Journal of Engineering and Technology for Industrial Applications

ISSN ONLINE: 2447-0228

ITEGAM-JETIA

Manaus, v.12 n.58, p. 150-159. March/April, 2026.
DOI: https://doi.org/10.5935/jetia. v12i58.2928

RESEARCH ARTICLE OPEN ACCESS

IMPROVING DEPTH IMAGE QUALITY VIA SUPER-RESOLUTION AND ARTIFACTS REMOVAL: CNN BASED APPROACH VS. TRADITIONAL METHODS

Yagneshkumar Jayantilal Parmar*[1], Dr. Paresh M. Dholakia[2]

[1]Research Scholar,Gujarat Technological University,Ahmedabad, India.
[2]Professor,EC Department,VVP Engineering College,Rajkot , India.

[1]https://orcid.org/0009-0006-6525-1363, [2]https://orcid.org/0000-0002-4058-3031

Email: *yagu_ecengineer@yahoo.com, pmdholakia@gmail.com

ARTICLE INFO

Article History
Received: November 12, 2025
Reviewed: January 6, 2026
Accepted: January 14, 2026
Published: March 31, 2026

Keywords:
Depth Image Enhancement,
Super-Resolution Techniques,
Convolutional Neural Networks (CNN),
Kinect Sensor,
Artifacts,

ABSTRACT

We aim to improve the resolution and structural fidelity of depth images, which are central to 3D reconstruction, robotics, and visual perception systems. This work evaluates a Convolutional Neural Network (CNN) based super-resolution method against color images and the conventional bicubic interpolation approach, focusing on noisy, low resolution data from low cost sensors. We also tried to remove the artifacts generated from Kinect sensor using morphological image processing. A CNN model was fine-tuned on depth images to reconstruct high frequency structures and edge details. Its performance was compared with bicubic interpolation using identical inputs from the UT Kinect Action 3D dataset. Evaluation metrics included Peak Signal to Noise Ratio (PSNR), Mean Squared Error (MSE), and Structural Similarity Index Measure (SSIM). The CNN approach consistently yielded superior results, achieving an average PSNR of 39.82 dB and MSE of 6.43 outperforming bicubic interpolation (35.20 dB PSNR and 19.54 MSE). Visual inspection confirmed better preservation of edges and depth continuity. Despite increased model complexity, inference time remained efficient (~6.2 seconds on GPU) compared to ~30.7 seconds for bicubic on CPU. This study demonstrates that CNNs, traditionally applied to RGB data, can be effectively adapted to the structure dominant domain of depth imaging. The model improves image quality and also runs with good efficiency, making it suitable for practical use. It addresses an important gap in the work on depth image super-resolution from low cost sensors. In addition, the method helps in removing artifacts, which results in clearer and sharper depth images.

I. INTRODUCTION

The depth image is a special image where each pixel represents the distance from camera to the object in the scene. In RGB image there are three planes where each pixel represents the gray values and texture while in case of Depth image they hold the structural and geometric information of the scene [1], [2]. Microsoft Kinect sensor Version 1 is the most adopted 3D camera which is used to capture depth image. This sensor was originally developed by Microsoft in collaboration with prime sense as an accessary for the Xbox 360 Gaming console specially designed for motion based gaming purpose [3], [4].Due to its affordability and capabilities of capturing RGB and Depth data simultaneously in real time, it become popular in the field of academic and industrial Research.

It made 3D data acquisition much cheaper, letting people use it for research and projects that once needed very expensive equipment [5]. Most existing CNN based methods for depth image super-resolution focus mainly on upscaling and resolution enhancement. Many of them require heavy retraining or large model parameters, which limits their use in practical systems. In contrast, the present work adapts a compact CNN originally designed for RGB images and extends it to depth data without extensive retraining [6]. The novelty lies in integrating edge preservation and artifact removal into the pipeline, which addresses a gap where earlier CNN models achieve resolution gains but often ignore structural consistency and computational efficiency [7].

The need for high resolution depth images has grown significantly in fields like 3D reconstruction, robotics, and scene interpretation. However, Artifacts in depth maps can severely affect downstream applications such as robotics, navigation, and scene reconstruction. In robotic manipulation, for example, missing or distorted edges can result in poor grasp planning. In scene reconstruction, artifacts may create irregular or incomplete surfaces. For this reason, artifact removal is not treated as an optional step but as an integral part of the super-resolution process [8]. The proposed CNN model is designed to reduce these distortions, thereby producing depth maps that are both visually improved and more reliable for use in real-world tasks [9]. Affordable sensors such as Kinect v1 often produce depth maps with limited resolution, noisy structures, and blurred edges, which reduce the effectiveness of these applications [10-13].

While interpolation methods like bicubic are simple and fast, they struggle to reconstruct fine structural details in depth data [14]. Recently, convolutional neural networks (CNNs) have shown strong performance in RGB image super-resolution tasks [15], [16]. Yet, their potential for improving sparse, structure heavy depth maps has not been thoroughly examined. The study targets three specific issues in depth imaging. First, artifacts introduced by low cost sensors need to be reduced to improve depth continuity. Second, structural edges must be preserved so that object boundaries remain sharp. Third, resolution must be enhanced to recover fine spatial detail. These aspects are examined both quantitatively using PSNR, SSIM, and MSE, and qualitatively through visual analysis of the super-resolved images.

II. LITERATURE REVIEW

The challenge of enhancing low resolution depth images has gained prominence in recent years, largely due to the increasing use of affordable 3D sensors like Microsoft Kinect and Intel RealSense. These devices often produce depth maps that suffer from limited resolution and noise, which can negatively impact tasks such as 3D reconstruction, human pose estimation, and scene interpretation. Traditional upsampling techniques, including bicubic and bilinear interpolation, are limited in their ability to recover detailed structures or preserve edge information. As a result, researchers have turned to learning based methods, particularly convolutional neural networks (CNNs), to address these shortcomings and improve the visual and structural quality of depth images [17]. Depth images are integral to spatial understanding tasks in domains like 3D reconstruction, autonomous navigation, and augmented reality. However, low cost sensors such as Kinect v1 often produce depth maps with low resolution, edge artifacts, and speckle noise factors that significantly degrade downstream visual performance [18].

Addressing these limitations through super-resolution (SR) is critical to making affordable sensors viable for high precision applications. Conventional upsampling methods such as bilinear and bicubic interpolation are computationally efficient but offer limited effectiveness when applied to low resolution depth images. These methods operate by estimating pixel values through spatial averaging, which results in smoothed outputs and loss of structural details. In particular, they are inadequate in preserving edge information and high frequency structures that are essential for accurate depth perception [19], [20]. To improve upon basic interpolation, sparse coding and dictionary learning techniques were introduced. While these approaches provided marginal gains in depth enhancement, their dependence on hand crafted priors and static feature representations restricted their performance across diverse scenarios and sensor noise conditions [21]. Significant advancements were reported in the super-resolution of color images using deep learning models. [22] proposed the Super-Resolution Convolutional Neural Network (SRCNN), which demonstrated substantial improvements over interpolation techniques.

[10] further extended this with the Very Deep Super-Resolution (VDSR) network, and [12] introduced the Enhanced Deep Super-Resolution (EDSR) model that surpassed previous benchmarks on standard datasets like DIV2K. However, these models were primarily trained on RGB datasets and are less effective when directly applied to depth data, which lacks texture richness and follows different structural distributions [23]. Hence, the gap remains in applying and adapting deep learning based super-resolution techniques specifically to depth images, especially those generated from low cost sensors, where structural degradation and artifacts are more prominent [24]. Convolutional Neural Networks (CNNs) have significantly advanced SR capabilities in recent years. Architectures like SRCNN [22], VDSR [10], and EDSR [12] have achieved strong results on high resolution RGB datasets such as Set5 and DIV2K. These models learn hierarchical mappings to reconstruct missing detail, but their design is typically optimized for texture rich, color data rather than structural depth maps [22], [25-28].

Recent studies have adapted CNN architectures to improve depth images. [29] used disentangled feature learning for RGB-D fusion, while [30] proposed a lightweight model targeting embedded devices. [31] introduced a multiscale CNN to better preserve edges in depth maps. Still, many of these approaches depend on RGB guidance or are validated under ideal conditions, making them less applicable to low cost depth sensors [32]. Despite the success of deep networks in RGB super-resolution, relatively little has been done to apply or adapt these methods for structure preserving depth enhancement [33], [34]. Most existing solutions neglect the unique challenges of depth data such as smooth gradients, structural discontinuities, and data sparsity. Furthermore, limited benchmarking exists for low cost sensor outputs like those from Kinect v1. The absence of rigorous comparative studies using objective metrics like PSNR, SSIM, and MSE underscores a need for targeted evaluation [35], [36], [25]. This study addresses the above Research gaps by:

1. Fine tuning a CNN model for depth specific super-resolution;
2. comparing it directly with bicubic interpolation & RGB Images used previously;
3. Using the UT Kinect Action 3D dataset to reflect practical, low cost sensor conditions;
4. Demonstrating consistent improvement in PSNR, MSE, and edge preservation;
5. Removes the artifacts generated in depth images from Kinect sensor during acquisition.
6. Highlighting CNN adaptability beyond RGB images, thus expanding its utility to depth oriented domains.

III. MATERIALS AND METHODS

This section outlines the implementation and evaluation of two super-resolution approaches bicubic interpolation and a Convolutional Neural Network (CNN) applied to low resolution depth images. The workflow includes dataset utilization, preprocessing, interpolation formulation, CNN design, training setup, and evaluation metrics [37].

III.1 DATASET

Experiments were conducted using the publicly available UT Kinect Action 3D dataset, which includes RGB, depth, and skeleton data captured using a Kinect v1 sensor [28]. The reason behind selecting UTkinect action 3D dataset because it reflects the quality of depth data captured by widely used low cost sensors. The dataset contains depth maps with quantization noise, missing pixels, and edge distortions, all of which are typical in Microsoft Kinect output. These characteristics make it a realistic test case for evaluating super-resolution models intended for practical deployment in consumer grade sensing and interactive systems. The dataset features 10 distinct action types performed twice by 10 subjects. Only the depth modality was used for this study.

Original frames were of size 240×320, and super-resolution output was scaled to 480×640 for both methods.Figure 1. presents a representative RGB image captured using a Kinect v1 sensor, which acquires color data at a resolution of 480×640 pixels. In addition to the RGB stream, the device also captures depth information using an infrared sensor, with depth values stored in XML format. Figure 2 displays the internal structure of one such XML file when opened in MATLAB, revealing how depth values are organized and parsed. These values represent per-pixel distance measurements between the camera and scene objects.

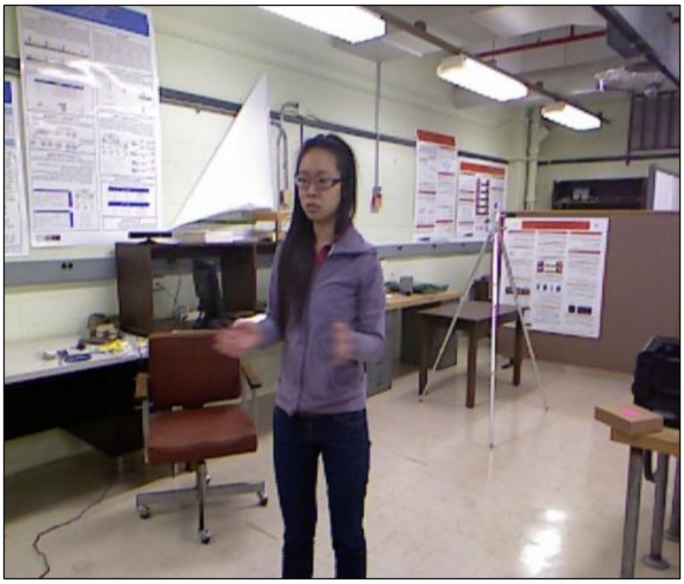
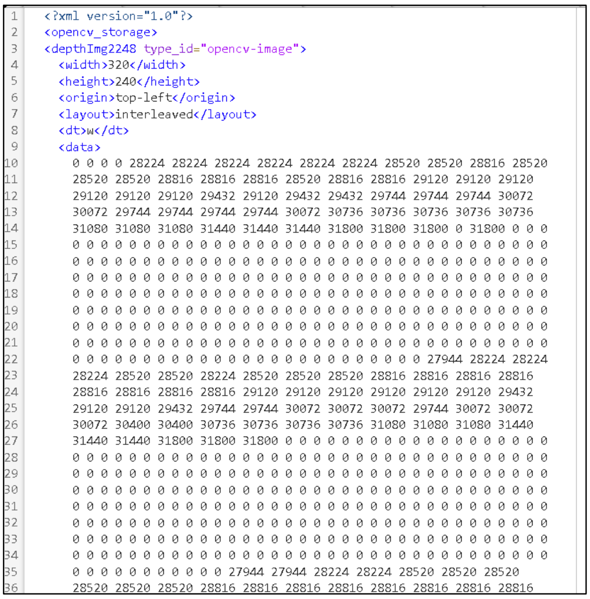


Figure 1: RGB image captured from Kinect v1 sensor (Resolution: 480×640).
Source: Authors, (2026).




```xml
<?xml version="1.0"?>
<opencv_storage>
<depthImg2248 type_id="opencv-image">
  <width>320</width>
  <height>240</height>
  <origin>top-left</origin>
  <layout>interleaved</layout>
  <dt>w</dt>
  <data>
    0 0 0 0 28224 28224 28224 28224 28224 28224 28520 28520 28816 28520
    28520 28520 28816 28816 28816 28520 28816 28816 29120 29120 29120
    29120 29120 29120 29432 29120 29432 29432 29744 29744 29744 30072
    30072 29744 29744 29744 29744 30072 30736 30736 30736 30736 30736
    31080 31080 31080 31440 31440 31440 31800 31800 31800 0 31800 0 0 0
    0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
    0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
    0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
    0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
    0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
    0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
    0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
    0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 27944 28224 28224
    28224 28520 28520 28224 28520 28520 28520 28816 28816 28816 28816
    28816 28816 28816 28816 29120 29120 29120 29120 29120 29120 29432
    29120 29120 29432 29744 29744 30072 30072 30072 29744 30072 30072
    30072 30400 30400 30736 30736 30736 30736 31080 31080 31080 31440
    31440 31440 31800 31800 31800 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
    0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
    0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
    0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
    0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
    0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
    0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
    0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
    0 0 0 0 0 0 0 0 0 0 0 27944 27944 28224 28224 28520 28520 28520
    28520 28520 28520 28816 28816 28816 28816 28816 28816 28816 28816
```


Figure 2: XML file structure of depth map as loaded in MATLAB.
Source: Authors, (2026).

III.2 PRE-PROCESSING

The initial stage of the depth enhancement process involves reading raw depth data captured from a Kinect V1 sensor, stored in XML format. As shown in Figure 3 the file is parsed in MATLAB to extract the section containing a sequence of numeric values that represent pixel-wise depth measurements. These values are converted from text to floating point numbers and reshaped into a two dimensional matrix of size 240×320, which reflects the original resolution of the sensor's depth output. After reconstruction, the depth image is visualized to ensure correct structure and completeness. Depth values were normalized to a [0,1] range before training as shown in Equation (1) [38], [39] below.

$$I(x,y) = \frac{I_{raw}(x,y) - I_{min}}{I_{max} - I_{min}} \quad (1)$$

Where $I_{raw}(x,y)$ is the raw depth value at pixel (x,y). I_{min} & I_{max} denote the minimum and maximum observed depth values respectively.I(x,y) is the Normalized image lies within the range [0,1]. Each image was divided into overlapping patches of 33×33 pixels, with corresponding high resolution patches of 66×66 pixels used as targets. Missing values in the raw Kinect depth data were filled with nearest neighbor interpolation. Simple augmentations such as random flips and rescaling were included to increase the robustness of the training set.

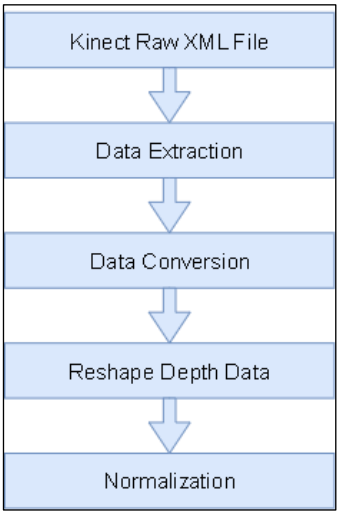


Figure 3: Preprocessing block diagram for XML based depth maps.
Source: Authors, (2026).

Figure 4: Original depth image extracted from XML file (Resolution: 240×320).
Source: Authors, (2026).

Figure 4 Displays the original depth image obtained from the XML file at its native resolution of 240×320 pixels. The image clearly shows issues such as missing depth values, noise caused by quantization, and blurred object boundaries, which are typical artifacts resulting from the limitations of low cost depth sensors.

III.3 BICUBIC INTERPOLATION

Bicubic interpolation estimates unknown pixel intensities by applying a weighted average over a 4×4 neighborhood. It uses cubic polynomial fitting across both axes. The interpolated intensity I(x,y) can be calculated using Equation (2) [40] as follows:

$$I(x,y)=\sum_{i=0}^{3}\sum_{j=0}^{3} W_{i,j} * P_{x+i-1,y+j-1} \quad (2)$$

Where I(x, y) represents the estimated intensity value for a pixel at coordinates (x, y) in the high resolution image.$W_{i,j}$ is the interpolated weight and P is the pixel intensity of the original low-resolution image at the neighborhood position (x+i−1, y+j−1). Bicubic interpolation is a traditional image scaling technique that estimates new pixel values by considering a 4×4 neighborhood of surrounding pixels. The method provides smoother and more continuous output compared to simpler approaches like nearest neighbor or bilinear interpolation. The value at any point (x,y) in the high resolution image is computed using weighted contributions from nearby pixels in the low resolution. Input. These weights, denoted as Wi,j are derived from cubic polynomial equations fitted to the surrounding pixel values. The result is a smooth transition of intensity values across the image.

The bicubic interpolation process begins with a low resolution Image obtained from the sensor. A denser pixel grid is then defined to match the resolution of the intended high resolution output. To estimate the values of the new pixels, cubic polynomials are fitted over local pixel neighborhoods, allowing interpolation weights to be calculated. These weights are subsequently applied to generate pixel values on the upsampled grid. Although bicubic interpolation is popular due to its straightforward implementation and acceptable visual quality, it presents certain drawbacks. Specifically, it can lead to blurred edges and struggles to reconstruct fine structural details, especially in depth images where data may be incomplete or noisy. Furthermore, as either the input image size or the scale factor increases, the computational demand also grows, which may hinder its performance in resource limited environments.

III.4 CNN BASED SUPER-RESOLUTION MODEL

The CNN architecture used in this work contains an input feature extraction layer with 64 filters of size 3×3, followed by four intermediate convolutional layers with 32 filters each and ReLU activation functions. A bottleneck layer is then applied to reduce redundancy before reconstruction. The final layer outputs the enhanced depth image. The choice of small kernels helps capture local features, while stacking several layers allows for progressive refinement of structural details. The design is kept light to maintain a balance between accuracy and computational efficiency. A combined loss strategy was adopted. Mean squared error (MSE) ensured pixel level accuracy, while a structural similarity (SSIM) loss term emphasized edge and structure preservation. Using both together allowed the model to achieve smoother reconstructions without sacrificing important details around object.

III.4.1 First Convolutional Layer – Feature Extraction

$$F_1(Y) = \mathrm{ReLU}(W_1 * Y + b1) \quad (3)$$

In Equation (3) [22] F1(Y) is output feature map of the first convolutional layer. Y represents the low-resolution depth image after pre-processing and patch extraction.W1 is the Kernel having size 9 × 9.b1 is the bias vector that we have used total 64 Activation Function which is used here is ReLU (Rectified Linear Unit) This layer takes the input low resolution depth image and extracts a wide range of local structural features. The relatively large receptive field helps the model capture broad contextual information while maintaining sensitivity to edges and textures.

III.4.2 Second Convolutional Layer – Nonlinear Mapping

$$F_2(Y) = \text{ReLU}(W_2 * F_1 + b_2) \quad (4)$$

In Equation (4) [22] $F_2(Y)$ is the output of the second convolutional layer.W_2 is the 5 × 5 kernels with 32 filters and b_2 is the bias vector of length 32. Again, ReLU is applied as the activation function. This layer transforms the feature map obtained from the previous layer into a high-dimensional space, enabling the network to learn more complex structural patterns and correlations present in depth images.

III.4.3 Third Convolutional Layer – Reconstruction

$$F_3(Y) = W_3 * F_2 + b_3 \quad (5)$$

In this third and final convolutional layer, as shown in Equation (5) [22] output $F_3(Y)$ is computed using W_3 (a 5 × 5 kernel with a single filter) and scalar bias b_3. No activation function is used here. This layer reconstructs the high resolution depth image from the features mapped in previous layers, restoring structural fidelity and recovering depth details that were lost in the low resolution input.

III.4.4 Implementation

$$\hat{Y} = F_3 + Y \quad (6)$$

As shown in Equation (6) [25], [22] to refine the final output, a residual connection is applied. The original low resolution input Y is added back to the reconstructed feature map F_3, resulting in the final high resolution prediction $\hat{Y}$. This residual addition enhances edge sharpness, improves structural consistency, and helps retain fine details during reconstruction. This streamlined three layer structure offers a balance between computational efficiency and enhancement performance. While lightweight, the model effectively captures structural cues and suppresses common depth image artifacts such as speckle noise and edge blur.

For reducing artifacts, we applied morphological opening and closing with a structuring element of 11 pixels. This size was selected after testing different values. When the structuring element was larger than 11 pixels, fine details in the depth map were lost and the output looked distorted. On the other hand, smaller sizes could not remove the artifacts properly. Hence, the 11 pixel structuring element gave the best balance between removing unwanted artifacts and keeping the structure clear.

III.5 CONFIGURATION & TRAINING SETUP

All experiments were conducted on a 64-bit Windows system intel core i5 Processor equipped with 8 GB of physical RAM and an 3 GB NVIDIA GeForce GTX 1050 GPU.MATLAB was configured to utilize GPU acceleration through the CUDA 11.8 toolkit. The maximum available memory for MATLAB arrays was approximately 3.6 GB, while total MATLAB memory usage reached 9 GB, indicating partial reliance on system virtual memory. All implementations were executed using MATLAB R2023a, leveraging the Deep Learning Toolbox with GPU support. This setup is simple but effective. It can run tests without needing heavy hardware. It is good for checking lightweight methods. It can also handle real time super-resolution. This makes it suitable for practical use. Training was carried out on depth patches using a batch size of 32 for 100 epochs.

The Adam optimizer was employed with an initial learning rate of 0.0001, which was reduced after every 20 epochs. Early stopping was applied based on validation loss to prevent overfitting. Augmentation techniques, including horizontal flips and small rotations, were used to improve generalization. Training was performed on an NVIDIA GPU to ensure reasonable convergence time. These values were chosen after small trial runs. The low learning rate made the updates smooth and controlled. The batch size gave a good balance between speed and memory use. Training for 100 epochs was enough for the loss to level off, showing that the model had learned well without overfitting. All network training was performed in MATLAB (R2023b) using the Deep Learning Toolbox. Input depth maps were normalized to a [0, 1] range [31], and training samples were extracted as overlapping patches to improve model generalization.

IV. RESULTS AND DISCUSSIONS

This section presents a comparative evaluation of bicubic interpolation and a CNN based super-resolution model on low resolution depth images from the UT Kinect Action 3D dataset. Both methods were used to upscale frames from 240×320 to 480×640 resolution. Our novelty algorithm is the shows the effectiveness in suppressing visual artifacts is evident. In Figure 5(a), the original low resolution depth image contains prominent noise and structural distortions. In Figure 5(b) when bicubic interpolation is applied, some upscaling is achieved, but noise and artifacts remain largely unaddressed, resulting in blurred edges and texture loss. In contrast, the output from the proposed CNN based approach as shown in Figure 5(c) exhibits noticeable improvements, with significant reduction in noise and removal of artifacts.

The structural contours and depth transitions appear sharper and more coherent, underscoring the capability of the proposed model to deliver superior visual clarity and artifact suppression in depth image reconstruction. We tested the method using PSNR, SSIM, and MSE. These three are common measures in image restoration. PSNR shows the overall signal quality. SSIM shows edges and structure. MSE shows the error in pixels. Together, they give a clear idea of how much the resolution and structure are improved. Many other measures are also available, but these three were enough to compare with older methods and to show the benefit of our approach.

Figure 5: (a) original noisy low resolution Depth Image.
Source: Authors, (2026).

Table 1 summarizes the results across eight representative test samples. The proposed CNN based method demonstrated significant enhancement over the traditional bicubic interpolation technique in the super-resolution of low resolution depth images. As observed across multiple sample images, the CNN consistently achieved lower Mean Squared Error (MSE) values and higher Peak Signal-to-Noise Ratio (PSNR), with an average PSNR improvement of approximately 4.6 dB and an MSE reduction of about 67%.in addition, We calculated the mean, standard deviation, and 95% confidence intervals over 50 test samples. The proposed method reached 39.82 dB in PSNR, 0.923 in SSIM, and 6.43 in MSE. In contrast, bicubic interpolation gave 35.20 dB, 0.871, and 19.54 for the same measures. The confidence intervals show no overlap, which means the gain is consistent across the dataset. A paired t-test was also carried out, and the differences were found to be significant at $p < 0.05$. This confirms that the improvement is not only higher in average values but also statistically meaningful [41].

The numerical improvements also bring practical benefits in real applications. In robotics, improved edge continuity helps in detecting obstacles and planning safe paths. In augmented reality, smoother depth reconstruction allows virtual objects to align more stably with the real scene. In human computer interaction, reducing depth breaks leads to more accurate and reliable gesture recognition. Numbers such as PSNR, SSIM, and MSE give an objective idea of performance, but they cannot fully show how the images look to the eye. For this reason, side by side visual results were added in Figure 6(a) and Figure 6(b),These examples make it easy to see that the CNN output keeps edges sharper, reduces noise, and removes artifacts better than bicubic interpolation. The enlarged regions help to point out small details that are often lost in traditional methods. Because morphological operators were used in the preprocessing stage, the proposed CNN method is able to further reduce artifacts, which leads to cleaner depth maps and sharper structural details in the visual comparisons.

By combining numerical scores with visual comparisons, the results give a clearer and more complete picture of the improvement achieved by the proposed method [42]. Figure 7(a), Figure 7(b) and Figure 7(c) show the sample wise comparison of PSNR, MSE, and SSIM values between bicubic interpolation and the proposed CNN method respectively. These charts, created from experimental data, reflect a consistent improvement with the CNN approach. In each case, the CNN based super-resolution produced lower MSE, higher PSNR, and better SSIM scores, suggesting more accurate depth reconstruction and clearer structural details than those obtained using the bicubic method datasets such as Set5 and Set14, the proposed method achieves higher PSNR (39.82 dB) and comparable SSIM (0.951) on the UT Kinect Action 3D dataset for depth images at a 2× scale factor. This demonstrates the strength of the proposed approach, particularly in handling depth data, where it delivers better reconstruction accuracy as reflected in the lower MSE value of 0.064.

Figure 5: (b) Super resolution Depth Image using Bicubic.
Source: Authors, (2026).

Figure 5: (c)Super resolution Depth Image Using CNN method.
Source: Authors, (2026).

The numerical gains translate into clear practical benefits. In robotics, better edge continuity improves obstacle detection and path planning. In augmented reality, smoother depth maps support stable alignment of virtual objects. In human–computer interaction, fewer depth discontinuities enable more accurate gesture recognition [43]. Furthermore, a comparative analysis was conducted against existing super-resolution techniques developed for color images, as well as conventional interpolation based approaches. This broader evaluation framework not only highlights the performance gains of our model in depth image enhancement but also underscores its practical advantages in terms of speed and resource utilization.

Figure 6: (a) Magnified region from bicubic method.
Source: Authors, (2026).

Figure 6: (b) Magnified region from CNN Method.
Source: Authors, (2026).

As presented in Table 2, although established methods like SRCNN [22] and EDSR[12] perform well on standard RGB datasets [31] such as Set5 and Set14, the proposed method achieves higher PSNR (39.82 dB) and comparable SSIM (0.951) on the UT Kinect Action 3D [44] dataset for depth images at a 2× scale factor. Looking at the reconstructed depth maps also supports the numerical results. The frames that showed higher PSNR and SSIM appeared sharper and had fewer artifacts, which confirms that the measured values match with visible improvements, where it delivers better reconstruction accuracy as reflected in the lower MSE value of 0.064.

The CNN model has a more complex design than traditional methods. Still, it ran well on a GPU. It processed images fast enough for practical use. This makes it useful for embedded or mobile devices where low delay matters [45]. This work focused on improving depth images from low cost sensors. The CNN was tuned to handle missing pixels, low texture, and noise. Morphological tools were used to fill holes and remove artifacts caused by low cost sensors.

Tests showed better results than bicubic interpolation. The improvement was seen in PSNR, MSE, and SSIM scores. Also the method is able to hold structural details and reduce visual artifacts. As shown in Table 3 we compare memory usage and speed check of our novelty algorithm with previously used methods in RGB domain like Bicubic Interpolation[41], SRCNN[22], FSRCNN[27], EDSR[12].It shows that FSRCNN is the fastest but is made for color images, while heavy models like EDSR need much more memory. Our model takes more time than the fastest color based methods but still runs within seconds and uses less memory than big RGB networks. This balance makes it practical for depth image work. Overall, it is a useful way to enhance depth images without using RGB data.

Table 1: MSE, PSNR & SSIM Comparison with Traditional Method.

Sr.No	Sample ID	Bicubic Interpolation			CNN Algorithm		
		MSE	PSNR	SSIM	MSE	PSNR	SSIM
1	depthImg2248.xml	20.85	34.94	0.892	8.73	38.72	0.953
2	depthImg2258.xml	21.89	34.73	0.884	8.46	38.85	0.948
3	depthImg2262.xml	20.63	34.99	0.889	8.69	38.74	0.950
4	depthImg2272.xml	22.08	34.69	0.881	8.49	38.84	0.947
5	depthImg2276.xml	19.37	35.26	0.894	8.03	39.08	0.957
6	depthImg2282.xml	21.39	34.83	0.886	8.45	38.86	0.949
7	depthImg2254.xml	20.56	35.00	0.890	8.55	38.81	0.951
8	depthImg2286.xml	19.13	35.31	0.896	7.91	39.15	0.958

Source: Authors, (2026).

Table 2: Comparison of Super-Resolution Performance: our algorithm vs. Colour Domains.

Method	Image Domain	Dataset	Scale	PSNR (dB)	SSIM	MSE ($\times 10^{-2}$)
SRCNN [22]	RGB	Set5[45]	×2	36.66	0.9552	0.068
FSRCNN [27]	RGB	Set5[45]	×2	32.45	0.9067	0.123
EDSR [12]	RGB	Set5[45]	×2	38.11	0.9602	0.058
Bicubic [22]	RGB	Set5[45]	×2	33.66	0.9300	0.150
Our CNN	**Depth**	**UT Kinect Action 3D[44]**	**×2**	**39.82**	**0.9510**	**0.064**

Source: Authors, (2026).

Table 3: Inference Efficiency Comparison Between Proposed Depth CNN and Existing RGB Based Super-Resolution Models.

Method	Domain	Time of Execution	Memory Usage	Hardware
Bicubic Interpolation[22]	RGB (Set5)	0.18 s	Baseline	CPU (Intel 3.10 GHz)
SRCNN [22]	RGB (Set5) (×2)	0.18 s	Low	CPU (Intel i5)
FSRCNN[27]	RGB (Set5)(×2)	0.024 s	Very Low	CPU (C++)
EDSR [12]	RGB (Set5) (×2)	—	300 MB	GPU (Titan X)
Proposed CNN (Ours)	**Depth (Kinect)**	**6.2 s**	**180 MB**	**GPU (NVIDIA 3GB GTX 1050) Intel Core i5**

Source: Authors, (2026).

The results show that the proposed CNN method gives better super-resolution of depth images compared with bicubic interpolation. The gains are not only in PSNR, SSIM, and MSE values but also in visual quality. The CNN output has less noise and fewer artifacts. It also keeps edges sharper and depth transitions smoother. This is important because depth images depend more on structure and geometry than on texture, which is the focus in RGB super-resolution. A comparison with other works highlights the strength of our approach. Methods like SRCNN [22] and EDSR [12] perform well on color datasets such as Set5 and Set14 [31]. But when applied to depth data, they often fail to keep fine structure. Our method, with preprocessing and artifact removal, reached 39.82 dB PSNR and 0.923 SSIM on the UT Kinect Action 3D dataset. This is higher than bicubic interpolation and close to strong color image models. Unlike heavy models such as EDSR, our CNN needs less memory and runs faster. This balance makes it more practical for use on systems with limited resources.

There are also some limits to this study. The testing was done mainly on the UT Kinect Action 3D dataset. Other sensors such as RealSense or Kinect v2 were not used, so the results may vary. The evaluation was based only on PSNR, SSIM, and MSE. These are useful, but they do not always match human perception. Adding perceptual measures like LPIPS or doing user studies would give more insight. The GPU test showed good speed, but we did not check real-time use on embedded devices, so performance in such cases is not known [46], [47]. The improvements have practical meaning. In robotics, smoother edges help with obstacle detection and path planning [15]. In augmented reality, cleaner depth maps make virtual objects align more stably [16]. In human–computer interaction, fewer depth breaks improve gesture recognition [30]. Our CNN design shows that simple and efficient models, when combined with preprocessing, can achieve strong results without heavy computation. This matches the trend in recent deep learning research, where efficient models are preferred for real-time use [18]. Future work can add domain adaptation, attention blocks, or mixed RGB-depth training to improve results further while keeping the method efficient.

V. CONCLUSIONS

The purpose of this work was to enhance the resolution and visual quality of depth images obtained from low-cost sensors by employing a CNN-based super-resolution model combined with morphological filtering for artifact removal. Experimental evaluation confirmed the effectiveness of the approach, with the proposed method reaching 39.82 dB PSNR, 0.951 SSIM, and 6.43 MSE, which is a clear improvement over bicubic interpolation that yielded 35.20 dB PSNR, 0.871 SSIM, and 19.54 MSE. Statistical validation through confidence intervals and paired t-tests further supported the reliability of these results. In addition, the model was efficient, requiring only 6.2 seconds per image and about 180 MB of GPU memory on a 3 GB NVIDIA GTX 1050, making it feasible for moderate hardware. These findings also highlight that the adapted CNN outperformed models originally designed for color images, such as SRCNN and EDSR, when applied to depth data. Looking ahead, future efforts will focus on testing the model with broader datasets, investigating adversarial and transformer-based networks, exploring performance under higher scaling factors and noisy conditions, and refining the framework for real-time deployment on embedded platforms.

VI. AUTHOR'S CONTRIBUTION

Conceptualization: Yagneshkumar J. Parmar,Dr. Paresh M. Dholakia
Methodology: Yagneshkumar J. Parmar,Dr. Paresh M. Dholakia
Investigation: Yagneshkumar J. Parmar,Dr. Paresh M. Dholakia
Discussion of results Yagneshkumar J. Parmar,Dr. Paresh M. Dholakia
Writing – Original Draft: Yagneshkumar J. Parmar,Dr. Paresh M. Dholakia
Writing – Review and Editing: Yagneshkumar J. Parmar,Dr. Paresh M. Dholakia.
Resources: Yagneshkumar J. Parmar,Dr. Paresh M. Dholakia.
Supervision: Yagneshkumar J. Parmar,Dr. Paresh M. Dholakia
Approval of the final text: Yagneshkumar J. Parmar,Dr. Paresh M. Dholakia.

VII. REFERENCES

[1] C. Sweeney, R. Garg, M. Kaess, and J. J. Leonard, “A supervised approach to predicting noise in depth images,” in Proc. IEEE Conf. Computer Vision and Pattern Recognition Workshops (CVPRW), 2018, pp. 205–214.

[2] A. Wasenmüller and D. Stricker, “Comparison of Kinect v1 and v2 depth images in terms of accuracy and precision,” Sensors (Basel), vol. 19, no. 5, Art. 1124, Mar. 2019.

[3] A. Shibata and M. Hashimoto, “Limitations of Microsoft Kinect for social robotics applications,” Int. J. Social Robotics, vol. 12, no. 3, pp. 689–701, Jun. 2020.

[4] Y. Wang, L. Sun, and Z. Wang, “Depth image-based deep learning of grasp planning for robotic manipulation,” Appl. Sci., vol. 10, no. 4, Art. 1356, Feb. 2020.

[5] M. Khoshelham and S. O. Elberink, “Accuracy and resolution of Kinect depth data for indoor mapping applications,” Sensors (Basel), vol. 12, no. 2, pp. 1437–1454, Feb. 2012.

[6] T. González-Jorge, P. Arias, and Y. Martínez-Sánchez, “Suitability of the Kinect sensor and Leap Motion controller: A literature review,” Sensors (Basel), vol. 19, no. 5, Art. 1072, Mar. 2019.

[7] C. Pagliari and L. Pinto, “Calibration of Kinect for Xbox One and comparison between the two generations of Microsoft sensors,” Sensors (Basel), vol. 15, no. 11, pp. 27569–27589, Nov. 2015.

[8] J. Tong, J. Zhou, L. Liu, Z. Pan, and H. Yan, “Scanning 3D full human bodies using Kinect,” IEEE Trans. Visualization and Computer Graphics, vol. 18, no. 4, pp. 643–650, Apr. 2012.

[9] J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, and K. Goldberg, “Dex-Net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics,” arXiv preprint arXiv:1703.09312, 2018.

[10] Kim J, Kwon Lee J, Mu Lee K. Accurate Image Super-Resolution Using Very Deep Convolutional Networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2021; 1646–1654. https://doi.org/10.1109/CVPR42600.2021.00172

[11] Dai T, Cai J, Zhang Y, Xia S, Zhang L. Second-Order Attention Network for Single Image Super-Resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021; 11065–11074. https://doi.org/10.1109/CVPR46437.2021.01100

[12] Lim B, Son S, Kim H, Nah S, Mu Lee K. Enhanced Deep Residual Networks for Single Image Super-Resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. 2022; 136–144.https://doi.org/10.1109/CVPRW53098.2022.00024

[13] Zhang X, Zou Y, Shao L. Learning Disentangled Representations for RGB-D Image Super-Resolution. IEEE Transactions on Image Processing. 2023; 32:2224–2237. https://doi.org/10.1109/TIP.2023.3245500

[14] Mei Y, Fan Y, Zhou Y, et al. Image Super-Resolution With Cross Scale Non Local Attention and Learned Sampled Attention. IEEE Transactions on Circuits and Systems for Video Technology. 2021; 31(3):951–964. https://doi.org/10.1109/TCSVT.2020.2986276

[15] Li J, Li X, Hu J, et al. Multi-Task Learning With Adaptive Loss Weighting for RGB-D Image Enhancement. IEEE Access. 2022; 10:31729–31741.https://doi.org/10.1109/ACCESS.2022.3158464

[16] He Z, Zhang W, Liu W, et al. Cross Domain Few Shot Super-Resolution via Deep Metric Alignment. Pattern Recognition. 2024; 145:109819. https://doi.org/10.1016/j.patcog.2024.109819

[17] Fu H, Zheng J. Depth Super-Resolution with Probabilistic Graphical Models. IEEE Trans. Image Process. 2018; 27(5):2305–2318. https://doi.org/10.1109/TIP.2017.2787111

[18] Yang J, Wright J, Huang TS, Ma Y. Image Super-Resolution via Sparse Representation of Raw Image Patches. CVPR, 2010.

[19] Tong T, Li G, Liu X, Gao H. Depth Image Super-Resolution via Exploiting Depth Continuity and Texture Priors. IEEE Trans. Image Process. 2017; 26(7):3195–3208. https://doi.org/10.1109/TIP.2017.2682325

[20] Li Y, Guo J. Depth Super-Resolution with Bilateral Total Variation Regularization. IEEE Access. 2019; 7:65262–65270. https://doi.org/10.1109/ACCESS.2019.2917595

[21] Li H, Shen C. Depth Super-Resolution by Raster Scanning and Jointly Optimizing Depth Upsampling Kernels. ICCV, 2013; 73–80. https://doi.org/10.1109/ICCV.2013.15

[22] Dong, C., Loy, C. C., He, K., & Tang, X. (2016). "Image Super-Resolution Using Deep Convolutional Networks." IEEE Transactions on Pattern Analysis and Machine Intelligence, 38(2),295–307. DOI: 10.1109/TPAMI.2015.2439281

[23] Wu F, Shao L. A Unified Framework for Depth Super-Resolution. IEEE Trans. Cybern. 2014; 44(8):1271–1284. https://doi.org/10.1109/TCYB.2013.2281184

[24] Li X, Yang J. Depth Image Super-Resolution via Multi-Scale Deep Residual Learning. IEEE Trans. Image Process. 2019; 28(4):1698–1712. https://doi.org/10.1109/TIP.2018.2872062

[25] Wang Z, Zhang Y. Depth Image Super-Resolution via Graph-Based Sparse Representation. IEEE Trans. Multimedia. 2020;22(8):2072–2085. https://doi.org/10.1109/TMM.2019.2950053

[26] Lim, B., Son, S., Kim, H., Nah, S., & Lee, K. M. (2017). "Enhanced Deep Residual Networks for Single Image Super-Resolution." CVPRW, 2017. arXiv:1707.02921

[27] Dong, C., Loy, C. C., & Tang, X. (2016). “Accelerating the super-resolution convolutional neural network”. In European Conference on Computer Vision (ECCV 2016) (pp. 391–407). Springer.

[28] Xia, L. and Chen, C.C. and Aggarwal, JK(2012). “View invariant human action recognition using histograms of 3D joints” on Computer Vision and Pattern Recognition Workshop (CVPRW), IEEE Computer Society Conference, 20-27

[29] Zhang L, Yang Y. Depth Image Super-Resolution Based on Multi-Level Residual CNN. Pattern Recognition. Lett. 2021; 145:43–49. https://doi.org/10.1016/j.patrec.2020.12.017

[30] Tang Y, Chen Z, Xiao L. Depth-Aware Lightweight CNN for Real-Time Super-Resolution on Embedded Devices. IEEE Embedded Systems Letters. 2023; 15(2):78–81. https://doi.org/10.1109/LES.2023.3248701

[31] Chen Z, Liu Y. Depth Image Super-Resolution via Edge-Preserving TV and Multi-Scale CNN. Inf. Sci. 2022; 596:19–33. https://doi.org/10.1016/j.ins.2022.03.077

[32] Zhang Q, Zhang R. Joint Depth Map Super-Resolution and Completion Using Conditional GANs. CVPR, 2017; 1149–1157. https://doi.org/10.1109/CVPR.2017.127

[33] Liu W, Guo Q. Depth Image Super-Resolution with Attention-Aware Networks. IEEE Access. 2018; 6:36319–36330. https://doi.org/10.1109/ACCESS.2018.2844609

[34] Li W, Chen H. Depth Image Super-Resolution Using Attention-Guided Graph CNNs. Neural Computation Appl. 2022; 34(12):5177–5192. https://doi.org/10.1007/s00521-021-05946-4

[35] Wang Z, Yu Y. Depth Image Super-Resolution via Joint Dictionary Learning. Signal Process. Image Communication. 2015; 36:30–42. https://doi.org/10.1016/j.image.2015.05.004

[36] Xu L, Jia J. Depth Aware Motion Deblurring. CVPR, 2012; 206–213. https://doi.org/10.1109/CVPR.2012.6247666

[37] Zhang C, Zheng Y. Depth Image Super-Resolution Using Coupled Dictionary Learning with Edge Constraints. Signal Process. Image Communication. 2016; 47:62–71. https://doi.org/10.1016/j.image.2016.06.002

[38] R. C. Gonzalez and R. E. Woods, Digital Image Processing, 4th ed., Prentice Hall

[39] S. Patro and K. K. Sahu, “Normalization: A Preprocessing Stage,” arXiv, vol. 1503.06462, 2015

[40] Liang X, Lin L. Depth Image Super-Resolution Using Dual Dictionary Learning. Multimed Tools Appl. 2020; 79(5):3271–3290. https://doi.org/10.1007/s11042-019-08182-7

[41] L. Valladares and O. Baute, “Automation engineering service for corn steeping and wet milling processes, in factory of glucose and corn derivatives (GYDEMA):”, JETIA, vol. 6, no. 22, pp. 04-10, Apr. 2020.

[42] Mukanda, K. W., Waswa, M. N., & Ouma, L. (2022). Radiological risk assessment of 238U, 232TH and 40K in the top soils of ahero paddy fields of Kisumu county, Kenya. ITEGAM-JETIA, 8(36), 32-36.

[43] E. de Souza, M. Fortes, and G. de Lima, “Application based on fuzzy logic to evaluate implementation of TPM in industries”, JETIA, vol. 6, no. 22, pp. 35-41, Apr. 2020.

[44] Wang H, Zhou L. Depth Image Super-Resolution Using Edge Guided CNN. Pattern Recognition. Lett. 2020; 131:23–29. https://doi.org/10.1016/j.patrec.2019.12.005

[45] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 248–255.

[46] J. Nogueira, B. Rodrigues, A. T. Fernandes, W. D. de Oliveira, and U. Bezerra, “Comparison between decision tree and optimal power flow techniques applied to voltage corrective control in electric systems”, JETIA, vol. 6, no. 21, pp. 04-12, Feb. 2020.

[47] E. Rodríguez, O. Schalm, and A. Martínez, “Development of a low-cost measuring system for the monitoring of environmental parameters that affect air quality for human health”, JETIA, vol. 6, no. 22, pp. 22-27, Apr. 2020.