We train the model with each up-sampling unit:
    1. Transposed convolution layer
    2. Nearest neighbor interpolation

Long exposure (ground truth)

Reconstructed image
Transposed convolution layer

Reconstructed image
Nearest neighbor interpolation

Conclusions

- Deep Learning has brought massive improvement to the field of low light imaging
- ResNet architecture were able to produce swifter inference. This is highly importance for real time systems such as video taken in low light
- We were able to reproduce the authors result on the “Sony” dataset
- We provided a proof of concept that the current architecture can deal with different datasets

Questions?