Super-Resolution Via Learning Sparse Representation

Haoze Wu haoze@jhu.edu
James Yu jyu132@jhu.edu
Thomas Yu tyu34@jhu.edu

Abstract

Super-resolution is a critical aspect of image processing, as it involves enhancing the resolution of imaging systems, which has significant implications in various fields such as medical imaging, surveillance, and satellite imagery analysis. The importance of researching this problem lies in its potential to improve the quality and clarity of images, enabling more accurate analysis and decision-making. With the advancement of technology, especially in the realm of neural networks and deep learning, there is a growing interest in exploring novel methods to tackle super-resolution challenges.

1. Introduction

Super-resolution imaging (SR) is a class of techniques that enhance or increase the resolution of an imaging system. Many ideas and algorithms are used to solve the problem, including compressed sensing, wavelet analysis, and neural networks especially in recent years. Classic algorithms like compressed sensing and wavelet analysis have been widely employed to enhance image resolution. Compressed sensing techniques leverage sparse signal representation to recover high-resolution details from lower-resolution data. Wavelet analysis, on the other hand, decomposes images into different frequency components, enabling the reconstruction of finer details. In recent years, the advent of data-driven algorithms, particularly those based on neural networks, has revolutionized super-resolution. Models like SRResNet[5], Swin2SR, and SRGAN utilize deep learning architectures to learn complex mappings from low-resolution to high-resolution images. These neural network-based approaches have demonstrated remarkable results, achieving substantial improvements in image quality and clarity across various applications, such as medical imaging, surveillance, and satellite imagery analysis. The fusion of traditional algorithms and cutting-edge neural network techniques continue to drive advancements in the field of super-resolution.

We plan to explore novel methods to solve the super-resolution problem and contribute to this field by initially exploring learning and classical model-driven algorithms to gain a strong foundation. Subsequently, we will delve into recent research to identify innovative approaches and potentially propose new methods to further advance super-resolution. We will provide a comprehensive survey of existing solutions, which will still be valuable in understanding the current state of the art in super-resolution imaging.

To evaluate the performance, we will use the PSNR (peak signal-to-noise ratio) and RMSE (root-mean-square error) to measure the quality of the super-resolved image by comparing it to the original high-resolution image. We will also use the SSIM (structural similarity index measure) to evaluate the structural similarity between the super-resolved image and the original high-resolution image. In addition, we will analyze the qualitative results of sample images, to provide insights of miscellaneous solutions.

2. Methods

Write down in point form some sentences on the data, methods, algorithms, software you are using for the project.

- Data: Set5, Set14
- Algorithms: Fast Image Super-resolution Based on In-place Example Regression[7]
- Software: OpenCV, Numpy, PyTorch

We manage to run the LapSRN model[4] (as baseline) and OpenCV resize interpolation APIs (including linear and bicubic schemes).

We use the method in a CVPR'08 paper "image super-resolution as sparse representation of raw image patches" [8], by Yang et al.

The goal of the paper is to solve the single-image super-resolution task: given a low-resolution image $\boldsymbol{Y}$, recover a higher-resolution image $\boldsymbol{X}$ of the same scene. And formally:

$$\boldsymbol{Y} = DH\boldsymbol{X} \tag{1}$$

where H represents a blurring filter, and D is a downsampling operator.

2.1. Background

$\boldsymbol{Y}$ and $\boldsymbol{X}$ can be divided into many corresponding image patches $\boldsymbol{y}$ and $\boldsymbol{x}$, such that every $\boldsymbol{y}$ should be recovered to $\boldsymbol{x}$. We denote that $\boldsymbol{x} \in \mathbb{R}^n$ is a high-resolution image patch, and $\boldsymbol{y} \in \mathbb{R}^k$ ($k < n$) is a low-resolution image patch, and

$$\boldsymbol{y} = L\boldsymbol{x} \tag{2}$$

The core intuition of the paper is to treat the high resolution image (patch) as a complete signal, while the low resolution image is the signal with noise and downsampling. Then the super-resolution problem is essentially a signal recovery problem.

The Nyquist-Shannon sampling theorem [6] shows that to reconstruct (in time domain) a signal with the highest frequency f in frequency domain, the minimal sampling rate is $\frac{1}{2f}$. The intuition of the theorem is that to determine a signal (in time domain)

$$y(t) = A\sin(\omega t + \varphi) \tag{3}$$

we need to determine three variables: A, ω, and φ. In the borderline case, they are exactly determined by 3 sampling points, so that the equation (3) can be solved. Hence, two sampling cycles (borderline case of getting 3 sampling points), in the worst case, are needed in the sampling for the highest frequency.

However, many signals are essentially "sparse", meaning that the signal only has a small number of components with different frequencies. Around 2004, researchers, including Emmanuel Candès and Terence Tao, found that given knowledge about a signal's sparsity, the signal can be reconstructed with much fewer samples than the Nyquist-Shannon sampling theorem requires [2]. The breakthrough is centered around a technique called *compressed sensing*.

Compressed sensing (or sparse representation) is used in K-SVD [1] and image denoising [3]. These two papers, in terms of the sparse representation, is the foundation of the paper I review.

Formally, based on (2), we can somehow design $\boldsymbol{D} \in \mathbb{R}^{n\times K}$ as an overcomplete dictionary of K prototype signal-atoms, and

$$\boldsymbol{y} = L\boldsymbol{x} = L\boldsymbol{D}\boldsymbol{\alpha}_0 \tag{4}$$

where $\boldsymbol{\alpha}_0 \in \mathbb{R}^K$ is a vector with very few ($<< K$) nonzero entries. $\boldsymbol{x} = \boldsymbol{D}\boldsymbol{\alpha}_0$ essentially indicates that $\boldsymbol{x}$ is a sparse signal.

This paper differs from related works [1, 3] in handling two coupled dictionaries instead. $\boldsymbol{D}_\hbar$ indicates the high-resolution patches, and $\boldsymbol{D}_\ell = L\boldsymbol{D}_\hbar$ indicates the low-resolution patches.

2.2. Algorithm

The algorithm consists of two steps:

- Based on (4), determine the sparse representation of local patch, i.e., determine $\boldsymbol{\alpha}_0$.
- Given the local patches, reconstruct the high-resolution image, i.e., enforcing the global reconstruction constraint.

The algorithm also requires two preparations before the computation:

- Prepare the dictionaries $\boldsymbol{D}_\hbar$ and $\boldsymbol{D}_\ell$. The theory of compressed sensing shows that randomly sampling among the training dataset would provide a good dictionary. In the evaluation and experiments, the dictionary is randomly sampled raw patches from photos, such as flower photos.
- Prepare a linear transformation F that extracts features from an image patch. The F used in the paper extracts 4 features, extracted by the derivative $\boldsymbol{f}_1 = [-1, 0, 1]$, $\boldsymbol{f}_2 = \boldsymbol{f}_1^\top$, $\boldsymbol{f}_3 = [1, 0, -2, 0, 1]$, and $\boldsymbol{f}_4 = \boldsymbol{f}_3^\top$.

The first step of the algorithm is to find the sparsest representation:

$$\min ||\boldsymbol{\alpha}||_0 \quad \text{s.t.} \quad ||F\boldsymbol{D}_\ell\boldsymbol{\alpha} - F\boldsymbol{y}||_2^2 \le \epsilon \tag{5}$$

The theory of sparse representation shows that minimizing the ℓ^1-norm is sufficient:

$$\min ||\boldsymbol{\alpha}||_1 \quad \text{s.t.} \quad ||F\boldsymbol{D}_\ell\boldsymbol{\alpha} - F\boldsymbol{y}||_2^2 \le \epsilon \tag{6}$$

With Lagrange multipliers:

$$\min \lambda||\boldsymbol{\alpha}||_1 + \frac{1}{2}||F\boldsymbol{D}_\ell\boldsymbol{\alpha} - F\boldsymbol{y}||_2^2 \tag{7}$$

Add a similar constraint for the compatibility of adjacent high-resolution patches:

$$\min \lambda||\boldsymbol{\alpha}||_1 + \frac{1}{2}\left|\left|\begin{bmatrix} F\boldsymbol{D}_\ell \\ \beta P\boldsymbol{D}_\hbar \end{bmatrix}\boldsymbol{\alpha} - \begin{bmatrix} F\boldsymbol{y} \\ \beta\boldsymbol{w} \end{bmatrix}\right|\right|_2^2 \tag{8}$$

The second step of the algorithm is enforcing the global reconstruction constraint. Note that the high-resolution image $\boldsymbol{X}_0$ stitching the patches computed in the first step of the algorithm might not satisfy the constraint (1), so we need to eliminate this discrepancy by projecting $\boldsymbol{X}_0$ onto the solution space of $DH\boldsymbol{X} = \boldsymbol{Y}$:

$$\boldsymbol{X}^* = \underset{\boldsymbol{X}}{\operatorname{argmin}} ||\boldsymbol{X} - \boldsymbol{X}_0|| \quad \text{s.t.} \quad \boldsymbol{Y} = DH\boldsymbol{X} \tag{9}$$

This optimization can be solved by the iterative back-projection method.

3. Results and Evaluation

Image	Bicubic	LapSRN	Sparse
Man	24.88	27.79	20.35
Pepper	27.28	30.78	21.52
Baboon	20.66	23.30	18.85
Face	29.37	32.38	27.49
Barbara	24.13	26.57	24.29
Foreman	26.28	30.34	27.57
Flowers	24.33	26.61	14.93
PPT3	21.36	23.92	18.88
Lenna	28.73	31.71	26.49
Bridge	23.59	25.89	21.79
Comic	20.77	23.02	21.02
Coastguard	24.69	26.65	22.14
Monarch	26.74	29.80	23.46
Zebra	23.54	26.56	20.40

Table 1. Peak Signal-to-Noise Ratio (PSNR) values for different images in Set14 using Bicubic, LapSRN, and Sparse upscaling methods. PSNR is a metric used to assess the quality of compressed or reconstructed images. Typically, PSNR scores range between 30 dB and 50 dB, with higher values indicating better image quality. The LapSRN SR images were obtained using the LapSRN model, which increases image resolution by x8 scaling.

For the super-resolution benchmark, a diverse range of datasets can be utilized to evaluate and compare the performance of super-resolution algorithms. The classical image super-resolution dataset includes the DIV2K validation set, along with popular benchmark datasets like Set5, Set14, BSD100, Urban100, and Manga109. As a survey of our results, we utilized results from the Set5 and Set14 datasets. We then compare our sparse representation model with a benchmark bicubic model and a state-of-the-art algorithm, LapSRN, using these two datasets to demonstrate the applications of our method on super-resolving real-world photos. The original HR images are compressed to a lower resolution by a factor such that the original HR images are x3 the size of the LR images. The LapSRN model upscales the LR image by x8 scaling while Bicubic and sparse upscale the image by x3 back to the size of the original HR image. The link to our SR implementation is https://github.com/functioner/cv-project-super-resolution.

Image	Bicubic	LapSRN	Sparse
Man	0.6415	0.8258	0.5838
Pepper	0.9713	0.9860	0.9276
Baboon	0.6766	0.8214	0.6665
Face	0.8882	0.9530	0.8219
Barbara	0.7879	0.8804	0.8027
Foreman	0.9248	0.9720	0.9328
Flowers	0.8514	0.9204	0.7140
PPT3	0.8635	0.9059	0.8243
Lenna	0.9732	0.9906	0.9687
Bridge	0.5378	0.7663	0.5816
Comic	0.7027	0.8319	0.7286
Coastguard	0.5936	0.8173	0.6009
Monarch	0.9641	0.9866	0.9493
Zebra	0.8771	0.9344	0.8409

Table 2. Structural Similarity Index (SSIM) values for different images in Set14 using Bicubic, LapSRN, and Sparse upscaling methods. SSIM is a metric used to measure the similarity between two images. It ranges between -1 and 1, with 1 indicating perfect similarity.

Image	Bicubic	LapSRN	Sparse
Man	14.53	10.40	24.49
Pepper	11.03	7.37	21.41
Baboon	23.64	17.44	29.11
Face	8.67	6.13	10.77
Barbara	15.84	11.97	15.56
Foreman	12.37	7.75	10.67
Flowers	15.49	11.91	45.73
PPT3	21.80	16.23	29.00
Lenna	9.33	6.62	12.08
Bridge	16.86	12.94	20.74
Comic	23.33	18.02	22.67
Coastguard	14.87	11.86	19.92
Monarch	11.74	8.25	17.12
Zebra	16.97	11.99	24.36

Table 3. Root Mean Square Error (RMSE) values for different images in Set14 using Bicubic, LapSRN, and Sparse upscaling methods. RMSE is a metric that measures the average magnitude of the differences between predicted and observed values. Lower values indicate better accuracy.

3.1. Comparisons with benchmark model and state-of-the-art model

We compared our sparse representation model with the benchmark Bicubic SR algorithm and the state-of-the-art SR algorithm, LapSRN[4]. We made our metric comparisons and carried out our experiments using the two

Figure 1. Examples of super-resolution on the flowers image from Set14 using the bicubic model (baseline), LapSRN model (state of the art), and our sparse representation model. The images include are: the original high-resolution image (top-left), the compressed low-resolution version of the image (top-right), super-resolution using the bicubic model (bottom-left), super-resolution using the LapSRN model (bottom-middle), and super-resolution using the sparse respresentation model (bottom-right). Despite the super-resolved image using sparse representation scoring much lower using the selected quantitative metrics, the image appears much clearer visually than the bicubic version. We suspect a significant change in color resulted in poor scores in PSNR, SSIM, and RMSE.

datasets, Set5 and Set14, which contain natural scenes. We evaluated the SR images using three commonly used qualitative metrics: PSNR (peak signal-to-noise ratio), SSIM (structural similarity index measure), and RMSE (root-mean-square error) of which the results are displayed in Table 1, Table 2, and Table 3 respectively. When looking at the metric results, our sparse representation model does not perform as well as the benchmark or state-of-the-art results. However, upon human observation and visual examination, the SR images using our sparse representation model consistently appear clearer and less blurry than the Bicubic SR images and occasionally is comparable to the LapSRN SR images. The blurriness factor, which are unrelated to resolution or image similarity, is not accounted for by the quantitative metrics and require qualitative analysis. We hypthesis that a significant change in color due to our sparse representation algorithm is responsible for lower performance in the quantitative metrics.

3.2. Discussion

Various factors contribute to the performance of our sparse representation model including image characteristics, training data, model complexity, and artifacts and overfitting. First, sparse representation models excel when the image contains structures or patterns that can be well represented sparsely. Images that contain textures, edges, or patterns that align with the sparsity assumption tend to perform better using the sparse representation SR model. Next, the complexity of the model and its ability to capture intricate details in images plays a significant role. Thus, sparse representation models may struggle with complex images or those containing a wide range of colors if the dictionary used for sparse coding is not diverse enough to capture the variations. Although Bicubic interpolation tends to produce smoother results, it also introduces blurriness or artifacts. On the other hand, sophisticated learning models may overfit to a specific type of data. The results of the SR images show that the Bicubic model introduced significant blurriness as compared to our sparse representation model that may not be reflected as severely in the quantitative metrics as the color change resulting from our model.

We have provided examples from two interesting image results from Set14: flowers and foreman. For flowers, our sparse representation model performed quite poorly in the quantitative metrics, even appearing to be an outlier as shown in Figure 3. This is likely due to the significant color change as previously mentioned. But, when looking at the SR images for a qualitative comparison, the observations re-

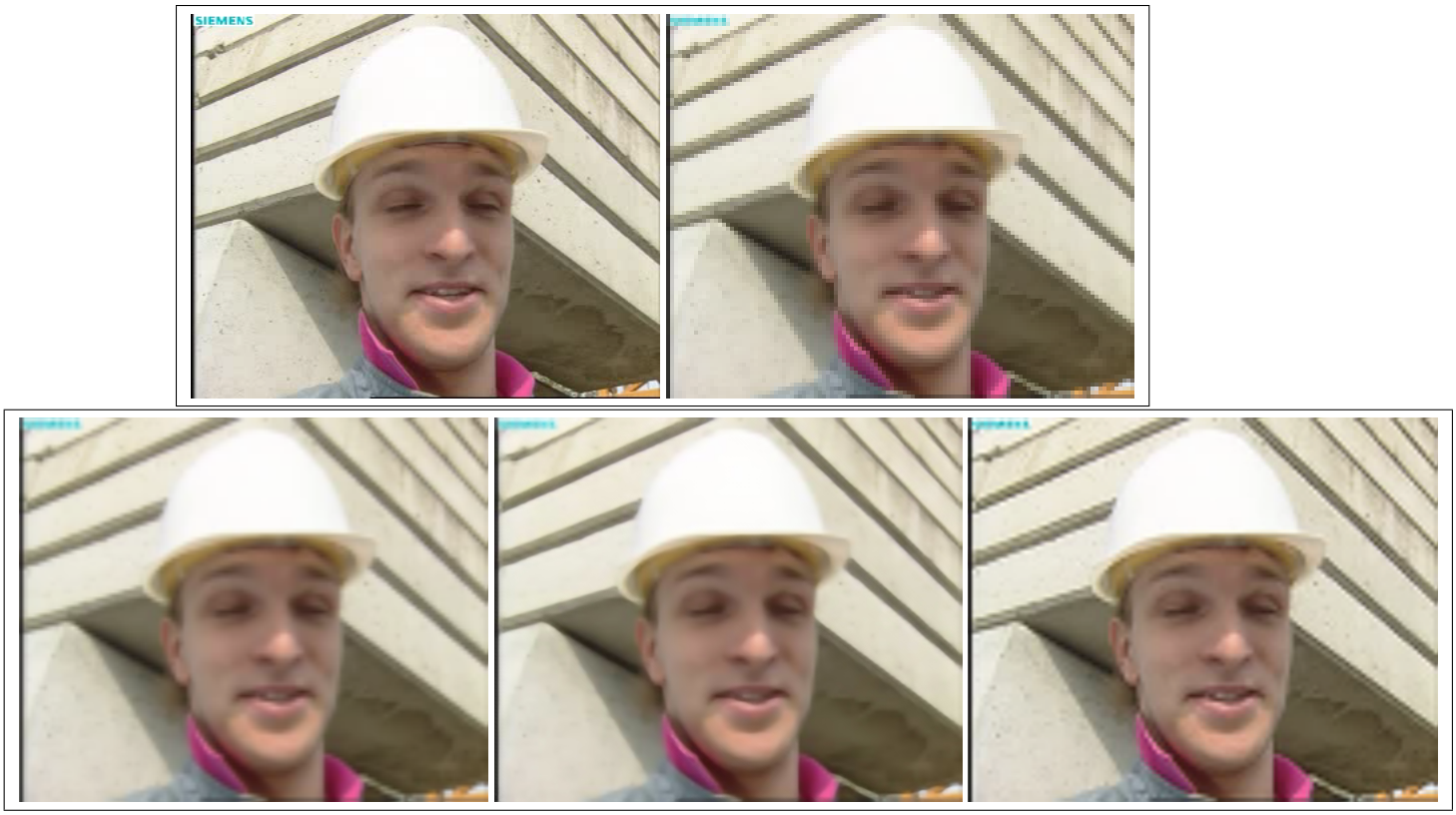

Figure 2. Examples of super-resolution on the foreman image from Set14 using the bicubic model (baseline), LapSRN model (state of the art), and our sparse representation model. The images include are: the original high-resolution image (top-left), the compressed low-resolution version of the image (top-right), super-resolution using the bicubic model (bottom-left), super-resolution using the LapSRN model (bottom-middle), and super-resolution using the sparse respresentation model (bottom-right). In this example, the super-resolved image using the sparse representation model scored much better using the quantitative metrics and the resulting image also appears much clearer with higher resolution than the other models.

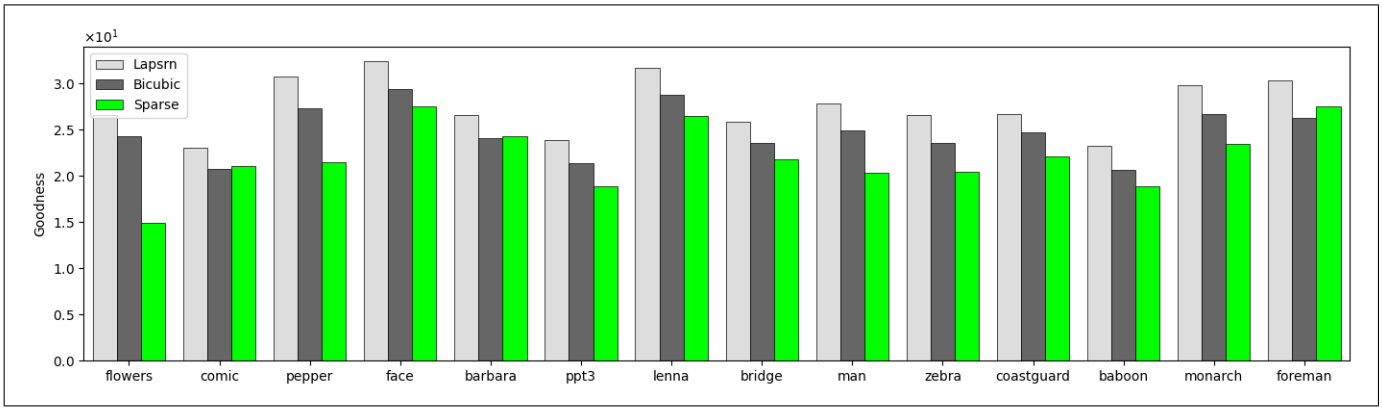

Figure 3. PSNR

fute the quantitative results. Despite the change in color, the flowers appear much clearer in our sparse SR image than the Bicubic SR image and possibly even the LapSRN SR image. Although specific details are changed, the image is sharper and easier to distinguish in details. For foreman, our sparse representation model scored better than Bicubic likely since there is a smaller range of colors in the image. Upon visual analysis, the sparse SR foreman image is almost indistinguishable from the original HR image despite the significant decrease in resolution in the LR image used

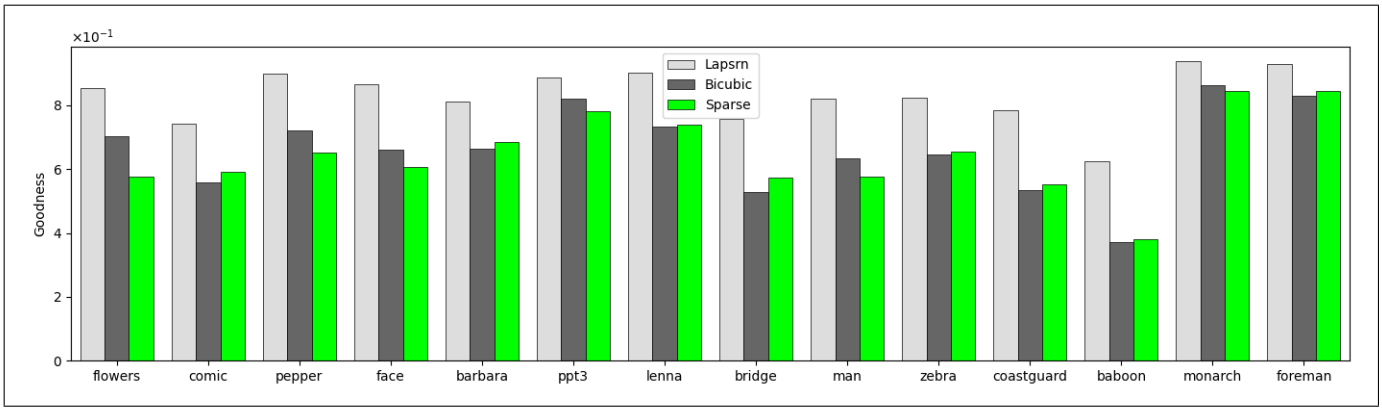

Figure 4. SSIM

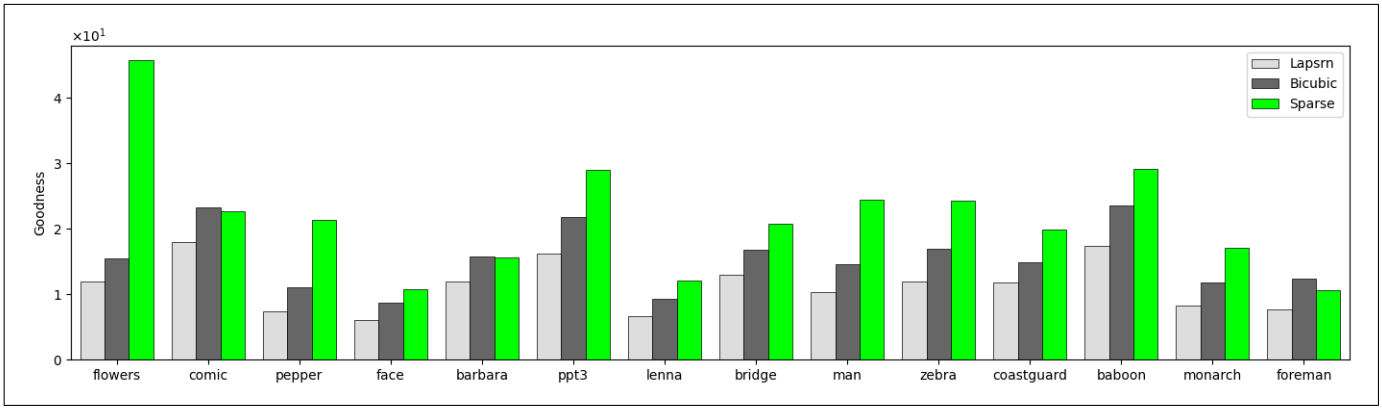

Figure 5. RMSE

as the original image. Additionally, it is very clear that LapSRN consistently outperforms Bicubic.

4. Conclusion

In conclusion, in exchange for visual sharpness and less blurriness, the sparse representation model provides SR images with a change in color detail. The sparse representation model performs best, particularly in quantitative measures, on images with less extreme changes in color.

For our sparse representation model, the super-resolution works well in some cases when the sparse representation is accurately modeled. However, there is still significant inconsistencies even in a single image. As shown in the foreman example, the person's face is very sharp and clear, but the edges of the wall in the back are jagged. The sparse representation model works well when there is not a significant shift in color. When there is a significant change in color in the image, the sparse SR image is still sharper, but the color details are changed significantly, as shown in the flowers example where there is color loss. The Bicubic model tends to score higher in the quantitative metrics because it blurs the LR image to increase resolution, in exchange for less sharpness.

With further work in our sparse representation model, there is potential to create an SR model that both performs well qualitatively as well as quantitatively. Our model currently creates sharper images, but does not consistently maintain colors and image details, so it is lacking in quantitative metrics.

References

[1] M. Aharon, M. Elad, and A. Bruckstein. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, 54(11):4311–4322, 2006. 2

[2] E. Candes, J. Romberg, and T. Tao. Stable signal recovery from incomplete and inaccurate measurements, 2005. 2

[3] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*, 15(12):3736–3745, 2006. 2

[4] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang. Deep laplacian pyramid networks for fast and accurate super-

resolution, 2017. 1, 3

[5] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi. Photo-realistic single image super-resolution using a generative adversarial network, 2017. 1

[6] C. Shannon. Communication in the presence of noise. *Proceedings of the IRE*, 37(1):10–21, jan 1949. 2

[7] J. Yang, Z. Lin, and S. Cohen. Fast image super-resolution based on in-place example regression. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1059–1066, 2013. 1

[8] J. Yang, J. Wright, T. Huang, and Y. Ma. Image super-resolution as sparse representation of raw image patches. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008. 1