COL783 Assignment-3 Report

Shreyansh Choudhary - 2018MT60794

April 17, 2022

Problem Statement

Super Resolution is the process of recovering a High Resolution (HR) image from a given Low Resolution (LR) image. An image may have a "lower resolution" due to a smaller spatial resolution (i.e. size) or due to a result of degradation (such as blurring). The basic idea behind SR is to combine the non-redundant information contained in multiple low-resolution frames to generate a high-resolution image. Single image super-resolution is heavily ill-posed since multiple HR patches could correspond to the same LR image patch.

We need to implement the following methods for Super Resolution in this assignment:

- Nearest Neighbors Interpolation
- BiCubic Interpolation
- Super Resolution using Single Image by Glasner et al.[1] Method

For further comparison and analysis, we need to implement the following extensions over the technique [1].

1. Use different Patch Similarity Measures to compare the Constructed Image
   - Gaussian Weighted SSD
   - Dot Product
   - Cosine Similarity
2. Enhance the predictions by Applying the method to flipped and rotated input images, and averaging the inverse of these transformations on the corresponding outputs. This often enhances the predictions on the input image.

Method

Related Work

Super Resolution methods can be broadly classified into 2 categories:

- The classical multi-image super resolution [2, 3, 4] - a set of linear constraints are imposed on intensity values of the high resolution image based on a number of low-resolution images of the same scene. Theoretically, the high resolution image can be recovered by solving the linear constraints given enough LR images are provided. However, Practically, this approach is numerically limited to small increases in resolution,

- Example based Super Resolution [5, 6, 7]- In example-based SR, correspondences between low and high resolution image patches are learned from a database of low and high resolution image pairs (usually with a relative scale factor of 2), and then applied to a new low-resolution image to recover its most likely high-resolution version. But this technique, unlike Classical methods, is not guaranteed to provide the true high resolution details.

Sophisticated methods for image up-scaling based on learning edge models have also been proposed [8, 9]. The main idea in these is to magnify the images in such a way that the sharpness of the edges is still maintained. While, The super resolution methods focus on finding the missing high-resolution details which are not explicitly found in any individual Low Resolution image.

An SR approach based on Patch repetitions was proposed by [10] for obtaining higher-resolution video frames, by applying the classical SR constraints to similar patches across consecutive video frames and within a small local spatial neighborhood. Their algorithm relied on having multiple image frames, and did not exploit the power of patch redundancy across different image scales.

In Classical SR framework, the high frequency information is assumed to be split across multiple low resolution images, whereas in Example based SR, this missing high res-information is assumed to be within the database patches, and is learned from the low-res/high-res pairs of examples.

Nearest Neighbor Interpolation - This is the fastest and simplest algorithm. This technique replaces every pixel at an unknown position with the nearest known pixel in the output.

Bicubic Interpolation : This considers 4x4 neighborhood of known pixels. The pixels which are closer to the one that's to be estimated are given higher weights as compared to those further away. A cubic spline is fit to interpolate the values. This has higher time complexity than Nearest Neighbor Interpolation. To fasten up the interpolation, we use the Interpolation kernel as defined by [11].

Super Resolution from Single Image

Looking at the limitations of both these family of methods, Glasner et al [1] proposed a single unified approach which combines the classical SR constraints with the example-based constraints, while exploiting (for each pixel) patch redundancies across all image scales and at varying scale gaps, thus obtaining adaptive SR with as little as a single low- resolution image. The approach is based on an observation that patches in the same image tend to recur many times inside the image redundantly, both within same scale as well as across different scales.

Patch Redundancy in a Single Image Natural images tend to contain repetitive visual content. Small patches tend to redundantly recur many times inside the image, within same scale as well as across different scales. We say that an input patch recurs on a different scale if it appears exactly same in a scaled down version of the image. For each low-res patch, we can find its high-res parent from the input image. These patches make "low-res/high-res" pairs.

The high-res parent of a found low-res patch provides an indication to what the (unknown) high-res parent of the source patch might look like. This forms the basis for Example-Based SR, even without an external database (enough recurring patches should be present in the image). Generally, there are too much recurring patches, even when we cannot perceive them. This is because these small patches can contain edges, corners etc, which are abundantly present in natural images.

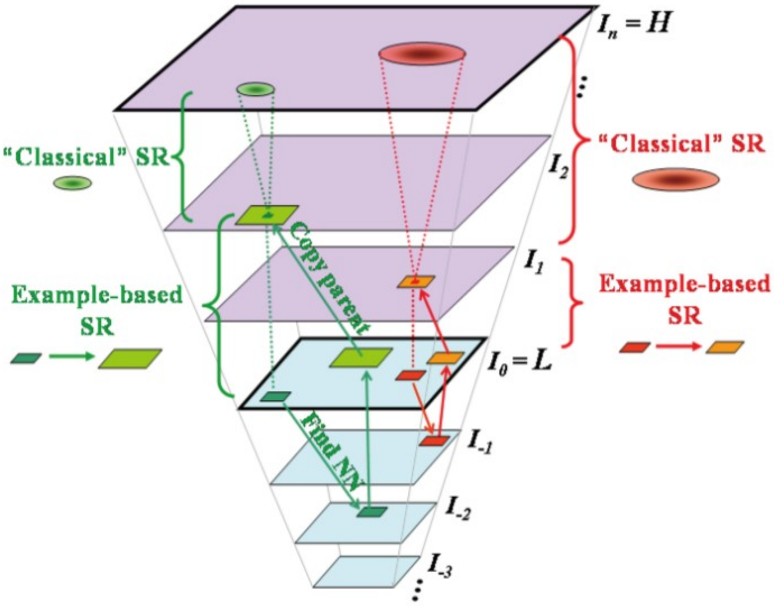


Figure 1: Image Pyramid, Unified approach

Unified Framework

Glasner et al. combines the classical Super resolution techniques and the Example based resolution techniques in the following way:

in-Scale Patch Redundancy The ideas here are build up using the framework Multi Image Super Resolution. We have a single Low Resolution Image, L which is degraded and sub-sampled version of an High Resolution Image, H with blur B i.e. $L = (H * B) \downarrow_s => L(q) = (H * B)(q) = \sum_{q_i \in Support(B)} H(q_i)B(q_i - q)$. As, we only have a single image, this system of linear equations is undetermined(the number of constraints is lesser than the number of variables).

But due to in-scale patch redundancy, we can create some more constraints. Let $p \in L$ be some pixel, and $\mathcal{P}$ be the surrounding patch, then there exist multiple similar patches $\mathcal{P}_1, \mathcal{P}_2, ...\mathcal{P}_k$ in L. These patches can be thought of as taken from k different low-resolution images of the same high resolution “scene”, thus inducing k times more linear constraints on the high-resolution intensities of pixels within the neighborhood of $q \in H$.

For increased numerical stability, each equation induced by a patch Pi is globally scaled by the degree of similarity of $\mathcal{P}_i$ to its source patch P. Thus, patches of higher similarity to P will have a stronger influence on the recovered high-resolution pixel values than patches of lower similarity.

Cross-Scale Patch Redundancy In Example based SR, correspondences between low and high resolution image patches are learned from a database of low and high resolution image pairs, and then applied to unseen/new low resolution image to recover its most likely high resolution version.

Let B be the blur kernel relating to low-res input Image L with the unknown high-res image H i.e $L = (H * B) \downarrow_s$. Let $I_0, I_1, ...I_n$ be the set of unknown images of increasing resolutions ranging from low res L to target high res H with the corresponding blur functions be $B_0, B_1, ...B_n$, such that $L = (I_l * B_l) \downarrow_{s_l}$, s_l denotes the relative scaling factor. When B is not known, we can approximate it with a gaussian in which case $B_l = B(s_l)$ are simply a cascade of gaussians whose variances are determined by s_l. Moreover, if s_l is power of some constant i.e. $s_l = \alpha^l$, we also have the following constraint on the high resolution image i.e. $I_l = (H * B_{n-l}) \downarrow_{s_{n-l}}$. Now consider, $I_0, I_{-1}, I_{-2}...I_{-m}$ denote a cascade of images of decreasing resolutions(scales), obtained from L using same Blur function i.e. $I_{-l} = (L * B_l) \downarrow_{s_l}$. Let $\mathcal{P}_l(p)$ be the surrounding patch in the image I_l for pixel.For a $p \in L$, we search for similar patches of in the low resolution cascade of images. Let $\mathcal{P}_{-l}(\bar{p})$ be such a matching patch in I_{-l}. Then its high-res parent $\mathcal{Q}_0(s.\bar{p})$ can be extracted from L. This provides a low-high res pair, which provides a prior on the appearance of the high res parent of the low res input patch, namely $\mathcal{Q}_l(s, p)$ in the high res unknown image I_l .

Unifying the two Using the Example-based SR, we can yield a large number of high-res patches Q_l, at each of high resolution levels I_l between L and H. Each such Q_l induces linear constraints on the unknown H. These constraints are like the linear constraints, we obtained in classical SR method.

Implementation

In my implementation, as directed by the authors of the papers, I have chosen the $s_l = \alpha^l$ and $\alpha = 2^{(1/3)}$, resulting in pyramid of height 7. I use 5x5 as the patch size. [12] Also, to find K-most similar patches, instead of using bulky KNN algorithm, I have used FAISS [13], by Facebook to find the K most similar patches. Note that FAISS is only developed for metrics - L2 distance and Inner product.

To use it for our case, I modified the patch p, element wise multiplying them with the square root of the Gaussian Kernel. To find the distance between 2 patches, say, p and q, define : p' and q' s.t. $p'(i, j) = \sqrt{G(i, j)}p(i, j)$ and $q'(i, j) = \sqrt{G(i, j)}q(i, j)$

Now the Gaussian weighted distance between p and q is same as L2 distance between p' and q', also the transformation from p to p' is one-one, we can use it to rank the patches based on Gaussian distance using FAISS.

Similar Transformation is applied to rank the patches according to the cosine similarity to use FAISS Indexing by inner product. $p' = p/||p||$ and $q' = q/||q||$

Results

We hereby present the results and their observations of the above mentioned techniques which were asked to implement.

- Nearest Neighbors Interpolation Fig.2,3,4- This is a decent upscaling technique, but as we can observe in the output images, When upscaling an image, multiple pixels of the same color will be duplicated throughout the image. This leaves squary patches in the output.

- BiCubic Interpolation Fig.5,6,7- The technique works fine for the Upscaling images, The end result of this image is quite blurry and is not as clear as the input image given.

- Super Resolution From Single Image Fig.8,9,10 - The technique is able to generalise the high resolution image from the low resolution image well, the end result for this is better than bicubic resolution, the edges and corners are well preserved.

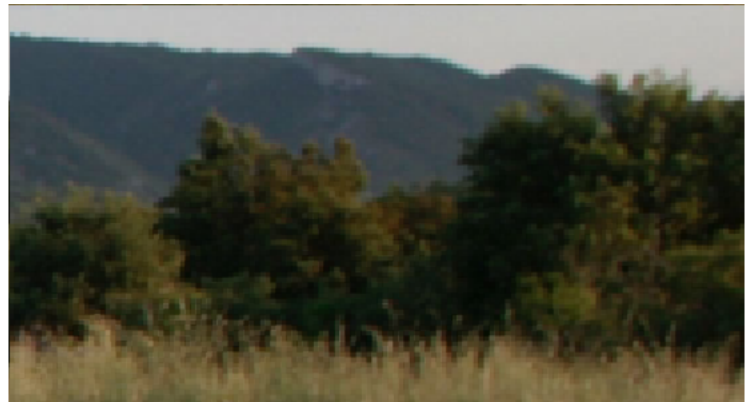
When we use different patch similarity measures to find K most similar patches, we observe that the Gaussian Weighted distances perform best.

The dot product Fig.11,12,13 as a similarity measure can vary a lot, which means when we are scaling the Patch constraint globally by the similarity, the image intensity value varies a lot leading to very high values to the image, causing the most of the values to be 1(or 255 in integer case).

In case of cosine similarity Fig.14,15,16, some patches are not properly rendered, this may be due to very small values cosine similarity takes, that it is not able to scale the weights at interpolated points that most of them are very small.

Enhanced Prediction Fig.17,18,19 - The geometric transformations included were - horizontal flip, vertical flip, both H and V flip, rotation by 90, 180, 270. After applying each of these transformations, we find their high resolution images, inverse of transformations and take the average. The predictions in each of these should be same at pixel-level. Assuming the noise is additive, averaging can cancel the noise and provide better predictions. This is well apparent in our results.

(2a)



(2b)

Figure 2: Nearest Neighbor Interpolation, from 276 x 149 to 552 x 296

(3a)

(3b)

Figure 3: Nearest Neighbor Interpolation, from 305 x 193 to 610 x 386

(4a)

(4b)

Figure 4: Nearest Neighbor Interpolation, from 242 x 131 to 487 x 261

(5a)

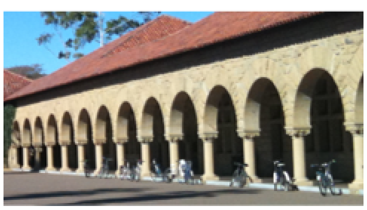
(5b)

Figure 5: Bicubic Interpolation, from 276 x 149 to 552 x 296

(6a)

(6b)

Figure 6: Bicubic Interpolation, from 305 x 193 to 610 x 386



(7a)

(7b)

Figure 7: Bicubic Interpolation, from 242 x 131 to 487 x 261

(8a)

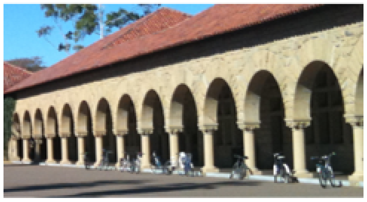
(8b)

Figure 8: SR method (Gaussian weighted L2), from 276 x 149 to 552 x 296

(9a)

(9b)

Figure 9: SR method (Gaussian Weighted L2), from 305 x 193 to 610 x 386



(10a)

(10b)

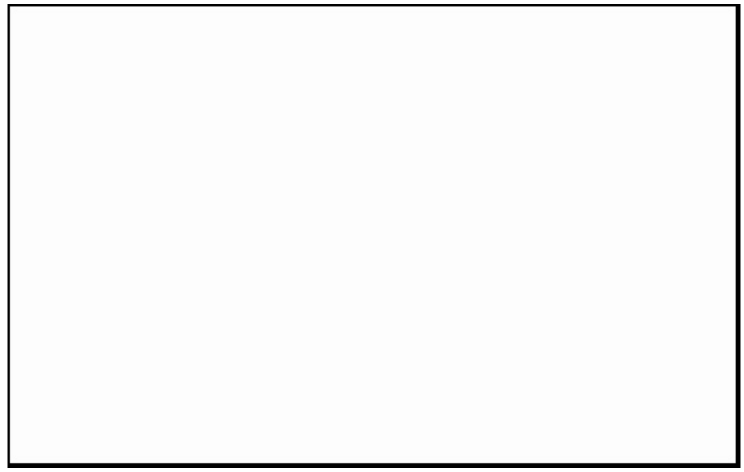
Figure 10: SR method (Gaussian Weighted L2), from 242 x 131 to 487 x 261

(11a)

(11b)

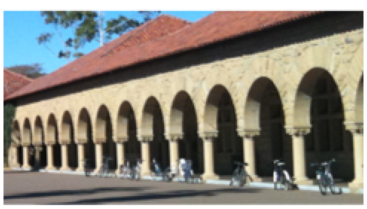
Figure 11: SR method (Dot Product), from 276 x 149 to 552 x 296

(12a)



(12b)

Figure 12: SR method (Dot product), from 305 x 193 to 610 x 386



(13a)

(13b)

Figure 13: SR method (Dot product), from 242 x 131 to 487 x 261

(14a)

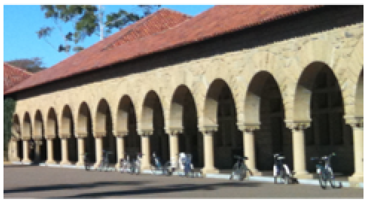
(14b)

Figure 14: SR method (Cosine Similarity), from 276 x 149 to 552 x 296

(15a)

(15b)

Figure 15: SR method (Cosine Similarity), from 305 x 193 to 610 x 386

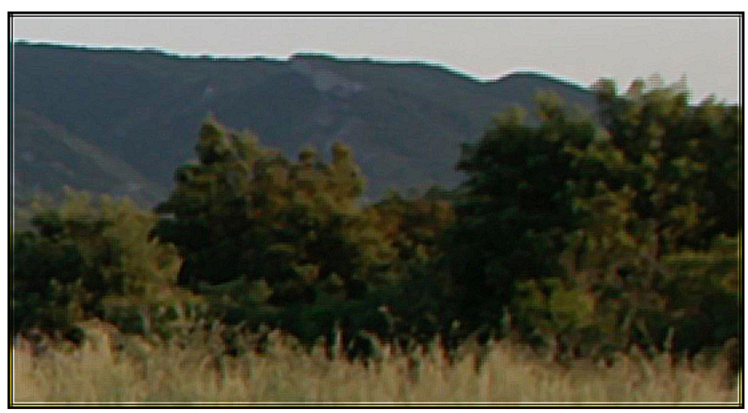


(16a)

(16b)

Figure 16: SR method (Cosine Similarity), from 242 x 131 to 487 x 261

(17a)



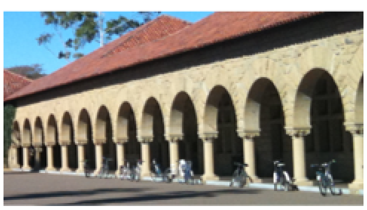
(17b)

Figure 17: Enhanced Predictions, from 276 x 149 to 552 x 296

(18a)

(18b)

Figure 18: Enhanced Predictions, from 305 x 193 to 610 x 386



(19a)

(19b)

Figure 19: Enhanced Predictions, from 242 x 131 to 487 x 261

References

[1] D. Glasner, S. Bagon, and M. Irani, "Super-resolution from a single image," in *2009 IEEE 12th International Conference on Computer Vision*, pp. 349–356, 2009.

[2] M. Irani and S. Peleg, "Improving resolution by image registration," *CVGIP: Graphical Models and Image Processing*, vol. 53, no. 3, pp. 231–239, 1991.

[3] S. Farsiu, M. Robinson, M. Elad, and P. Milanfar, "Fast and robust multiframe super resolution," *IEEE Transactions on Image Processing*, vol. 13, no. 10, pp. 1327–1344, 2004.

[4] C. D, "Image mosaicing and super-resolution."

[5] S. Baker and T. Kanade, "Hallucinating faces," in *Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580)*, pp. 83–88, 2000.

[6] W. T. Freeman, T. R. Jones, and E. C. Pasztor, "Example-based super-resolution," *IEEE Computer Graphics and Applications*, vol. 22, pp. 56–65, 2002.

[7] W. Freeman and E. Pasztor, "Learning low-level vision," *Proceedings of the Seventh IEEE International Conference on Computer Vision*, 1999.

[8] R. Fattal, "Image upsampling via imposed edge statistics," *ACM SIGGRAPH 2007 papers*, 2007.

[9] J. Sun, Z. Xu, and H. Shum, "Image super-resolution using gradient profile prior," *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2008.

[10] M. Protter, M. Elad, H. Takeda, and P. Milanfar, "Generalizing the nonlocal-means to super-resolution reconstruction," *IEEE Transactions on Image Processing*, vol. 18, no. 1, pp. 36–51, 2009.

[11] R. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 29, no. 6, pp. 1153–1160, 1981.

[12] D. Barash, "Sr by example," *GitHub repository*, 2013.

[13] J. Johnson, M. Douze, and H. Jégou, "Billion-scale similarity search with GPUs," *IEEE Transactions on Big Data*, vol. 7, no. 3, pp. 535–547, 2019.