Frame Booster

Kamil Barszczak

Frame interpolation

First image

Image to predict

Second image

First image

Image to predict

Second image

First image

Image to predict

Second image

The dataset used during the training

- Vimeo90K - triplet (~ 51000 samples, 448x256x3, 35 GB)

First image

Image to predict

Second image

- Custom data (~ 1000 samples, 256x144x3, 1.3 GB)

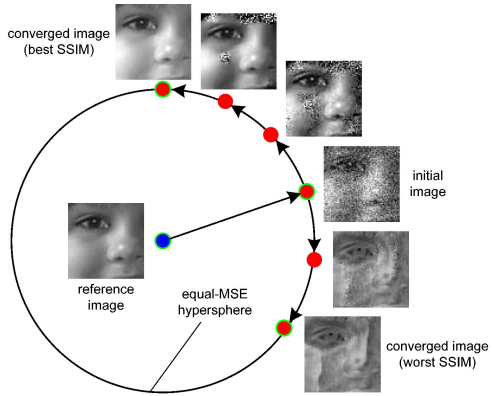
First image

Image to predict

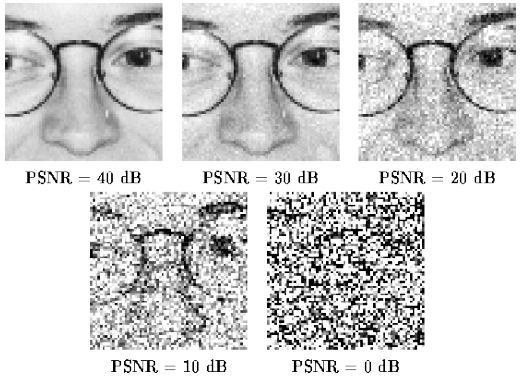
Second image

Loss function

SSIM:



PSNR:



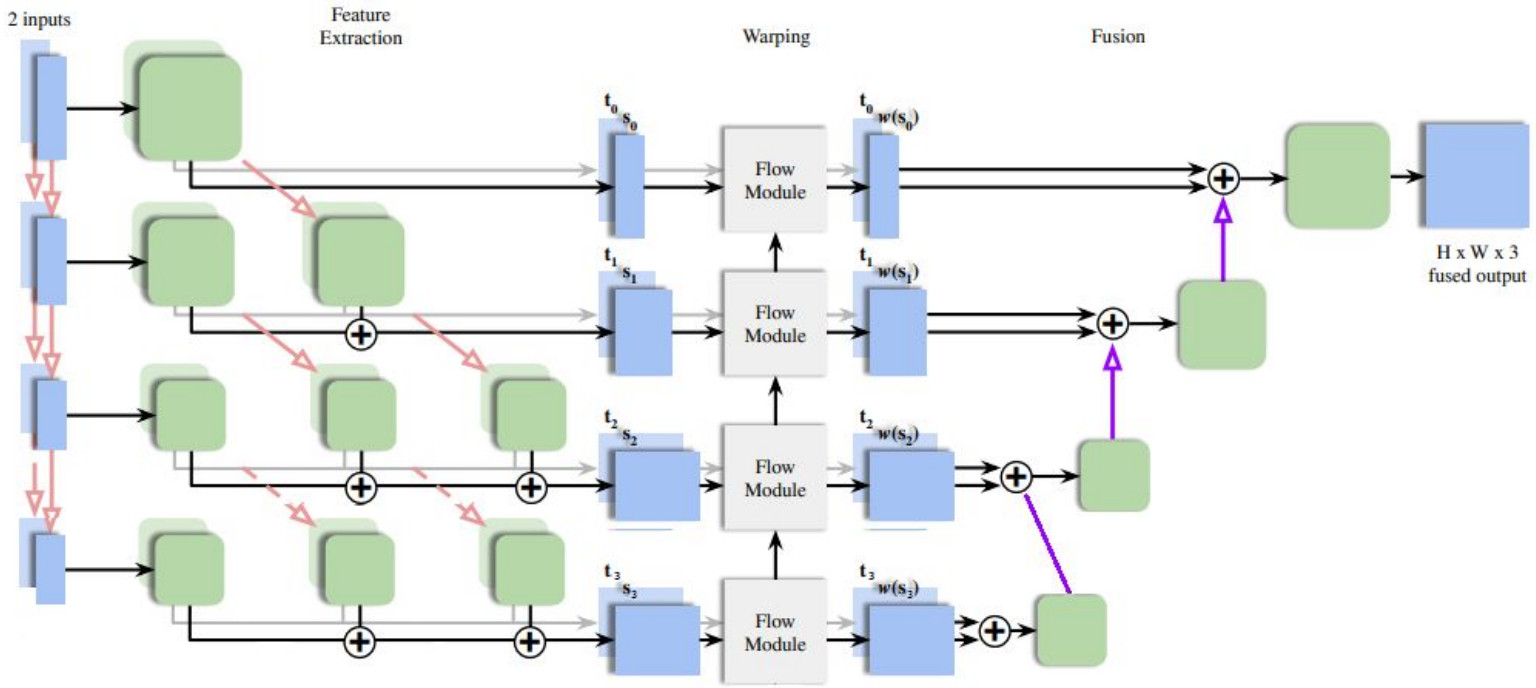
L1: $$|\mathbf{x}|_1 = \sum_{r=1}^{n} |x_r|.$$

L2: $$|\mathbf{x}| = \sqrt{\sum_{k=1}^{n} |x_k|^2},$$

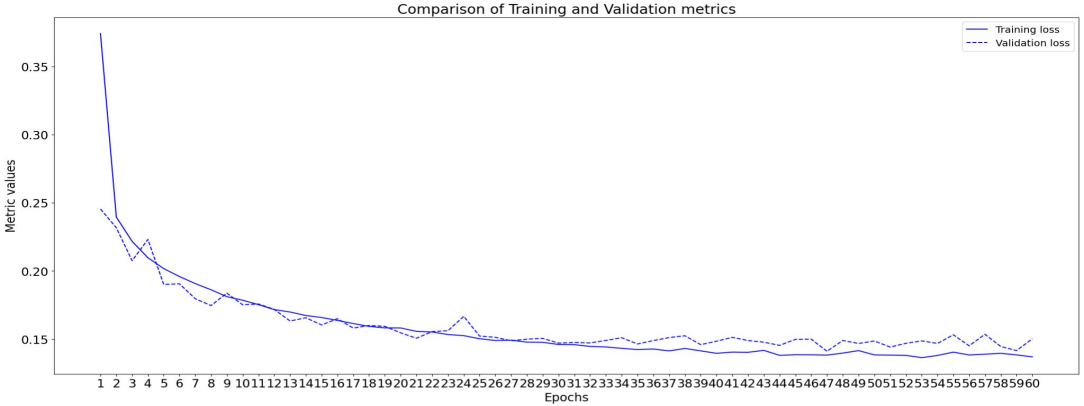
Loss = SSIM + PSNR + 5.0*L1 + 10.0*L2

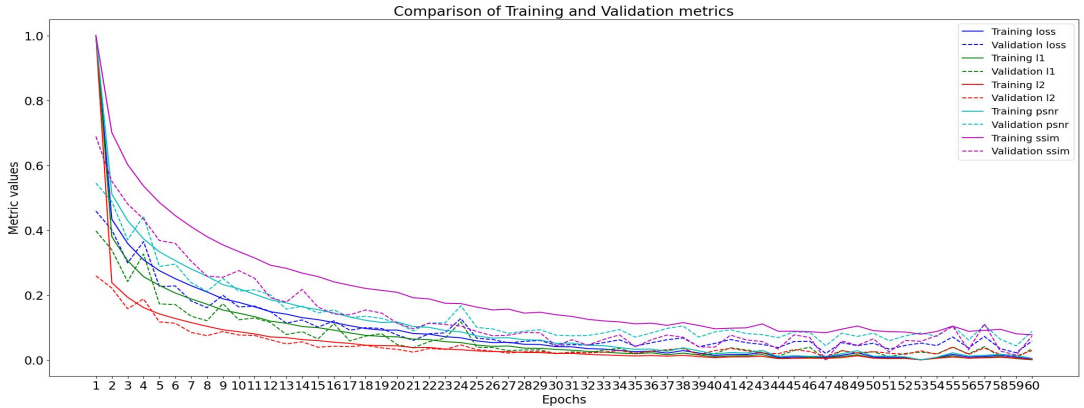
FBNet V1

FBNet v1


2 inputs
Feature Extraction
Warping
Fusion
Flow Module
Flow Module
Flow Module
Flow Module
H x W x 3 fused output

Initial model tests





Initial model tests

First frame

Predicted and Ground-truth difference

First frame

Predicted and Ground-truth difference

Predicted frame

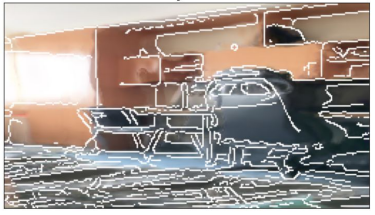
Ground-truth frame

Predicted frame

Ground-truth frame

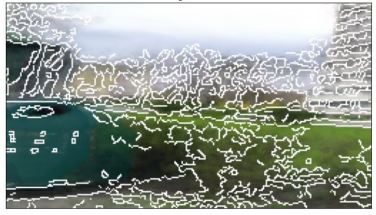
Second frame

Edge shift



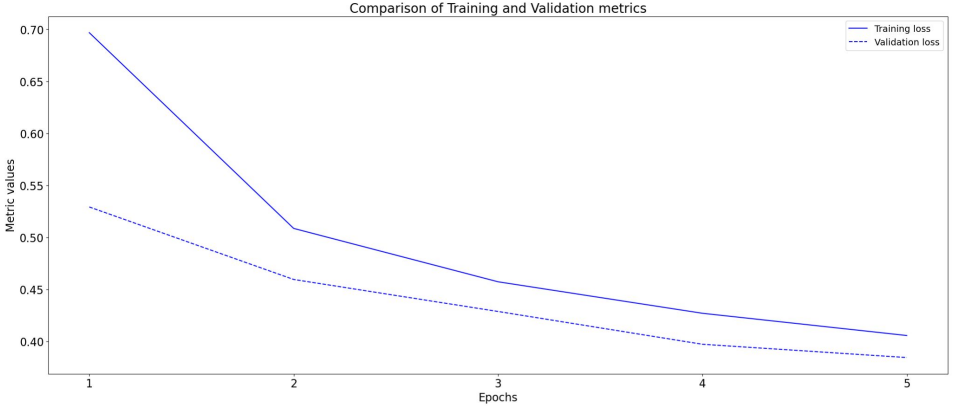
Second frame

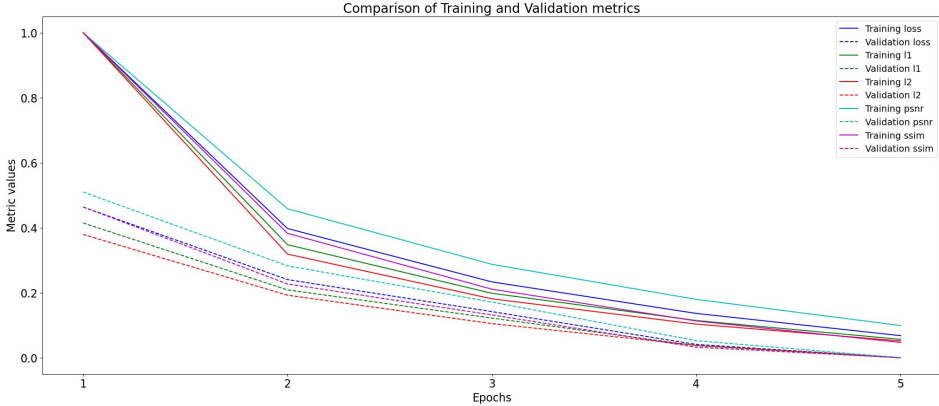
Edge shift



Trening

- TFRecords for processing 19000 training samples (25.5 GB)
- Nadam optimizer (lr = 0.0001, clipvalue=1.0, clipnorm=1.0)
- Epoch time: 22 minutes (1xGTX970 4GB, i5 4690K)
- Model parameters: 1 199 971

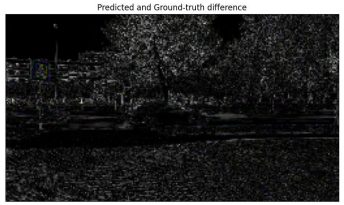


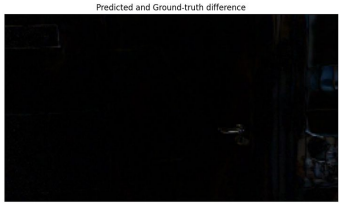


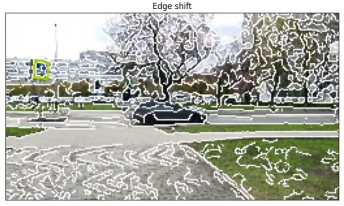
Results

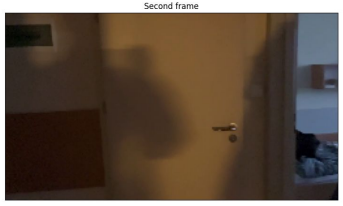
Results for a test set of Vimeo90K in resolution 144 x 256 px

- PSNR: 30.44 (SOTA: 36.76)
- SSIM: 0.9096 (SOTA: 0.9800)









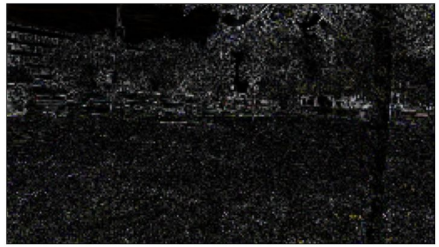
Results

First frame

Predicted and Ground-truth difference

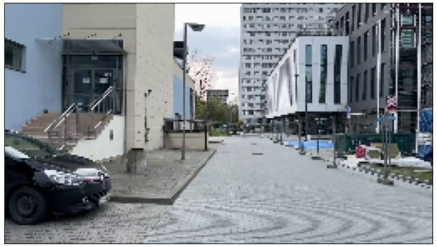
First frame

Predicted and Ground-truth difference



Predicted frame

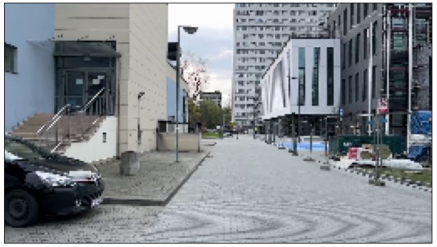
Ground-truth frame



Predicted frame

Ground-truth frame

Second frame



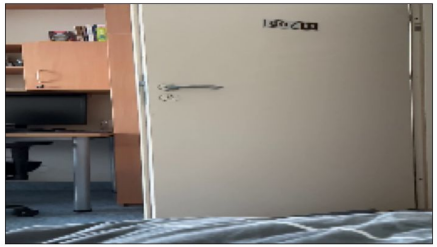
Edge shift

Second frame

Edge shift

Results

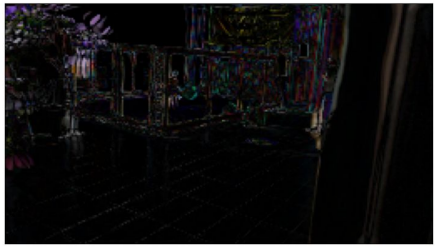
First frame



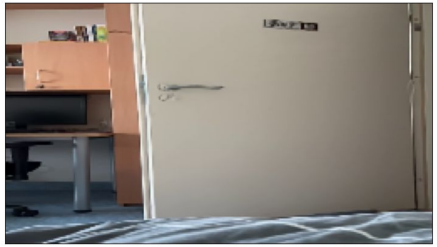
Predicted and Ground-truth difference

First frame

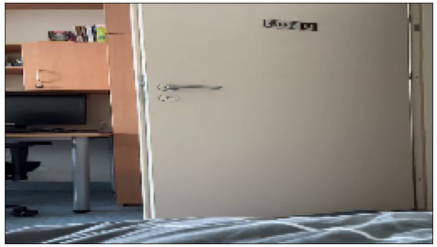
Predicted and Ground-truth difference



Predicted frame



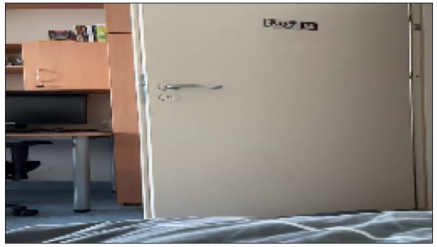
Ground-truth frame



Predicted frame

Ground-truth frame

Second frame



Edge shift

Second frame

Edge shift

Model issues

First frame

Predicted and Ground-truth difference

First frame

Predicted and Ground-truth difference

Predicted frame

Ground-truth frame

Predicted frame

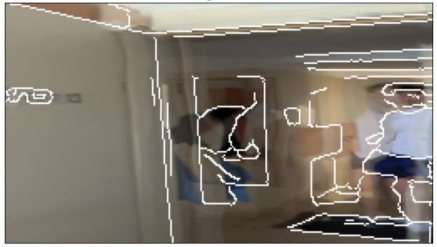
Ground-truth frame

Second frame

Edge shift

Second frame

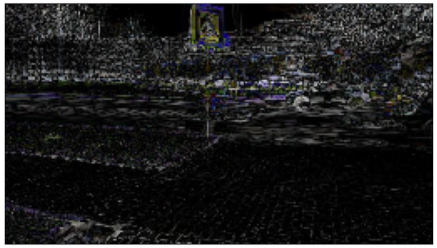
Edge shift



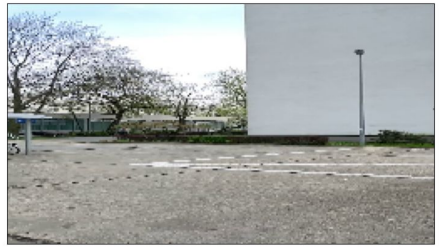
Model issues

First frame

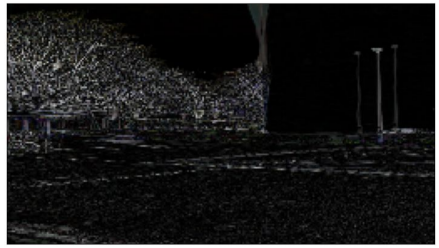
Predicted and Ground-truth difference



First frame

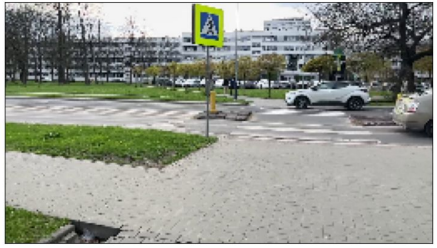


Predicted and Ground-truth difference



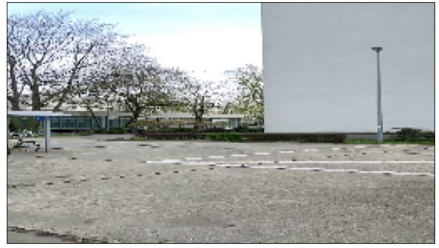
Predicted frame

Ground-truth frame



Predicted frame

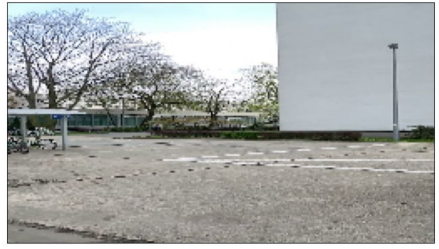
Ground-truth frame



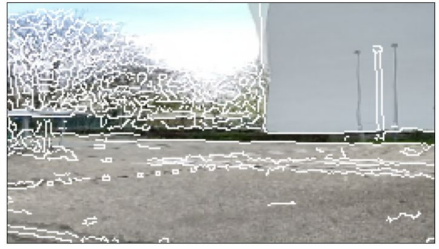
Second frame

Edge shift

Second frame



Edge shift

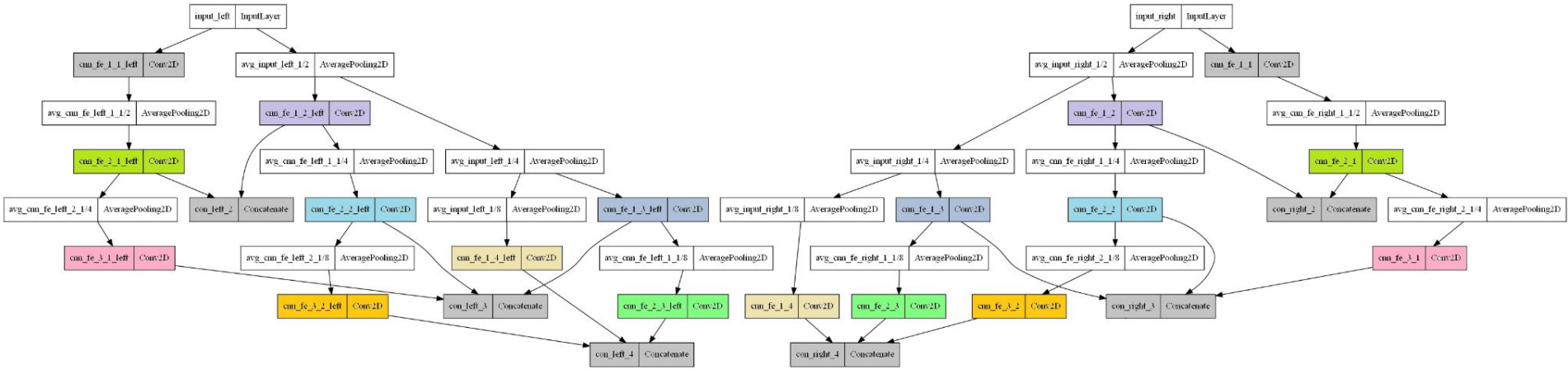


FBNet V3

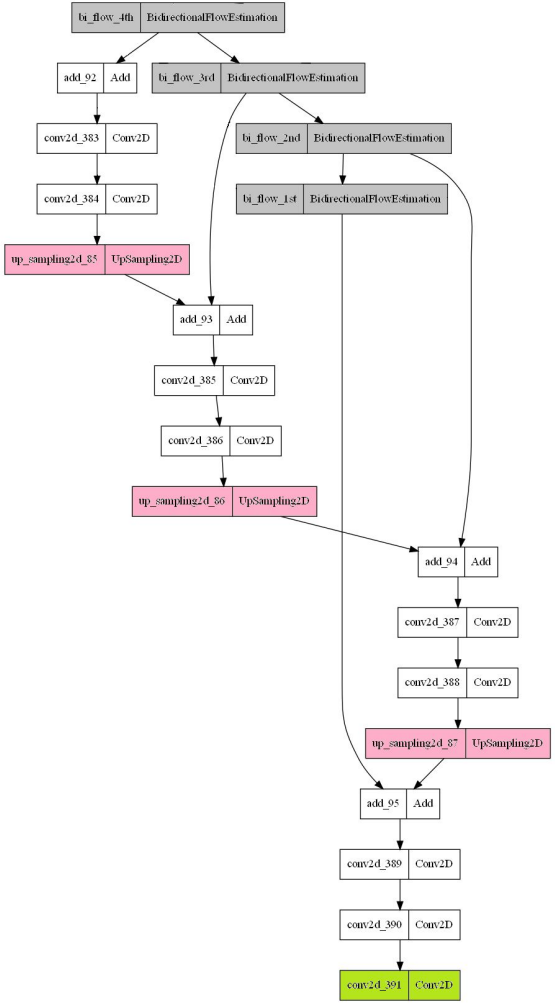
Modifications

1. Changed sharing policy for CNN and BidirectionalFlowEstimation layers
2. Fixed loss function issues
3. Prepared custom fit function
4. Used LeakyReLU activation for each hidden layer
5. Increased size of FeatureExtraction layer
6. Increased FlowEstimation layer context
7. Added net visualization

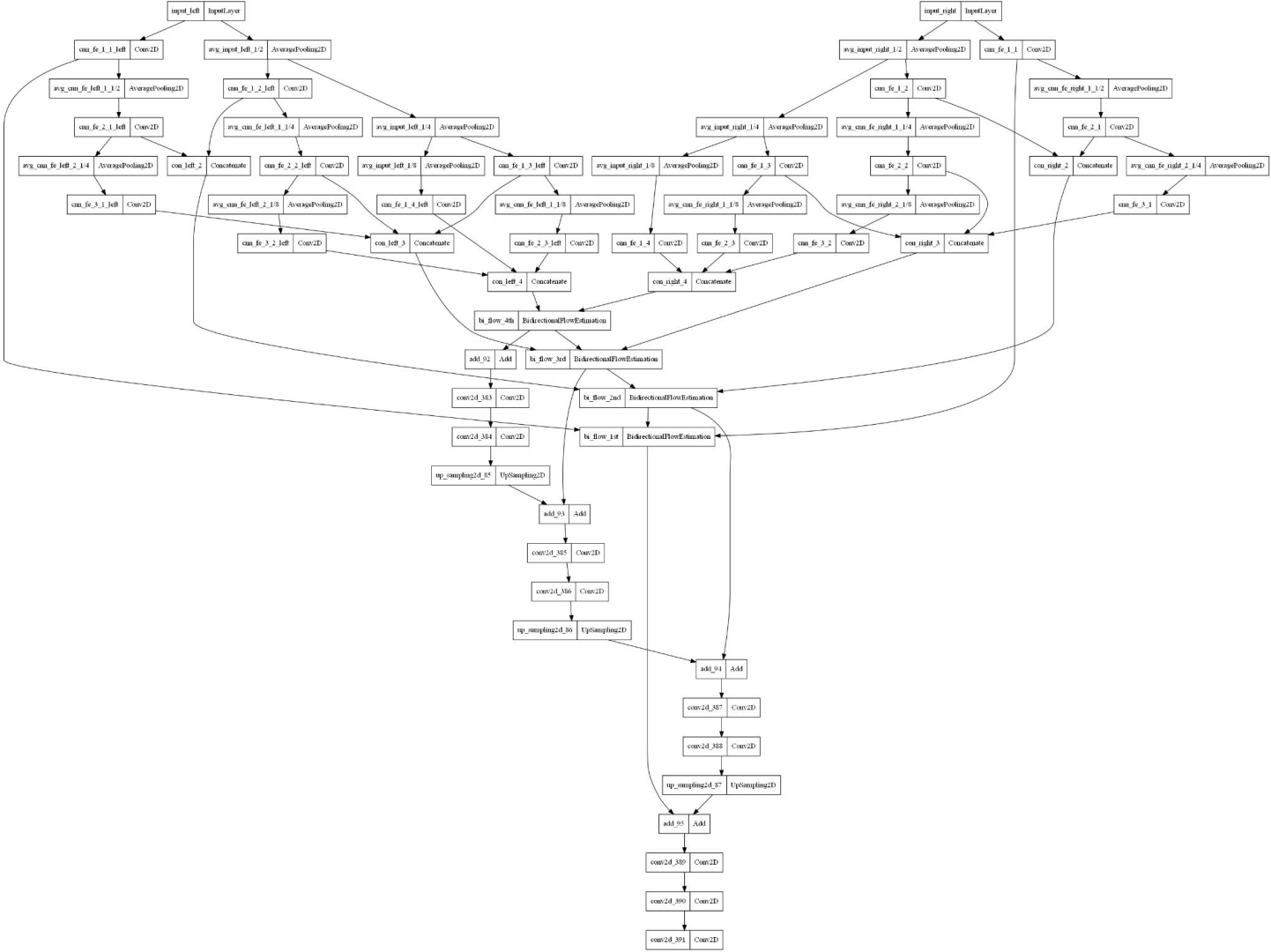
FBNet v3 - encoder



FBNet v3
- decoder

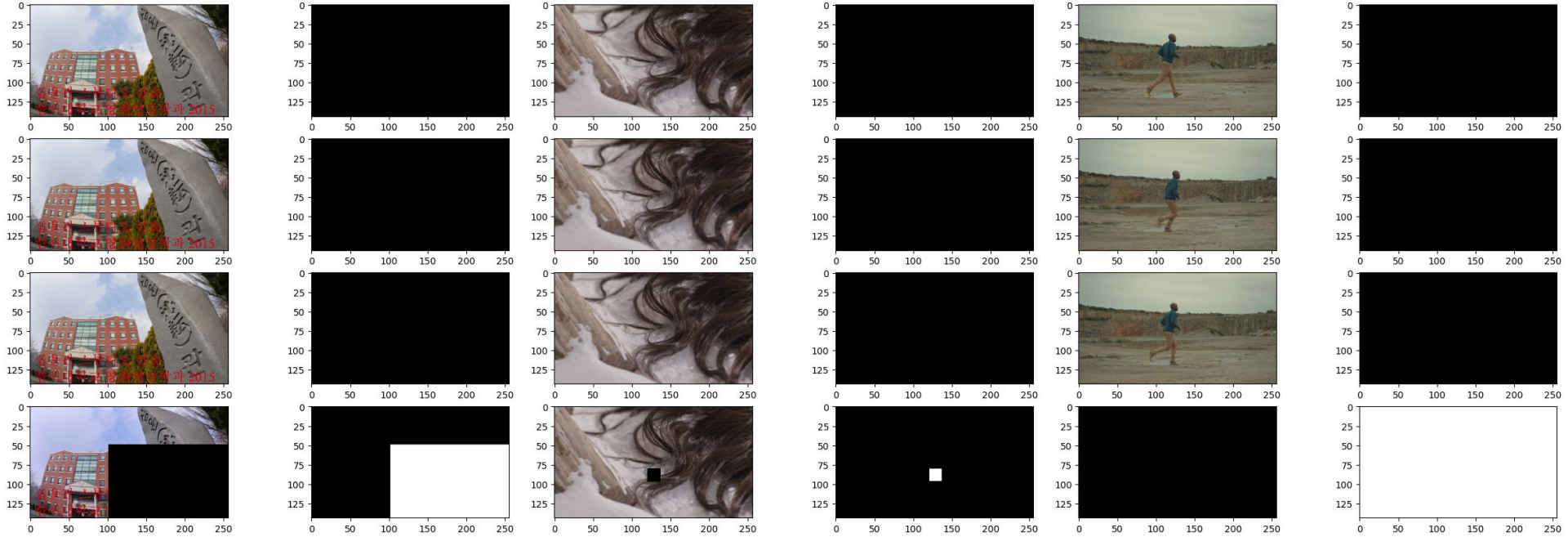


FBNet v3


input_left InputLayer
input_right InputLayer
cnn_fe_1_1_left Conv2D
avg_input_left_1/2 AveragePooling2D
avg_input_right_1/2 AveragePooling2D
cnn_fe_1_1 Conv2D
avg_cnn_fe_left_1_1/2 AveragePooling2D
cnn_fe_1_2_left Conv2D
cnn_fe_1_2 Conv2D
avg_cnn_fe_right_1_1/2 AveragePooling2D
cnn_fe_2_1_left Conv2D
avg_cnn_fe_left_1_1/4 AveragePooling2D
avg_input_left_1/4 AveragePooling2D
avg_input_right_1/4 AveragePooling2D
avg_cnn_fe_right_1_1/4 AveragePooling2D
cnn_fe_2_1 Conv2D
avg_cnn_fe_left_2_1/4 AveragePooling2D
con_left_2 Concatenate
cnn_fe_2_2_left Conv2D
avg_input_left_1/8 AveragePooling2D
cnn_fe_1_3_left Conv2D
avg_input_right_1/8 AveragePooling2D
cnn_fe_1_3 Conv2D
cnn_fe_2_2 Conv2D
con_right_2 Concatenate
avg_cnn_fe_right_2_1/4 AveragePooling2D
cnn_fe_3_1_left Conv2D
avg_cnn_fe_left_2_1/8 AveragePooling2D
cnn_fe_1_4_left Conv2D
avg_cnn_fe_left_1_1/8 AveragePooling2D
avg_cnn_fe_right_1_1/8 AveragePooling2D
avg_cnn_fe_right_2_1/8 AveragePooling2D
cnn_fe_3_1 Conv2D
cnn_fe_3_2_left Conv2D
con_left_3 Concatenate
cnn_fe_2_3_left Conv2D
cnn_fe_1_4 Conv2D
cnn_fe_2_3 Conv2D
cnn_fe_3_2 Conv2D
con_right_3 Concatenate
con_left_4 Concatenate
con_right_4 Concatenate
bi_flow_4th BidirectionalFlowEstimation
add_92 Add
bi_flow_3rd BidirectionalFlowEstimation
conv2d_383 Conv2D
bi_flow_2nd BidirectionalFlowEstimation
conv2d_384 Conv2D
bi_flow_1st BidirectionalFlowEstimation
up_sampling2d_85 UpSampling2D
add_93 Add
conv2d_385 Conv2D
conv2d_386 Conv2D
up_sampling2d_86 UpSampling2D
add_94 Add
conv2d_387 Conv2D
conv2d_388 Conv2D
up_sampling2d_87 UpSampling2D
add_95 Add
conv2d_389 Conv2D
conv2d_390 Conv2D
conv2d_391 Conv2D

Training issues

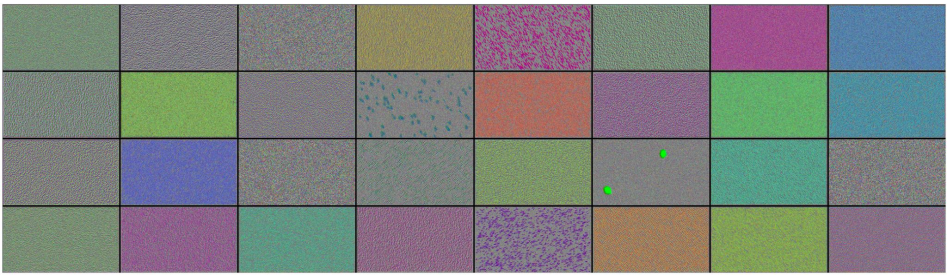
1. Issue related to @tf.function
2. Problems with training on GPU

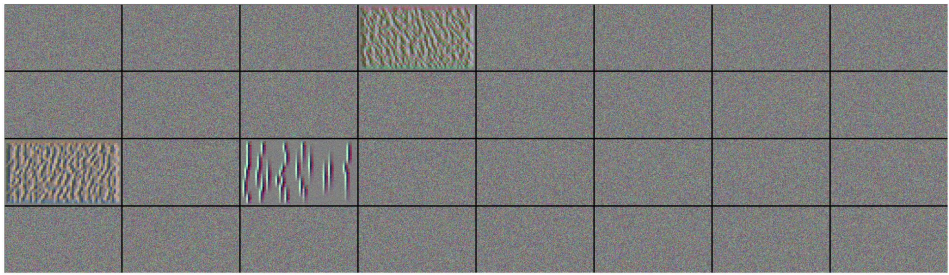


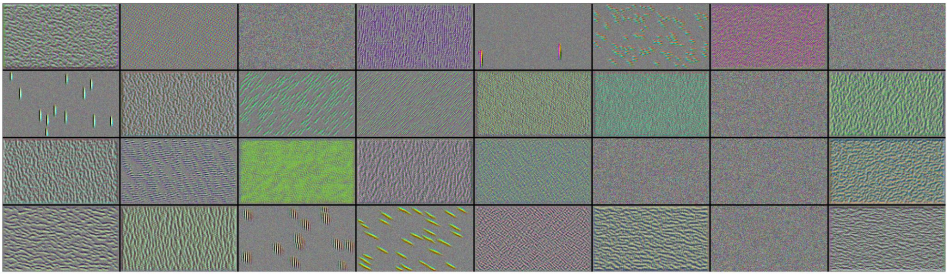
Training results

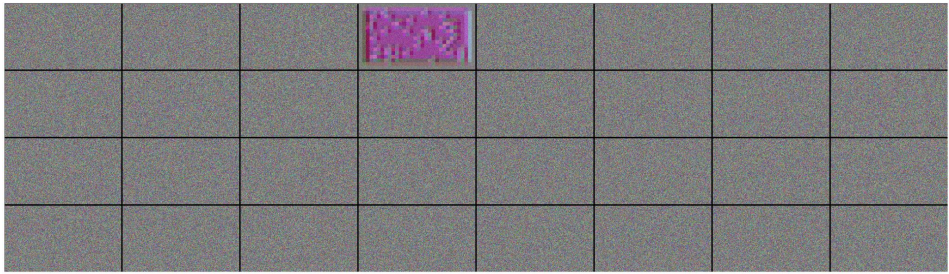
- Loss: 0.3569 (15% better result than model v2)
- PSNR: 31.636 (4% better than model v1 and 2.5% better than model v2)
- SSIM: 0.9348 (3% better than model v1)

Encoder conv2d filters (1_1-1_4)

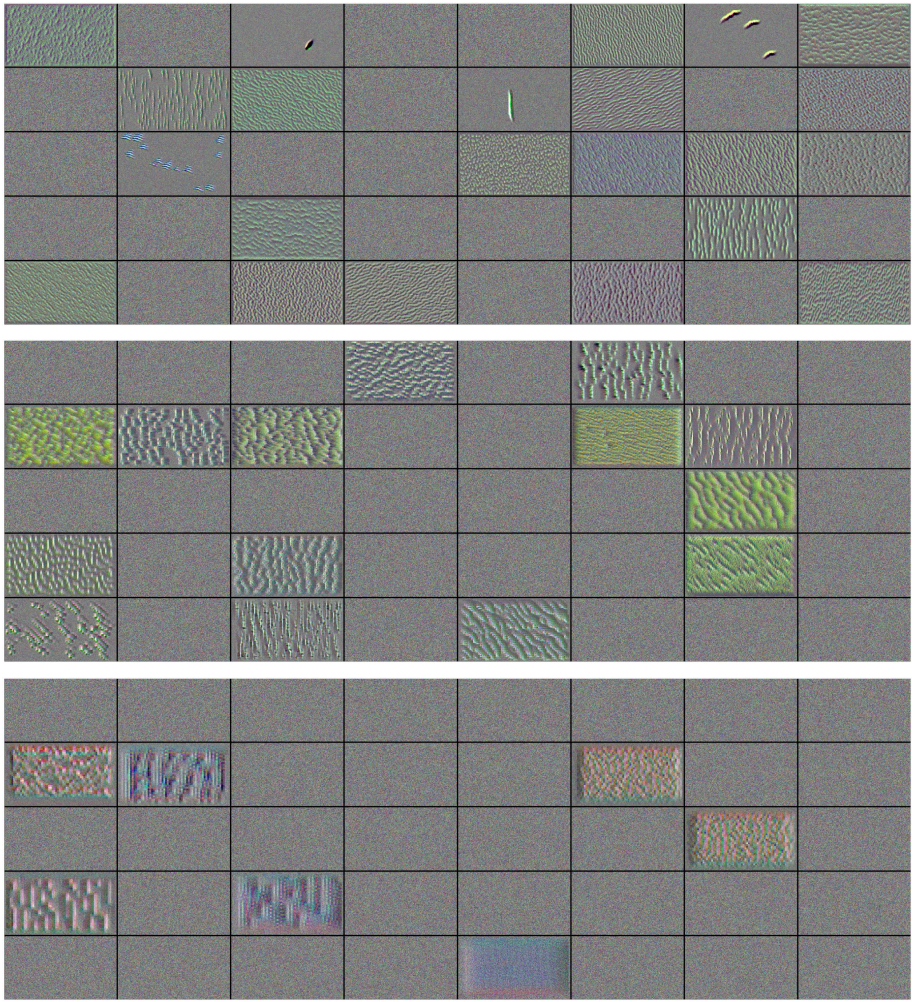




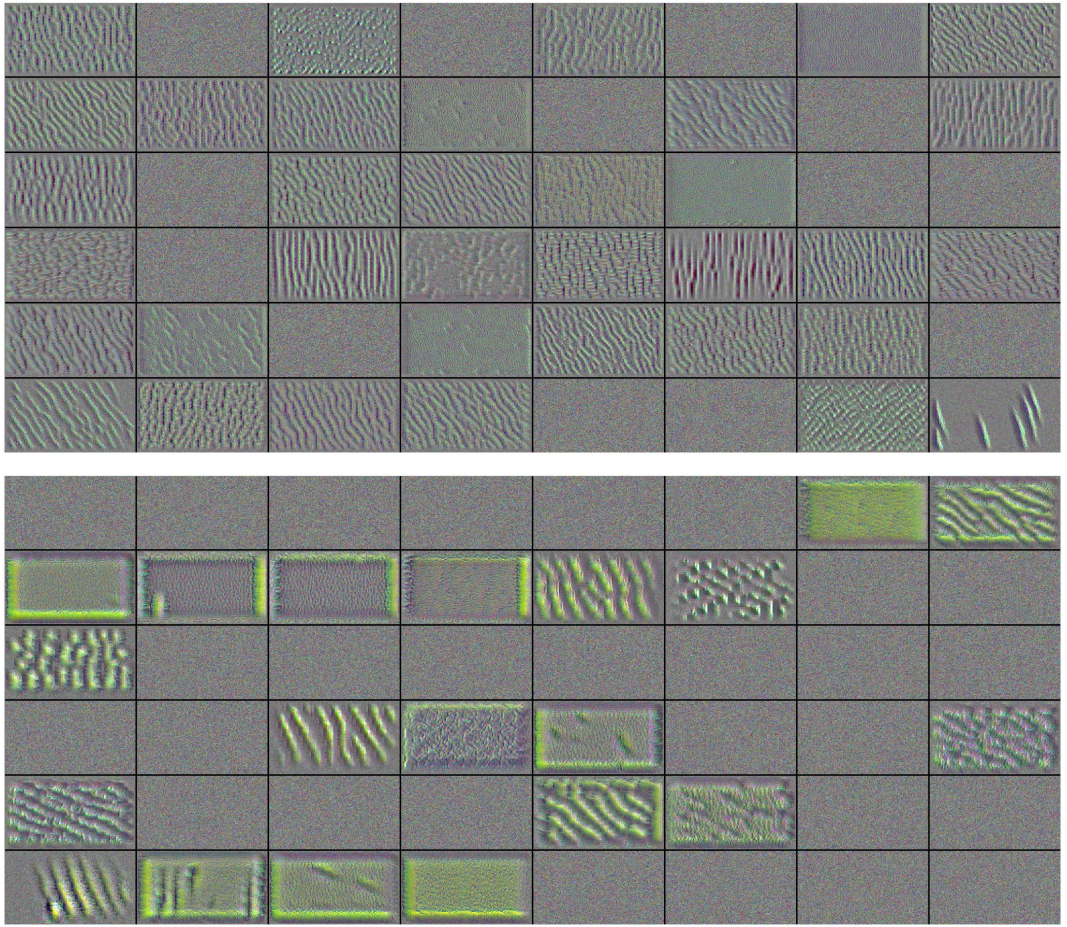




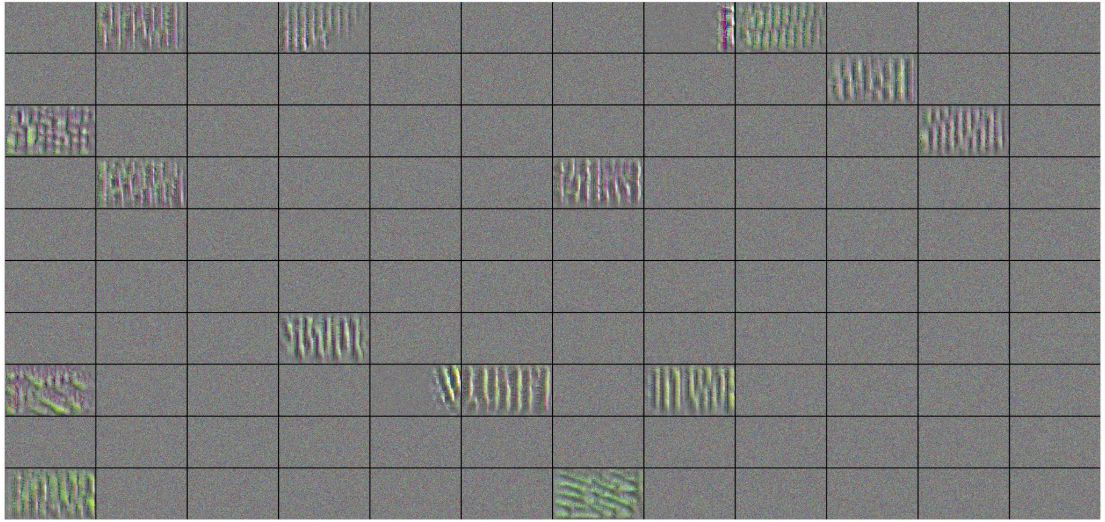
Encoder conv2d filters (2_1-2_3)

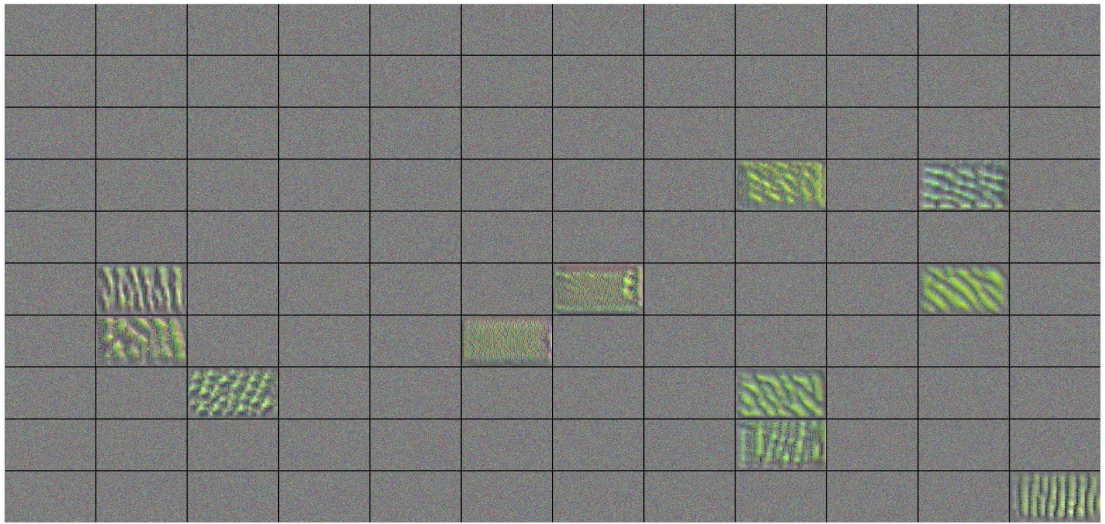


Encoder conv2d filters (3_1-3_2)

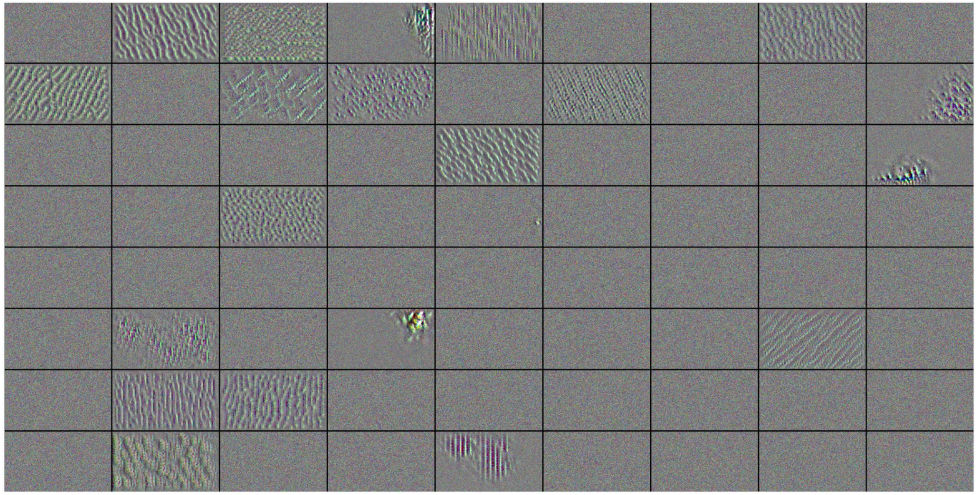


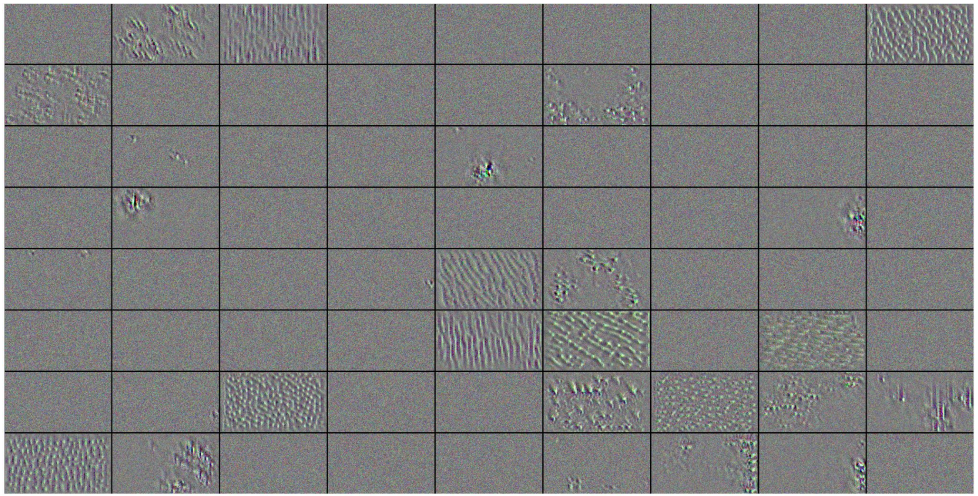
Decoder conv2d filters (30, 31)



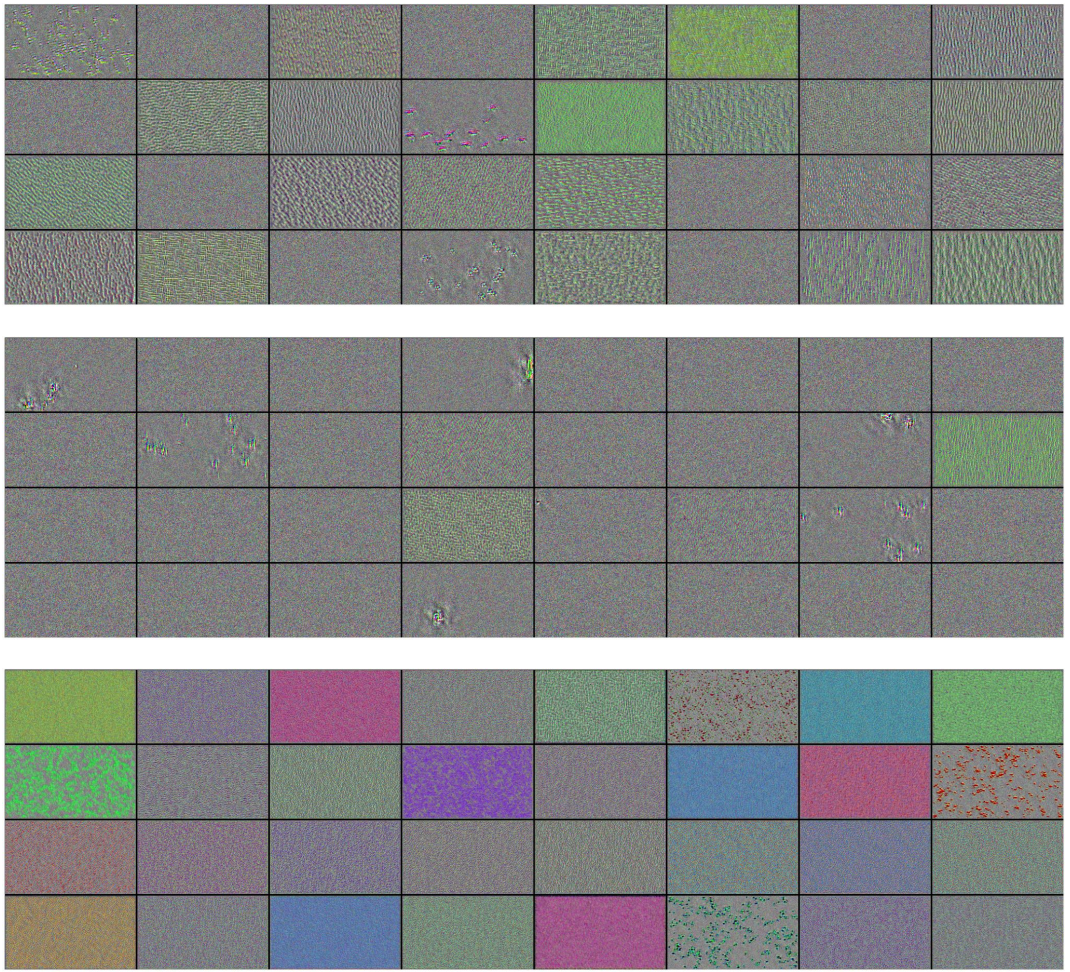


Decoder conv2d filters (32, 33)

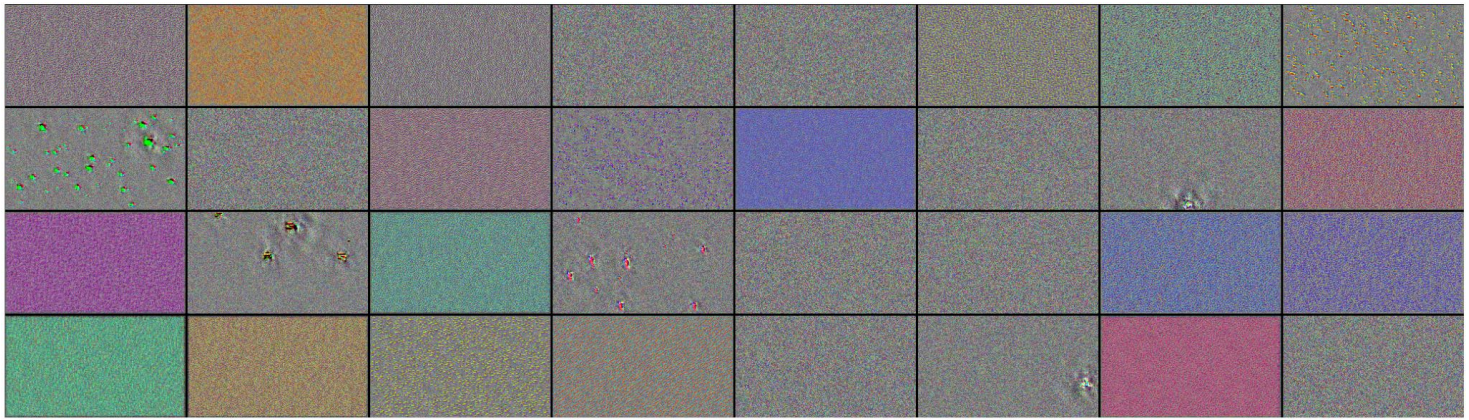


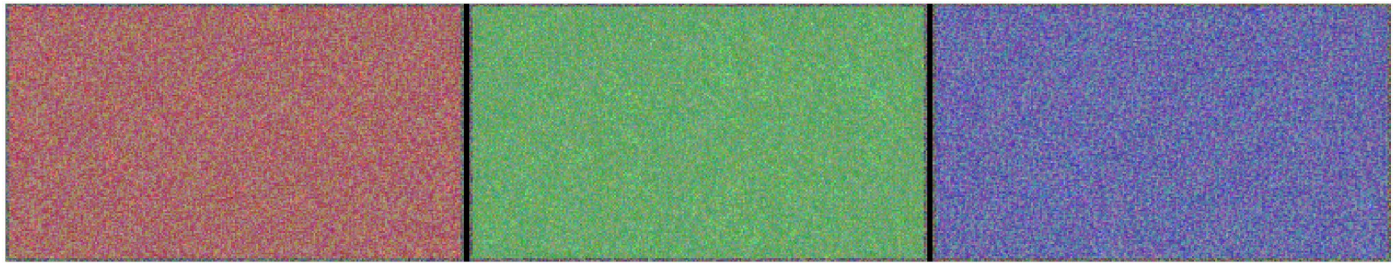


Decoder conv2d filters (34, 35, 36)

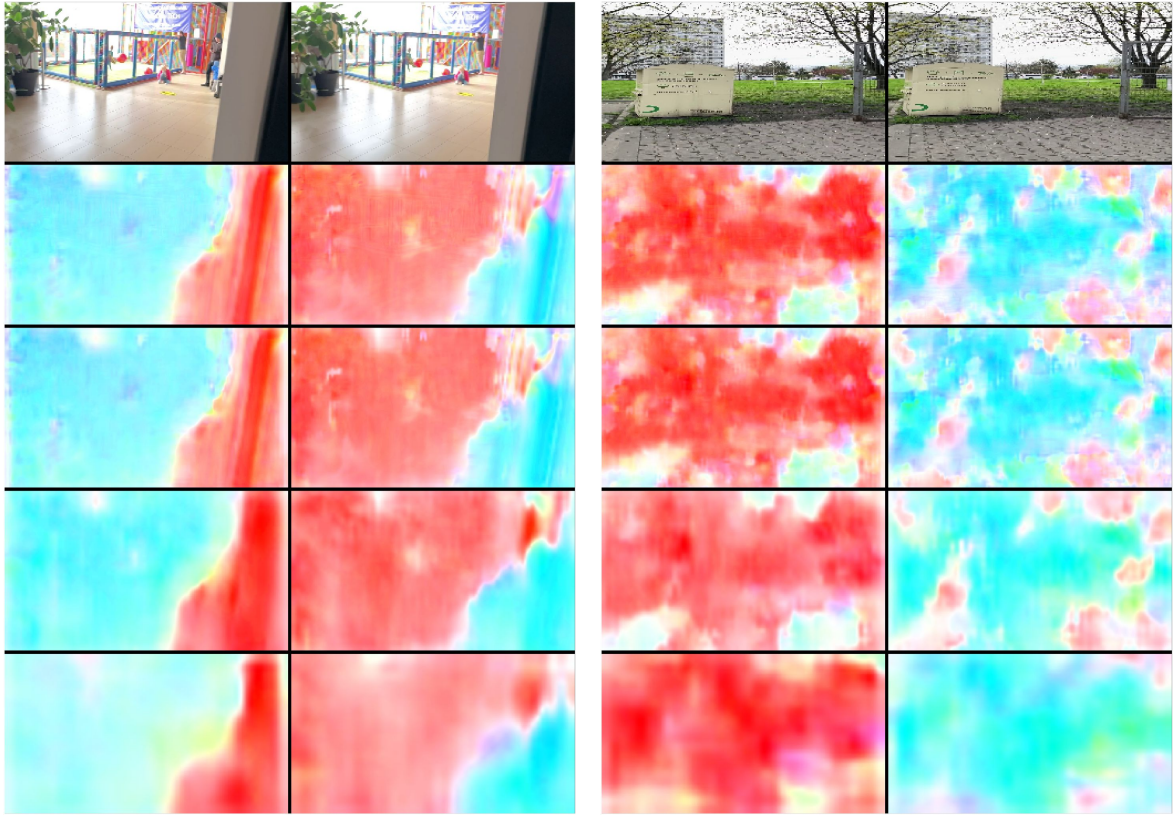


Decoder conv2d filters (37, output)





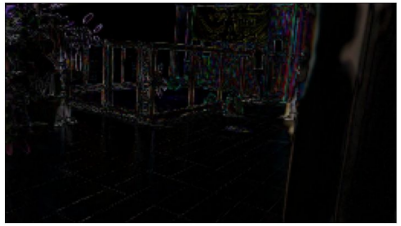
Predicted optical flow



Predicted images

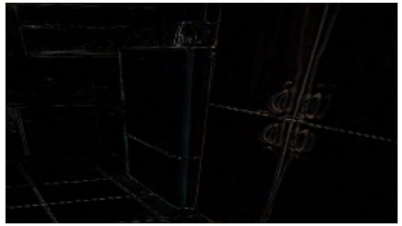
First frame

Predicted and Ground-truth difference



First frame

Predicted and Ground-truth difference



Predicted frame

Ground-truth frame

Predicted frame

Ground-truth frame

Second frame

Edge shift

Second frame

Edge shift

Project development

- Fix described model issues

Resources

1. Single Image Super Resolution with deep convolutional neural networks
2. Real-Time Intermediate Flow Estimation for Video Frame Interpolation
3. Depth-Aware Video Frame Interpolation
4. BiFormer: Learning Bilateral Motion Estimation via Bilateral Transformer for 4K Video Frame Interpolation
5. Attention is all you need
6. Video Frame Interpolation via Adaptive Convolution
7. Large Motion Frame Interpolation
8. FILM: Frame Interpolation for Large Motion
9. Multi-view Image Fusion
10. Perceptual Losses for Real-Time Style Transfer and Super-Resolution
11. Image Style Transfer Using Convolutional Neural Networks
12. Exploring Motion Ambiguity and Alignment for High-Quality Video Frame Interpolation
13. PWC-Net: CNNs for Optical Flow Using Pyramid, Warping, and Cost Volume