FIRN: Fast Invertible Rescaling Net

Junjae Lee
qwoqwo200@gmail.com

Abstract
Invertible Rescaling Net (IRN) [1] modeled the downscaling and up-scaling process using Invertible Neural Networks (INN) [1,22,26] instead of upsampling to the traditional Single-image super resolution (SISR) method. As a result, it showed significantly improved performance than the previous method. However, apart from its high performance, IRN requires a lot of computation. Hence, to improve this, we replace the existing dense block [2] with Pixel Attention Distillation Block (PADB). In addition, we use Charbonnier loss [43] instead of Mean Absolute Error (MAE) for the existing reconstruction loss. Through these improvements, we trade off the high performance and speed of the existing architecture and achieve higher performance than the lightweight SR model using the conventional method. In addition, by improving the perceptual loss and adversarial loss, we achieve perceptually satisfactory results than the model using the IRN+ method.

Key words: Deep Learning, Invertible Neural Networks, image super resolution, image processing

1 Introduction

SISR is a work of approximating high-resolution (HR) images based on low-resolution (LR) images. And since SRCNN [4], it has been evolved in the direction of using deep convolution neural network (CNN). In addition, SISR has been evolved in the direction of increasing Peak Signal-to-Noise Ratio (PSNR) [9,12,44,45,46]. However, there was a limit to the method of approximating HR image based on the LR image reduced by the traditional (e.g. bicubic [5]) method, and to solve this problem, methods of integrating the upscaling and downscaling of the image were proposed. [1,6,7,8] And IRN [1] succeeded in achieving high performance through an attempt to capture the characteristics of lost information by connecting two processes in this way. However, this model has the disadvantage of high computational cost apart from high performance.

Our first goal is to keep the high performance of the IRN as much as possible while reducing the computational cost. To do this, we replace the existing IRN with dense block [2] with PADB. In addition, we improve the existing restoration loss. Our second goal is to produce perceptually satisfactory results than the model using the IRN+ method. For this purpose, perceptual loss and adversarial loss are improved.

2 Related Work

SRCNN [4] first proposed SISR through deep learning. And SRCNN outperforms the traditional method. Various models that improve SRCNN appear. VDSR [9] improves performance by using deeper model, SRResNet [10] uses residual block [44] and SR–DenseNet [11] used a dense block [2]. and RDN [12] showed high performance by combining the residual block and the dense block. However, these methods had the disadvantage of high computational load apart from the high performance. And while maintaining such high performance as much as possible, there have been many attempts to reduce computational load [13,14,15,16,17,38,43]. Among them, IMDN [15] showed high performance compared to low computational load. RFDN [16], which recently improved IMDN, and PAN [17] using pixel attention have been proposed.

Existing image downscaling methods tend to omit details and make images smooth. In order to solve this problem, various downscaling methods have been proposed [18,19,20,21,8]. Among them, the method of training the existing SISR model through the content adaptive resampler achieved higher
performance than the existing method, however still has limitations [1]. Recently, IRN[1] used INN [1,22,26] to overcome these limitations and achieve high performance.

Usually, SR models are trained with the goal of increasing PSNR. However, it does not mean that an image with a high PSNR is a perceptually good image [10]. Hence, the existing SR model produced a perceptually poor image, and to solve this, SRGAN [10] using Generative adversarial network (GAN) [27] and perceptual loss [28]. ESRGAN [29] introduces a deeper architecture consisting of residual-in-residual dense blocks. It also improves SRGAN by using the feature map before the activation function in the VGG19-conv54 [45] and using a RaGAN [30]. CESRGAN [23] improves ESRGAN using Cosine Contextual (CCX) loss [24].

3 Proposed Methods
3.1 Network Architecture

In order to create a lighter model than the existing IRN [1], the original block is replaced with the proposed PADB. The structure of the invertible neural network blocks (InvBlock) remains as shown on the left in Fig. 1, but the existing dense block [2] is replaced with a new block as shown on the right in Fig. 1. The proposed new block is based on residual feature distillation block [16] as shown in Fig. 2, replaces contrast-aware channel attention layer [15] with pixel attention block [17]. In addition, a convolution layer is used before PADB. The new InvBlock effectively reduces the number of parameters of the existing IRN. Models using this new InvBlock have fewer parameters than the existing IRN while maintaining the existing performance as much as possible. The model using this new InvBlock is called FIRN.

Fig. 2: PADB used in the proposed FIRN
3.2 Reconstruction loss

Previously, MAE was used as the restoration loss to measure the pixel-by-pixel similarity between the original image and the restored image, because the MAE achieved a higher PSNR than the mean square error. [1] However, there is still room to improve the loss, so we use Charbonnier loss [43] as the reconstruction loss to achieve a higher PSNR. As shown in Tab. 2, when using the Charbonnier loss, a higher PSNR was achieved than MAE. The Charbonnier loss is defined as:

$$L_{\text{recon}}(\hat{y}, y) = \frac{1}{N} \sum_{i=1}^{N} \sum_{s=1}^{L} \rho \left( y_s^{(i)} - \hat{y}_s^{(i)} \right),$$

where $\rho(x) = \sqrt{x^2 + \varepsilon^2}$, and $\varepsilon = 10^{-6}$. In addition, $y$ and $\hat{y}$ mean the original image and the generated image, respectively.

3.3 RaGAN

In the existing SR field, using GAN [27] tends to make the existing image more realistic [10]. And when using a RaGAN, it tends to make it more realistic than the existing GAN [29]. Therefore, FIRN+ uses the RaGAN [30]. The discriminator loss is then defined as:

$$L_{D_{Ra}} = -E_y [\log(D_{Ra}(y, \hat{y}))] - E_{\hat{y}} [\log(1 - D_{Ra}(\hat{y}, y))],$$

The adversarial loss for generator is then defined as:

$$L_{G_{Ra}} = -E_y [\log(1 - D_{Ra}(y, \hat{y}))] - E_{\hat{y}} [\log(D_{Ra}(\hat{y}, y))].$$

Where $y$ and $\hat{y}$ mean the original image and the generated image, respectively. $D$ and $G$ mean discriminator and generator, respectively.

3.4 perceptual loss

perceptual loss [28] leads to more realistic images than existing content loss [10]. And, using feature maps pulled before the activation function in the VGG19-conv54 [45] generate more satisfactory results than after [29]. And using CCX loss [24] instead of existing perceptual loss generate more visually satisfactory results [23]. Therefore, for FIRN+, we use CCX loss. CCX loss is defined as:

$$L_{\text{CCX}}(\phi(\hat{y}), \phi(y)) = -\log \left( \frac{1}{N} \sum_j \max_i A_{ij} \right),$$

$$A_{ij} = \frac{e^{(1-d_{ij}/h)}}{\sum_k e^{(1-d_{ik}/h)}},$$

$$d_{ij} = \min_k d_{ik} + \varepsilon,$$

$$d_{ij} = \frac{(x_i - r) \cdot (y_j - r)}{||x_i - r||_2 \times ||y_j - r||_2},$$

where $h > 0$ is a bandwidth parameter, $\varepsilon = 10^{-6}$ and $r = \frac{1}{N} \sum_j y_j$. $y$ and $\hat{y}$ mean the original image and the generated image, respectively. $\phi$ denotes the feature mapping from before the activation function in the VGG19-conv54.

the overall loss of FIRN+ is defined as:

$$L_{\text{FIRN+}} = \lambda_1 L_{\text{recon}} + \lambda_2 L_{\text{guide}} + \lambda_3 L_{G_{Ra}} + \lambda_4 L_{\text{CCX}},$$

Where $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ are coefficients for balancing different loss conditions. In addition where $L_{\text{guide}}$ is defined as:
\[ L_{\text{guide}}(\hat{y}_{LR}, y_{LR}) = \frac{1}{N} \sum_{i=1}^{N} (\hat{y}_{LR} - y_{LR})^2, \]

Where \( y_{LR} \) and \( \hat{y}_{LR} \) are the original LR image and the generated LR image, respectively.

## 4 Experiments

### 4.1 Data

The training set used uses DF2K, which incorporates DIV2K [31] and Flickr2K [32]. DF2K contains 3450 high-quality 2k resolution images. In addition, four data sets for evaluation (Set5[33],Set14 [34],BSD100[35],Urban100[36]) are used. Using this data set, PSNR and SSIM [20] are evaluated in the Y channel displayed in the YCbCr (Y, Cb, Cr) color space.

### 4.2 Training Details

We train and compare model using two downscaling modules. downscaling module has 8 InvBlocks. The training set used uses DF2K, which incorporates DIV2K, Urban100, and Flickr2K are evaluate in the Y channel displayed in the YCbCr (Y, Cb, Cr) color space.

### 4.3 Results

**Table 1**: Quantitative evaluation results of various downscaling and upscaling methods for image reconstruction for data set Set5, Set14, BSD100, and Urban100 (PSNR/SSIM): Red/Blue : Best/Second Best

<table>
<thead>
<tr>
<th>Upscaling&amp;Downscaling</th>
<th>Scale</th>
<th>Param</th>
<th>Set5</th>
<th>Set14</th>
<th>B100</th>
<th>Urban100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bicubic&amp;Bicubic</td>
<td>4x</td>
<td>0.05M</td>
<td>30.48/0.8628</td>
<td>27.59/0.7581</td>
<td>26.90/0.7101</td>
<td>24.52/0.7221</td>
</tr>
<tr>
<td>SRCNN&amp;Bicubic</td>
<td>4x</td>
<td>0.01M</td>
<td>30.51/0.8657</td>
<td>27.59/0.7550</td>
<td>26.90/0.7104</td>
<td>24.52/0.7224</td>
</tr>
<tr>
<td>SRResNet&amp;Bicubic</td>
<td>4x</td>
<td>1.51M</td>
<td>32.17/0.8951</td>
<td>28.61/0.7823</td>
<td>27.89/0.7365</td>
<td>26.12/0.7871</td>
</tr>
<tr>
<td>EDSR&amp;Bicubic</td>
<td>4x</td>
<td>40.7M</td>
<td>32.62/0.8984</td>
<td>28.94/0.7891</td>
<td>27.79/0.7437</td>
<td>26.86/0.8080</td>
</tr>
<tr>
<td>RCAN&amp;Bicubic</td>
<td>4x</td>
<td>15.6M</td>
<td>32.63/0.9002</td>
<td>28.87/0.7889</td>
<td>27.77/0.7436</td>
<td>26.82/0.8087</td>
</tr>
<tr>
<td>ESRGAN&amp;Bicubic</td>
<td>4x</td>
<td>16.3M</td>
<td>32.74/0.9012</td>
<td>29.00/0.7915</td>
<td>27.84/0.7455</td>
<td>27.03/0.8152</td>
</tr>
<tr>
<td>SAN&amp;Bicubic</td>
<td>4x</td>
<td>15.8M</td>
<td>32.64/0.9003</td>
<td>28.92/0.7908</td>
<td>27.87/0.7436</td>
<td>26.79/0.8086</td>
</tr>
<tr>
<td>RDN&amp;Bicubic</td>
<td>4x</td>
<td>22.2M</td>
<td>32.47/0.8990</td>
<td>28.81/0.7871</td>
<td>27.72/0.7419</td>
<td>26.61/0.8028</td>
</tr>
<tr>
<td>VDSR&amp;Bicubic</td>
<td>4x</td>
<td>0.66M</td>
<td>31.35/0.8838</td>
<td>28.01/0.7624</td>
<td>27.26/0.7251</td>
<td>25.18/0.7524</td>
</tr>
<tr>
<td>DRCN&amp;Bicubic</td>
<td>4x</td>
<td>1.77M</td>
<td>31.53/0.8854</td>
<td>28.02/0.7670</td>
<td>27.23/0.7233</td>
<td>25.14/0.7510</td>
</tr>
<tr>
<td>LapSRN&amp;Bicubic</td>
<td>4x</td>
<td>0.81M</td>
<td>31.54/0.8850</td>
<td>29.19/0.7720</td>
<td>27.32/0.7280</td>
<td>25.21/0.7560</td>
</tr>
<tr>
<td>IDN&amp;Bicubic</td>
<td>4x</td>
<td>0.55M</td>
<td>31.82/0.8903</td>
<td>28.25/0.7730</td>
<td>27.41/0.7297</td>
<td>25.41/0.7632</td>
</tr>
<tr>
<td>MemNet&amp;Bicubic</td>
<td>4x</td>
<td>0.67M</td>
<td>31.74/0.8893</td>
<td>28.26/0.7723</td>
<td>27.40/0.7281</td>
<td>25.50/0.7630</td>
</tr>
<tr>
<td>IMDN&amp;Bicubic</td>
<td>4x</td>
<td>0.71M</td>
<td>32.21/0.8948</td>
<td>28.58/0.7811</td>
<td>27.56/0.7353</td>
<td>26.04/0.7838</td>
</tr>
<tr>
<td>RFDN&amp;Bicubic</td>
<td>4x</td>
<td>0.55M</td>
<td>32.24/0.8952</td>
<td>28.61/0.7819</td>
<td>27.58/0.7360</td>
<td>26.11/0.7858</td>
</tr>
<tr>
<td>PAN&amp;Bicubic</td>
<td>4x</td>
<td>0.73M</td>
<td>32.13/0.8948</td>
<td>28.61/0.7822</td>
<td>27.52/0.7356</td>
<td>26.11/0.7854</td>
</tr>
<tr>
<td>EDSR&amp;CAR</td>
<td>4x</td>
<td>52.8M</td>
<td>33.88/0.9174</td>
<td>30.31/0.8382</td>
<td>29.15/0.8001</td>
<td>29.28/0.8711</td>
</tr>
<tr>
<td>IRN</td>
<td>4x</td>
<td>4.35M</td>
<td>36.19/0.9451</td>
<td>32.67/0.9015</td>
<td>31.64/0.8826</td>
<td>31.41/0.9157</td>
</tr>
<tr>
<td>FIRN(ours)</td>
<td>4x</td>
<td>0.32M</td>
<td>33.55/0.9189</td>
<td>30.01/0.8452</td>
<td>29.56/0.8301</td>
<td>27.43/0.8406</td>
</tr>
</tbody>
</table>

We compare the proposed FIRN with various SR methods for upscaling factor 4x, including SRCNN[4],FSRCNN[38],SRResNet[10],EDSR[39],RCAN[40],ESRGAN[29],SAN[41],RDN[12],VDSR[9],DRCN[42],LapSRN[43],IDN[13],MemNet[14],IMDN[15],RFDN[16],PAN[17], CAR[8] and IRN[1]. It can be seen from the Tab. 1 that the proposed FIRN has a high performance while using a parameter of 320k.

### 4.4 Ablation Study
Table 2: Performance comparison by type of restoration loss (MAE, Charbonnier), red indicates the best performance.

<table>
<thead>
<tr>
<th></th>
<th>Set5</th>
<th>Set14</th>
<th>B100</th>
<th>Urban100</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAE</td>
<td>33.51/0.9190</td>
<td>29.98/0.8454</td>
<td>29.55/0.8300</td>
<td>27.39/0.8399</td>
</tr>
<tr>
<td>Charbonnier</td>
<td>33.55/0.9189</td>
<td>30.01/0.8452</td>
<td>29.56/0.8301</td>
<td>27.43/0.8406</td>
</tr>
</tbody>
</table>

An experiment is conducted to analyze the performance according to the restoration loss. As shown in Tab. 1, FIRN has higher PSNR when using Charbonnier loss than when using MAE loss.

In order to study the effect of each component in the proposed FIRN+, we correct several losses in FIRN applied the method of IRN+ and compare the effects. The overall visual comparison is illustrated in Fig. 3. A detailed discussion is provided as follows.

If CCX was used instead of existing perceptual loss, the texture became sharper than before, as observed by CESRGAN, resulting in satisfactory results.

Similar to the observation result of ESRGAN by applying a RaGAN, the texture becomes sharper and more visually satisfactory than before.

Fig. 3: Overall visual comparison to show the effect of each component in FIRN. Each model was compared after 100k times training based on pre-trained FIRN.

As you can see from Fig. 3, improving losses can give you more visually satisfactory results than before.
5 Conclusion

We propose a lighter FIRN while maintaining the performance of the existing IRN as much as possible. We introduced PADB to make the network lighter. In addition, Charbonnier loss is used as the reconstruction loss. We succeeded in creating a lighter network while maintaining the existing performance as much as possible.

FIRN+ improves the existing perceptual loss and adversarial loss. Through these improvements, it generates a more perceptually satisfactory image than FIRN using existing method.

6 References


