International Journal of Computer Vision and Signal Processing, 15(1), 67-73(2025) ORIGINAL ARTICLE

Refining NeRF: The Power of High-Resolution Omnidirectional Vision

Sho Hasegawa*

Faculty of Engineering
Shibaura Institute of Technology, Tokyo, Japan

Chinthaka Premachandra

Faculty of Engineering
Shibaura Institute of Technology, Tokyo, Japan

Abstract

As NeRF technology becomes more widely adopted, research has increasingly focused on enhancing its performance. Despite its potential, NeRF still faces several challenges, including long training times, high computational demands, and lower accuracy compared to other 3D reconstruction methods such as photogrammetry. A critical step in NeRF generation is camera pose estimation, which typically involves extracting features such as object boundaries and corners from captured images. We found that using an omnidirectional camera can reduce shooting time while still enabling accurate NeRF generation, even when the camera lacks a high-performance image sensor. In this study, we aimed to improve the quality of camera pose estimation in order to enhance the accuracy of NeRF generation by increasing the resolution of partitioned omnidirectional images and improving the definition of object boundaries. Our experiments demonstrated that these improvements effectively reduced noise in the generated NeRFs and improved their overall accuracy. Therefore, our findings suggest that even with consumer-grade devices, such as general omnidirectional cameras, it is possible to generate a more accurate NeRF space by incorporating the proposed processing.

Keywords: 3D Generation, NeRF, Omnidirectional Image, Image Resolution Improvement, Object Boundary Enhancement

IJCVSP

ISSN: 2186-1390 (Online)
http://cennser.org/IJCVSP

Article History:
Received: 9/4/2025
Revised: 27/8/2025
Accepted: 1/11/2025
Published Online: 23/11/2025

1. INTRODUCTION

In NeRF generation, the camera's pose from multiple viewpoints (i.e., its position and orientation) is estimated first by performing feature matching across the input images. Based on the estimated poses and input images, an MLP (Multi-Layer Perceptron) is used to learn the scene, generating four-dimensional outputs such as density and color, which represent the volumetric properties of the environment. The color values along a desired viewing direction are computed by accumulating color and density values, resulting in an image from that virtual viewpoint. Thus, NeRF represents both the 3D geometry and the appearance of a scene from arbitrary viewpoints.

In our study, input images are first processed to enhance object boundaries and increase overall resolution. Camera poses are then estimated from the processed images, and a NeRF model is generated based on the obtained parameters. The resulting model is used to reconstruct the scene in 3D space. Further details are provided in the following chapter.

Photogrammetry, a traditional method for reconstructing 3D shapes from multiple images, has been studied since the mid-19th century. More recently, Neural Radiance Fields (NeRF) have gained significant attention as a deep learning-based alternative. The primary differences between NeRF and conventional photogrammetry lie in generation time and the range of objects that can be accurately represented. Photogrammetry often requires several hours to capture and process a single scene [1]. In contrast, NeRF leverages deep learning techniques and can reconstruct scenes with shorter capture times. The required

*Corresponding author
Email addresses: ag20038@shibaura-it.ac.jp (Sho Hasegawa), chintaka@sic.shibaura-it.ac.jp (Chinthaka Premachandra)

time can be reduced to just a few minutes, depending on the complexity of the scene.

However, photogrammetry struggles with objects that lack clear contours, such as reflective, refractive, or metallic surfaces, due to difficulties in detecting and matching features. In contrast, NeRF can learn how the appearance of a scene changes depending on the viewing angle. As a result, it is capable of reconstructing complex visual phenomena, including reflections and transparent objects such as glass, by modeling view-dependent effects.

NeRF has strong potential in mixed reality (MR) applications, and its use has expanded into everyday contexts. Notable examples include McDonald's commercials, artistic works produced by modeling collectives, and various entertainment applications. Furthermore, services such as Luma AI have made NeRF generation more accessible to general users. With the increasing availability of NeRF tools, research is shifting toward improving usability and performance [2]–[3]. Current challenges include long training times, high computational costs, and lower accuracy in complex scenes compared to traditional photogrammetry. In addition to efforts to improve the core NeRF model, several studies have focused on enhancing the accuracy of camera pose estimation, which is an essential step in NeRF reconstruction. Other studies have explored combining NeRF with complementary techniques [2][4].

The accuracy of NeRF generation is strongly influenced by camera hardware. For example, using a DSLR with a large sensor and a high-quality lens can better capture high-frequency image components, resulting in more accurate NeRF reconstructions. While high-end cameras are suitable for research purposes, making NeRF more accessible requires achieving similar levels of accuracy using consumer-grade devices.

To address this issue, our study proposes improving NeRF reconstruction using images captured by an omnidirectional camera, which has gained popularity in recent years. Many smartphones are now equipped with cameras capable of capturing omnidirectional images, making this form of photography increasingly common. In the field of NeRF research, several studies have investigated the integration of NeRF with omnidirectional imagery [5]. In our approach, as shown in Fig.1, images are first captured using an omnidirectional camera and then segmented into multiple regions. Each region is processed to enhance both resolution and the definition of object boundaries. These enhanced images are used in the camera pose estimation stage to improve the accuracy of estimated positions and orientations. This leads to more accurate virtual viewpoint synthesis and ultimately enhances the overall quality of the NeRF.

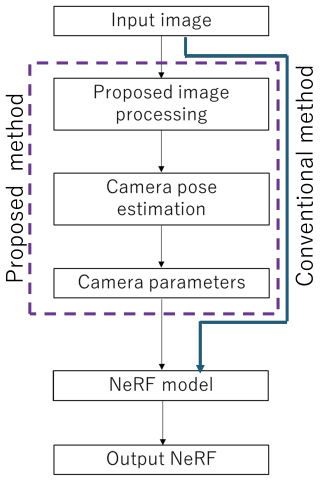

Figure 1: Processing flow of NeRF including proposed approach

2. Camera Pose Estimation and NeRF Space Learning

NeRF generation requires input images along with the external parameters of the camera, specifically its position and orientation at the time the image was captured. Estimating these parameters is known as camera pose estimation. Since this information is not usually recorded in the image metadata or by the camera itself, it must be estimated using other methods. One such method is Visual Simultaneous Localization and Mapping (VSLAM) [6], which estimates the camera's pose while simultaneously mapping the surrounding environment based on input images. By performing camera pose estimation, it becomes possible to determine the viewpoint and position of the camera during image or video capture. Camera pose estimation is commonly performed using marker boards such as those provided by ArUco. However, in this study, we use a software-based approach for estimation.

2.1. Camera Pose Estimation

In this study, we use COLMAP [7], a widely adopted tool in conventional NeRF pipelines, for camera pose estimation. COLMAP is an open-source Structure-from-Motion (SfM) framework that estimates camera poses from multiple images. Specifically, it employs a feature point detection algorithm known as SIFT (Scale-Invariant Feature Transform) [8]. COLMAP first detects object features such as edges and corners that are commonly observed across images captured from different viewpoints. It then estimates the camera poses by matching these features within the overlapping regions of the images. Finally, the NeRF scene is constructed using the estimated camera poses along with the corresponding input images.

2.2. Learning Strategies for NeRF

The estimated camera pose information is utilized to train the NeRF representation. In this study, we imple-

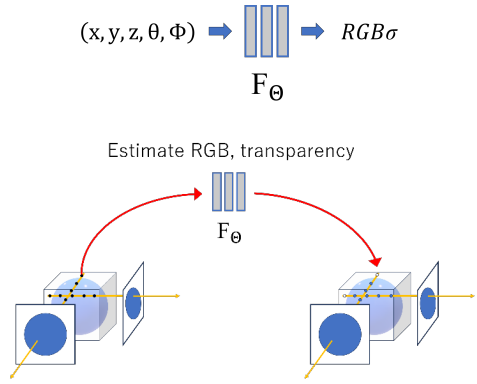

Figure 2: Schematic diagram of NeRF model

ment our method using NVIDIA's instant-ngp [9]. The primary objective of NeRF is to train a Multi-Layer Perceptron (MLP) [10], as illustrated in Fig.2. This MLP takes as input a five-dimensional vector comprising the 3D position coordinates (x,y,z) and the viewing direction (θ, ϕ) and outputs the volume density (σ) along with the object material's color and transparency in terms of RGB values.

However, instant-ngp, used in this study, applies Multiresolution Hash Encoding (MHE) to accelerate the training process and improve rendering quality [9]. MHE converts low-frequency inputs such as 3D position vectors and viewing directions into high-frequency signals such as RGB values and density, which represent the color and transparency of an object material. Although similar encoding techniques have been used in previous studies, MHE achieves faster performance because it is highly parallelizable. In addition, its flexibility allows it to be applied to other systems, including SLAM.

In the MHE process, the input is first mapped to indices on a multi-resolution grid (Fig.3). Feature vectors are then obtained by linearly interpolating from these indices. These feature vectors are combined with other inputs, such as the viewing direction, to form a final input vector. This vector is then used to train a neural network (NN) that represents the scene.

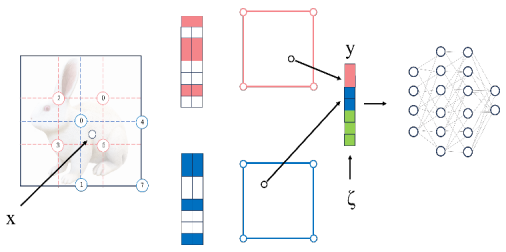

Figure 3: (1) Hashing of voxel vertices (2) Look up (3) Linear (4)Concentration (5)NN interpolation

3. Shooting Environment and Processing to Improve Resolution for NeRF

In this study, a frame is extracted from a video and divided into four images, each covering a 90-degree field of view. The original image size is 1920 × 1080 pixels. These images are then downscaled to 50 percent of their original size (960 × 540 pixels) using the Lanczos resampling method, followed by the application of a sharpening filter. After that, the processed images undergo a super-resolution process using SwinIR [11] to improve their resolution. To reduce processing time, a low-resolution filter is applied once before the super-resolution step. SwinIR is a super-resolution model based on the Swin Transformer [12], a deep learning architecture capable of hierarchically extracting image features. This model enables high-quality denoising and upscaling while compensating for the loss of high-frequency details caused by the initial segmentation of the omnidirectional image. In this study, SwinIR is used to enhance blurred edges and to improve the accuracy of camera pose estimation.

3.1. Image capturing and segmentation

In this study, the RICOH THETA X omnidirectional camera manufactured by RICOH, as shown in Fig.4, is used as the imaging device. This camera captures 360-degree omnidirectional images by combining two hemispherical images taken with ultra-wide-angle fisheye lenses mounted on the front and back of the camera. The images are then corrected for distortion, as illustrated in Fig.5. While this method offers the advantage of capturing images in all directions, it also presents several challenges. One issue is that the omnidirectional image contains a large amount of visual information, which causes a significant loss in resolution when the distance between the camera and the object increases. Additionally, due to the use of ultra-wide-angle lenses, the captured image becomes increasingly distorted toward the edges as the distance from the center grows [13]. To address these challenges, the distortion-corrected 360-degree omnidirectional image is divided into four uniformly segmented images, as shown in Fig.6. This step enables the adaptation of the captured omnidirectional imagery to the NeRF generation model, which is typically designed for standard two-dimensional camera images. In this study, a 1-minute and 14-second video was recorded. From this video, 78 frames were extracted and divided into four segments each, resulting in a total of 312 input images.

3.2. Image sharpening

In this study, a sharpening filter is applied to the input images to improve the accuracy of camera pose estimation using COLMAP. A sharpening filter enhances image shading and emphasizes the edges of object boundaries by increasing the rate of change in pixel values within the image. Since COLMAP estimates camera poses based on image features such as edges detected across multiple viewpoints, applying a sharpening filter helps to improve the

Figure 4: The camera used (RICOH THETA X)

Figure 5: Example image frame from omnidirectional video

precision of feature matching and, consequently, the accuracy of camera pose estimation. This enhancement is expected to contribute to better performance in NeRF reconstruction. The sharpening filter [14] used is shown in Fig.7. In this process, since color images are used, the sharpening filter is applied separately to each of the RGB channels. Since the application of this filter also enhances noise components in the image, it is important to address noise processing as well. In this study, after applying the sharpening filter, the images are processed using a super-resolution model, which is explained in the next section. Figures 8 and 9 show the images before and after the sharpening filter is applied, respectively. As shown in Fig.9, the edges of objects in the image are more clearly emphasized after processing.

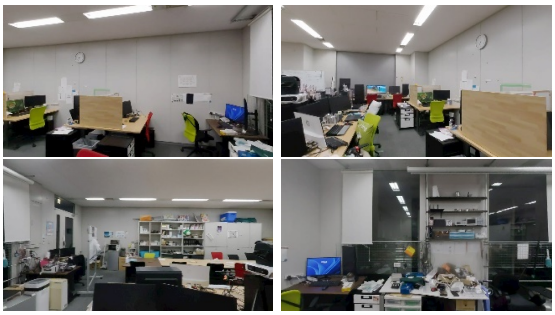

Figure 6: Example of segmented image frame from omnidirectional camera

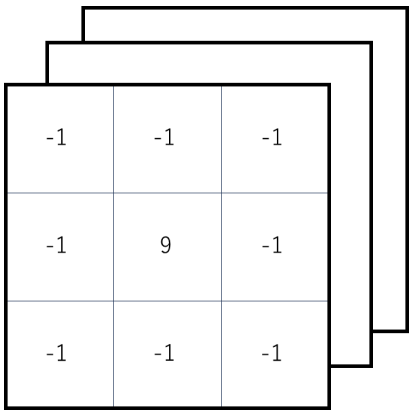

Figure 7: Image smoothing filter used

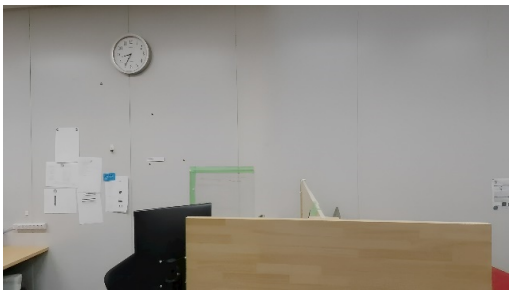

Figure 8: Before applying the sharpening filter

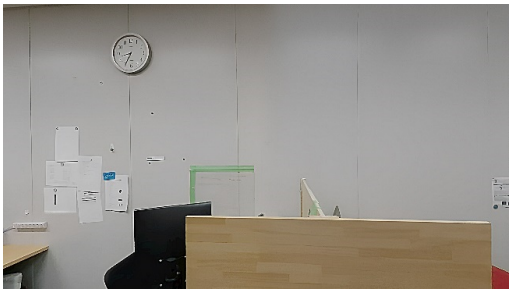

Figure 9: After applying the sharpening filter

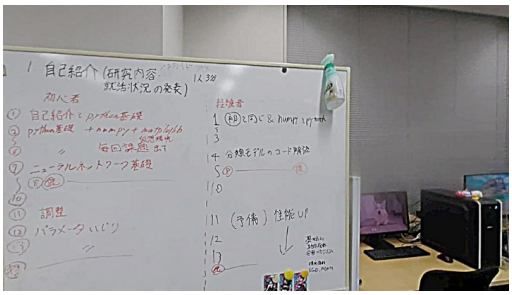

Figure 10: An example before applying super-resolution process

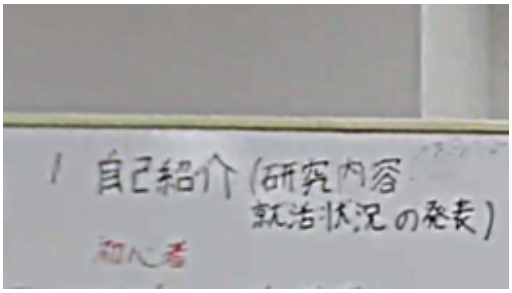

Figure 11: Cropped image part from Fig.10

3.3. Object boundary enhancement

Super-resolution techniques can be broadly categorized into two types: multiple-image and single-image high-resolution conversion. The model used in this study is a single-image super-resolution model, which is trained using a large number of paired low- and high-quality images to learn how to generate high-resolution outputs. This enables the conversion of an input image into a higher-resolution version. The development of super-resolution techniques using convolutional neural networks (CNNs) began with the introduction of SRCNN [15], which utilizes a three-layer CNN. More recently, Transformer-based models have emerged, achieving higher accuracy; however, they generally require longer processing times [9]. These models are widely used in applications such as display monitors and image generation AI. In this study, we apply super-resolution processing using SwinIR [10] to images enhanced with sharpening filters. This technique improves image quality primarily by increasing the pixel count in low-quality images. SwinIR was chosen because it provides a more accurate representation of real-world scenes compared to GAN-based models like Real-ESRGAN [16]. Furthermore, since SwinIR is Transformer-based, it demonstrates strong performance in noise reduction, allowing it to suppress the noise that often increases alongside high-frequency components. Figures 10 and 11 show an example image before applying the super-resolution process and a cropped portion of the image from Figure 10, respectively. Figures 12 and 13 present the corresponding results after applying the super-resolution process. As seen in these figures, the resolution of the entire image is enhanced, and the object boundaries, in particular, are more clearly emphasized.

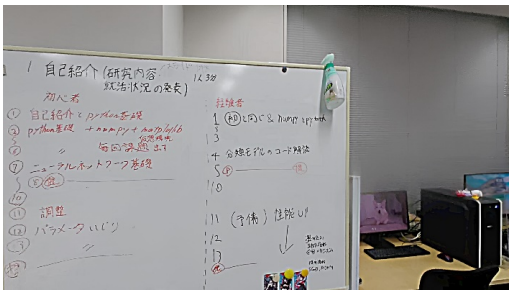

Figure 12: . After applying super-resolution process to image in Fig.10

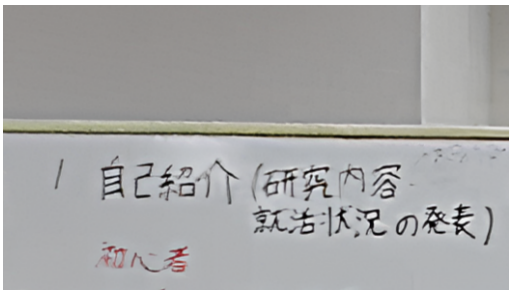

Figure 13: Same cropped image area as Fig.11 after applying the super-resolution process.

4. Implementation Experiments

4.1. Dataset preparations

A RICOH THETA X was carried around the room at a height of approximately 1 meter and used to capture video footage for about 1 minute and 14 seconds, from which 78 frame images were extracted. A dataset of 312 images was then created through segmentation, sharpening filter processing, and super-resolution processing. Table 1 shows

Table 1: Specifications of the PCs used

CPU	Intel(R)Core(TM)i7-12700CPU
memory	80G
GPU	NVIDIA GeForce RTX 3070

Table 2: Software used

OS	Windows11
Python	3.11.5
OpenCV	4.8.1.78

the specifications of the PC used for this processing, and Table 2 lists the software used.

4.2. Evaluation procedure

In this study, PSNR and SSIM are used as evaluation metrics. PSNR (Peak Signal-to-Noise Ratio) measures the ratio between the maximum possible signal power and the power of noise, and it is commonly used to assess image quality in the field of image processing. Generally, a higher PSNR value indicates lower noise levels in the image, and therefore, better image quality. However, since PSNR only accounts for overall noise levels, it may not distinguish between cases where noise is uniformly distributed and cases where significant noise is localized in a specific area. As a result, an image with a relatively low PSNR value may still appear to have high visual quality to the human eye. Equation 1 shows the calculation of PSNR, where MAX represents the maximum possible pixel value of the image, and MSE denotes the mean squared error.

$$\mathrm{PSNR} = 10 \cdot \log_{10} \left(\frac{\mathrm{MAX}^2}{\mathrm{MSE}} \right) \tag{1}$$

SSIM (Structural Similarity Index) [14] is a metric that evaluates image quality based on comparisons of luminance, contrast, and structural information. It was developed to address the limitation of PSNR in capturing perceptual differences, as images with similar PSNR values can still appear visually different to the human eye. Like PSNR, SSIM is widely used in the field of image processing. A higher SSIM value indicates greater similarity in structural aspects such as brightness and contrast, and therefore suggests better image quality. SSIM is defined by Equation 2, where l(x,y), c(x,y), s(x,y) are terms comparing luminance, contrast, and structure, respectively, and α, β, γ are positive constants

$$\mathrm{SSIM}(x, y) = [l(x, y)]^{\alpha} \cdot [c(x, y)]^{\beta} \cdot [s(x, y)]^{\gamma} \tag{2}$$

4.3. Experimental results

The NeRF space was trained for 35,000 iterations using Instant-NGP, and its accuracy was evaluated using PSNR and SSIM, depending on whether the proposed image processing was applied to the input images. Since NeRF results can vary between training runs, the average values from five independently trained NeRF models were used to improve the reliability of the evaluation.

Table 3: Accuracy of NeRF (result value after 5 times of generation)

	Without proposed processing	With proposed processing
PSNR	25.971	**26.124**
	25.867	**26.111**
	25.950	**26.108**
	25.971	**26.082**
	25.901	**26.092**
SSIM	0.88168	**0.88285**
	0.88058	**0.88337**
	0.88076	**0.88234**
	0.88193	**0.88243**
	0.88073	**0.88261**

(a) NeRF generation without applying the proposed processing

(b) NeRF generation with applying the proposed processing

Figure 14: An example of NeRF generation improvement

Table 3 presents the PSNR and SSIM values computed for NeRF spaces generated with and without the proposed image processing applied to the input images. As shown in Table 3, both PSNR and SSIM values are higher when the proposed image processing is applied. Furthermore, Figs. 14 and 15 show examples of NeRF, generated using the images without and with the proposed processing method, respectively. Specifically, the image regions inside the red rectangles show noticeable differences. These results confirm that the accuracy of NeRF image generation can be improved by incorporating the proposed processing

(a) NeRF generation without applying the proposed processing

(b) NeRF generation with applying the proposed processing

Figure 15: An example of NeRF generation improvement

method.

5. CONCLUSIONS

In this study, we aimed to improve the accuracy of NeRF by proposing the use of omnidirectional images and an input image enhancement processing method. The proposed processing enhances the input images by increasing resolution and emphasizing object boundaries, which in turn improves the accuracy of camera pose estimation and subsequently the quality of the generated NeRF space. This improvement is attributed to the enhanced visibility of edges and corners in the processed images. Our findings suggest that even with consumer-grade devices, such as general omnidirectional cameras, it is possible to generate a more accurate NeRF space by incorporating the proposed processing. We believe this approach can contribute to the broader adoption and accessibility of NeRF technology.

References

[1] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, R. Ng, Nerf: Representing scenes as neural radiance fields for view synthesis, Communications of the ACM 65 (1) (2021) 99–106.

[2] C. Deng, C. Jiang, C. R. Qi, X. Yan, Y. Zhou, L. Guibas, D. Anguelov, et al., Nerdi: Single-view nerf synthesis with language-guided diffusion as general image priors, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2023, pp. 20637–20647.

[3] L. Xu, Y. Xiangli, S. Peng, X. Pan, N. Zhao, C. Theobalt, B. Dai, D. Lin, Grid-guided neural radiance fields for large urban scenes, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 8296–8306.

[4] Y. Chen, G. Lee, DBARF: Deep bundle-adjusting generalizable neural radiance fields, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2023, pp. 24–34.

[5] C. Choi, S. Kim, Y. Kim, Balanced spherical grid for egocentric view synthesis, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2023, pp. 12663–12673.

[6] A. J. Davison, I. D. Reid, N. D. Molton, O. Stasse, Monoslam: Real-time single camera slam, IEEE transactions on pattern analysis and machine intelligence 29 (6) (2007) 1052–1067.

[7] L. Johannes, J. Michael, Pixelwise view selection for unstructured multi-view stereo, in: Proceedings of the European Conference on Computer Vision (ECCV), 2016, pp. 501–518.

[8] D. G. Lowe, Distinctive Image Features from Scale-Invariant Keypoints, International Journal of Computer Vision 60 (2) (2004) 91–110.

[9] T. Müller, A. Evans, C. Schied, A. Keller, Instant neural graphics primitives with a multiresolution hash encoding, ACM transactions on graphics (TOG) 41 (4) (2022) 1–15.

[10] D. E. Rumelhart, G. E. Hinton, R. J. Williams, Learning representations by back-propagating errors, nature 323 (6088) (1986) 533–536.

[11] J. Liang, J. Cao, G. Sun, K. Zhang, L. Van Gool, R. Timofte, Swinir: Image restoration using swin transformer, in: Proceedings of the IEEE/CVF international conference on computer vision, 2021, pp. 1833–1844.

[12] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, Swin transformer: Hierarchical vision transformer using shifted windows, in: Proceedings of the IEEE/CVF international conference on computer vision, 2021, pp. 10012–10022.

[13] S. K. Nayar, Catadioptric omnidirectional camera, in: Proceedings of IEEE computer society conference on computer vision and pattern recognition, IEEE, 1997, pp. 482–488.

[14] J. Canny, A computational approach to edge detection, IEEE Transactions on pattern analysis and machine intelligence (6) (2009) 679–698.

[15] C. Dong, C. C. Loy, K. He, X. Tang, Image super-resolution using deep convolutional networks, IEEE transactions on pattern analysis and machine intelligence 38 (2) (2015) 295–307.

[16] X. Wang, L. Xie, C. Dong, Y. Shan, Real-esrgan: Training real-world blind super-resolution with pure synthetic data, in: Proceedings of the IEEE/CVF international conference on computer vision, 2021, pp. 1905–1914.