(Visual) Attention is All You Need: An Exploration on Attention-Based Image Super-Resolution

Athiya Deviyani
Department of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
adeviyan@cs.cmu.edu

Udaikaran Singh
Department of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
udaikars@andrew.cmu.edu

1 Introduction

Image super-resolution is in essence the task of interpolation of unknown pixels when upscaling an image. A naive approach to this task is to spread out the pixels into the higher-resolution space and estimate mixing pixel values with nearby pixels, such as taking a weighted average of neighboring pixels. However, this tends to fail in reproducing images believable to the human eye. Neural networks improve on this naive approach by leveraging a prior distribution of training images in order to hallucinate missing values.

Although super-resolution is in practice a unsupervised task, it can be transformed into a supervised problem by randomly downsampling high resolution images. This downsampling allows us to recreate the setting of having low resolution images, but now with ground truth labels. With such ground truth labels, we can effectively train a model in a supervised fashion for this task.

The primary goal of single image super-resolution (SISR) is to recover a high resolution (HR) image from its degraded low resolution (LR) counterpart [1]. The task still proves to be a challenge as there is a limited number of naturally LR-HR image pairs, so several methods have been proposed to artificially generate LR images from a HR image by using downsampling kernels and adding noise. Current state-of-the-art techniques in the image super-resolution domain involve the use of CNN-based models. With SRCNN, Dong et al. [2] introduced the first deep to address the SISR with only three convolution layers. Models such as EDSR [3] are introduced shortly after, and will be discussed further in the methods and models section. For this project, we would like to investigate the performance of a attention-based models in the image super-resolution task.

In recent years, transformers have shown success in the natural language processing field, leveraging self-attention to model long-range dependencies in an input sequence. They have also addressed several of the limitations of RNNs and CNNs, such as scalability, parallelizability and complexity. Even more recently, transformers have shown promising results in the field of computer vision, particularly in popular tasks such as object detection and image classification which were previously dominated by CNNs [4].

We will use the DIV2K dataset [5] to train our single-image super-resolution models. The dataset contains 800 high definition high resolution images and low resolution images that are bicubically downscaled by 2, 3, and 4 downscaling factors. We will then test the performance of our models against external datasets that are commonly used in image super-resolution tasks. We will use the Peak Signal to Noise Ratio (PSNR) [6] and Structural Similarity Index Measure (SSIM) [7] metrics on the external dataset to check how our model generalizes to real-world LR images.

2 Background

In our midway report, we aimed to explore the current approaches to super-resolution and establish a baseline for our experimentation. The ultimate goal of our project is to demonstrate that the use of attention blocks can be used as a potential improvement to the model. Particularly, our conclusion was primarily that most state of the art approaches in the field utilized convolutional neural networks or convolutional neural networks in additional to another technique for this task, such as using a generative adversarial network. We also looked to utilize an extremely naive approach of using a sharpening kernel in order to establish a non-machine learning baseline for the task of interpolation.

In addition to exploring the current modeling approaches for this task, we found that the metrics used for the task of super-resolution are primarily PSNR and SSIM. The problem of super-resolution is inherently ill-posed as there is not a well-defined solution to the generation of high resolution images. However, these metrics typically are used as heuristics to the goodness of the upscaling by the network. Some papers went beyond using these heuristics by using human subjects to judge the realism of the upscaling [8]. This use of human judgement reflects the ultimate goal of recreating realistic images; however, due to time and resource limitations, we settled to using heuristic metrics measurements and visual inspection to evaluate our performance.

Lastly, we found in our preliminary experimentation that reported results on the DIV2K dataset tended to be varied and difficult to reproduce. Even when using pre-trained models on the DIV2K dataset, our resulting PSNR values tended to be much lower than the reported values in papers and code repositories. This indicates that this task requires a large amount of hyperparameter tuning.

3 Related Work

3.1 Deep Learning in Super-Resolution

In upscaling, the primarily task is the interpolation between pixels in the high resolution image. Super-resolution aims to leverage prior knowledge about images in order to add information not present in the low resolution image in order to achieve this interpolation. Prior to deep learning, methods included image statistic methods, edge prediction methods, and example based methods [9]. However, the introduction of deep learning case from the use SRCNN, as shown in figure 1, which created a simple multi-layer convolutional neural network [10]. This demonstrated that this CNN architecture has theoretical equivalencies with sparse-coding methods in which patches of the images are learned based on encoding of patches within the a dataset of images. Neural networks improves on the sparse-coding methods by learning the appropriate weightings of patches in an end-to-end method using deep learning.

3.2 EDSR

Building on this introduction to of convolutional neural networks, Enhanced Dynamic Residual Networks for Single Image Super-Resolution (EDSR) improved on previous models by applying residual blocks to the model architecture [3]. EDSR fundamentally borrows from ResNet [11] by introducing skip connections present as residual blocks. Skip connections allows for a better flow of information within the network by adding outputs from previous layers. This alleviates the issue of a transformation within a hidden layer of a model potentially degrading important information stored up that point in the neural network's forward pass. EDSR primarily motivated our use of residual blocks in our modeling.

3.3 Iterative Methods for Super-Resolution

In additional to the use of residual blocks to allow for quicker training of the convolutional network, another improvement to a rudimentary CNN model is the use of iterative upscaling. Specifically, this iterative approach is found in LapSRN [12] (Deep Laplacian Pyramid Networks), more formally called as progressive reconstruction. In this network architecture, there are are two parallel branches; in the first, there is a Laplacian pyramid network constructed to predict residuals from the upscaling and the second model to perform normal upscaling using CNN. LapSRN differs from traditional CNN architectures by appling a reconstruction loss to different layers of the network. This allows for

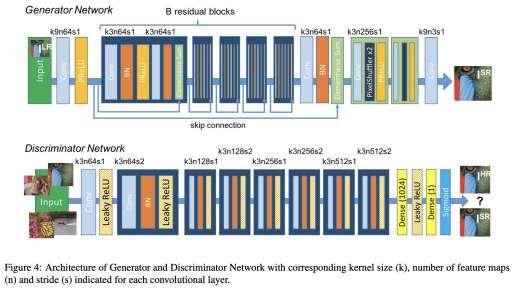
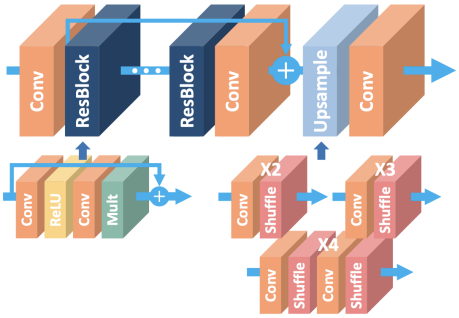


Figure 4: Architecture of Generator and Discriminator Network with corresponding kernel size (k), number of feature maps (n) and stride (s) indicated for each convolutional layer.

(a) SRGAN architecture [13]



(b) Single-Image SR EDSR architecture [3]

Figure 1: Architecture Designs of state-of-the-art CNN Models

the model to avoid overfitting to outliers by learning intermediate layers in a supervised manner. This also informs the use a single architecture for different levels of upscaling by simply truncating the existing network.

3.4 Generative Adversarial Networks for Super-Resolution

Generative Adversarial Networks (GAN) were invented for the unsupervised learning task of learning the distribution of the input space and generating synthetic data [14]. In a GAN, there are two distinct architectures: a generator network and a discriminator network. The generator looks to generate data from a random sampling. The discriminator looks to learn to differentiate between real and synthetic data. By using the generator and a real-world dataset, we can implicitly construct a dataset with a binary label of being real or not. GANs explicitly define an adversarial task in which the discriminator and generator look to outperform each other. This implicitly causes the generator to learn a proper distribution of the input data space in order to generate data mimicking real-world data.

In previous framings of the super-resolution problem, the task has been defined as a supervised learning problem in which we look to reconstruct a high resolution image that has been randomly downscaled (usually using bicubic scaling). However, super-resolution can also be seen as a problem of learning the distribution of high-resolution images. An example of this in action is using SRGAN [13].

3.5 Visual Transformers

The attention module was created for the transformer network in the paper "Attention is all you need" [15]. Transformers have been used heavily in the Natural Language Processing field as an encoder-decoder for sequence to sequence tasks like machine translation and text generation. Underlying transformers' success is the advancement of the attention module in lieu of convolutional layers and recurrence relationships. The attention module effectively acts as a "look-up" by dynamically learning a weighting of the encoding of the sequence. This weighting attributes an importance to each part of the sequence, which may be effective in the decoder by allowing for a wider search in the sequence compared to previous models.

Given that transformers were first introduced in 2017, most of the early applications were for the NLP domain. More recently, transformers have been adopted for tasks in the computer vision domain. In a visual transformer (ViT), the image is split into patches and treated as a sequence on which the attention modules can be applied. For example, for benchmark image recognition tasks like ImageNet and Cifar-100, ViT's were found to perform similarly well as convolutional neural networks [4]. However, many of the results were mixed; ViT's tended to perform poorly compared to a CNN when using smaller datasets. However, when using a larger dataset, the performance was comparable to state of the art convolutional neural networks used for image recognition tasks. Even so, training a ViT to a stable state in various tasks still remains a challenge today.

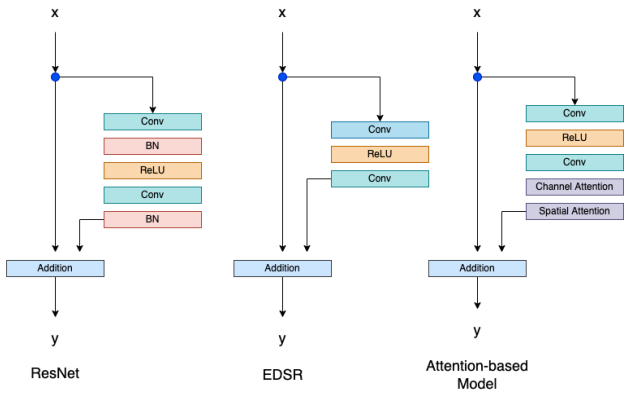


Figure 2: Residual Block Diagram

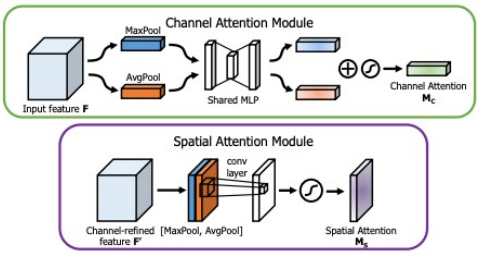


Figure 3: Convolutional Block Attention Module

4 Methods

4.1 Full attention-based models

In our first attempt, we looked to make an entirely attention-based architecture for super-resolution. This is in the same vein as the "Attention is all you need" [15] paper. The motivation for this design decision is based on the success of ViT on other computer vision tasks, particularly image recognition tasks [4]. We presumed that the success in other computer vision tasks may translate to super-resolution. However, we found that using an attention-only model failed to reach results anywhere close to the state of the art.

The failure of an attention-only model could be attributable to a multitude of reasons. To begin, the inductive bias for a ViT is far less useful for a computer vision setting than a CNN. When using a CNN, the use of learned kernels allows that are translationally invariant. However, a ViT is learning relationships between patches on the image globally. This global nature of the attention nodes both allows for both more expressive models and more difficult to learn models. Also, ViT in the literature have been found to give varying and sometimes unpredictable results on computer vision tasks [4].

Due to this poor performance, we did not pursue the idea of implementing a completely attention-based model for super-resolution. Rather, we looked towards hybrid models that combined convolution layers and attention modules; this hybrid model is even promoted as a potential possibility for exploration in the paper introducing visual transformers [4].

4.2 Attention Augmented Convolutional Networks

Rather than using a fully attention-based model, we opted for using a hybrid model in which we have an attention module after a convolutional layer. Our intuition is that rather than learning a linear mapping between the kernels, the attention module would be allow for a more expressive feature mapping than a dense layer. We hypothesize that this expressiveness would lead to an improve on the previous baseline models. Our approach is similar to Convolutional Block Attention Module (CBAM) [16], but we differ in that we do not train our convolutional and attention model separately. Rather,

we train them jointly. In order to experiment our hypothesis, we modified a variety of established super-resolution architectures in order to test if an attention module improves on previous baselines.

We have used a combination of convolution layers with channel and spatial attention maps to build our Convolutional Block Attention Module. Given an intermediate feature map $\mathbf{F}$ with C channels, height H and width W, the CBAM sequentially infers a 1D channel attention map $\mathbf{M_c}$ of size Cx1x1 and a 2D spatial attention map $\mathbf{M_s}$ of size 1xHxW. When combined, the whole process can be formally defined as:

$$\mathbf{F}' = \mathbf{M_c}(\mathbf{F}) \otimes \mathbf{F}$$
$$\mathbf{F}'' = \mathbf{M_s}(\mathbf{F}') \otimes \mathbf{F}'$$

In the above formulation, $\otimes$ denote the element-wise multiplication operation, where the attention values are broadcasted accordingly. This formulation is described visually in Figure 3.

The channel attention map is generated by initially doing global average pooling of the feature map $\mathbf{F}$. The resulting channel vector $\mathbf{F_c}$ is of size Cx1x1 and is passed through a small multiplayer perceptron of one hidden layer of the dimension C/r, where r is the reduction ration for the hidden channel. Finally, a batch normalization layer is added to the multilayer perceptron. The same process is done with maximum average pooling of the feature map $\mathbf{F}$, in an attempt to have more distinctive channel features. This process is formally defined as:

$$\mathbf{M_c}(\mathbf{F}) = \sigma(MLP(AveragePool(\mathbf{F})) + MLP(MaxPool(\mathbf{F})))$$

The spatial attention map is generated by taking the input feature map F and generating two intermediate feature maps generated by global average pooling and max pooling. These feature maps are concatenated and passed through a 1-dimensional convolutional block of 7x7 kernel size. This process is formally defined as:

$$\mathbf{M_s}(\mathbf{F}) = \sigma(f^{7x7}(Concat(AveragePool(\mathbf{F}), MaxPool(\mathbf{F})))$$

We will be using these elements in the Convolutional Attention Block Module as the attention blocks to be integrated in existing single-image super-resolution models.

4.2.1 EDSR with Attention

For our EDSR model, we have used the architecture described in the original paper [3], with 16 residual blocks with 64 filters. At the end of each residual block, we have added the Convolutional Attention Block Module. The final model has 1.52M parameters.

4.2.2 ESPCN with Attention

The Efficient Sub-pixel Convolutional Neural Network (ESPCN) [17] attempts to solve a problem faced by the CNN-based approaches such as SRCNN and VDSR by removing the need to use interpolatiomn methods to upsample the low-resolution image. Instead, ESPCN increases the resolution at the very end of the network and the upscaling is handled by the last layer. This allows ESPCN to perform single-image super-resolution on-par with the state-of-the-art methods with fewer computational complexity and cost. For our implementation, we have used 3 convolution layers, where the first layer has 64 filters and a kernel size of 5x5, the second layer has 32 filters and a kernel size of 3x3, and the final layer has a kernel of size 3x3. We have added the Convolutional Attention Block Module after each convolution layer.

4.2.3 SRGAN with Attention

We have implemented SRGAN as described in the original paper [13], where the resulting model has 1.55M parameters. The model is trained with the VGG54 context loss. This loss function is used to improve the performance by comparing more high level features of the image through looking at the intermediate activation of the pre-trained VGG-19 network.

The generator module of SRGAN follows the SRResNet architecture, with 16 residual blocks with 64 filters. The discriminator comprises of 8 discriminator blocks containing a single convolution layer

where the filter sizes increase from 64, 128, 256, to 512 each two-block discriminator block pairs. The second block in each pair has a stride value of 2.

4.2.4 LAPSRN with Attention

We have implemented the LapSRN architecture introduced by its original authors [12], consisting of 27 layers with residual blocks, progressive reconstruction and the Charbonnier loss function used for training. The authors argue that the Charbonnier loss function is much more robust against outliers compared to the L2 loss (mean squared error) which is used on most single-image super-resolution models. We have added the Convolutional Block Module after each residual block in the feature extraction section of LapSRN.

4.3 Data and Training

For our training and testing, we used the DIV2K dataset [5]. This dataset is composed of 800 high resolution images in the training set and 100 images in both the validation and test set. Although different forms of downsampling are supported, we only focused on 4x bicubic downsampling for our experimentation. For our training process, we used Adam optimizer with a learning rate of 0.001 and learning rate decay in all experimentation. We also applied random augmentations to the images in the training data to help with generalization. The models were trained on a large Amazon EC2 G4 instance equipped with an NVIDIA T4 GPU and 3 vCPUs. The SRGAN took around 13 hours to train, while the remaining models only took around 5 hours. We have trained the non-attention and attention-based models separately.

5 Results

5.1 Metrics

For the evaluation of our models, we used PSNR and SSIM, which are formally defined below. These are typically used measurements of the reconstruction error of an image. The task in super-resolution is more appropriately characterized by the ability to create believable upscaled images. However, with the existence of ground-truth images in our problem setting, we can approximate this goal by the reconstruction error.

$$PSNR(x,y) = \frac{10\log_{10}[\max(\max(x),\max(y))]^2}{(x-y)^2} \qquad SSIM(x,y) = \frac{(2\mu_x\mu_y+C_1)+(2\sigma_{xy}+C_2)}{(\mu_x^2+\mu_y^2+C_1)(\sigma_x^2+\sigma_y^2+C_2)}$$

5.2 Results

Our results are shown in Figures 4, 5, and 6. It shows primarily that training with an additional attention module shows minimal improvement over the baseline. Looking at the resolved images in Figure 6, it can be seen that the reconstructions using attentions tend to have less artifacts of sharp white lines and less emphasis on elements with softer lines. It appears that the attention-based models emphasize less on resolving harder lines, however after a closer look we can observe that they resolve finer lines better within an image. However, the differences are very subtle, and this corresponds to the minute differences between the non-attention and attention-based models in Figure 4.

It can also be seen that the SRGAN produces reconstructions are are generally smoother than the other models, but the differences between the models is very slight. Generally, all the models outperform our sharpening kernel baseline, but that is a very naive approach to this problem.

5.3 Evaluation

Our results show that there is a general small improvements with by adding attention modules to state of the art models. We found that in our training, we were not able to reproduce the advertised results of the state of the art models; however, we think this is likely due to a lack of hyper-parameter search in our models. Due to our experimentation involving relatively large models and datasets, we had very limited hyper-parameter search or neural architecture search.

Therefore, it is difficult to conclude that using additional attention modules will lead to a significant improvement for the task. However, we did find that using attention did improve the heuristic values

Model	PSNR	SSIM
EDSR	28.7709	0.7965
EDSR augmented w/ attention	28.8371	0.7938
SRGAN	29.3513	0.78279
SRGAN augmented w/ attention	30.4334	0.7974
ESPCN	27.3786	0.7612
ESPCN augmented w/ attention	27.4080	0.7619
LAPSRN	28.3517	0.78884
LAPSRN augmented w/ attention	28.3681	0.7895

Figure 4: Results on DIV2K Test Set

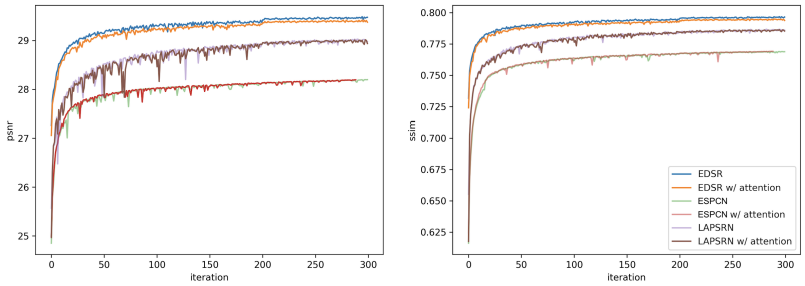


Figure 5: PSNR and SSIM of EDSR, ESPCN and LAPSRN on validation set during training

across the board for all our architectures. This indicates inconclusively that the attention module is able to at reproduce the results of convolutional networks with potential to beat those baselines with further tuning. We expected that using the additional attention modules would improve upon our baseline models. Our experimental results reflected this expectation, but we had generally very limited improvement from the baseline models.

6 Discussion and Analysis

The limitations of our approach is primarily that we did not use have a large architecture or hyper-parameter search in order to fine-tune our architecture. This was primarily due to the fact that we had limited time and computational resources.

Due to this limitation on computational resources, we make a general assumption that attention blocks are most appropriate after the convolutional layers in a residual block. The use of residual blocks comes from the success of models like EDSR [3], and the choice of placing the attention block after the convolutional layers comes primarily from work on CBAM [16]. We do not have testing to theoretically or empirically support these design decisions.

Our experimentation does provide an insight to the machine learning community by demonstrating that adding the attention module to existing state-of-the-art single-image super-resolution models provides small improvements to the baseline state of art models.

6.1 Future Work

For future work, we think further testing should be done in implementing a complete visual transformer (ViT) for this task. Our experimentation showed that using attention modules alongside feature mappings from convolutional layers can be learned by a model in order to improve performance. However, this does not give definitive evidence for the effectiveness that utilizing attention modules is appropriate for this task. Alongside this would likely require more testing around possible architecture decisions and hyper-parameter tuning.

In addition, so other forms of experimentation that would be effective given more time and resources is testing the effectiveness of attention modules in settings of differing dataset sizes and different levels of upscaling (2x, 3x, 8x, etc.) in order to see which settings attention modules are more appropriate. It was seen in earlier papers [4] that the ViT had varying performances on different datasets, so maybe a similar phenomena can be shown in the super-resolution setting.

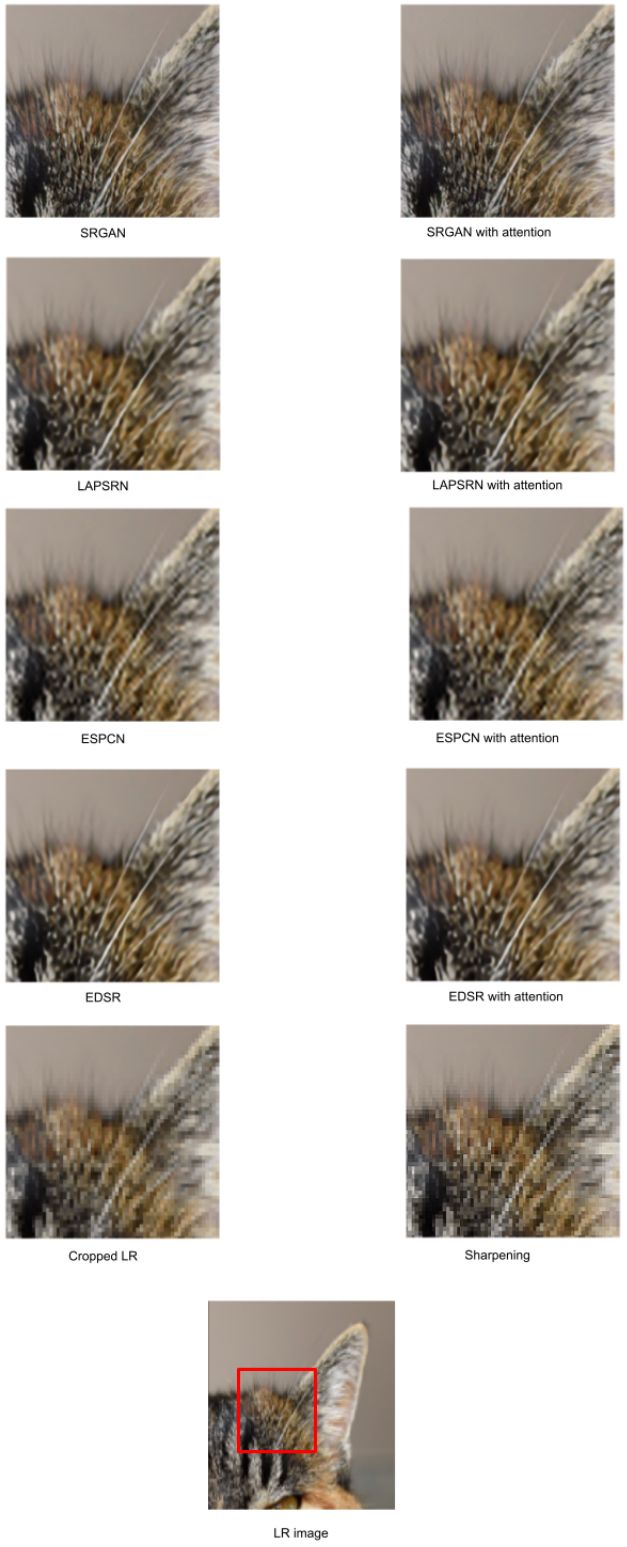


Figure 6: Super-resolved images obtained through various super-resolution methods

A Appendix

Implementations of our models and training can be found on the following GitHub repositories:

- `https://github.com/athiyadeviyani/super-resolution`
- `https://github.com/UdaikaranSingh/10707_project`

References

[1] Chih-Yuan Yang, Chao Ma, and Ming-Hsuan Yang. Single-image super-resolution: A benchmark. In *European conference on computer vision*, pages 372–386. Springer, 2014.

[2] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *CoRR*, abs/1501.00092, 2015. URL `http://arxiv.org/abs/1501.00092`.

[3] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. *CoRR*, abs/1707.02921, 2017. URL `http://arxiv.org/abs/1707.02921`.

[4] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *CoRR*, abs/2010.11929, 2020. URL `https://arxiv.org/abs/2010.11929`.

[5] Andrey Ignatov, Radu Timofte, et al. Pirm challenge on perceptual image enhancement on smartphones: report. In *European Conference on Computer Vision (ECCV) Workshops*, January 2019.

[6] Osama S. Faragallah, Heba El-Hoseny, Walid El-Shafai, Wael Abd El-Rahman, Hala S. El-Sayed, El-Sayed M. El-Rabaie, Fathi E. Abd El-Samie, and Gamal G. N. Geweid. A comprehensive survey analysis for present solutions of medical image fusion and future directions. *IEEE Access*, 9:11358–11371, 2021. doi: 10.1109/ACCESS.2020.3048315.

[7] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. doi: 10.1109/TIP.2003.819861.

[8] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *arXiv:2104.07636*, 2021.

[9] Daniel Glasner, Shai Bagon, and Michal Irani. Super-resolution from a single image. In *2009 IEEE 12th International Conference on Computer Vision*, pages 349–356, 2009. doi: 10.1109/ICCV.2009.5459271.

[10] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *CoRR*, abs/1501.00092, 2015. URL `http://arxiv.org/abs/1501.00092`.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. URL `http://arxiv.org/abs/1512.03385`.

[12] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Fast and accurate image super-resolution with deep laplacian pyramid networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[13] Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew P. Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network. *CoRR*, abs/1609.04802, 2016. URL `http://arxiv.org/abs/1609.04802`.

[14] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014. URL `https://arxiv.org/abs/1406.2661`.

[15] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017. URL `http://arxiv.org/abs/1706.03762`.

[16] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. CBAM: convolutional block attention module. *CoRR*, abs/1807.06521, 2018. URL `http://arxiv.org/abs/1807.06521`.

[17] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1874–1883, 2016.