MRI Super-Resolution Through Generative Degradation Learning*

Supplementary Document

Yao Sui[1,2], Onur Afacan[1,2], Ali Gholipour[1,2], and Simon K. Warfield[1,2]

[1] Harvard Medical School, Boston, Massachusetts, United States
[2] Boston Children's Hospital, Boston, Massachusetts, United States
{yao.sui,onur.afacan,ali.gholipour,simon.warfield}@childrens.harvard.edu

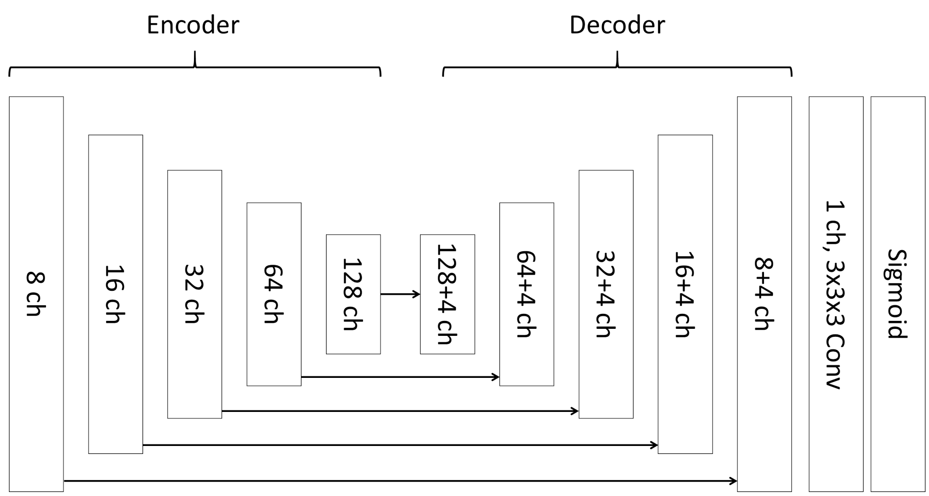
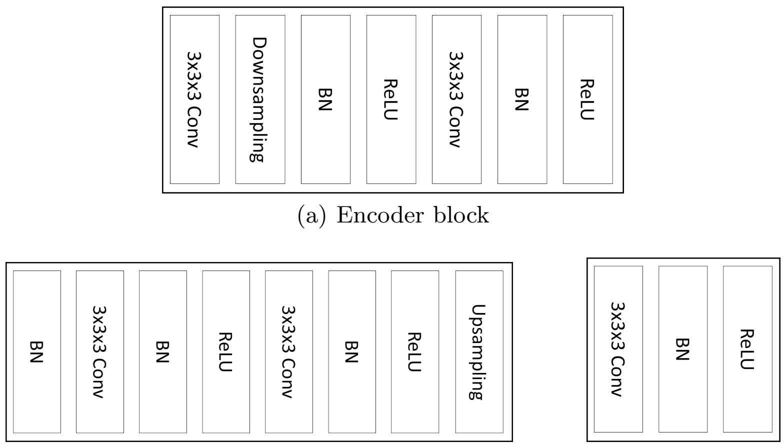


Fig. 1. Architecture of the high-resolution generative network.

This supplementary document provides the implementation details of our super-resolution reconstruction approach. As shown in Fig. 1, the generative function $f_{\boldsymbol{\theta}}$ in Eq. (3) is realized by an encoder-decoder architecture, followed by a convolutional layer that integrates the data from all channels into an image with a single channel and a sigmoid layer that rescales the voxel intensities in the

* This work was supported in part by the National Institutes of Health (NIH) under grants R01 NS079788, R01 EB019483, R01 EB018988, R01 NS106030, R01 EB031849, IDDRC U54 HD090255, S10OD025111; a research grant from the Boston Children's Hospital Translational Research Program; a Technological Innovations in Neuroscience Award from the McKnight Foundation; a research grant from the Thrasher Research Fund; and a pilot grant from National Multiple Sclerosis Society under Award Number PP-1905-34002.

range of $[0, 1]$. The channels of the output data of each encoder and decoder block are depicted in the corresponding boxes. Skip connections from the encoder to the decoder, denoted by the arrows in Fig. 1, are leveraged and provide additional 4 channels of data concatenated to the decoder outputs.



(a) Encoder block

(b) Decoder block (c) Skip block

Fig. 2. Implementations of the blocks in the high-resolution generative network.

The high-resolution generative network contains 5 encoder blocks, 5 decoder blocks, and 5 skip blocks. The implementation of each block is shown in Fig. 2. A reflection padding strategy is utilized in all convolutional layers. The downsampling layers isotropically decimate the inputs by a factor of 2, while the upsampling layers interpolate the inputs by a factor of 2 using a trilinear method.

Fig. 3. Architecture of the blur kernel learning network.

The generative function $g_{\boldsymbol{\omega}_j}$ is implemented by a fully connected network, as shown in Fig. 3. The network input is a one-dimensional Gaussian profile. The dropout layers randomly zero some of the elements of the input data with a probability of 25% using samples from a Bernoulli distribution.