SRFeat

SRFeat: Single Image Super-Resolution with Feature Discrimination

SRFeat - Objective

- Super-resolution
- GAN-based method
- Use multiple discriminator

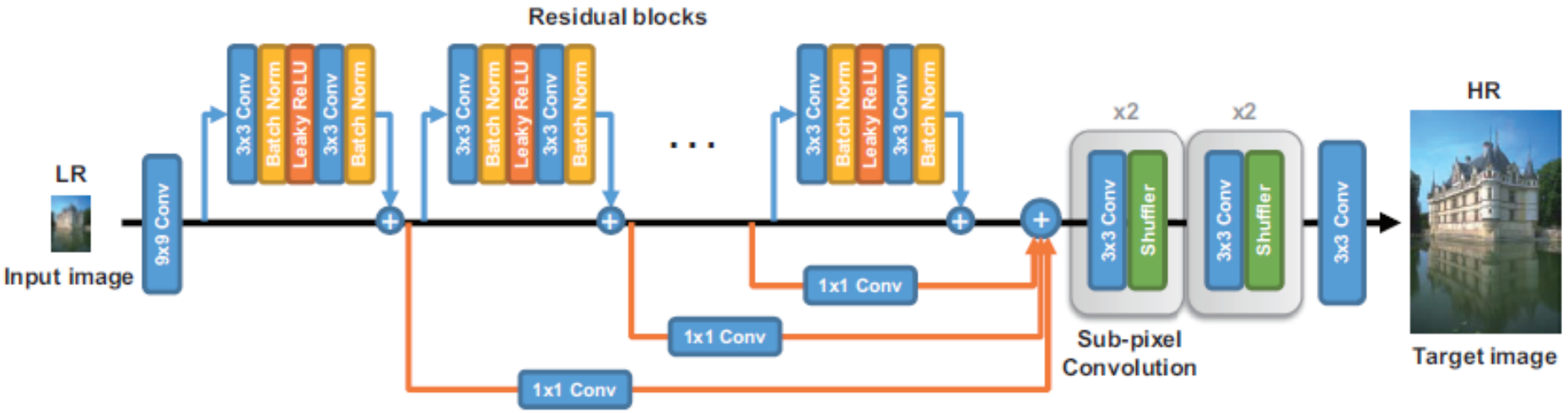
SRFeat - Contribution

- New Generator network
  - Residual block & long-range connection

- Use feature discriminator
  - Previous GAN often make meaningless HR feature(not realistic)
  - Make realistic structural component

SRFeat - Architecture

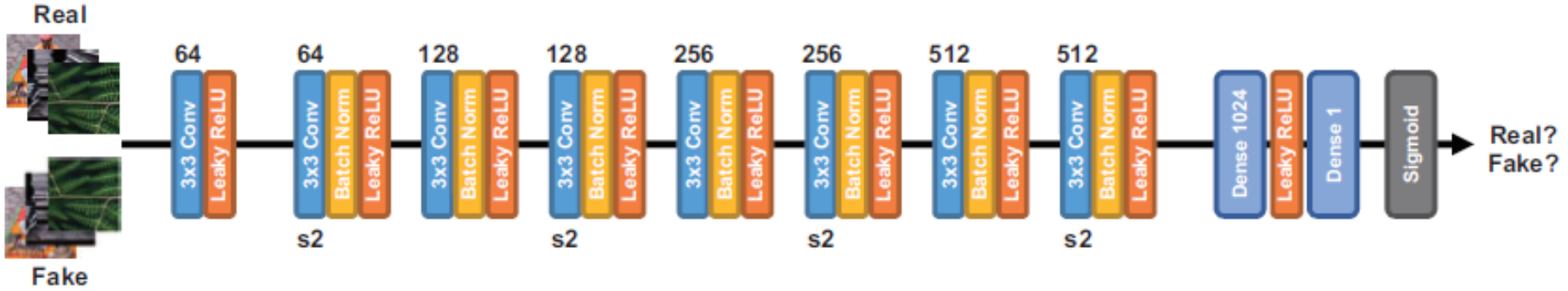
- Generator



- Generating high-resolution image
- Stacked 16 residual blocks(128 channels) + 15 long-range connections
- 2 Sub-pixel convolution blocks (for upscaling)

SRFeat - Architecture

- Discriminator



- Discriminate generated/real image
- Stacked conv blocks + Dense layer
- Use two discriminators
  1. Image discriminator : pixel domain
  2. Feature discriminator : structural components

SRFeat - Implementation

- Training (2 phases)
  1. Pre-training (with MSE loss)

     <Dataset>

     - Use COCO 2017 dataset (116620 images)
     - Apply center crop : 296 X 296
     - HR/LR : 296 X 296 / 74 X 74

     <Hyperparameters>

     - Batch size : 9 (≈ 12958 iterations/epoch)
     - Epochs : 20 (about 260 thousand iterations)
     - Learning rate : $10^{-4} \rightarrow 10^{-5}$ (10 epoch~) $\rightarrow 10^{-6}$ (15 epoch~)

SRFeat - Implementation

- Training (2 phases)

  * Loss function – MSE loss

$$L_{MSE} = \frac{1}{WHC} \sum_{i}^{W} \sum_{j}^{H} \sum_{k}^{C} (I_{i,j,k}^{h} - I_{i,j,k}^{g})^2.$$

  - Use Image pairs
  - Sum of difference between two corresponding pixels in images
  - Guarantees high PSNR, but not guarantees high perceptual quality!

SRFeat - Implementation

- Training (2 phases)

  2. GAN training

     <Dataset>

     - Use DIV2K Dataset (800 -> 104000 , augmented by the author's MATLAB script)
     - Apply random flip, rotation, crop
     - HR/LR : 296 X 296 / 74 X 74

     <Hyperparameters>

     - Batch size : 5 (20800 iterations/epoch)
     - Epochs : 5 (about 104,000 iterations)
     - Learning rate : $10^{-4} \rightarrow 10^{-5}$ (3 epoch~) $\rightarrow 10^{-6}$ (last epoch)

SRFeat - Implementation

- GAN Training

  * Loss function – perceptual loss

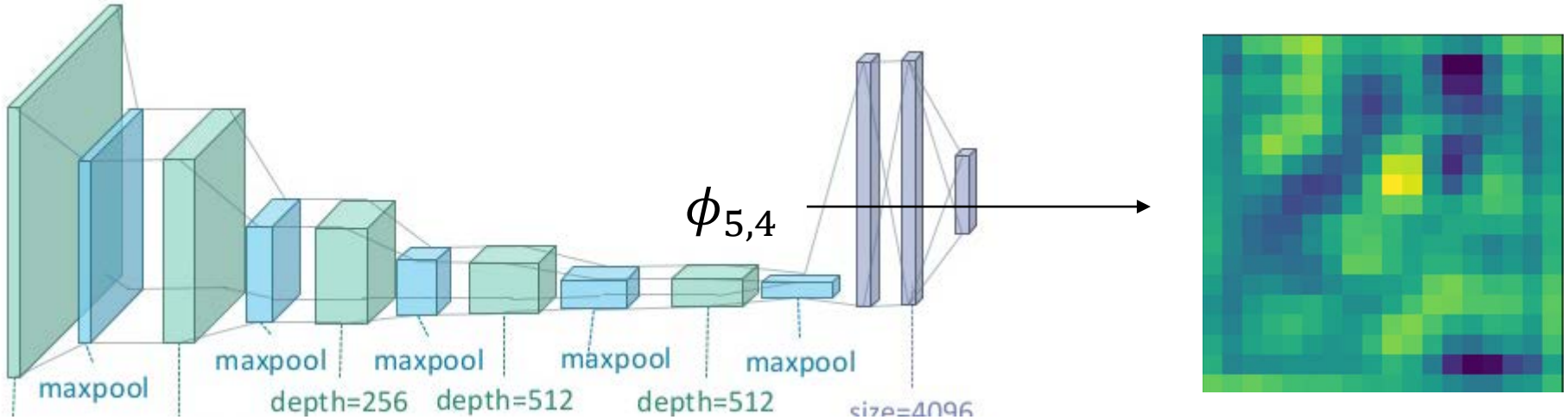
$$L_p = \frac{1}{W_m H_m C_m} \sum_{i}^{W_m} \sum_{j}^{H_m} \sum_{k}^{C_m} \left(\phi_{i,j,k}^{m}(I^h) - \phi_{i,j,k}^{m}(I^g) \right)^2$$

  - MSE about Image feature pairs
  - We can measure feature-wise difference
  - Can improve perceptual quality

SRFeat - Implementation

- GAN Training

  * Loss function – perceptual loss



SRFeat - Implementation

- GAN Training

  * Loss function – Image Discriminator

$$L_a^i = -\log\left(d^i\left(I^g\right)\right), \qquad \text{and}$$

$$L_d^i = -\log\left(d^i\left(I^h\right)\right) - \log\left(1 - d^i\left(I^g\right)\right)$$

  - Pixel-wise discriminator
  - Discriminate fake/real image
  - Can improve perceptual quality significantly

SRFeat - Implementation

- GAN Training

  * Loss function – Feature Discriminator

$$L_a^f = -\log\left(d^f\left(\phi^m\left(I^g\right)\right)\right), \qquad \text{and}$$

$$L_d^f = -\log\left(d^f\left(\phi^m\left(I^h\right)\right)\right) - \log\left(1 - d^f\left(\phi^m\left(I^g\right)\right)\right)$$

  - Feature-wise discriminator
  - Discriminate fake/real image features
  - Can improve realistic(meaningful) structural feature

SRFeat - Implementation

- GAN Training

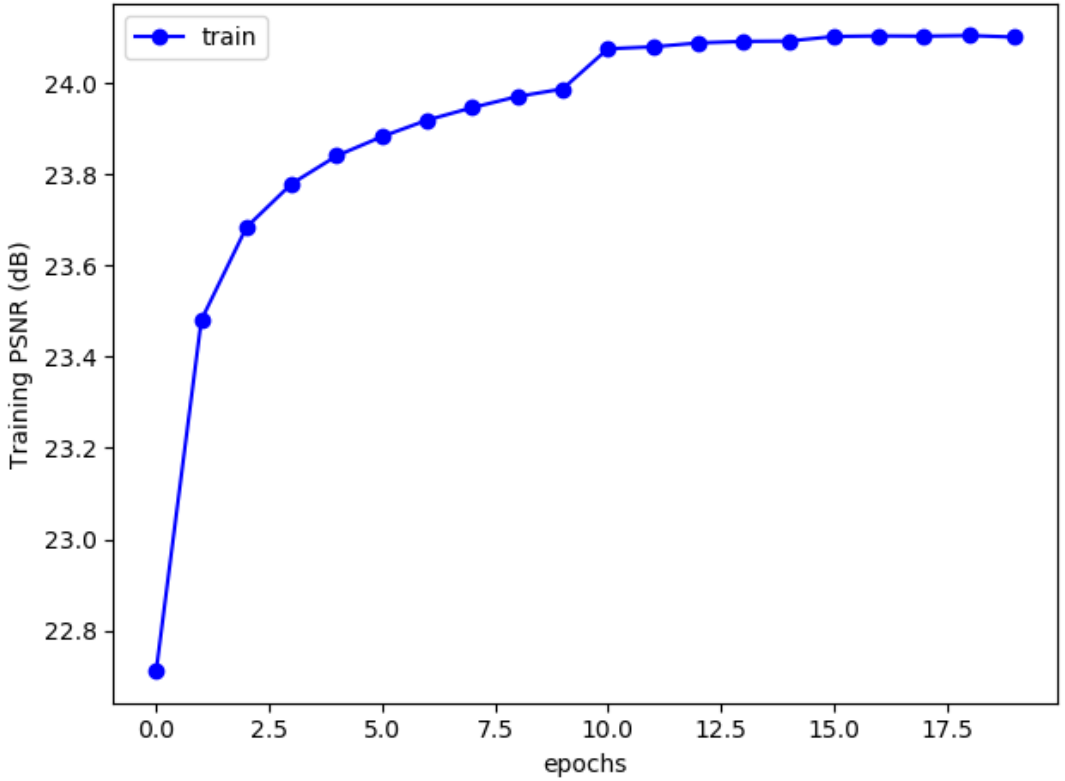
  * Loss function - Final

$$L_g = L_p + \lambda \left(L_a^i + L_a^f \right)$$

- $\lambda = 10^{-3}$
- Pixel values of input feature maps for L_p and L_a^f are scaled to $1/12.75$ of its original value.

SRFeat - Result

- Training Curve (pre-training)



Validation result

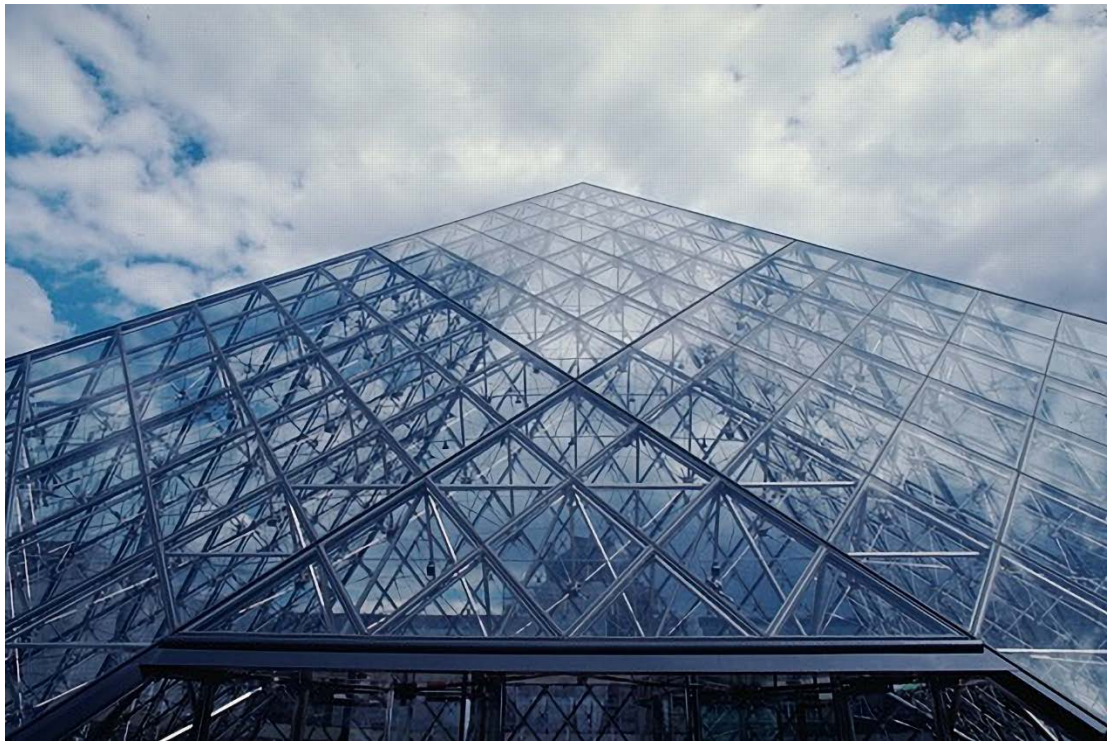
Dataset	PSNR(dB)
BSD100	25.78
Set14	26.38
Set5	30.56

SRFeat - Result

- Fully-Trained Network

<pretrain>

<Fully-train>



SRFeat - Result

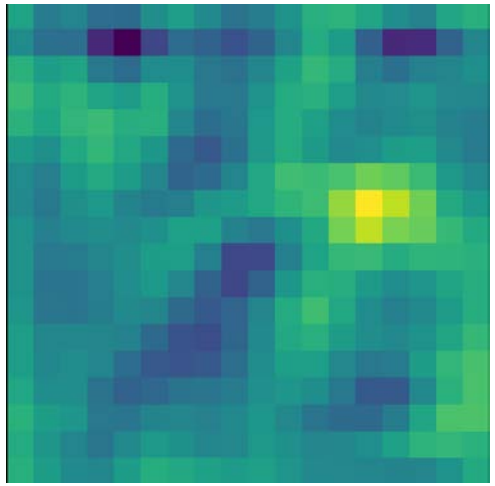
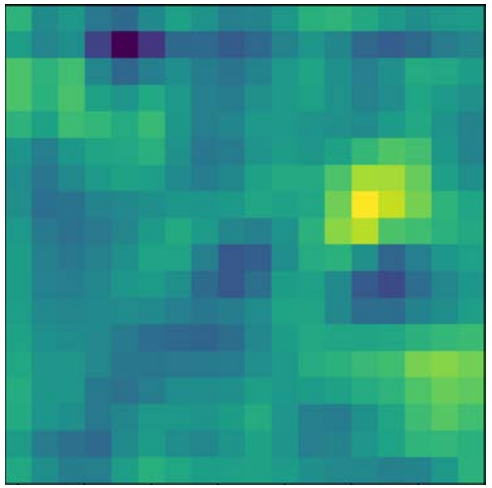
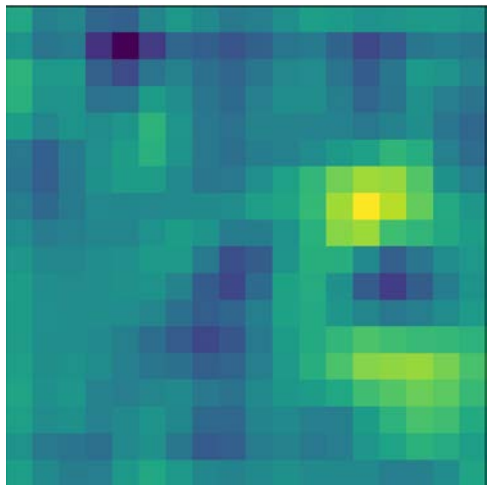
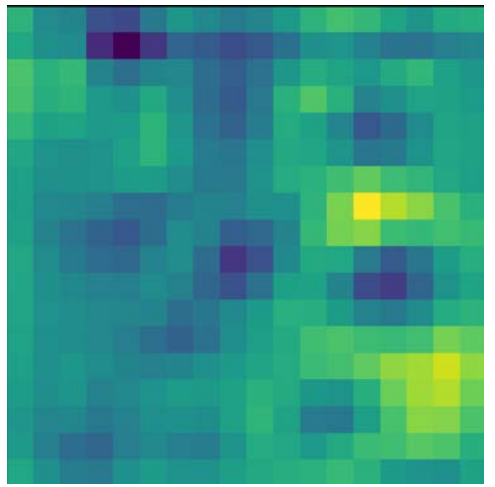
- Fully-Trained Network

<pretrain>

<Fully-train>

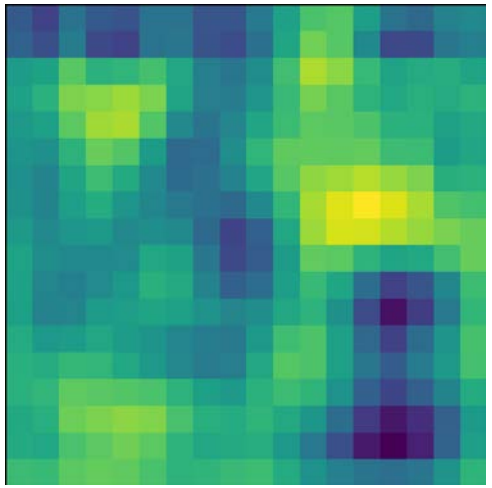
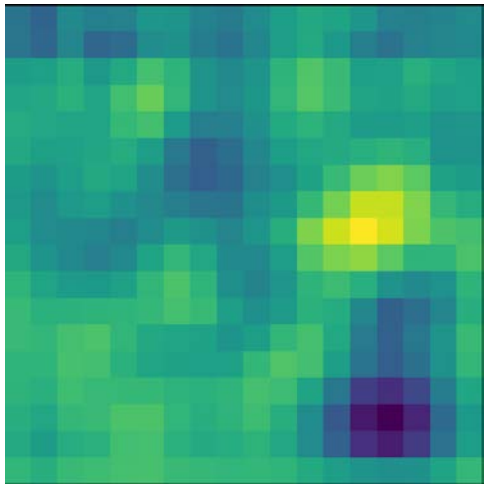
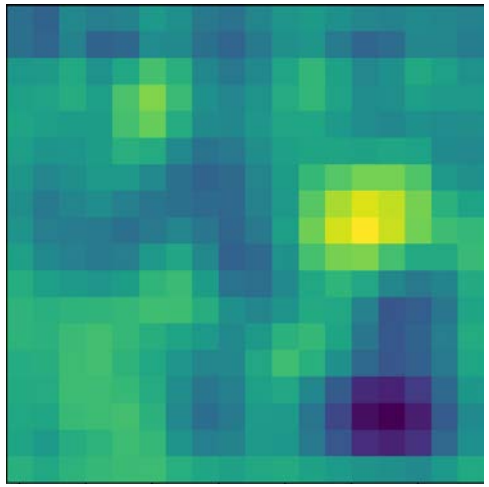
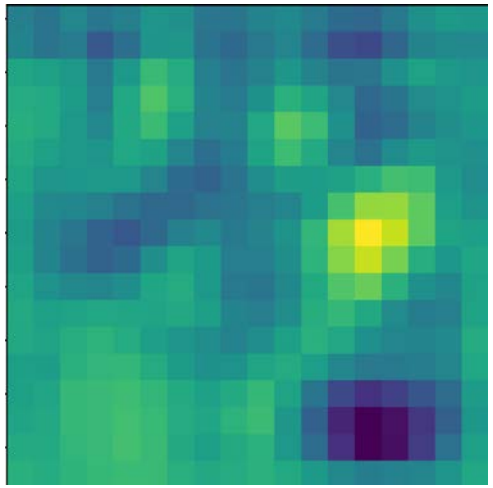
SRFeat - Verification

- Verification of the effectiveness of the feature discriminator

Version	Pretrained	Trained with Discriminators(epoch 3)	Trained with Discriminators(final)	Target
Feature				

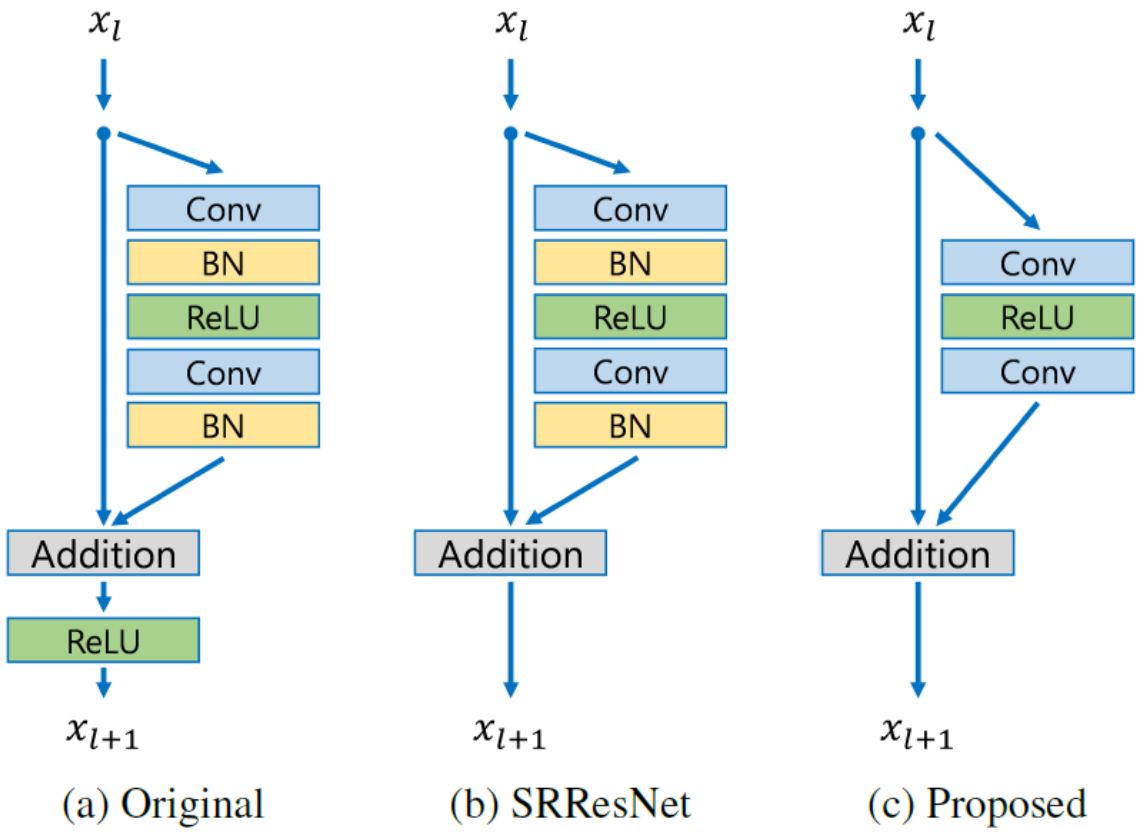
SRFeat - Verification

- Verification of the effectiveness of the feature discriminator

Version	Pretrained	Trained with Discriminators(epoch 3)	Trained with Discriminators(final)	Target
Feature				

SRFeat - Improvement

- Enhanced Resblock (NotBN block)



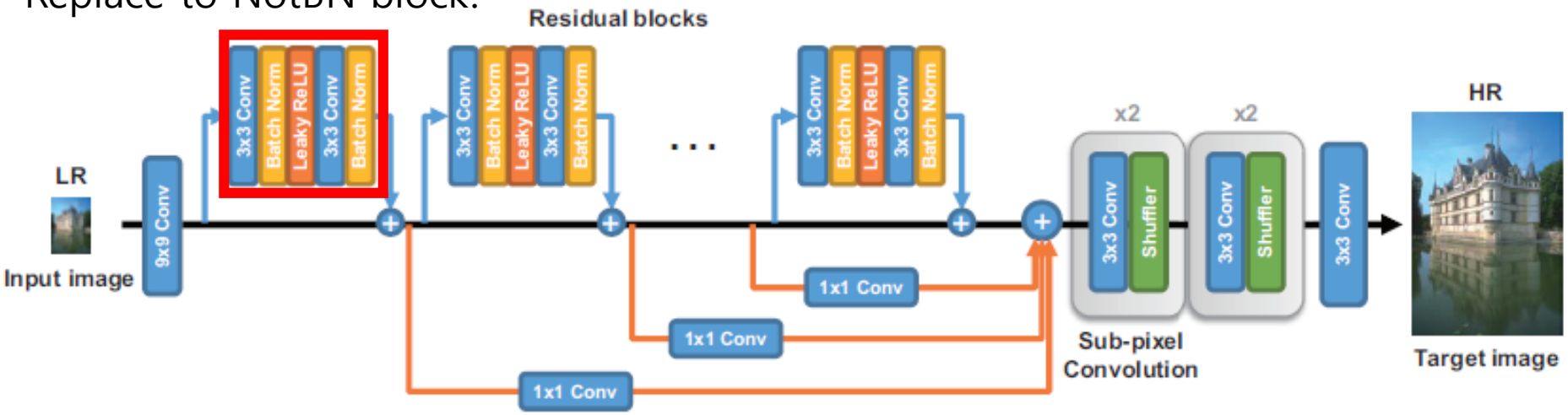
- From EDSR paper (CVPRW 2017)
- BN layers get rid of range flexibility from networks!
- So, let's remove them!

Lim et al., Enhanced Deep Residual Networks for Single Image Super-Resolution

SRFeat - Improvement

- NotBN Generator

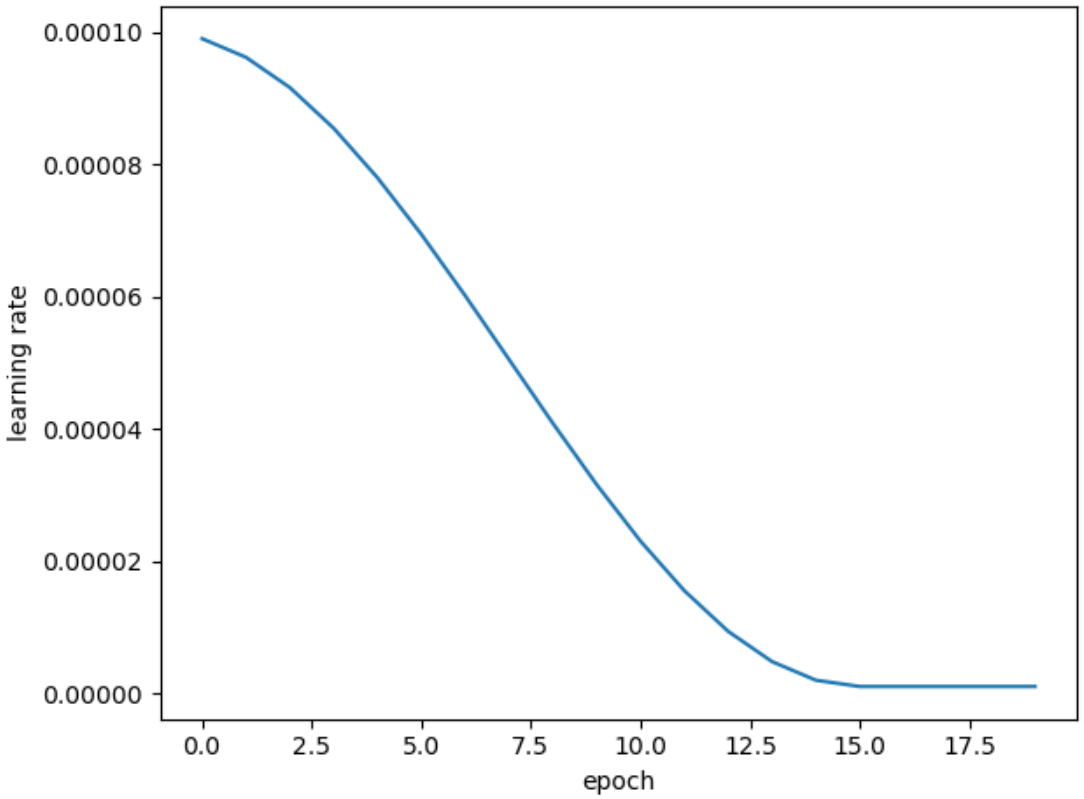
Replace to NotBN block!



- Generating high-resolution image
- Stacked 16 NotBN residual blocks + 15 long-range connections
- 2 Sub-pixel convolution blocks (for upscaling)

SRFeat - Improvement

- Cosine Annealing

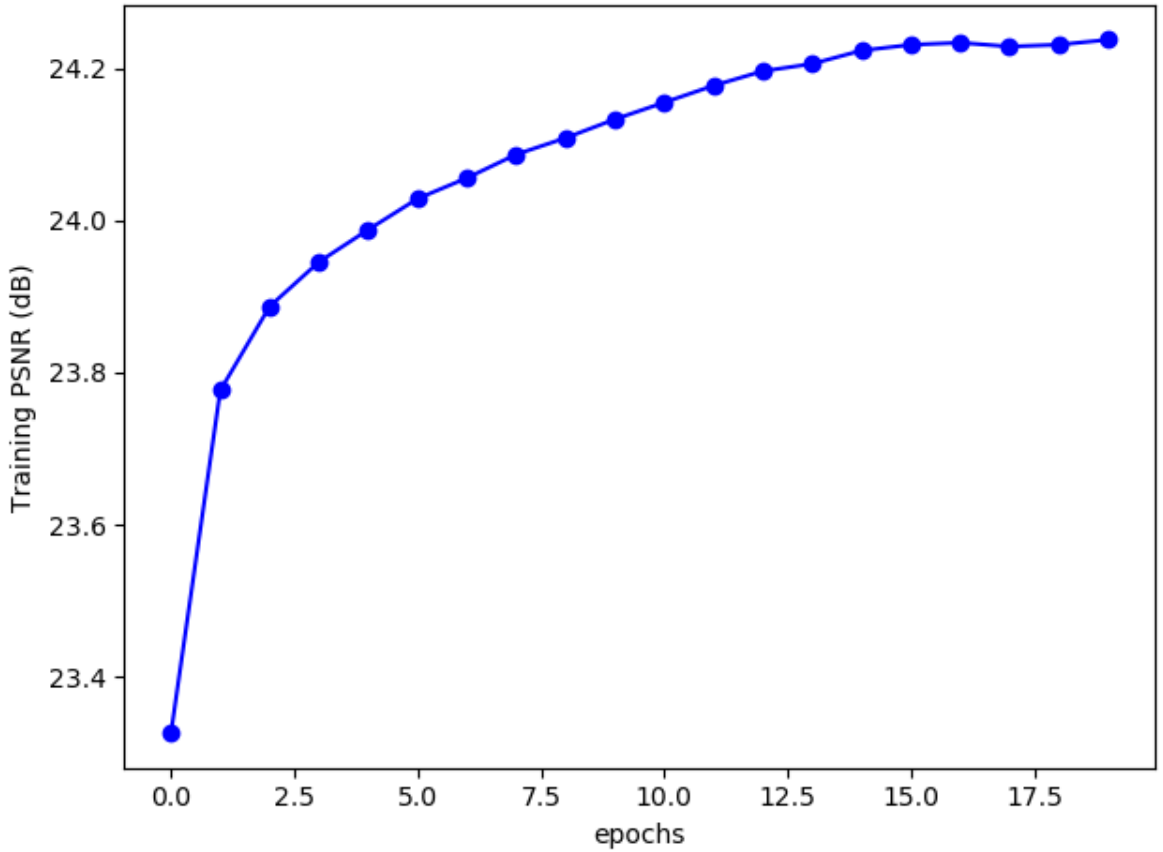


- From SGDR paper (ICLR 2017)
- Change lr along the cosine curve
  1. More stable training
  2. Get better performance (probably)
- In this problem, I set the learning rate to reach the 10^{-6} at 15 epoch.

Ilya Loshchilov, Frank Hutter, "SGDR: Stochastic Gradient Descent with Warm Restarts"

SRFeat - Improvement

- Training Curve (NotBN pre-training)



Validation result

Dataset	PSNR(dB)	
	BN	NotBN
BSD100	25.78	25.82
Set14	26.38	26.44
Set5	30.56	30.72

SRFeat - Improvement

- Fully-Trained Network

<Original>

<NotBN>

SRFeat - Improvement

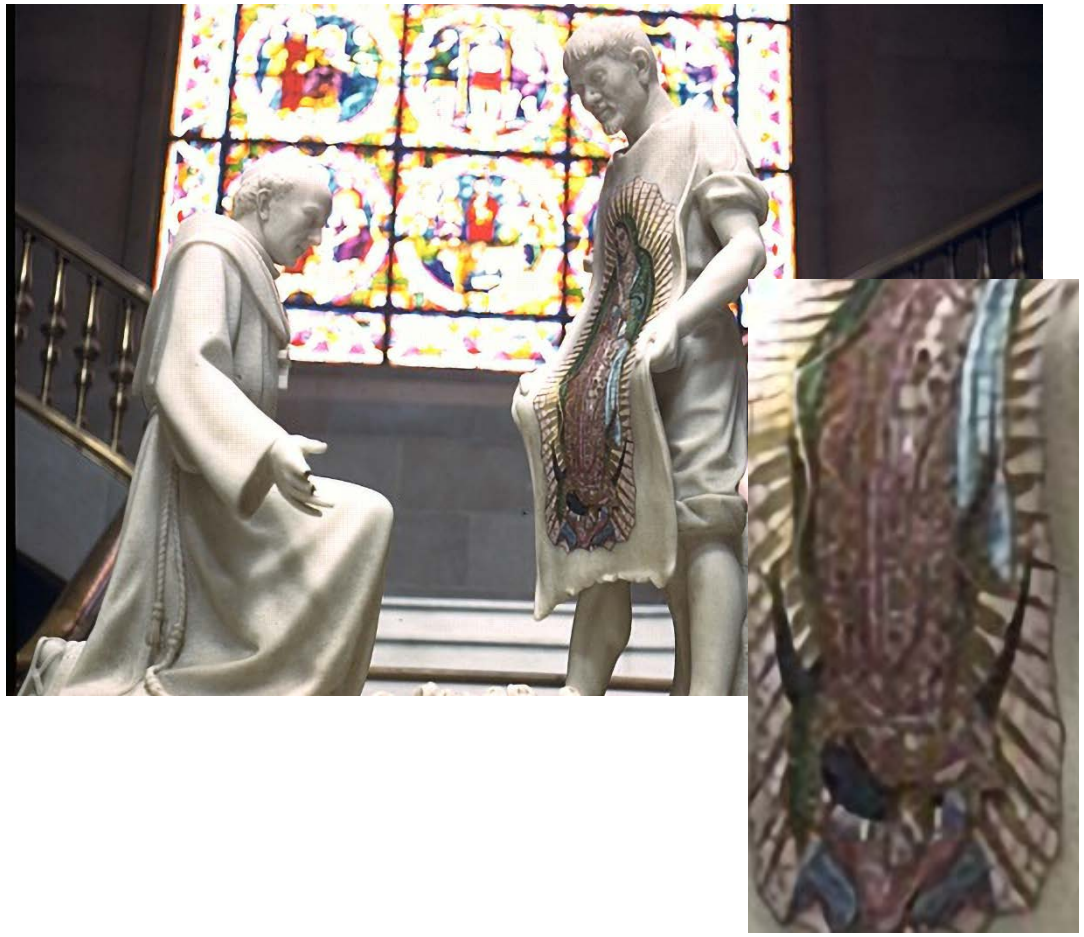
<Original>

<NotBN>

SRFeat - Improvement

- Fully-Trained Network

<Original>



<NotBN>