Final Project: Different approaches towards Image Super Resolution

Dhruvil Parikh
CS5330: Pattern Recognition and
Computer Vision
Northeastern University
Boston, USA
parikh.dh@northeastern.edu

***Abstract*—Image Super Resolution is essential in the field of Computer Vision. It has a plethora of real world applications. The central purpose of Super Resolution is the generation of higher resolution image from lower resolution image offering a high pixel density and thereby providing a more detailed scene in the original frame. It is necessary at times as low-resolution images saves storage space, are easier to process, easier to transfer, requires low computation as compared to their high resolution counterparts. High resolution is of utmost importance in medical imaging for diagnosis. Other applications are in zooming to the specific area of interest in the image wherein high resolution is essential for example forensic analysis, surveillance and satellite imaging. This project aims to study different approaches towards Image Super Resolution and through it, gain a better understanding of the working of Filters, Convolutional Neural Networks and Generative Adversarial Networks.**

Keywords—Image Super Resolution, CNN, GAN, SR, EDSR, ESPCN, SRCNN, FSRCNN, LapSRN, Deep Learning

I. Introduction

The setup required for high resolution imaging is expensive and it might not always be feasible give the inherent limitations of the sensor and other optics manufacturing technology. Such problems can be solved with the help of image super resolution. It costs less and the existing systems can still be utilized.

It is based on the idea of combining the low-resolution or noisy sequence of images for the generation of a high-resolution image. The purpose is to increase the pixel density in order to provide more and accurate information of the scene for further processing.

The sparse coding based method is one of the examples of external example-based Super Resolution methods. This method is a result of involving multiple steps along the pipeline. Patches are cropped from the original image and then preprocessed and normalized. Later, they are encoded by a low resolution dictionary. The sparse coefficients are fed into a high resolution dictionary. In this manner, the high resolution patches are reconstructed. The patches that are overlapping are combined by weighted average in order to produce the output image. This method pays attention to learning and optimization of the dictionaries or building of mapping functions with increased efficiency but the rest of steps are not optimized.

Image Super Resolution using Deep Convolutional Networks aims to solve this issue. The authors claim that the pipeline is equivalent to the deep convolutional neural network. The convolutional neural network in this case learns an end-to-end mapping between low and high resolution images.

SRCNN is a fully convolutional neural network. The structure is designed in a way that its simple while also providing superior efficiency over traditional and other state-of-the-art example-based methods.

After the training of SRCNN model from scratch, I have performed 3x Image Super Resolution using an improved an accelerated version of SRCNN called Fast SRCNN or FSRCNN, 4x Image Super Resolution using Enhanced Deep Residual Networks for Single Image Super-Resolution (EDSR) and Efficient Sub-Pixel Convolutional Neural Network (ESPCN) and 8x Image Super Resolution using Image Super-Resolution with Deep Laplacian Pyramid Networks or LapSRN. I have also added an iterative approach to image super resolution as a reference to the state-of-the-art techniques.

We will see the methods implemented, the datasets used, the cleanup or the preparation of the dataset as well as results in the upcoming sections.

II. Related Works

[The research papers referenced can be found in the reference section at the end of the Report]

A. Traditional Approaches

The traditional approaches to Image Super Resolution use interpolation techniques which are based on sampling theory. They exhibit various limitations in predicting realistic and detailed textures.

Next generation works aimed to learn the mapping functions from Low-Resolution Images to High-Resolution Images. These methods utilized the techniques ranging from neighbor embedding to sparse coding based techniques. They are thoroughly evaluated in Yang et al.'s work.

The internal example-based methods used the similarity of an input image with itself and generated patches from the input. After it was first proposed, it has had several improved variations which are proposed to again accelerate the process of implementation.

The external example-based methods aim to learn mapping functions from low resolution image patches to high resolution image patches from external datasets. They focus on ways of learning a compact dictionary or manifold to determine relations between low to high resolution image patches. The learned representation can then help in upscaling an image.

The work proposed by Freeman et al. presented the low and high-resolution patch in the form of pairs where the corresponding high resolution patch was used for the reconstruction of the image.

Chang et al. proposed a manifold embedding technique which was according to him, was an alternative to the NN strategy. The NN is taken to a sophisticated sparse coding form in Yang et al.'s work. Random Forest, Kernel Regression, Anchored Neighborhood Regression are some other mapping functions proposed to improve the mapping speed and accuracy. Some of the state-of-the-art methods for Image Super Resolution include sparse-coding-based-methods and it's several improvements.

Many algorithms for SR focus on single-channel or gray-scale image super resolution. To deal with color images, they are first transformed to a different color space like YUV or YCbCr and then the super resolution is applied only on the luminance channel. Some efforts have been made to perform Super resolution on all three channels and then combine the results to produce a final image.

B. CNN Based Approaches

Super Resolution methods have demonstrated excellent results by optimizing various steps involved in the process like extraction of features, non-linear mapping and image reconstruction.

The VDSR network showed an observable improvement over the SRCNN method by increasing the number of layers to twenty from three. It adopts the global residual learning model to predict the differences between the original high-resolution image and the bicubic up sampled Low resolution image instead of using the actual pixel values for training a deeper network and providing a fast convergence speed at the same time. Wang et al. uses another approach of combining the domain of sparse coding with a deep convolutional neural network and goes on to train a cascade network to up sample images progressively (SCN). Kim et al. introduces a network that has multiple recursive layers called DRCN with up to 16 recursions. The approach using DRNN trains a 52-layer network by extending approach of local residual learning in ResNet with deep recursion. The above methods however use bicubic interpolation to process the input low resolution images before actually feeding them into the deep CNNs, which gives a rise to the computational costs and also needs a large amount of memory.

To solve this problem and achieve decent speed in real time scenarios, the ESPCN method is introduced to extract feature maps in the low-resolution space and then replacing the bicubic up sampling operation with an efficient sub-pixel operation (convolution) which is also known as pixel shuffling.

The FSRCNN proposes a similar idea and uses a CNN in the shape of an hourglass with transposed convolutional layers for up sampling.

ESPCN and FSRCNN compensate for speed by having limited network capacities for the learning of complex mappings. In addition to this, the above-mentioned methods up sample features or images in one step and use only a single supervisory signal form the target. This design causes training difficulties for larger scales.

The LapSRN model, on the other hand, progressively up samples input images on different pyramid levels and utilize multiple losses got the prediction of sub-band residuals at each level, which leads to proper reconstruction, even for larger scales.

Nearly all CNN-based methods for Image Super Resolution works with the L2 loss function which more often than not lead to smooth results. This does not sit well with human perception.

C. Laplacian Pyramid Based Approaches

This technique has been used in Computer Vision for several applications including but not limited to texture synthesis, semantic segmentation, edge-aware filtering and even image bending.

Denton et al. used this technique in conjunction with Generative Adversarial Network based on a Laplacian Pyramid framework called LapGAN.

It is a similar approach to the LapSRN model I have used but differs in its objectives. The LapGAN model is to build synthetically diverse natural images from random noise as an input and the sample inputs while the LapSRN aims to perform image super resolution on the provided input image. The architecture designs also differ.

D. GAN Based Approaches

GANs are relatively new but their applications in Image Super Resolution have given rise to certain state of the art models like SR3 approach for image refinement. They have application in several image reconstruction and synthesizing problems including but nor limited to face completion, 3D face generation, face super resolution, image inpainting, etc.

It was first introduced by Ledig et al. for learning the process of image super resolution in a natural manner. The ResNet is used as a generative network and training is done using the combination of perceptual, L2 and adversarial loss. The output images produced in this manner have lower PSNR but are more natural and plausible.

III. METHODS

A. SRCNN

1) The Process

Considering, a single low-resolution image, the image is upscaled to the required size using bicubic interpolation. The goal is to recover an image from this interpolated image that is as close to the ground truth as possible.

Let's say that the ground truth image is A and the bicubic interpolated image is B. We want to learn a mapping from B to A which consists of the following operations:

- The first operation is to extract patches from the lower resolution image B and convert that to a high dimensional vector. This particular vector now consists of feature maps.
- The second operation is non-linear mapping from each high dimensional vector to another high dimensional vector. The mapped vector is theoretically the representation of a patch in high resolution. The vectors combined represent a set of feature maps.

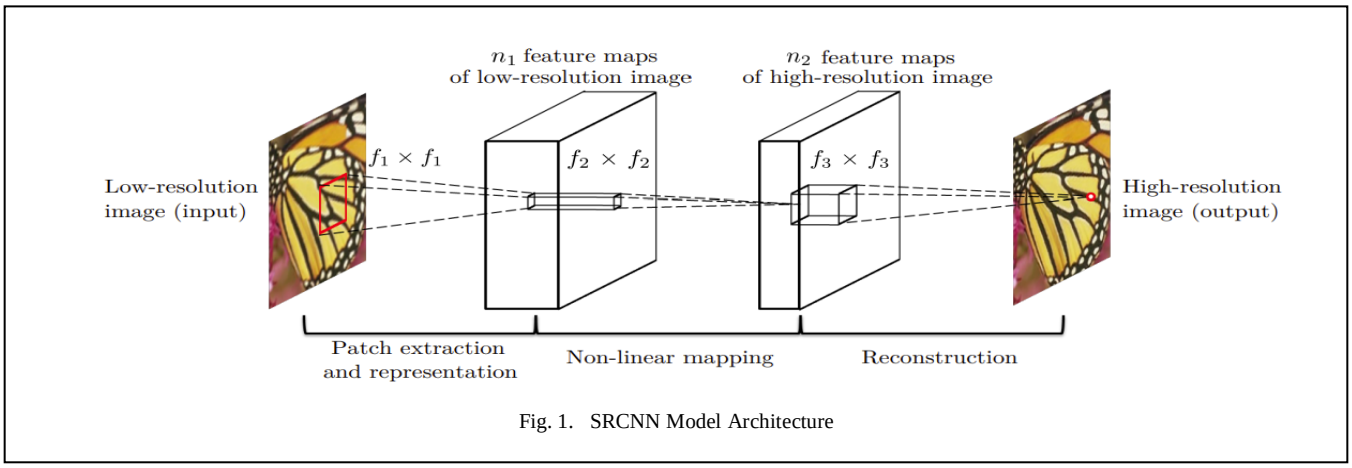

Fig. 1. SRCNN Model Architecture

- This operation is the reconstruction that collects the above high-resolution patch-wise representations to give rise to the final high resolution image. This image should be as close to the ground truth image as possible.

2) Architecture

The SRCNN is a fully convolutional network with three convolutional layers

- The first layer is a convolution layer with 64 filters of kernel size (9, 9).
- RelU activation is applied.
- The second layer is a convolution layer with 32 filters of kernel size (1, 1).
- RelU activation is applied.
- The third layer is a convolution layer with the number of filters equal to the depth of image which is 3.
- RelU activation is applied.

Note that this is a fully convolutional layer and does not have a traditional output. It outputs an image and so the number of filters should be equal to the dept or the number of channels as the output image is required in the BGR format like that of the input.

3) Training

The model has to learn an end-to-end mapping function F requiring the estimate and update of the network parameters like W1, B1, W2, B2, W3, B3. This can be attained by minimizing the computed loss between the ground truth images and the corresponding reconstructed images.

Given a data set of high-resolution images A and the reconstructed images B, I use the MSE or Mean Squared Error loss function.

Using this particular function to compute loss favors a high PSNR which is a widely used metric to evaluate the image restoration quality on a quantifiable basis.

One of the reasons why CNNs are better is that the model can adapt to various loss function metrics as long as it is relevant, and the parameters will adjust amongst themselves while training. This is something that cannot be achieved with the traditional methods

Stochastic Gradient Descent (SGD) minimizes the loss with standard backpropagation.

The filter weights and biases are initialized randomly from a Gaussian Distribution with mean equal to zero and standard deviation of 0.001 for weights and 0 for biases.

For the purpose of training, the ground-truth images are prepared according to the input of our Super Resolution Convolutional Neural Network. They are then processed fed into the network. The sub-images then fed into the network generate an output of patches which are then processed (as they are overlapping) to form a complete output image. The loss function computes the loss by comparing this output image with the ground truth image and the model starts learning. The output image is also processed for removing black areas (borders and edges) before the loss is computed.

The model learns the mapping between high-resolution image and low-resolution image in this manner.

4) Dataset

The dataset used for training the model was a subset of ukbench dataset. Considering that the paper was unclear on whether the model performed well on big dataset and the limitations of my system, I extracted a 100 images from the ukbench dataset and trained my model on it.

B. EDSR

In CNN, the performance of the model can be enhanced by stacking multiple layers or by increasing the number of filters.

Let's say that the number of layers or the depth is D, the number of feature channels or width is W, it roughly occupies a memory of O(DW) with the parameters O(DW^2). Increasing W instead of D can maximize the capacity of model when there's a limit to the computational resources.

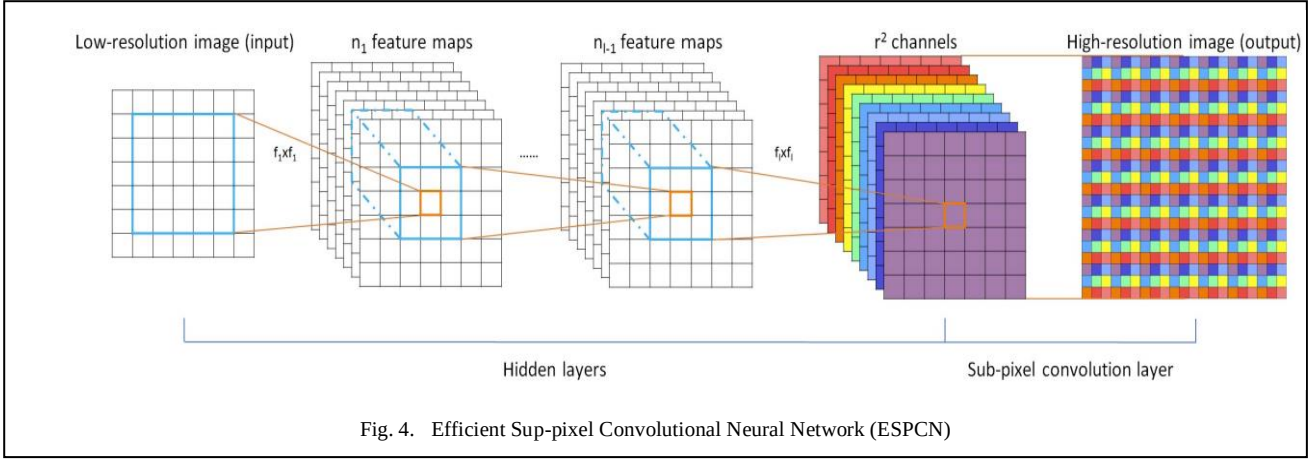

Fig. 4. Efficient Sup-pixel Convolutional Neural Network (ESPCN)

Increasing the number of feature maps above a threshold would however destabilize the training procedure. This was resolved by the authors by adopting the residual scaling with factor 0.1. Each of the residual scaling block had constant scaling layers placed after the last of the convolution layers. This innovative idea stabilizes the training procedure in a significant manner when using a large number of filters. The author proposed a network with residual blocks. The structure has similarity with the SRResNet but does not have RelU activation outside of the residual blocks. 64 feature maps are used for each and every convolution layer. In the final model, the authors expanded the model to D = 32 and W = 256 with 0.1 as the scaling factor.

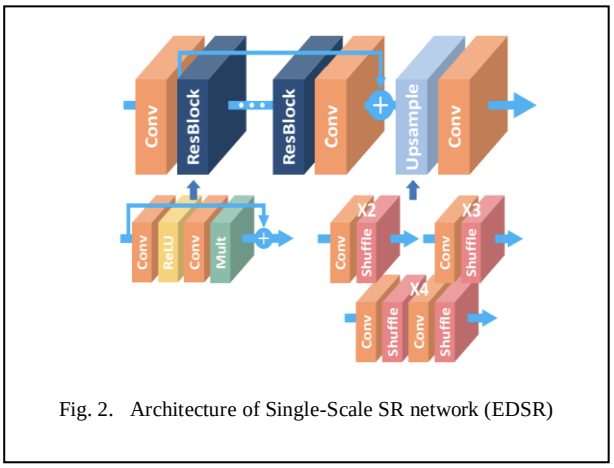

Fig. 2. Architecture of Single-Scale SR network (EDSR)

C. *ESPCN*

This upscales a low-resolution image with a fractional stride in the low-resolution space. It can naively be implemented with the help of perforation, interpolation or un-pooling from the low-resolution space to high resolution space with a stride of 1 in the high-resolution space. The computational cost is increased by these implementations since it happens in the high-resolution space.

The innovative part of this network is that the periodic shuffling can be avoided in training time. The training data can be pre-shuffled to match the output of the layer instead of implementing the periodic shuffling in training time. This proposed layer if faster by a factor of log2(r^2) to deconvolution layer in training and by a factor of r^2 when compared to implementation using different forms of upscaling before convolution.

D. *FSRCNN*

The FSRCNN is an upgraded network of SRCNN called Fast SRCNN. It differs from the original SRCNN in mainly 3 aspects.

- It adopts the original low-resolution image as input without having to perform bicubic interpolation.
- FSRCNN replaces the non-linear mapping step in SRCNN by three different steps namely the shrinking, mapping and the expanding step.
- FSRCNN uses smaller filter sizes with a deeper network structure.

This improvements of FSRCNN over SRCNN gives better performance with a lower computational cost.

E. *LapSRN*

The network architecture is based on the Laplacian Pyramid framework. The model receives input in the form of a low resolution image and progressively predicts residual images on the log2(S) pyramid levels, S being the up scaling scale factor.

The LapSRN model has two branches:

- Feature Extraction
- Image Reconstruction

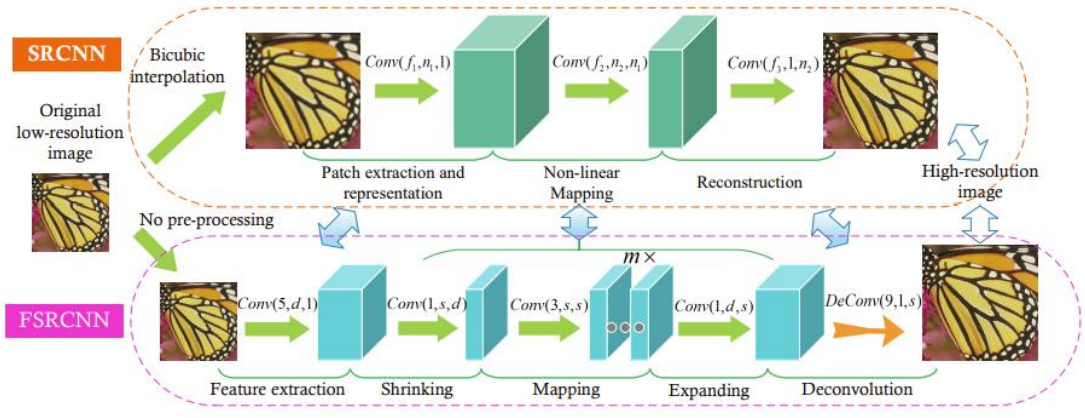

Fig. 3. SRCNN vs FSRCNN Model Architectures

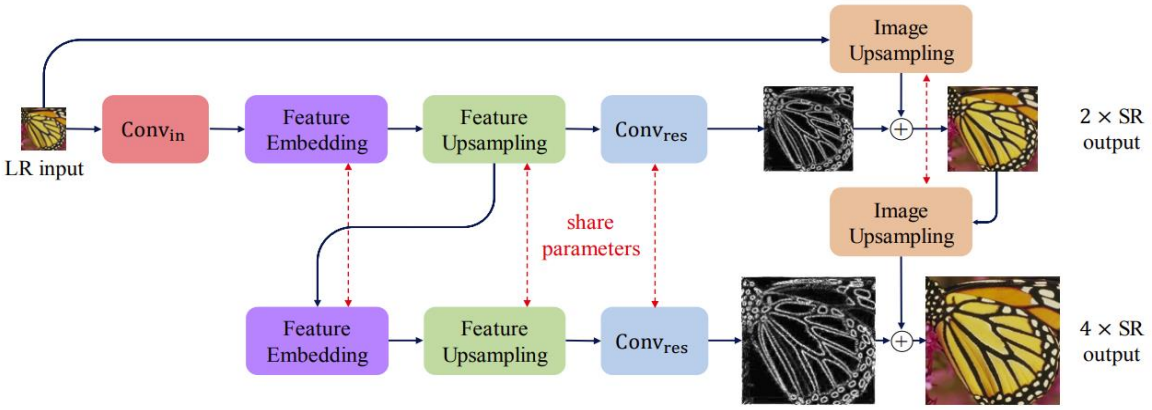

Fig. 5. LapSRN Model Architecture

IV. Experiments and Results

.Fig. 6. Training Loss on 50 epochs

Let us first look into the training loss of the SRCNN. We can observe that the curve was extremely shift at first which says that the model improved a lot in the first few epochs and then it becomes almost constant as the number of epochs progresses.

On further experimentation, I find out that the training loss nearly goes constant after 10 epochs and so the final model is trained with number epochs equal to 10.

Fig. 7a. Original Image

In order to get a good idea of how all these model networks performed along the way, I have shows results of the same image that was fed into all the networks

A. *SRCNN*

Fig. 8a. (above) Image upscaled using bicubic interpolation
Fig. 8b.(below) Image Super Resolution using SRCNN (Note the black edges at the border)

B. *EDSR*

Fig. 9 Image Super Resolution using EDSR

C. ESPCN

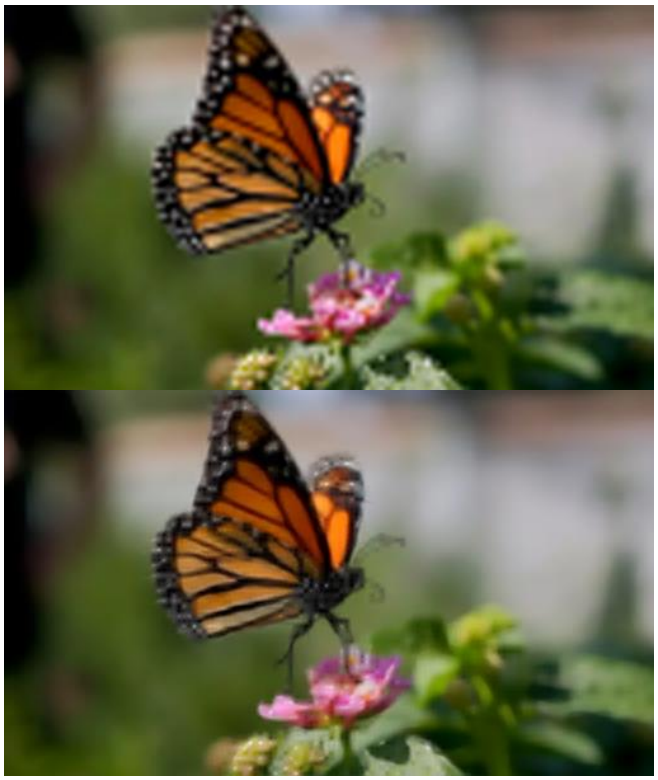

Fig. 10a. (above) Image upscaled using bicubic interpolation
Fig. 10b.(below) Image Super Resolution using ESPCN

D. FSRCNN

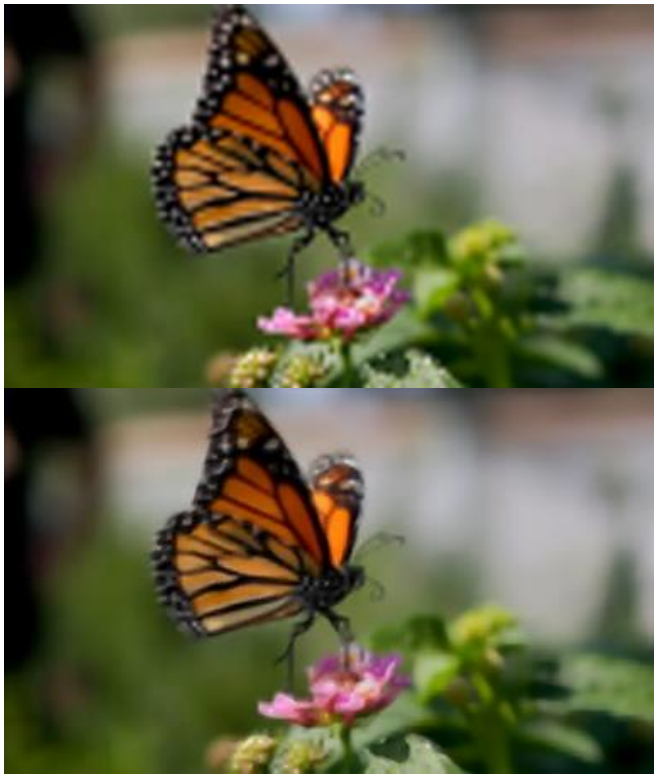

Fig. 11a. (above) Image upscaled using bicubic interpolation
Fig. 11b.(below) Image Super Resolution using FSRCNN

E. LapSRN

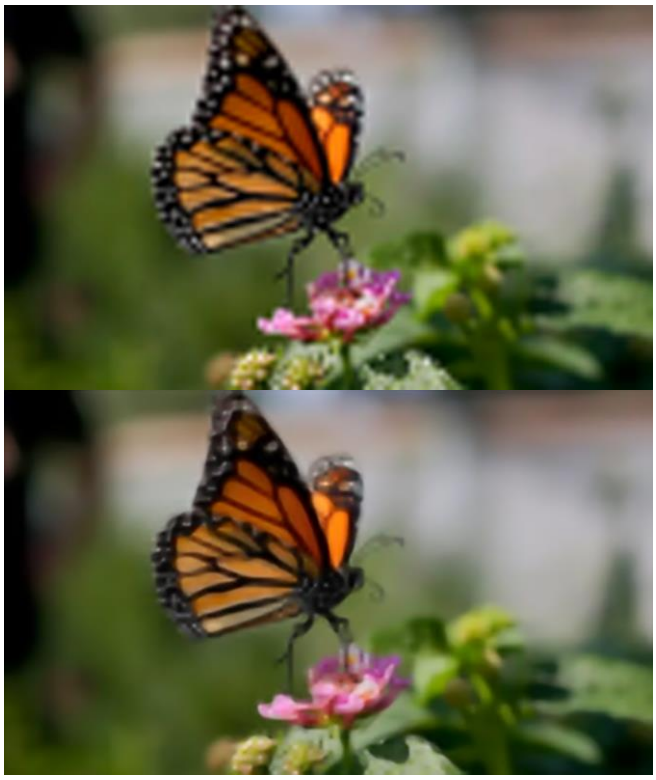

Fig. 12a. (above) Image upscaled using bicubic interpolation
Fig. 12b.(below) Image Super Resolution using LapSRN

V. Discussion and Summary

The images represented in the above sections can be compared qualitatively but it does not seem to be that intuitive on a visual basis.
For the SRCNN model, it can be said that there is a clear observable difference between the image that is bicubic interpolated and the image that is fed into the model. The black edges are not being removed that accurately which is because of the padding but there are a lot more black edges that have been completely removed from between the image using post processing techniques.

Looking at the other results, it can be said that the output is effectively better upscaled than the bicubic interpolated image and that the following models are able to upscale it as per the following level:

- SRCNN: 2x
- FSRCNN: 3x
- EDSR: 4x
- ESPCN: 4x
- LapSRN: 8x

LapSRN is found to be the most effective given the current comparison.

VI. Future Works

GAN has improved a lot in the recent times and the SR3 approach for iterative image refinement is truly intriguing. I am currently in the process of understanding GANs and how they work but I wasn't able to completely reimplement it.

I would like to implement the SR3 approach of iterative image refinement which the author claims to be better than all the above techniques.

It is interesting as the upscaled image that is generated is generated completely from random Gaussian Noise Signals.

ACKNOWLEDGMENT

I would like to acknowledge Professor Bruce Maxwell for this project would most certainly not be possible without his excellent teaching style and lucid explanation of concepts that I have learnt throughout the course.
I would also like to acknowledge the TAs for their constant support throughout the semester.
I also want to acknowledge Adrian Rosebrock's blogs that provided me with helpful conceptual and supplemental information.

REFERENCES

The first 5 references are the references to the papers proposing the architectures that I have reimplemented. The remaining paper references were referenced in the LapSRN paper whose related literature I followed as it is the latest model architecture and described all the relevant work done in this field before. So, the remaining references while not completely utilized were worthy of mention as they allowed my to go through the literature review.

[1] Lim, Bee & Son, Sanghyun & Kim, Heewon & Nah, Seungjun & Lee, Kyoung Mu. (2017). Enhanced Deep Residual Networks for Single Image Super-Resolution. 1132-1140. 10.1109/CVPRW.2017.151.

[2] Shi, Wenzhe & Caballero, Jose & Huszár, Ferenc & Totz, Johannes & Aitken, Andrew & Bishop, Rob & Rueckert, Daniel & Wang, Zehan. (2016). Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network. 10.1109/CVPR.2016.207.

[3] Dong, Chao & Loy, Chen Change & Tang, Xiaoou. (2016). Accelerating the Super-Resolution Convolutional Neural Network. 9906. 391-407. 10.1007/978-3-319-46475-6_25.

[4] Lai, Wei-Sheng & Huang, Jia-Bin & Ahuja, Narendra & Yang, Ming-Hsuan. (2017). Fast and Accurate Image Super-Resolution with Deep Laplacian Pyramid Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence. PP. 10.1109/TPAMI.2018.2865304.

[5] Kim, Jiwon & Lee, Jung & Lee, Kyoung Mu. (2015). Accurate Image Super-Resolution Using Very Deep Convolutional Networks.

[6] J. Yang, J. Wright, T. Huang, and Y. Ma, "Image super-resolution as sparse representation of raw image patches," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[7] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 2861–2873, 2010.

[8] R. Timofte, V. De Smet, and L. Van Gool, "A+: Adjusted anchored neighborhood regression for fast super-resolution," in *Asia Conference on Computer Vision*, 2014.

[9] C.-Y. Yang and M.-H. Yang, "Fast direct super-resolution by simple functions," in *IEEE International Conference on Computer Vision*, 2013.

[10] S. Schulter, C. Leistner, and H. Bischof, "Fast and accurate image upscaling with super-resolution forests," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[12] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[13] P. Fischer, A. Dosovitskiy, E. Ilg, P. Hausser, C. Hazırbas¸, V. Golkov,¨ P. van der Smagt, D. Cremers, and T. Brox, "FlowNet: Learning optical flow with convolutional networks," in *IEEE International Conference on Computer Vision*, 2015.

[14] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 2, pp. 295–307, 2015.

[15] Z. Wang, D. Liu, J. Yang, W. Han, and T. Huang, "Deep networks for image super-resolution with sparse prior," in *IEEE International Conference on Computer Vision*, 2015.

[16] J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[17] J. Kim, J. K. Lee, and K. M. Lee, "Deeply-recursive convolutional network for image super-resolution," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[18] Y. Tai, J. Yang, and X. Liu, "Image super-resolution via deep recursive residual network," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[19] W. Shi, J. Caballero, F. Huszar, J. Totz, A. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network." in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[20] C. Dong, C. C. Loy, and X. Tang, "Accelerating the superresolution convolutional neural network," in *European Conference on Computer Vision*, 2016.

[21] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Deep laplacian pyramid networks for fast and accurate super-resolution," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[22] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017.

[23] G. Freedman and R. Fattal, "Image and video upscaling from local self-examples," *ACM Transactions on Graphics (Proceedings of SIGGRAPH)*, vol. 30, no. 2, p. 12, 2011.

[24] C.-Y. Yang, J.-B. Huang, and M.-H. Yang, "Exploiting selfsimilarities for single frame super-resolution," in *Asia Conference on Computer Vision*, 2010.

[25] D. Glasner, S. Bagon, and M. Irani, "Super-resolution from a single image," in *IEEE International Conference on Computer Vision*, 2009.

[26] A. Singh and N. Ahuja, "Super-resolution using sub-band selfsimilarity," in *Asia Conference on Computer Vision*, 2014.

[27] J.-B. Huang, A. Singh, and N. Ahuja, "Single image superresolution from transformed self-exemplars," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[28] W. T. Freeman, T. R. Jones, and E. C. Pasztor, "Example-based super-resolution," *IEEE Computer Graphics and Applications*, vol. 22, no. 2, pp. 56–65, 2002.

[29] M. Bevilacqua, A. Roumy, C. Guillemot, and M. L. Alberi-Morel, "Low-complexity single-image super-resolution based on nonnegative neighbor embedding," in *British Machine Vision Conference*, 2012.

[30] H. Chang, D.-Y. Yeung, and Y. Xiong, "Super-resolution through neighbor embedding," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2004.

[31] K. I. Kim and Y. Kwon, "Single-image super-resolution using sparse regression and natural image prior," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 6, pp. 1127– 1133, 2010.

[32] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *International Conference on Curves and Surfaces*, 2010.

[33] R. Timofte, V. Smet, and L. Gool, "Anchored neighborhood regression for fast example-based super-resolution," in *IEEE International Conference on Computer Vision*, 2013.

[34] R. Timofte, E. Agustsson, L. Van Gool, M.-H. Yang, L. Zhang, B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "NTIRE 2017 challenge on single image super-resolution: Methods and results,"
in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017.

[35] P. J. Burt and E. H. Adelson, "The Laplacian pyramid as a compact image code," *IEEE Transactions on Communications*, vol. 31, no. 4, pp. 532–540, 1983.

[36] D. J. Heeger and J. R. Bergen, "Pyramid-based texture analysis/synthesis," in *22nd Annual Conference on Computer Graphics and Interactive Techniques*, 1995.

[37] S. Paris, S. W. Hasinoff, and J. Kautz, "Local laplacian filters: Edge-aware image processing with a laplacian pyramid." *ACM Transactions on Graphics (Proceedings of SIGGRAPH)*, vol. 30, no. 4, p. 68, 2011.

[38] G. Ghiasi and C. C. Fowlkes, "Laplacian pyramid reconstruction and refinement for semantic segmentation," in *European Conference on Computer Vision*, 2016.

[39] E. L. Denton, S. Chintala, and R. Fergus, "Deep generative image models using a laplacian pyramid of adversarial networks," in *Neural Information Processing Systems*, 2015.