FSRCNN

* Introduce a deconvolution layer at the end of the network, then the mapping is learned directly from the original low-resolution image (without interpolation) to the high-resolution one.
* Second, we reformulate the mapping layer by shrinking the input feature dimension before mapping and expanding back afterwards.
* Then, they (Dong et al., 2016b) adopted a deconvolutional layer at the end of the network in FSRCNN, which can reduce computation cost. However, the deconvolution operation is easy to introduce checkerboard artifacts (Odena et al., 2016). Different from the first two methods, Shi et al. (2016) adopted a novel upsampling method, called pixel shuffle, which is designed specifically for SR problem without extra parameters.
* Third, we adopt smaller ﬁlter sizes but more mapping layers.
* In SRCNN, as a pre-processing step, the original LR image needs to be upsampled to the desired size using bicubic interpolation to form the input. Thus, the computation complexity of SRCNN grows quadratically with the spatial size of the HR image (not the original LR image). For the upscaling factor n, the computational cost of convolution with the interpolated LR image will be n2 times of that for the original LR one. This is also the restriction for most learning-based SR methods [1, 3-7]. If the network was learned directly from the original LR image, the acceleration would be signiﬁcant, i.e., about n2 times faster.
* The large SRCNN (SRCNNEx) [2] has 57,184 parameters, which are six times larger than that for SRCNN (8,032 parameters).
* The proposed FSRCNN is different from SRCNN mainly in three aspects. First, FSRCNN adopts the original low-resolution image as input without bicubic interpolation.
* A deconvolution layer is introduced at the end of the network to perform upsampling.
* Second, The non-linear mapping step in SRCNN is replaced by three steps in FSRCNN, namely the shrinking, mapping, and expanding step.
* Third, FSRCNN adopts smaller ﬁlter sizes and a deeper network structure. These improvements provide FSRCNN with better performance but lower computational cost than SRCNN.
* FSRCNN can be decomposed into ﬁve parts–feature extraction, shrinking, mapping, expanding and deconvolution.
* The ﬁrst four parts are convolution layers, while the last one is a deconvolution layer.

