Superresolution and Satellite Track Removal in Astronomical Images

Preliminary Report (Team 16)

Arumugam Swetha
EE20BTECH11005
ee20btech11005@iith.ac.in

Satya Aditi Dhaaipule
EE20BTECH11046
ee20btech11046@iith.ac.in

Peram Sree Keerthan Reddy
EE20BTECH11050
ee20btech11040@iith.ac.in

Vibhuvan Kumar Reddy Mukkamalla
EE20BTECH11055
ee20btech11055@iith.ac.in

Abstract

This report discusses a method for improving the quality of astronomical images, which are often blurry, noisy, and have satellite tracks that obscure important information. High-resolution images are reconstructed using a super-resolution reconstruction technique based on wavelet transform which is an expectation-maximization reconstruction process to improve the accuracy and visual quality of the reconstructed image. Then the two algorithms, RAST and Hough Transform, are used to remove the satellite tracks. The effectiveness of the methods are evaluated using different metrics like PSNR, MSE, SSIM and Homogeneity.

1. Introduction

In the 1950s, the space program provided high-resolution images of Earth, the Moon, and Mars, but technical issues degraded them. Digital image restoration emerged to extract more information by applying signal processing techniques. The first application was in images of the comet Halley. Image processing is now routine in astronomical observations. Recent advances in Machine Learning and Deep Learning have given researchers more tools to explore and understand our nearby celestial neighbors. Therefore, image processing is undoubtedly one of the most important pre-processing and analytical phases in this pipeline.

The goal of this paper is to construct high-resolution astronomical images from severally blurred and noisy low-resolution images and then remove satellite trails. We can then enhance the image to improve image contrast quality. We then finally compare the output images with the original images using different metrics.

This manuscript is structured as follows. The motivation behind the work is described in Sect. 2. The analysis methodology is described in Sect. 3, and we conclude in Sect. 4.

2. Motivation

Space telescopes are limited in their physical resolution by physical phenomena such as the atmosphere and the interstellar medium, as well as system features such as lens aperture and CCD array qualities. These systems gather multiple images of celestial entities, which are often distorted by mechanical vibrations and the satellite's movement. The first objective is to use Superresolution image reconstruction [11] to create a new, high-resolution image from these low-resolution, noisy, and shifted observations to identify image details and structures not readily discernible in the raw data.

A study [3] revealed that the number of Hubble images distorted by passing satellites has increased due to private satellites such as Starlink, producing unremovable long, bright streaks and curves of light. The likelihood of spotting a satellite in a Hubble image between 2009 and 2020 was only 3.7%, but it increased to 5.9% in 2021, an increase that experts claim is related to Starlink. To address this issue, two methods, Recognition by Adaptive Subdivision of Transformation Space (RAST) [1] and Hough Transform [5], are employed to identify and remove satellite traces in images. The results obtained from these two methods are then compared after enhancing images to highlight hidden features.

3. Methodology

3.1. Wavelet-Based Superresolution

The process of super-resolution image reconstruction involves creating a higher-resolution image from a set of low-resolution, rotated, shifted, and noisy observations. To achieve this, we use a wavelet-based method that can effectively handle Gaussian noise in astronomical image processing. Wavelets and multiresolution analysis are particularly suitable for this task since they can provide accurate and sparse images of smooth regions with abrupt changes or singularities.

This method utilizes the EM algorithm to address image deconvolution by combining the discrete wavelet transform's efficient image representation with the diagonalization of the convolution operator obtained in the Fourier domain. The algorithm alternates between an E-step based on the Fast Fourier Transform (FFT) and a DWT-based M-step, which requires O(NlogN) operations per iteration, where N is the number of pixels in the super-resolution image.

y_k is modeled as a shifted, rotated, down-sampled, noisy version of the super-resolution x.

The movement of the instrument causes the shift and rotation, and the blur is caused by the point spread function (PSF) of instrument optics and the integration done by the CCD array. The downsampling models the change in resolution between the observations and the desired super-resolution image.

We have

$$y_k = DBS_k x + n_k \tag{1}$$

where D is the Down-sampling operator, B is the Blurring operator, S_k is the Shift and Rotation operator for kth observation, and n_k is Additive White Gaussian Noise with a variance of σ^2.

If we replace the DBS_k operator with H, the equation now becomes

$$y = Hx + n \tag{2}$$

Here, however, H is unknown because of the unknown shifts and rotations of the observations. If H was known, we can get x if H is invertible.

3.1.1 Registration of the Observations

We register the observed low-resolution images to one another using a Taylor series expansion (See [7]). Let f_1 and f_2 be the continuous images underlying the sampled images y_1 and y_2, respectively, where f_2 is shifted and rotated version of f_1.

$$f_2(t_x, t_y) = f_1(t_x cosr - t_y sinr + s_x, t_y cosr + t_x sinr + s_y) \tag{3}$$

where s_x, s_y is the shift and r is the rotation. A first-order Taylor series approximation of f_2 is

$$\hat{f}_2(t_x, t_y) = f_1(t_x, t_y) + (s_x - t_x r - \frac{t_y r^2}{2})\frac{\partial f_1}{\partial t_x} + (s_y - t_y r - \frac{t_x r^2}{2})\frac{\partial f_1}{\partial t_y} \tag{4}$$

Let $\hat{y_2}$ be a sampled version of $\hat{f}_2$: then y_1 and y_2 can be registered by finding the s_k, s_y and r which minimize $||y_1 - \hat{y_2}||_2^2$, where $||x||_2^2 = \sum_i x_i^2$. This minimization is calculated with an iterative procedure which ensures that the motion being estimated in each iteration is small enough for Taylor series approximation.

A low-resolution image is registered to $DBS\hat{x}$. Taylor series can then produce highly accurate results. However, in low SNR scenarios, where confidence in registration parameter estimates may be low, the estimates can be further refined at each iteration of the proposed EM algorithm.

3.1.2 Multiscale Expectation-Maximization

We introduce an unobservable or missing data space. The direct inverse can be split into two subproblems: computing the expectation of unobservable data (as though no blurring or downsampling took place) and estimating the image from this expectation.

We derive an EM algorithm that consists of linear filtering in the E-step and image denoising in the M-step.

The Gaussian observation model can be written with respect to the DWT coefficients θ, where x = Wθ and W denotes the inverse DWT operator (See [9]).

$$y = HW\theta + n \tag{5}$$

The noise in the observation model can be decomposed into two different Gaussian noises, white and non-white:

$$n = \alpha H n_1 + n_2 \tag{6}$$

where α is a positive parameter, and n_1 and n_2 are independent zero-mean Gaussian noise with covariances $\sum_1 = I$ and $\sum_2 = \sigma^2 I - \alpha^2 HH^T$, respectively.

The Gaussian observation model can be rewritten as

$$y = H(W\theta + \alpha n_1) + n_2 \tag{7}$$

Thus, we get a sequence of x for t = 0, 1, 2...

3.1.3 E-step

This set updates the estimate of the missing date using the relation:

$$\hat{z}^{(t)} = E[z|y, \hat{\theta}^{(t)}] \tag{8}$$

This can be reduced to a Landweber [8] iteration:

$$\hat{z}^{(t)} = \hat{x}^{(t)} + \frac{\alpha^2}{\sigma^2} H^T (y - H\hat{x}^{(t)}) \tag{9}$$

3.1.4 M-step

We update the estimate of the superresolution image x. In the Gaussian case, this constitutes updating the wavelet coefficient vector θ according to

$$\hat{\theta}^{(t+1)} = argmin_{\theta}\{\frac{||W\theta - \hat{z}^{(t)}||_2^2}{2\alpha^2} + pen(\theta)\} \quad (10)$$

and setting $\hat{x}^{(t+1)}$ = W$\hat{\theta}^{(t+1)}$. This optimization can be performed using any wavelet-based denoising procedure.

For example, under an i.i.d Laplacian prior, $pen(\theta) = -logp(\theta) \propto \tau||\theta||_1$ (where $||\theta||_1 = \Sigma_i|\theta_i|$ denotes the l_1 norm), $\hat{\theta}^{(t+1)}$ is obtained by applying a soft-threshold function to the wavelet coefficients of $\hat{z}^{(t)}$.For the simulations presented in this paper, we applied a similar denoising method described in [4], which requires O(N) operations.

3.2. Satellite Tracks Removal

3.2.1 Recognition by Adaptive Subdivision of Transformation Space

In this section, we describe the implementation of the RAST algorithm for detecting lines in gray-scale images. Prior to applying the RAST algorithm, image pre-processing is performed on FITS gray-scale images to convert them into Portable BitMap (PBM) images, where binarization is accomplished through global thresholding [6]. In order to remove noise and CCD faults from the images, they are next subjected to the morphological operation "Opening" (Erosion followed by Dilation).

The RAST algorithm takes sample points as input features, where the points correspond to the (x,y) coordinates of the white pixels in the binarized image and attempts to fit an optimal line on these sample points in the image. The algorithm is based on a parametric model, described by line parameters (r, θ), where r is the distance of the line from the origin and θ is the angle between the perpendicular and the x-axis.

$$xcos\theta + ysin\theta = r \quad (11)$$

The algorithm is implemented using hierarchical and adaptive subdivision of the space of line parameters to search for the optimal parameter values, which starts with a large region and iteratively implements a recursive subdivision of the parameter space [2] until a solution is found using a quality heuristic to find the best-fit line, the algorithm terminates when the upper bound on the quality of the match falls below a specified threshold [2]. The algorithm returns a list of parameter values r and θ, and a new image is reconstructed, wherein the pixels of the detected line are replaced with suitable intensities.

The different parameters that can be set for the algorithm include the epsilon (eps) parameter, which defines the distance up to which a point can contribute to the line; the tolerance (tol) and angle tolerance (atol) parameters, which specify the allowed deviation of the parameters r and θ, respectively, from their optimal values. The default value of the tolerance parameter is set to `0.1`, the value of the angle-tolerance parameter is set to `0.001`, and the value of the epsilon parameter is set to `2`. The quality parameter which specifies the minimum acceptable quality of a line as given by

$$q(\vartheta, P) = \sum_{k=1}^{N} max(0, 1 - \frac{d_k^2}{\varepsilon^2}) \quad (12)$$

where ϑ is the set of parameters (r, θ), P is set of points, N is total number of points and d_k is the distance of kth point from the line.

The implementation steps of the RAST algorithm are as follows:

1. We choose an initial region T in parameter space (r, θ) containing all range of parameter values, then define a priority queue Q, where the priority of the queue Q is the quality of best match in that region, now T is then inserted into the priority queue Q.

2. We pop the element of the highest priority from the queue Q.

3. If the quality of the best match in the extracted region T is less than the minimum quality threshold, then you end the algorithm.

4. If the size of the region T is less than the tolerance size of the region dimensions r and θ, then the region T is a solution, and we will go to step 3.

5. If the size of the region T is greater than the minimum size of the tolerance interval for a solution, then it is divided into two sub-regions, Ta and Tb, and these sub-regions are pushed into the priority queue Q, and the algorithm continues to step 3.

3.2.2 Hough Transform

The Hough Transform is an image processing technique that can detect linear and other features in an image. It works by transforming the data from the point space to the Hough space, which represents lines [5]. In the Hough space, each point corresponds to a line in the original space that is perpendicular to the line passing through the center of the data space and inclined at a specific angle.

To apply the Hough Transform, the following steps are taken:

1. Compute discrete values for r_i and θ_i in the appropriate range for the image. The range of r is from -R to R (where R is the resultant distance), and the range of θ is from -$\pi/2$ to $\pi/2$.

2. Create an array of size r_n by θ_n, where r_n is the number of r values and θ_n is the number of θ values. This array is an accumulator or a three-dimensional histogram.

3. For each pixel in the image, loop through each θ value. Compute r for each θ value and select the nearest discrete r value.

4. Increment by one the appropriate [r,θ] element in the accumulator. The more counts an element has in the accumulator, the more points there are on that particular line.

5. Once all pixels have been processed, the accumulator will contain peaks at the [r,θ] values corresponding to the lines in the image.

6. If the number of counts in a particular element exceeds a certain value, then the [r,θ] value associated with that element is considered to be a line in the image.

The Hough Transform is particularly useful when traditional edge detection techniques fail, such as when the edges are not well defined or broken. It can also be used to detect other features in an image, such as circles or ellipses, by modifying the accumulator to represent those shapes [5].

For the removal of the line detected using the above two algorithms, we can use something as simple as Bilinear Interpolation (where the line's new pixel is detected using a distance-weighted average of four neighboring pixel values). We can explore other methods, we will develop on this in the next report.

3.3. Image Enhancement

Image enhancement (IE) is one of the major image processing approach and used for enhancing the original image quality, which improves the contrast of the image by sharpening the edge pixels intensity. Initially, the image is denoised using the Modified median filter (MMF) filter [10] and then enhanced with a technique suitable for the image. This will be done if deemed required.

3.4. Performance Analysis

The performance of the different approaches are measured using PSNR, MSE, SSIM, WPSNR, homogeneity, and contrast [10].

3.4.1 MSE

The Mean Squared Error metric is used to compute the difference between the original and enhanced images, and it is represented as:

$$MSE(I,J) = \frac{1}{NM}\sum_{i=0}^{N-1}\sum_{j=0}^{M-1}(I(i,j) - J(i,j))^2 \quad (13)$$

3.4.2 Contrast

It defines the statistical features that measures the variance between intensity and the original image.

$$C_0 = \sum_{i=0}^{N-1}\sum_{j=0}^{M-1} C_A(i,j) \quad (14)$$

where $C_A(i,j)$ is a total co-appearance for levels of intensity i and j.

3.4.3 PSNR

It is exploited to evaluate variance among the original and enhanced images, and it is represented as:

$$PSNR = 10log_{10}\left[\frac{L^2}{MSE(I,J)}\right]dB \quad (15)$$

where L is the maximum allowable gray value.

3.4.4 Homogenity

This metric provides the image intensity similarity. That means a large value for homogeneity provides a large quality of the image. It is expressed as:

$$H_0(I,J) = \sum_{i=0}^{N-1}\sum_{j=0}^{M-1}\frac{C_A(i,j)}{1+|i-j|} \quad (16)$$

3.4.5 SSIM

The structural similarity (SSIM) index between image I and J is given by

$$SSIM_{I,J}(i,j) = L_{I,J}(i,j)C_{I,J}(i,j)S_{I,J}(i,j) \quad (17)$$

where,$L_{I,J}(i,j)$ is the measure of local luminance similarity, $C_{I,J}(i,j)$ is the measure of local constrast similarity, $S_{I,J}(i,j)$ is the measure of local structure similarity.

4. Conclusion

Astronomical images are crucial for understanding the universe, but they often are severally blurred and contain noise and obscuring lines called satellite tracks, which can interfere with identifying important objects. To address this, we first re-construct High-resolution images using super-resolution reconstruction [11] and use algorithms like

RAST [1] and Hough Transform [5] to remove the satellite trails.

We first use a novel approach to image super-resolution based on wavelet transform in the presence of Gaussian noise. The method combines the efficient image representation offered by the discrete wavelet transform with the diagonalization of the convolution operator obtained in the Fourier domain. The proposed algorithm alternates between an E-step based on the Fast Fourier Transform and a DWT-based M-step, with O(NlogN) operations per iteration.

Then we use two algorithms:

1. **RAST**, which can detect straight satellite tracks in images by taking coordinates of white pixels and fitting an optimal line to them using hierarchical and adaptive subdivision of the space of line parameters r and θ. The algorithm returns a list of optimal parameter values of r and θ, which are used to reconstruct a new image with suitable intensities.

2. **Hough Transform**, which is used to detect linear and other features in an image by transforming the data from point space to Hough space. It involves creating an accumulator, a three-dimensional histogram, and computing r and θ values in the appropriate range. This method is particularly useful in cases where traditional edge detection techniques fail and can be adapted to detect other features like circles or ellipses.

We then will finally compare the images obtained using MSE, PSNR, Homogeneity, and SSIM to evaluate the methods used.

References

[1] Haider Ali, Christoph H. Lampert, and Thomas M. Breuel. Satellite tracks removal in astronomical images. In José Francisco Martínez-Trinidad, Jesús Ariel Carrasco Ochoa, and Josef Kittler, editors, *Progress in Pattern Recognition, Image Analysis and Applications*, pages 892–901, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg. 1, 5

[2] Thomas M. Breuel. Finding lines under bounded error. *Pattern Recognition*, 29(1):167–178, 1996. 3

[3] Kruk S. et al. The impact of satellite trails on hubble space telescope observations. *Nat Astron (2023)*. 1

[4] Mário AT Figueiredo and Robert D Nowak. Wavelet-based image estimation: An empirical bayes approach using jeffrey's noninformative prior. *IEEE Transactions on Image Processing*, 10(9):1322–1331, 2001. 3

[5] Owen D Giersch and John A Kennewell. Automated analysis of satellite trails in astronomical images. 1, 3, 4, 5

[6] R. E. Woods R. C. Gonzalez. Digital image processing, 2nd edition. prentice-hall, inc., 2002. 3

[7] Michal Irani and Shmuel Peleg. Improving resolution by image registration. *CVGIP: Graphical Models and Image Processing*, 53(3):231–239, 1991. 2

[8] L. Landweber. An iteration formula for fredholm integral equations of the first kind. *American Journal of Mathematics*, 73(3):615–624, 1951. 2

[9] Stéphane Mallat. *A wavelet tour of signal processing*. Elsevier, 1999. 2

[10] S. Navaneetha Krishnan, D. Yuvaraj, Kakoli Banerjee, P Joel Josephson, T CH Anil Kumar, and Mohamed Uvaze Ahamed Ayoobkhan. Medical image enhancement in health care applications using modified sun flower optimization. *Optik*, 271:170051, 2022. 4

[11] R. M. Willett, I. Jermyn, R. D. Nowak, and J. Zerubia. Wavelet-Based Superresolution in Astronomy. In Francois Ochsenbein, Mark G. Allen, and Daniel Egret, editors, *Astronomical Data Analysis Software and Systems (ADASS) XIII*, volume 314 of *Astronomical Society of the Pacific Conference Series*, page 107, july 2004. 1, 4