Computational Astrophotography using Image Processing

Arnav Jalan
Computer Science and Engineering
Shiv Nadar University
Email: aj713@snu.edu.in

Arnav Aditya
Computer Science and Engineering
Shiv Nadar University
Email: aa716@snu.edu.in

Aditya Raghuram
Computer Science and Engineering
Shiv Nadar University
Email: ar694@snu.edu.in

Jaideep Singh
Computer Science and Engineering
Shiv Nadar University
Email: js452@snu.edu.in

Abstract—**This paper presents an exploration into the realm of low-cost astrophotography through the application of image processing techniques. Traditional astrophotography has been characterized by its high costs and technical barriers, limiting access to amateur astronomers. However, recent advancements in image processing algorithms have enabled the enhancement and extraction of valuable information from raw astronomical images obtained with consumer-grade equipment. This paper delves into the potential of computational astrophotography in improving image quality and increasing resolution of the celestial horizon. Through a series of equations and mathematical methods, the efficacy of various image processing algorithms, including image thresholding, Otsu's method, bilateral filtering and more is demonstrated.**

1. Introduction

Astrophotography, the practice of capturing images of celestial objects using DSLR cameras, has gained widespread popularity among enthusiasts and professionals alike. However, the pursuit of this hobby often comes with significant financial barriers, particularly stemming from the need to counteract the Earth's rotation during long exposure times. As celestial objects traverse the sky during these exposures, star streaks—bright lines tracing their trajectories—can mar the quality of the images, detracting from their accuracy. While some photographers intentionally incorporate these streaks for artistic effect, they pose a challenge for those seeking to produce precise representations of the night sky.

To address this issue, astrophotographers commonly rely on moving camera and telescope mounts to compensate for the Earth's rotation, ensuring that celestial objects remain steady within the frame. However, the cost of these specialized mounts can be prohibitive, placing the practice of astrophotography out of reach for many aspiring enthusiasts. In light of these challenges, this paper proposes an alternative approach: leveraging computational techniques to mitigate the effects of Earth's rotation during long-exposure imaging. By harnessing the power of digital image processing, we aim to eliminate the need for expensive rotating mounts, thereby making astrophotography more accessible and affordable to a broader audience. Through the application of computational methods, we seek to refine the process of capturing and post-processing long-exposure images, ultimately enhancing the accessibility and inclusivity of this captivating hobby.

2. Related Work

In current astrophotography practices, compensating for the Earth's rotation often involves physically moving the camera along with the Earth. Nevertheless, efforts have been made to alleviate the necessity for such physical adjustments. Notably, a technique has been proposed that remaps star streaks to polar coordinates, ensuring a consistent point-spread function for all stars in the image. Although primarily aimed at star localization for identification purposes, this approach holds promise for our objective of mitigating star streaks [4]. Additionally, Richardson-Lucy deblurring techniques have been employed to address quality degradation in star images. This method iteratively refines the estimate of the original image, effectively reducing blurring effects caused by various factors such as atmospheric turbulence or optical imperfections [2]. Furthermore, maximally sparse optimization approaches have been explored to enhance image fidelity by minimizing noise and artifacts. By promoting sparsity in the image domain, these techniques facilitate the extraction of meaningful information from noisy or degraded images [1]. These endeavors collectively underscore the ongoing exploration of computational methods to enhance the accuracy and clarity of astrophotographic imagery, aligning closely with the objectives of this study.

3. Data Collection

In current astrophotography practices, compensating for the Earth's rotation often involves physically moving the camera along with the Earth. Nevertheless, efforts have been made to alleviate the necessity for such physical adjustments. Notably, a technique has been proposed that remaps star streaks to polar coordinates, ensuring a consistent point-spread function for all stars in the image. Although primarily aimed at star localization for identification purposes, this approach holds promise for our objective of mitigating star streaks [4]. Additionally, Richardson-Lucy deblurring techniques have been employed to address quality degradation in star images. This method iteratively refines the estimate of the original image, effectively reducing blurring effects caused by various factors such as atmospheric turbulence or optical imperfections [2]. Furthermore, maximally sparse optimization approaches have been explored to enhance image fidelity by minimizing noise and artifacts. By promoting sparsity in the image domain, these techniques facilitate the extraction of meaningful information from noisy or degraded images [1]. These endeavors collectively underscore the ongoing exploration of computational methods to enhance the accuracy and clarity of astrophotographic imagery, aligning closely with the objectives of this study.

4. Image Segmentation

To turn the lines of stars into points, we need to use a process called image segmentation. This helps us isolate the star streaks. The star streaks can have different levels of brightness and color, and these can even change along the length of a single streak. Also, even though the sky is usually quite dark at night, there can be areas that are slightly brighter or have different colors. Lastly, there are often objects in the front of the image that we need to separate from the star streaks in the back.

4.1. Thresholding

We carry out three distinct thresholding methods to isolate the star streaks from the rest of the image. Each technique is executed separately before being merged to reduce the likelihood of undesired segmentations being carried forward, as depicted in Figure 1. Initially, we employ global thresholding by examining the grayscale versions of the original images. If the pixel value on the black and white image exceeds a global threshold, we regard it as part of a star streak in the color image. This approach enables us to capture the brightest star streaks while preserving the color data. Nevertheless, this might result in large sections of images being classified as stars and removed from the image if their grayscale values are excessively high. To rectify this, we conduct a second pass through the image and restore the pixels to their original values if the amount of information removed in a specific window surpasses a certain percentage, as illustrated in Figure 2.

Next, we apply Otsu's method, a clustering-based image thresholding technique, to effectively distinguish the image's background from the foreground while segmenting a different group of star streaks [3]. Using a binary version of the original image, Otsu's method determines an optimal threshold that maximizes the intra-class variance and minimizes the inter-class variance. The stars are then segmented based on this computed threshold and eliminated from the image. Once again, to reduce the quantity of dark patches, we iterate through the image and restore the removed pixels to their original values if an excessive amount of information was removed in a specific area. Lastly, we execute thresholding by examining the HSV values converted from the RGB image. Transforming the image to HSV representation enables us to separately consider the hue, saturation, and value, unlike in the RGB space. This aids us in detecting some of the more faint streaks in the image. After converting to the HSV space, we inspect the value array and only segment out the pixels that have a higher value compared to a modified mean. Similar to the previous thresholding techniques, we post-process the resulting image to eliminate the dark spots.

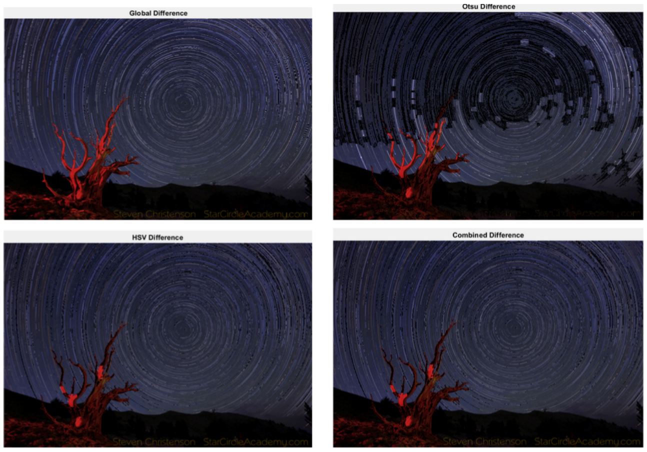

Figure 1. Different thresholding methods

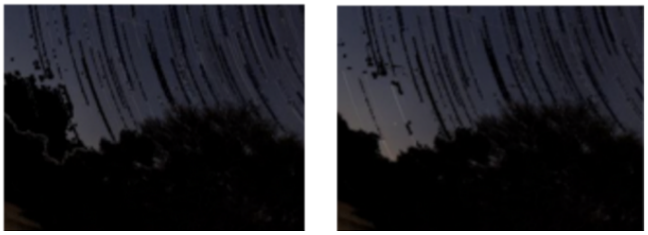

Figure 2. Patch removal: before vs. after

4.2. Width Estimation

Considering that individual star streaks could vary in brightness and color along their path, we also incorporate the neighboring pixels that are within a certain gradient from the pixels that were eliminated by the thresholding methods. This enables us to capture a more representative segment of the star streak while leaving less behind. Even though the brightness tends to be highest in the center of the streak, dimmer portions along the width of the star are also part of the streak. This method also helps account for varying streak widths, as opposed to just using a defined region size of neighbors, as depicted in Figure 3.

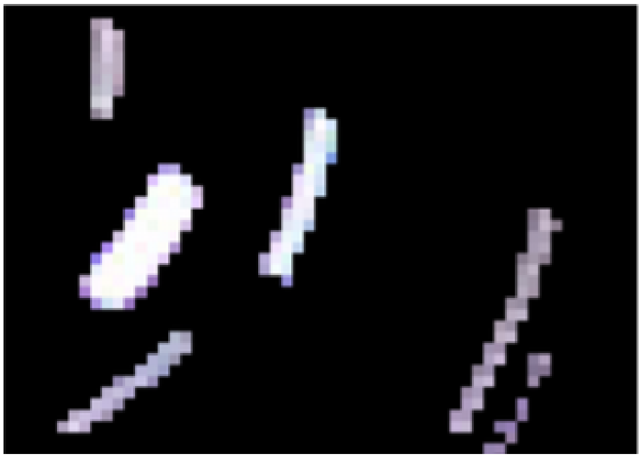

Figure 3. Width estimation

4.3. Combining Methods

After executing the three independent thresholding methods and accurately estimating the star widths, we merge the resulting star streaks to maximize the number of streaks we remove from the image segmentation. As an additional verification, we examine the amount of information that was removed from the original image before combining the results. Since certain methods would perform better than others on images with varying parameters such as illumination and quality, we only combine those that removed a balanced amount of pixel values to avoid losing too much information compared to the original image, as depicted in Figure 1. We also experimented with other segmentation techniques like edge detection, but the outcomes were even poorer than those of global thresholding. This might be because the star streaks are densely packed in the data images. Other issues included numerous artifacts along the foreground/background boundaries, particularly along the horizon. As a result, we opted for balancing multiple thresholding methods, which yielded significantly better results.

5. Star Streak Modification

5.1. Finding the Polar PSF (Point Spread Function)

The star streaks are caused by the rotation of the earth and form a circular shape around the celestial north pole as displayed in figure 4. The point spread function (PSF) is not uniform because it is caused by many factors that contribute to the blurring effect in an image. For instance, lens aberrations, diffraction, pixel size, and sensor noise can all affect the shape of the PSF. This means that the PSF can vary depending on the optics and imaging system used for capturing the image. The non-uniformity of the PSF is caused by the Earth's rotation and atmospheric effects which translate into a time-varying PSF that changes with the camera's exposure time and position. So, we transform the star streak image into polar coordinates with the origin at the celestial pole to have each individual streak appear as an approximately horizontal line of a uniform length. We have chosen to instead test our procedure exclusively on images in which the celestial pole lay within the image, and to have the user manually indicate the location of the axis by clicking on it when prompted.

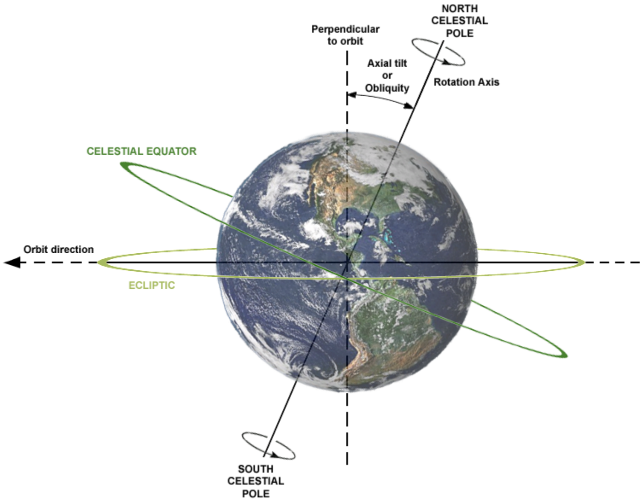

Figure 4. Celestial Pole

5.2. Star Clustering

After transforming the streaks into polar coordinates, we were able to convert them into points. However, the star streaks in the images tested did not form uniform circles, and so did not form perfectly uniform lines in polar coordinates. So, we implemented a recursive algorithm where each bright pixel is recognised and its neighbors are added to a cluster given that they aren't surrounded by dark pixels. This process is repeated until the complete cluster has been formed. Next, the cluster is compressed into a single pixel by averaging the values of all pixels in the cluster. Subsequently, the value of each cluster is set to zero (except for the center pixel which is set to zero). This process is repeated until each cluster has been identified.

[Recursive Algorithm] Input image I, coordinates of a bright pixel (x, y) Cluster of connected bright pixels AddClusterAddCluster FnFunctionend I, x, y Base case $I(x, y)$ is dark or already in cluster Add current pixel to cluster AddPixelToCluster(x, y) Recursive case each neighboring pixel (x', y') of (x, y) in I (x', y') is bright and not in cluster I, x', y' Recursive algorithm for forming a cluster of connected bright pixels

5.3. Star replacement

The blurred image (with removed dark patches) is combined with the image where the star streaks have been

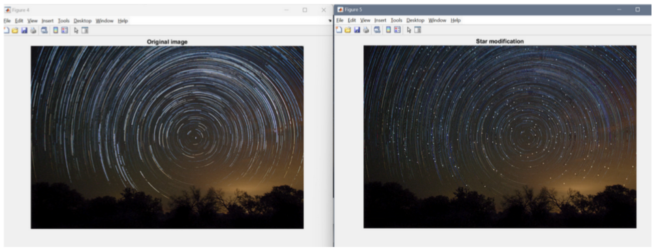
Figure 4
File Edit View Insert Tools Desktop Window Help
Original image
Figure 5
File Edit View Insert Tools Desktop Window Help
Star modification

Figure 5. Star Modification

transformed to pixels. Then, the start pixels are clustered and the compressed image is converted back to Cartesian coordinates, with the origin again at the celestial pole. Next, the stars are correctly placed and the overlapping stars are removed.The image containing only the star pixels is convolved with a 3-pixel wide Gaussian kernel in order to cause the stars to appear broader. The standard deviation of the Gaussian was made to depend on the size of the image, so as to avoid creating huge stars in a small image of the sky. After this widening and filtering of the star image, the background image and the star image are simply added to each other.

Equation for the 2D Gaussian kernel:

$G(x, y) = e^{-\frac{x^2+y^2}{2\sigma^2}} \frac{}{2\pi\sigma^2}$

6. Image Post Processing

6.1. Hole filling

After identifying and removing the bright streaks of stars from the image, we encounter empty regions where the stars were located. These gaps disrupt the visual continuity of the image and need to be addressed. To fill these voids, we first tackle each line of pixels where stars were removed. By replacing the empty spaces with the average value of the surrounding pixels, we ensure that the transition between the starry areas and the background is smoother. This step prevents abrupt changes in pixel intensity and helps maintain the natural appearance of the sky. Next, for the remaining darkened areas in the image, we employ a two-dimensional averaging function. This technique is necessary to effectively fill in larger gaps and create a more cohesive night sky. By considering the surrounding pixels in both horizontal and vertical directions, we achieve a more comprehensive interpolation, resulting in a visually pleasing and seamless image overall.

6.2. Smoothening and Noise Filtering

After addressing the gaps, residual noise persists in segments of the night sky within the images, occasionally causing abrupt color transitions between adjacent pixels. To mitigate this, we employ a bilateral filter leveraging a Gaussian distribution to refine the image. Mathematically, the bilateral filter is defined as:

$$B(I) = \frac{1}{W_p} \sum_{p \in \Omega} G_{\sigma_s}(||p - p'||) \cdot G_{\sigma_r}(|I(p) - I(p')|) \cdot I(p')$$

Here, $B(I)$ represents the filtered image, W_p is the normalization factor, Ω denotes the spatial domain, G_{σ_s} and G_{σ_r} are Gaussian functions with standard deviations σ_s and σ_r controlling the spatial and intensity domains respectively, and $I(p)$ represents the intensity of pixel p. This technique is selectively applied to images with sufficiently high resolution to prevent adverse impacts on the newly introduced point stars. The modified window size W is adjusted accordingly to ensure minimal interference with the stars. One notable advantage of the bilateral filter is its ability to preserve edge sharpness while smoothing the image, unlike a basic median filter. Mathematically, this is achieved by incorporating both spatial and intensity domains into the filtering process, allowing for fine control over the degree of smoothing applied to different regions of the image. This approach is crucial as it aims to retain the crisp edges present in the foreground while achieving a smoother appearance for the night sky in the background. Additionally, the point stars are rendered with a softer and more seamlessly integrated look against the backdrop of the night sky.

7. Results and Discussion

Qualitatively, the images generated by our algorithm exhibit visually pleasing attributes, exemplified in Figure 8. Stars, once localized at the center of streaks in the original image, maintain their color while seamlessly blending into the background sky, thereby mitigating any harsh transitions caused by streak removal. This process proves effective even in images characterized by complex color gradients. While the qualitative preservation of edges post-bilateral filtering appears satisfactory, accurate quantification is challenging due to the absence of ground truth images. However, some limitations persist; imperfect segmentation results in residual streaks, particularly faint ones, remaining in the processed images. Additionally, sections of the foreground with high brightness may erroneously be interpreted as stars, introducing false positives. The filling-in process, although functional, occasionally yields a somewhat blotchy background. Moreover, the uniformity in shape and size of all stars contrasts with the natural variability observed in the night sky. Quantitatively, assessing the efficacy of star streak removal involved measuring changes in image characteristics, such as variance, before and after processing. A decrease in variance would indicate successful streak removal. Additionally, our algorithm demonstrated efficient performance, with processing times averaging 20 seconds for medium-sized images (around 500k pixels) and 80 seconds for large, high-quality images (around 3 million pixels).

8. Conclusion

We successfully developed an algorithm capable of mitigating the majority of star streaks present in an image, substituting them with accurately-colored point-stars. Although some streaks persist, resulting in a notably smoother sky compared to the unedited image, the application of denoising techniques through bilateral filtering enhances the overall aesthetic quality while preserving the spatial distribution of stars. To further refine our approach, we propose exploring alternative image segmentation methods, potentially leveraging machine learning algorithms to enhance the differentiation between star streaks and background elements. Presently, all stars are rendered with uniform shapes, which diverges from their actual appearance; hence, future enhancements could involve implementing more realistic visualization techniques, possibly incorporating width estimation algorithms. Moreover, improving strategies for addressing gaps left in the image post-streak removal warrants investigation. While constrained from acquiring original images for this study, future endeavors should involve testing our algorithm on raw images directly captured by a camera, bypassing any post-processing steps.

Acknowledgments

The authors would like to express their sincere gratitude to Professor Saurabh Janardan Shigwan for his invaluable guidance, encouragement, and support throughout the duration of this research project. His expertise, insights, and unwavering dedication have significantly contributed to the success and completion of this study. Additionally, the authors extend their appreciation to Teaching Assistant Tanvir Singh for his assistance, feedback, and technical expertise, which have been instrumental in shaping the direction and methodology of this work. Their contributions are deeply appreciated and acknowledged with heartfelt gratitude.

9. References

[1] B. D. Jeffs and M. Gunsay. Restoration of blurred star field images by maximally sparse optimization. IEEE Transactions on Image Processing, 2(2):202– 211, April 1993.

[2] L. W. H. W. Y. H. Laili Su, Xiaopeng Shao. Richardson-lucy deblurring for the star scene under a thinning motion path. 9501, 2015.

[3] N. Otsu. A threshold selection method from gray-level histograms. IEEE Transactions on Systems, Man, and Cybernetics, 9(1):62–66, Jan 1979.

[4] B. Sease and B. Flewelling. Polar and spherical image transformations for star localization and rso discrimination. 01 2015.