IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 24 July 2023, accepted 7 August 2023, date of publication 21 August 2023, date of current version 11 September 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3307198

APPLIED RESEARCH

Conditional Generative Adversarial Network Model for Conversion of 2 Dimensional Radiographs Into 3 Dimensional Views

NITESH PRADHAN[1], VIJAYPAL SINGH DHAKA[2], (Member, IEEE), GEETA RANI[2], (Member, IEEE), VIVEK PRADHAN[3], EUGENIO VOCATURO[4,5], (Member, IEEE), AND ESTER ZUMPANO[4,5]

[1]Department of Computer Science and Engineering, The LNM Institute of Information Technology, Jaipur, Rajasthan 302031, India
[2]Department of Computer and Communication Engineering, Manipal University Jaipur, Jaipur 303007, India
[3]Department of Orthopedics, CKS Hospital, Jaipur 302013, India
[4]Department of Informatics, Modeling, Electronics and Systems (DIMES), University of Calabria, 87036 Arcavacata, Italy
[5]CNR NANOTEC, National Research Council, 87036 Rende, Italy

Corresponding author: Geeta Rani (geeta.rani@jaipur.manipal.edu)

This work was supported by the Department of Informatics, Modeling, Electronics and Systems (DIMES), University of Calabria, under Grant SIMPATICO_ZUMPANO.

ABSTRACT The inefficacy of 2-Dimensional techniques in visualizing all perspectives of an organ may lead to inaccurate diagnosis of a disease or deformity. This raises a need for adopting 3-Dimensional medical images. But, the high expense, exposure to a high volume of harmful radiations, and limited availability of machinery for capturing images are limiting factors in implementing 3-Dimensional medical imaging for the whole populace. Thus, the conversion of 2-Dimensional images to 3-Dimensional images gained high popularity in the field of medical imaging. However, numerous research works offer the potential for the reconstruction of 3-Dimensional images. But, none of these provides the visualization of all angles of view from 0° to 360° for a 2-Dimensional input image such as X-ray and dual-energy X-ray absorptiometry. Also, these techniques fail to handle noisy and deformed input images. The purpose of this research is to propose a tailored Conditional Adversarial Network Model for the translation of 2-Dimensional images of bones into their corresponding 3-Dimensional view. The model is preceded by pre-processing techniques for dataset cleaning, noise removal, and converting the dataset to a uniform format. Further, the efficacy of the model is improved by determining the optimal values of model parameters, employing the customized activation function, and optimizers. Additionally, the visual quality of the generated 3-Dimensional images is evaluated to showcase the degree of quality degradation while translating. The experimental results obtained on the real-life datasets collected from hospitals across India prove the efficacy of the proposed model in generating 3-Dimensional images. The generated images are similar in quality to the input image and also effective in retaining the information available in an input image.

INDEX TERMS Conditional generative adversarial network, deep learning, X-ray, 3-dimensional view, 2-dimensional imaging.

I. INTRODUCTION

Early and correct diagnosis of a disease or deformity is an important step before the treatment of patients. An inaccurate diagnosis can lead to the wrong treatment of disease and may increase the death rate [1]. The introduction of medical imaging techniques such as Computerized Tomography scan (CT scan), Magnetic Resonance Imaging (MRI), Ultrasound rays, and Dual-energy X-ray Absorptiometry (DXA), etc. have improved the visualization of the anatomy of organs and tissues. Therefore, these have become important for the accurate diagnosis of an infection or deformity in one or more body parts. The medical imaging techniques are broadly categorized into 2-Dimensional (2-D) imaging such as X-ray and

The associate editor coordinating the review of this manuscript and approving it for publication was Sung-Min Park.

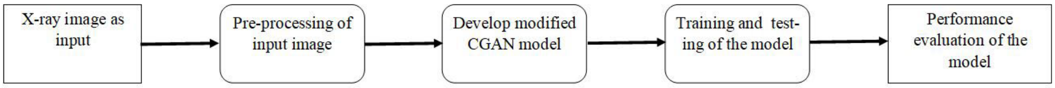

X-ray image as input
Pre-processing of input image
Develop modified CGAN model
Training and testing of the model
Performance evaluation of the model

FIGURE 1. Work flow of the proposed system.

DXA and 3-Dimensional (3-D) imaging such as CT scan and MRI [2]. The 2-D imaging techniques display a 3-D structure into a 2-D form. So, the horizontal and vertical displacement of the structure is based on its distance from the film [2]. 2-D imaging also causes the superimposition of the left and right-hand sides of the structure. Thus, 2-D images give low precision in locating the deformities in organs. Also, these techniques fail to mark the well-defined boundaries of organs and the depth of infection or deformity. The 2-D images also fail to display the precise anatomical structure of an organ [3]. On the other hand, 3-D imaging techniques display the 3-D structure of an organ. Thus, these techniques overcome the above-stated drawbacks and provide better visualization of an organ. The CT scan imaging produces multiple slices of an image. For the correct diagnosis of a disease or deformity, the system applies a reconstruction technique and produces a complete image of an organ [2]. The lesion far from the cross-section captured in the CT scan may be ignored [4]. Also, the cost of CT scan imaging is high. Due to the high cost of machinery, CT scan imaging is not available in each health center. Another popularly used modality for 3-D imaging is MRI scanning. It is also successful in overcoming the drawbacks of 2-D imaging techniques. But, it is more time-consuming than X-rays and CT-scan. It also requires sedation in children [5]. Further, MRI scanning is not feasible for patients with implanted organs. The cost of MRI imaging is even higher than CT scan imaging [6].

The discussion in [7] and [8] shows that 3-D images provide better visualization than 2-D images for a fractured or damaged part. But, the high cost and low availability of 3-D imaging techniques raise the need for low-cost technological alternatives of presenting the 3-D view of an organ. In this manuscript, the authors propose a tailored Conditional Adversarial Network (CGAN) Model for the translation of 2-D images of bones into their corresponding 3-D view. They apply pre-processing techniques on input images for noise removal, conversion to a uniform format, resizing, and normalization etc. They pass the pre-processed 2-D X-rays images to the CGAN model for conversion into their corresponding 3-D view. The workflow of the proposed system is shown in Figure 1.

The major objectives of this manuscript are as follows:

1) To apply pre-processing techniques to X-ray images collected from different sources.
2) To develop a system for the conversion of X-ray images into their equivalent 3-D images.
3) To develop a system for displaying the desired angle view from 0° to 360° for the converted image.
4) To provide a low-cost alternative to 3-D imaging techniques.
5) To ensure the good visual quality and information preservation of the generated 3-D images.

II. RELATED WORKS

The importance of converting 2-D images to 3-D images attracted researchers to develop new techniques and improve the existing techniques of conversion. The researchers applied Direct Linear Transformation (DLT) [9], Free From Deformation (FFD) [10], Statistical Shape Model (SSM) [11], Non-Stereo Corresponding Points (NSCP) algorithm [12], Deep Convolutional Neural Network (DCNN) [13], Laplacian Surface Deformation (LSD) [14], Iterative Closest Point (ICP) algorithm [15], and Partial Least Square Regression (PLSR) [16] for the conversion of 2-D to 3-D images.

Wei et al. presented the comparison of CT scans and X-ray images in the work proposed in [17]. Based on the comparison, they claimed that X-ray imaging is preferred over CT scan due to its low cost and less exposure to harmful radiations [17]. But, the visualization of disease in CT scan is better than in X-ray images. So, they proposed the 3-D recreation system for femoral shaft shape. To serve this purpose, they used the numerical morphology strategy for boundary detection. They identified the central point of the femur shaft edge and computed its three coordinates using a stamp point in the two headings of the X-ray image.

Similarly, Le Bras et al. in [18] applied 3-D CT scan remaking and 3-D stereo radiographic reconstruction methods for the reconstruction of the proximal femur. They considered the parameters such as image procurement, determination of volume, and scout view for the recreation. The authors used the Slice Omatic software to obtain the 3-D view of a femur. They used the Non-Stereorediography Corresponding Countur (NSCC) algorithm for developing the 3-D shape of a femur. The performance evaluation of these methods on 25 proximal femur images shows that the stereo radiographic reconstruction method reports the mean P2S error lower than 2.0 mm. To reduce the P2S error, Akkoul et al. proposed a model for 3-D proximal femur surface recreation [19]. They used the pseudo-stereo matching procedure. They used three cadaveric proximal femora scanned with CT scan and X-ray imaging. The model completes the recreation into seven steps. In the first step, the model uses the projection display to identify the angle between two X-ray images of a femur. In the second step, it uses active contours for determining the boundary of the femur. At the next step, it finds the coordinating points between 2-D shapes. The authors applied the city-obstruct, the chess-board, and euclidian 2-D spatial separation for finding the coordinating points. They claimed that euclidian separation gives the best accuracy. The authors employed Thales hypothesis to locate three points on the surface of an image. They used these points to calculate the right angle. Finally, they referred to these angles to produce 3-D point clouds. At the next step, the model uses the Iterative Closed Point (ICP) for 3-D rigid registration. ICP does not require local feature extraction. It can be generalized to N-Dimensional space and is suitable for parallel architectures. However, ICP is robust and stable but, it is based on the initial assumptions that may degrade its performance. Also, ICP needs pre-processing techniques for triangulation, mesh simplification, and generating 3-D trees. This technique requires high computation time for finding the closest point pairs.

Baka etal. [11] applied the Statistical Shape Model (SSM) for posture estimation and construction of a 3-D bone surface from X-ray images. In this approach, the authors used the concept of edge determination. This approach has the potential of capturing the global shape of the object of interest rather than reducing it to a set of fixed geometric measurements such as lengths and angles. But, it is challenging to determine the landmark correspondence over a set of bone shapes in the training dataset. Further, the number of identifiable landmarks in long bones is insufficient to represent the bone shape. Gamage etal. [20] proposed a technique for the conversion of 2-D radiographs into a patient-specific 3-D bone model. In the first step, the technique extracts the edge points from 2-D X-ray images. These edge points detect the boundary of the femur. Now, the non-rigid registration is done between the edges recognized in the X-ray images and contour points projected. At the next step, the technique uses the translational field to distinguish the anterior and lateral viewpoints of the 3-D anatomy. At the last step, it constructs the 3-D translational field through a Thin Plate Spline (TSP) based insertion and the 3-D generic anatomical data.

Lee etal. [21] proposed the model for constructing the 3-D shape of the femoral bone from its X-ray images. The model uses the 3-D referential approach to join the anatomical parameters viz. neck length, femoral length, head offset length, the anatomical axis, and sagittal radius. The authors applied nonlinear regression for calculating the inner position and range of the femoral head. They used elliptical regression for calculating the center point of the anatomical axis. Laporte et al. [22] applied the Direct Linear Transformation (DLT) and Non-Stereorediography Corresponding Points algorithm (NSCP) for the 3-D reconstruction of bones. But, these algorithms are effective for the ceaseless shape such as knee joint. The DLT technique does not require multiple images to calculate the distortion boundaries. Therefore, it has low computational complexity. However, its computation time is low but still, its use is limited due to the identification of the small number of corresponding anatomical landmarks on the radiographs. Moreover, DLT encounters real-time challenges such as it requires a large calibration frame to include the space of motion. The small frame may lead to extrapolation and, hence inaccurate computation of coordinates. Also, post-calibration alteration in settings may lead to inaccurate results. The NSCP technique employed in [22], reports more accurate results due to consideration of more number of control points. But, this technique lacks in marking the specific anatomical landmark points. This makes the technique unsuitable for bony structures with continuous shapes. Moreover, manual identification of landmarks is time-consuming and complex. Thus, it becomes unacceptable for clinical purposes.

Kolta et al. [23] proposed the technique for remaking the 3-D form of human bone from X-ray and DXA images. They considered 20 samples of proximal femur of human males and 5 samples of females. The samples were collected from the age range of 83 to 103 years. The authors used the contour detection method for generating 3-D shapes from 2-D images. They applied the NSCC algorithm for the initial matching and then applied the nonlinear deformation to minimize the gap between the 2-D projections DXA image. They achieved the mean error of 0.8 mm which is lower than the 2.1 mm error rate reported in [23].

Goli et al. [24] proposed an automated system for a vehicle. They experimented with the heuristic approach as well as with the whale optimization algorithm (WOA). The authors claimed that their model's efficiency and accuracy is better than particle swarm optimization and ant colony optimization algorithm. Even though, out of the heuristic approach and WOA, WOA methods give the best solution for a given automation problem.

Karade and Ravi proposed the LSD technique for 3-D femur reconstruction from bi-planer X-ray images [14]. This deformation technique is fast, robust, and easy to control. Thus, it is useful for creating interactive applications. But, this technique requires the explicit setting of smoothing to create a satisfactory base mesh. The meshes with complex details may require multiple levels of multiresolution hierarchy to correctly handle the details. Han proposed the DCNN [13] approach for classification and 3-D model design. The deep learning architecture proposed for the analysis of 2-D images can be easily adapted to 3-D models by merely replacing the 2-D up-convolutions in the decoder with 3-D up-convolutions. But, this approach is computationally expensive due to the cubical increase in the number of convolutions on the 3-D space.

The extensive study of the works proposed in the literature shows that the researchers majorly applied DLT [9], FFD [10], SSM [11], NSCP algorithm [12], DCNN [13], LSD [14], ICP algorithm [15], and PLSR [16] for the conversion of 2-D to 3-D images. Each technique has its advantages and limitations.

The comparative analysis of these techniques is shown in Table 1. The first column includes the year, the second column presents the names of researchers, the third column gives the name of techniques, the fourth column presents the application(s) of the technique, the fifth column highlights

TABLE 1. Comparative analysis of techniques proposed in literature for conversion of 2-D to 3-D.

Year	Authors	technique used	Application(s)	Advantage(s)	Disadvantage(s)
2010	Zhang, B., Sun, S., Sun, J., Chi, Z.,	Direct Linear Transformation	3-D recreation of the femur bone	For distortion of boundaries, DLT requires less number of images. Thus, it has low computational complexity.	DLT requires uniform distribution of more control points for improving accuracy. Also, it requires two cameras for capturing different views of an image which adds to the cost of conversion. Moreover, it is difficult to sync the two cameras.
2011	Koh, K., Kim, Y. H., Kim, K., Park, W. M	Free From Deformation	To procure the patient-specific deformation	It has the potential to calculate the control points of a shape at a low cost.	It is difficult to apply the FFD technique to the complex anatomy of the body. Also, it is found ineffective in handling the deformed input images.
2011	Baka, N., Kaptein, B. L., de Bruijne, M., van Walsum, T., Giphart, J. E., Niessen, W. J., Lelieveldt, B. P	Statistical Shape Model	Posture estimation and shape reconstruction of a 3-D bone surface.	It requires a limited number of parameters for the identification of the shape of an object.	Determining the landmark correspondence point of a bone is a time-consuming task.
2013	Zheng, G	Partial Least Square Regression	3-D reconstruction of volumetric intensity	It can easily handle the multi-collinearity between the independent points at the time of 3-D reconstruction.	PLSR technique needs a large number of data for training.
2015	Mitulescu, A., Semaan, I., De Guise, J. A., Leborgne, P., Adamsbaum, C., Skalli, W	Non-Stereo Corresponding Points	To reproduce the 3-D shape from the biplanar radiographs	It reports high accuracy due to the usage of more number of control points.	Its accuracy becomes low for Joint bones such as knee and elbow Joints, etc.
2015	Karade, V., Ravi, B	Laplacian Surface Deformation	3-D femur reconstruction from bi-planer X-ray images, and shape correspondence calculation for deformation	It requires low computation time due to less number of parameters.	It requires smoothing for generating a satisfactory mesh shape.
2019	Han, X., Laga, H., Bennamoun, M	Deep Convolutional Neural Network	Classification, and 3D model Design	DCNN is fault-tolerant. Therefore, one corrupted neuron does not affect the performance of other neurons.	Fine-tuning of hyper-parameters is non-trivial. Requires a large dataset for training. Also, this technique is ineffective for spatially invariant to the input data.

the advantages of the technique and the last column uncovers the drawbacks of the technique.

III. MATERIALS AND METHODS

In this section, the authors explain the dataset used and the methodology adopted for the conversion of 2-D images into 3-D images.

Dataset: For conducting the experiments, the authors collected the dataset from RG Stone Urology Laparoscopy Hospital. They collected 63140 X-rays and CT slice images from different branches of RG hospitals across India. The dataset contains 20410 images of knee bones, 22190 images of elbow bones, and 20540 images of bones of the lower limb. The authors divided the dataset into training, validation, and testing datasets. The training set contains 75% images of the total dataset. It contains 15307, 16642, and 15405 images from knee, elbow, and lower limb respectively. To avoid the class imbalance issue, it is important to use an equal number of images in each class. A system trained on such a dataset becomes equally efficient in recognizing images of each class. The testing dataset contains 25% of the total dataset size. This dataset contains 5102 knee bones, 5547 images of elbow bones, and 5135 images of bones of the lower limb. The validation dataset includes 20% images of each category.

The authors experienced the following challenges in the collected dataset:

1) Digital Imaging and Communications in Medicine (DICOM) images of the same bone may vary in quality, size, and resolution. The variation is dependent on the exposure of the radiations directed through the subject.
2) The dimensions such as height, width, and depth of DICOM images are different for various patients.
3) The DICOM images contain noise such as air, fat, soft tissues, etc. The noisy data may lead to the problem of over-fitting and under-fitting during the training of the model.

A. PRE-PROCESSING

A CT scan image is a collection of X-ray images captured at different angles. Each X-ray image stored in the DICOM format displays only a part of the bone rather than the complete bone. But, the deep learning model requires the image of a complete bone as an input. Therefore, multiple

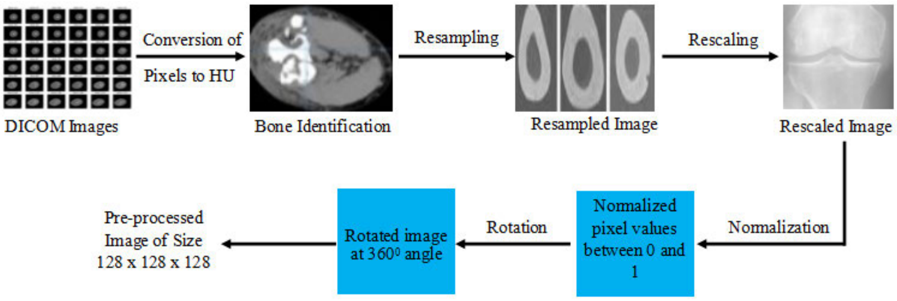


FIGURE 2. Pre-processing of the collected dataset.

DICOM images of CT slices of a bone are collected in the DICOM files. These images are merged to get the image of a complete bone. The merging of images may cause noise due to variations in shapes, sizes, and resolutions. Therefore, the authors loaded all DICOM files from the folder into a list. They applied the pre-processing operations such as noise removal, resampling, rescaling, and normalization as shown in Figure 2. These operations address the challenges identified in the raw data collected from the hospital.

1) NOISE REMOVAL

The collected DICOM images contain noise such as air, soft tissues, fat, etc. In this research, the authors need to apply the deep learning model only on the bone or its part. Therefore, it becomes essential to remove noise from the DICOM images and locate the bone. There is a measurable difference in the values of Hounsfield units (HU) for different types of information recorded in an image. Exemplified as the value for air is 1000 HU, for fat is −120 to −90 HU, for soft tissues it varies from +100 to +300, and for bone the values lie in the range from +700 to +3000 HU [25]. The difference in HU values of noise and bone is useful to recognize and locate a bone in the DICOM image. Therefore, the authors converted the voxel values of the DICOM images into HU as defined in equation 1. They used the rescale slope and rescale intercept stored in the DICOM header to perform the linear transformation.

$$\mathrm{HU} = (\mathrm{Gray\_Value} \times \mathrm{Slope}) + \mathrm{Intercept} \tag{1}$$

2) RESAMPLING

The DICOM files collected from the hospital differ in width, height, and depth. This information is available in the DICOM headers. The authors applied resampling on the available information and converted all the DICOM images into the ratio of 1 × 1 × 1 mm. Now, all the CT scan slices are uniform in height, width, and depth. This is useful in bringing uniformity to the dataset collected from various sources.

3) RESCALING

The resampled images are uniform in height, width, and depth but still, there may be variations in the number of pixels in each dimension. Merging the DICOM images of different dimensions may create white spaces in between two or more images. This white space acts as a noise and may degrade the performance of the model. Therefore, the authors rescaled each image to the uniform dimensions of 128 × 128 × 128 before giving them as inputs to the deep learning model.

4) NORMALIZATION

The pixel values of input images may vary and can disrupt the learning process of a Neural network. Also, the higher pixel values increase the computation cost and slow down the learning process. Thus, the authors divided the value of each pixel with the largest pixel value to normalize them between 0 and 1. The pixel normalization is applied across all channels regardless of the actual range of pixel values of an image.

5) AUGMENTATION

Deep learning neural networks require a huge dataset for training. But, labeling the medical data needs a substantial amount of time and effort from health experts and radiologists. So, it is challenging to get labelled dataset of medical images. Data augmentation techniques are useful in generating more images from the available samples. Hence, these are used to increase the size of the dataset. In this research, the authors applied data augmentation techniques such as angle rotation, and axis rotation. Also, to convert a 2-D medical image such as an X-ray into its corresponding 3-D view, it is mandatory to include images of each angle from 0° to 360° in the training dataset. But, the authors did not receive the image for each angle in the dataset collected from the hospital. Therefore, they applied the angle and axis rotation operations on the normalized images of bones. This increased the dataset size as well as fulfilled the requirement of training the model at images of each possible angle.

B. ARCHITECTURE OF CGAN

For generating the 3-D view from the 2-D view of a medical image, the authors used Conditional Generative Adversarial Network (CGAN) [26]. The CGAN is a deep-learning model comprising a generator and a discriminator. The generator and discriminator are contenders of each other. The generator generates a random image corresponding to the input image

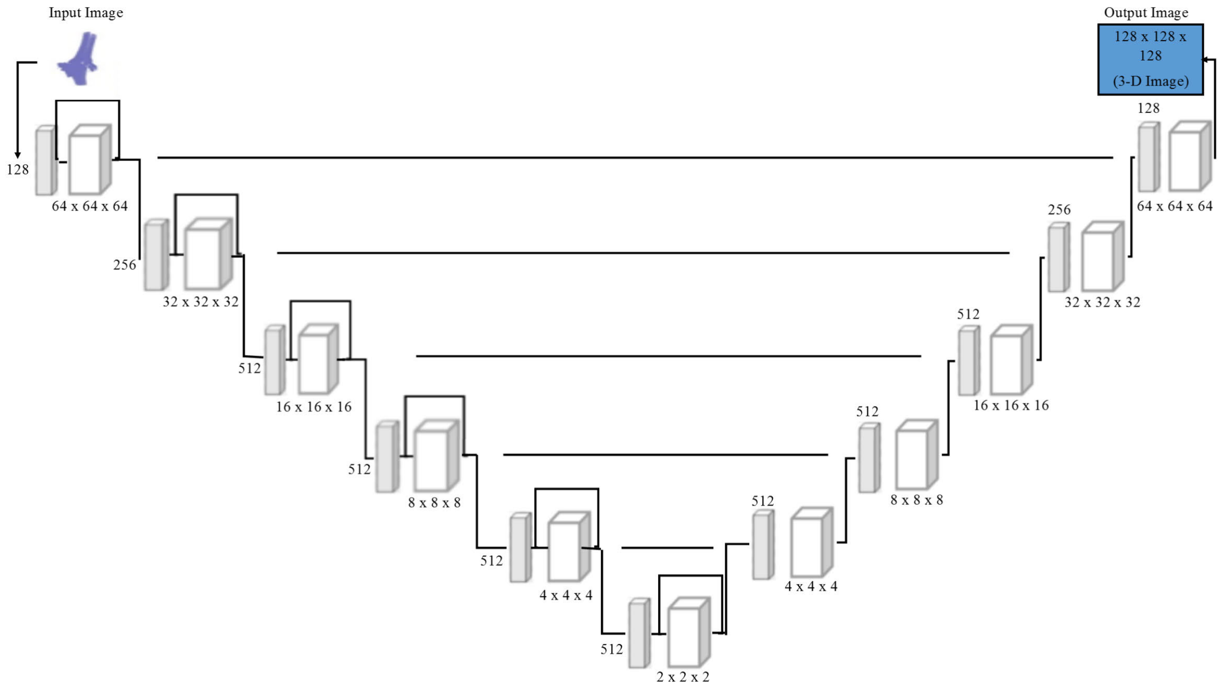


FIGURE 3. **Architecture of proposed CGAN model.**

given to it. The discriminator compares the output image of the generator with the input image and detects whether the generated image is real or fake. The generator tries to mislead the discriminator by generating the fake image whereas the discriminator drives the generator to generate an image similar to the input image.

The basic architecture of GAN does not require output labels at the discriminator end. Therefore, it adopts the unsupervised learning approach. But, in this research, the authors need to present each view from 0° to 360° corresponding to a specific angle view, say 0°. Thus, they added the output labels viz. real images and fake images at the discriminator end. They also assigned the condition to the generator for generating the images of all views from 0° to 360° or the desired view at any angle. This changed the unsupervised learning approach to supervised learning.

In this research, the authors tailored the architecture of Generative Adversarial Network (GAN) to design the CGAN architecture according to the dataset available and the output required. The architecture is similar to the Unet [27] network and comprises three paths viz. contracting path, bottleneck, and expanding path. In this architecture, the authors used the kernel size as 4×4 and a stride of 2 at each layer. They employed the LeakyReLU activation function for non-linear transformation. This function enhances the learning of the network by considering negative values [28]. So, the network becomes efficient in performing complex tasks. The authors used the skip connections [29] to connect the stages of the contracting and expanding paths. The network uses different channels for extracting the features from the input image. In this architecture, the authors used Conv3d to get the 3-D image as discussed below. The proposed architecture of CGAN is shown in Figure 3.

1) CONTRACTING PATH

At this path, the network performs convolutions for down-sampling of the input image from the dimensions 128×128 to $4 \times 4 \times 4$. Initially, a 2-D image of size 128×128 is given as input to the network. At the first layer, the network contains 28 channels to extract the features. It gives the feature map of dimensions $64 \times 64 \times 64$. At the second layer, it uses 256 channels each of dimension $32 \times 32 \times 32$. At the next layer, it uses 512 channels each of dimensions $16 \times 16 \times 16$. Now, the number of channels remains 512 for the next two layers but, the dimensions of feature maps are reduced to half at each layer. Thus, in the next layer, the dimensions become $8 \times 8 \times 8$ that are reduced to $4 \times 4 \times 4$ at the last layer of the contracting path. During this step, the model converts the 2-D view (128×128) to the 3-D view ($4 \times 4 \times 4$) of an input image and extracts the important features.

2) BOTTLENECK PATH

At this stage, the network uses 512 channels each of dimension $2 \times 2 \times 2$ to reduce the computational cost of the whole network. The layer at the bottleneck path acts as a connecting link between contracting and expanding paths. It receives the input from the contracting path and passes it to the expanding path.

3) EXPENDING PATH

The information retrieved in the form of features extracted at the contracting phase is useful in generating a 3-D view from the 2-D view of an image. The expanding path uses this information and generates the 3-D view of an image. It doubles the dimensions at each layer until the achieves the dimensions of the input image (128 × 128 × 128). At the first layer, the network uses 512 feature maps each of dimensions 4 × 4 × 4. At the next layer, the dimensions of each feature map double and become 8 × 8 × 8. At the third layer, the network increases the dimensions to 16 × 16 × 16 for 512 feature maps. At the fourth layer, the number of channels is reduced to 256 but, the dimensions of each channel are further increased to 32 × 32 × 32. At the last layer, the network uses 128 feature maps each of dimensions 64 × 64 × 64 and gives a 3-D image of size 128 × 128 × 128 as shown in Figure 3.

C. TRAINING PARAMETERS

The performance of deep learning models is directly related to their training. The training of these architectures is dependent on the size of the dataset, quality of the dataset, values of model parameters and hyper-parameters for the network, activation functions employed, and the number of output classes. In this section, the authors discuss the model parameters and hyper-parameters used to fine-tune the modified GAN architecture.

1) MODEL PARAMETERS

a: LOSS FUNCTION

The loss function is used to calculate the error of the model. The error calculated at each iteration is back-propagated to alter the weights of neurons. The model learns from the updated weights and tries to minimize the value of the loss function. The lower value of the loss function indicates better training of the network. The authors employed the following loss functions in network architecture.

L1 LOSS

It is the pixel-to-pixel difference between the image generated by the generator and the target image. Its definition is given in equation 2. In this equation, y_i is an instance of the target image, x_ij is the input pixel and w_ij is the weight of a neuron. L1 loss is the least absolute deviation that is used to decide which function should be minimized during learning from the dataset. The L1 loss gives higher gradients to the small values of loss and updates the weights and biases of different layers for the training of the network.

$$L1Loss = \sum_{i=0}^{N}(y_i - \sum_{j=0}^{M}(x_{ij}w_{ij}))^2 \tag{2}$$

Binary Cross Entropy Loss

Binary Cross-Entropy (BCE) loss is the negative of the logarithmic function as defined in equation 3, In this equation, y_i is the actual label (0 or 1). P is the predicted probability for the class and N is the total number of samples. The discriminator uses this loss function to distinguish the image generated by the generator and its corresponding input image.

$$BCELoss = -\frac{1}{N}\sum_{i=1}^{n}(y_i log(p) + (1 - y_i) log(1 - p)) \tag{3}$$

b: OPTIMIZERS

The continuous update in the weights of neurons is required for training the network. Therefore, the networks employ the optimizers such as Stochastic Gradient Descent (SGD) [30], Root Mean Square Propagation (RMSProp [31]), Adaptive Moment Estimation (Adam) [31], Adaptive Gradient (AdaGrad) [31] and Adadelta [30]. A brief description of these optimizers is given below.

The SGD optimizer chooses one random sample rather than a batch of the dataset. It requires low memory as it computes only 1 point at a time. But, this optimizer needs more time to complete 1 epoch of training due to random samples. The RMSProp optimizer is an adaptive learning rate method. It automatically adjusts the learning rate and sets the different learning rates for each parameter. The adagrad optimizer performs higher updates for the infrequent parameters and smaller updates for the frequent parameters. Therefore, it deals with sparse datasets [31]. Also, it has a continuous decaying learning rate throughout the training. The learning rate automatically becomes infinitesimally small after a set of iterations. Thus, it does not require manual fine-tuning of the learning rate. The optimizer adadelta overcomes the problem of monotonic and continuous decreases in the learning rate observed in AdaGrad.

In this manuscript, the authors employed an Adam optimizer that adopts the features of both the RMSProp and AdaGrad optimizers. Similar to the RMSProp, it uses squared gradients to scale the learning rate. Further, it takes advantage of momentum by using the moving average of the gradient instead of the gradient itself. To calculate the value of the current gradient it uses the values of the past gradient. For calculating the momentum, it adds a fraction of the previous gradient to the current gradient. Adam optimizer calculates the mean and variance of the moment. It uses an exponentially decaying average of past gradients (mt) and past squared gradients (vt) as defined in equations 4 and 5 respectively.

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \tag{4}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \tag{5}$$

In equations 4 and 5, g_t shows the value of the loss at the i^{th} iteration. The term β_1 and β_2 are the forgetting factors for the mean and non-centered variance of the gradient respectively. The values of β_1 and β_2 are set to 0.5 and 0.999 based on the set of experiments conducted and the research works presented in [31].

c: ACTIVATION FUNCTION

Activation functions calculate the weighted sum of neurons and add bias to it. Based on the calculated value, it identifies the neurons to be activated. It introduces the non-linear

TABLE 2. Impact of hyper-parameter.

	α=0.0001 and η=100		α=0.0003 and η=100	
Epoch	Loss of Discriminator	Loss of Generator	Loss of Discriminator	Loss of Generator
15	0.003261359	1.923183461	1.399525284	1.773968554
30	0.015893746	0.912701669	1.325775861	1.531654061
45	0.005482937	1.615594452	1.345080018	1.629260665
60	0.004738476	1.574684221	1.71176064	1.840917689
75	0.025739576	1.565683832	1.445257663	1.722736474
90	0.003689274	1.621834521	1.46692287	1.618364288
105	0.004278347	1.776031188	1.46692287	1.768832596
120	0.005719283	1.672490924	1.486529231	1.871892124
135	0.014857625	1.021035152	1.49918746	1.912135252
150	0.006283746	1.476518832	1.52298784	1.614006907

transformation to the input. Hence, it is useful in enhancing the learning of the network and improving its efficacy for performing complex tasks. In the proposed architecture, the authors used three activation functions viz. Tangent Hyperbolic function (tanh), LeakyReLU, and sigmoid [28]. The tanh is a nonlinear activation function used in the architecture of the generator to introduce the non-linearity in the generated image. Its value lies in the range from −1 to +1. Due to zero centroid functionality, the tanh activation function makes the optimization easier. The gradient of this activation function is stronger than other activation functions as its derivatives are steeper.

LeakyReLU activation function is used in the architectures of both the generator as well as the discriminator. It gives the solution to the problem of dying ReLU [28]. In this problem, the network considers both the negative as well as positive values of the gradient. Thus, the resultant value of the gradient becomes zero. The sigmoid activation function is used in the architecture of discriminators for binary classification. The discriminator detects whether the generated image is real or fake. Its '0' value indicates the fake image and '1' indicates the real image.

d: WEIGHT INITIALIZATION

At the initial step, random weights are assigned to the neurons of the deep learning model. The random and non-uniform initialization of weights to different neurons may lead to incorrect training. For example, if the model automatically selects the pixel values of an image as the initial weights of its neurons, then the pixel values corresponding to the bones will be higher than its neighboring pixels in the image. Thus, a part of the image that contains one or more bones will contribute higher weights to the neuron than the remaining part(s). This can lead to wrong and imbalanced training of the network for different components captured in an image. Therefore, it becomes important to normalize the weights of the neuron. The authors normalized the weight of each neuron between 0 and 1. Also, they initialized convolutional layers with a normal distribution of mean as 0.0 and a standard deviation of 0.02. The authors used normal distribution with a mean value of 1.0 and a standard deviation of 0.02 for batch norm layers. They initialized the Bias as '0' to avoid the asymmetry that may be caused by random numbers. Now, the weights are continuously updated by using the concept of back-propagation. The normalization reduces the computation time of the model and is also responsible for its fast convergence [32].

To update the weights of the neurons in the network, the concept of back-propagation plays an important role. Here, the current weight of a particular neuron depends on the previous weights. Therefore, the authors initialized convolutional layers with a normal distribution of mean as 0.0 and the standard deviation as 0.02. Similarly, for batch norm layers, the authors used normal distribution with a mean value of 1.0 and a standard deviation of 0.02. Bias is initialized '0' to avoid the asymmetry that may be caused by random numbers.

2) HYPER PARAMETERS

Hyper-parameters are the adjustable parameters such as the regularization parameter (α) and learning rate (η) of the neural network model. Their values are not estimated from the dataset. These parameters are fine-tuned and preset before the actual training of the model. It is important to optimize the performance of the model.

a: REGULARIZATION PARAMETER (α)

The regularization parameter adds a penalty term to the loss function to fine-tune it according to the size of the dataset. Its low value causes the problem of over-fitting whereas its high value leads to the under-fitting of the model. The optimal value of this parameter resolves the problem of over-fitting and under-fitting. In this research, the authors conducted a set of experiments and set the value of α as 100 for achieving the optimum performance of the proposed model. The impact of the regularization parameter on the convergence and performance of the model is shown in Table 2 and Table 3.

b: LEARNING RATE (η)

The learning rate is the change in the value of weights during each epoch of training. It determines the response of the model to the error computed. The model updates its weights after each iteration of error computation. Its low value indicates that neurons take a long time for reaching the optimum solution. Thus, the model has high computational complexity. The high value of the learning rate leads to the instability of the model. Therefore, it is important to find

TABLE 3. Impact of hyper-parameter.

	α=0.0001 and η=95		α=0.0005 and η=105	
Epoch	Loss of Discriminator	Loss of Generator	Loss of Discriminator	Loss of Generator
15	0.018273691	1.712217223	0.002749374	0.951103746
30	0.004192229	1.517771087	0.052716354	1.647412506
45	0.00829912	1.721967102	0.003102839	1.528650848
60	0.008362715	1.818164916	0.000411877	1.837485488
75	0.004292922	1.823518371	0.007188976	1.728184738
90	0.189237452	1.712540239	0.009912912	1.683780975
105	0.047183726	1.593438713	0.291002933	1.481736907
120	0.007238461	1.611489771	1.021938235	1.912913513
135	0.004343782	1.743519985	0.005728374	1.885609008
150	1.284736491	1.911729362	0.008819723	1.728563415

the optimal value of the learning rate. The research works presented in [31] recommend the values of learning rate between 0.0 to 1.0. Based on the values recommended in [31] and a set of experiments conducted, the authors observed that the model proposed in this manuscript achieved its optimum performance at 0.0002 value of the learning rate. Beyond this value, there is a quick drop in the value of the loss function. The impact of the learning rate on the convergence of the model is shown in Table 2 and Table 3.

D. TRAINING OF CGAN MODEL

In this section, the authors explain the training mechanism of the proposed model. The authors give 3-D input images for training the model. The corresponding generated image is evaluated against the 3-D input image by calculating the loss function. The sample input image is shown in figure 3. Once the model is trained, its performance is evaluated using the dataset comprising 2-D images. It is evident from the results shown in figures 4, 5, and 6 that the model is effective in the conversion of 2-D image to its corresponding 3-D image. The sequence of steps involved in the training is shown in Algorithm 1.

Algorithm 1 Training Procedure of the Model

```
1) $\mathrm{D}\left(x \bigoplus y_{real}\right) \rightarrow P\{0, 1\} \Rightarrow 1$
2) $\mathrm{G}(\mathrm{x}) \rightarrow y_{fake}$
3) $\mathrm{D}\left(x \bigoplus y_{fake}\right) \rightarrow P\{0, 1\} \Rightarrow 0$
4) $L_{CGAN}(\mathrm{G}, \mathrm{D}) = E_{x,y_{real}}[\log \ \mathrm{D}(\mathrm{x}, y_{real})] + E_{x,y_{fake}}[\log(1 - \mathrm{D}(\mathrm{x}, \mathrm{G}(\mathrm{x})))]$
5) $L_{L1}\ (\mathrm{G}) = \mathrm{E}_{x,y_{real}} = [|y_{real} - \mathrm{G}(\mathrm{x})|]$
6) Total Loss $= arg\, \min_G \max_D L_{CGAN}(G, D) + \lambda\ L_{L1}\ (G)$
7) Back propagate total loss and calculate the gradients.
8) Adam optimizer updates the weights and bias of the network.
9) Repeat steps 1 to 8 until total loss becomes minimum.
```

At the first step, a 2-D input image $'x'$ and the target image y_{real} are given as input to the discriminator of the model. Both images are similar, so the discriminator gives the probability approximate to 1. This indicates the image is real. In the second step, the input image is given to the generator (G). Initially, it generates a fake image for the given input image. Now, the input image and the generated fake image are forwarded to the discriminator (D). The discriminator compares the generated image with the target image. Both, the inputs of the discriminator are fake counterparts of each other. So, the discriminator gives the value of probability (P) approximately 0. In the fourth step, the model calculates the values of its loss (L) functions. It calculates the values of BCE loss for the generator as well as discriminator using the equation shown in step 4 of algorithm 1. Based on the value of BCE loss the discriminator distinguishes the generated image and its corresponding real image. At the next step, the model calculates the value of L1 Loss for the generator using the equation given in step 5 of algorithm 1. Now, the model calculates the value of the total loss as given in step 6 of algorithm 1. In the next step, the model back propagates the value of the total loss for calculating the gradients of the model. In the last step, the Adam optimizer with an adaptable learning rate updates the parameters viz. weight and bias of the network. The above-discussed procedure is iterated until the value of the loss function becomes minimum. The minimum value of the loss function approximates to '0' indicating that the model is trained effectively to generate the 3-D view of the 2-D input image.

IV. RESULTS

In this research, the authors conducted experiments on RTX 2080 Graphics Processing Unit (GPU) with 64 GB RAM and a 2 TB hard disk. The GPU runs with the Ubuntu 18.04 operating system.

The model is trained for 150 epochs. At each epoch, the variation in the value of the loss function is observed for the generator and discriminator. The experiments were conducted using different values of hyperparameters. The values of loss functions obtained at different values of hyperparameters are shown in Table 2, and Table 3. Table 2, shows the impact of α without changing the value of η. It is evident from the values of loss function obtained at α as 0.0001 that the discriminator and generator reach the minimum of values of loss functions as 0.003261359 and 0.912701669 respectively. Whereas, Table 3 shows the collaborative impact of α as 0.0001 and η as 95. On these values, the discriminator and generator report the lowest values of the

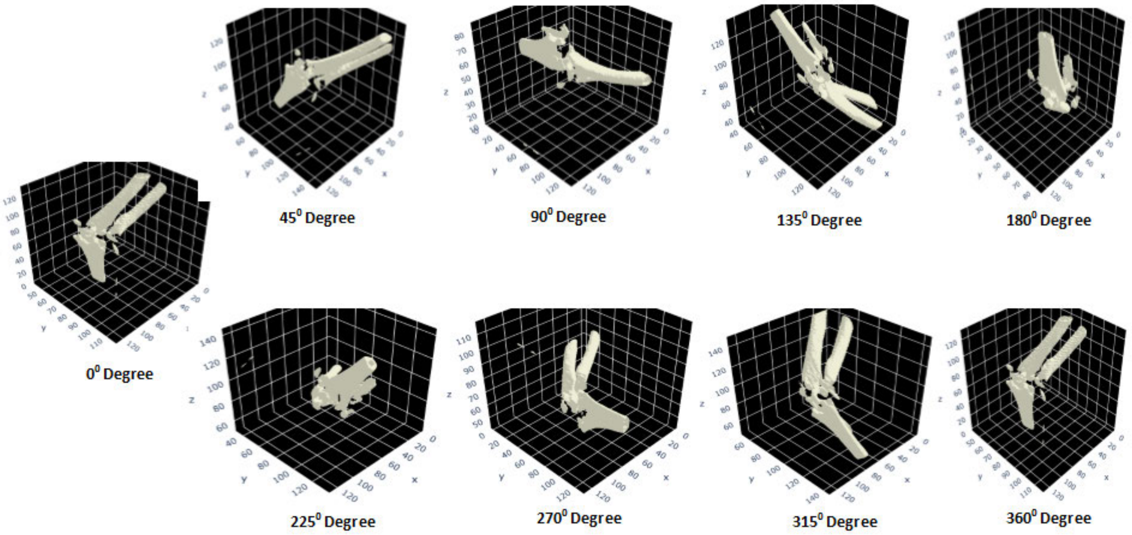

0^0 Degree, 45^0 Degree, 90^0 Degree, 135^0 Degree, 180^0 Degree, 225^0 Degree, 270^0 Degree, 315^0 Degree, 360^0 Degree

FIGURE 4. 3-D view of knee bone at different angles.

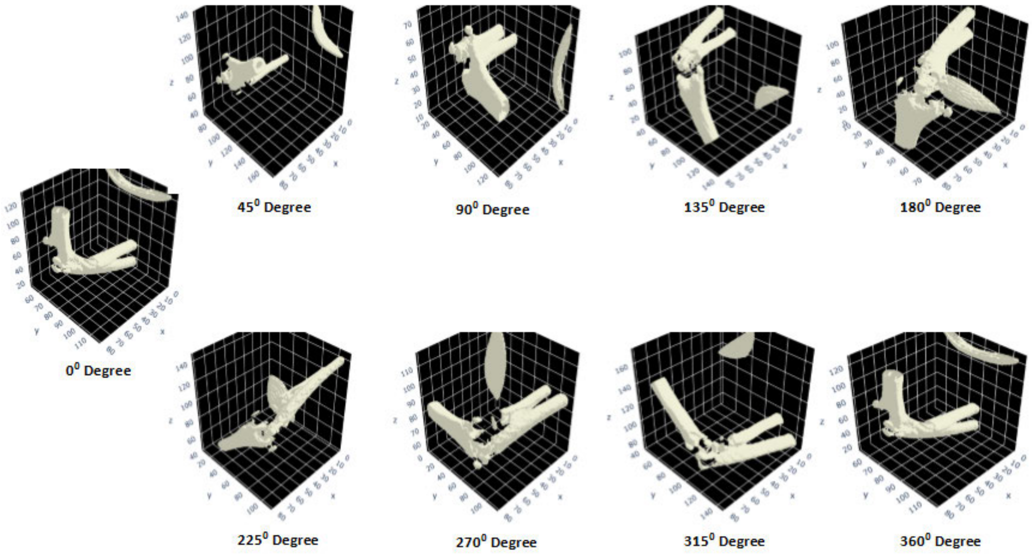

0^0 Degree, 45^0 Degree, 90^0 Degree, 135^0 Degree, 180^0 Degree, 225^0 Degree, 270^0 Degree, 315^0 Degree, 360^0 Degree

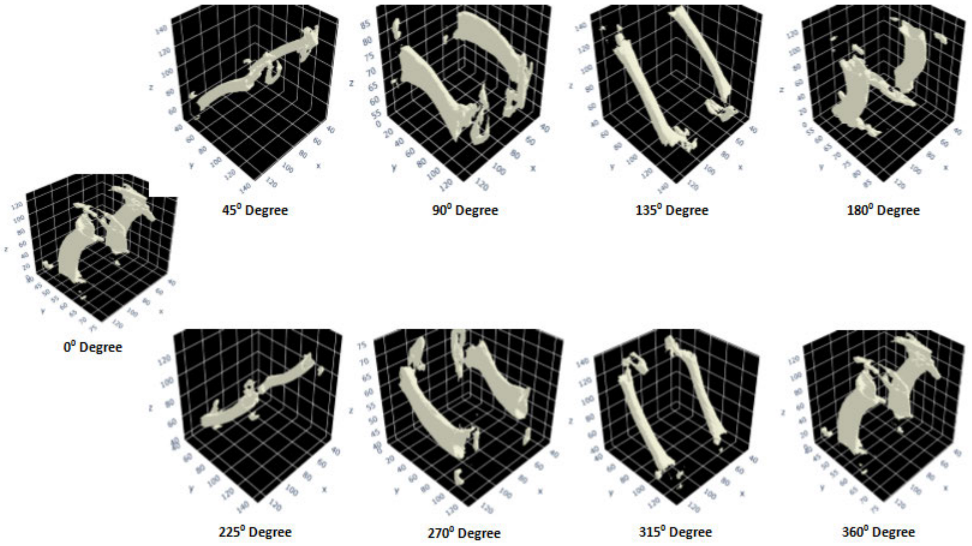
FIGURE 5. 3-D view of elbow bone at different angles.

loss function as 0.004192229 and 1.517771087 respectively. Similarly, it is apparent from Table 3 that both the generator and discriminator report the lowest values of loss function as 0.0014094849 and 0.9195939898 respectively. The values of the loss function obtained at 100 and 0.0002 values of α and η respectively, are lower than the values obtained at other values of hyperparameters. The values of loss functions reported at these optimal values through different epochs are shown in Table 3. The first column of the table contains the epoch number, the second column shows the values of the loss function calculated by the discriminator, and the last column presents the values of the loss function reported by the generator.

The values given in table 3 demonstrate that the generator gives the minimum value of loss function when the model is trained at 135 epochs. Whereas, the discriminator reports the minimum value of loss function on training the model at 30 epochs. To identify the optimum number of epochs for the conversion of a 2-D image into the 3-D view, the authors observed the values of the loss function as well as the 3-D view at different epochs. They observed that training the model for 135 epochs gives the complete and the most accurate 3-D view of a 2-D image. The model generates all the views from 0° to 360°. In Figures 4,5 and 6, the authors present the sample outputs at 0°, 45°,90°,135°, 180°, 225°, 270°, 315°, and 360° for the knee bone, elbow bone, and bone of lower limb respectively.

V. EVALUATION OF IMAGE QUALITY

The evaluation of merely the 3-D view and the loss functions is not sufficient to prove the efficacy of the proposed model. The visual quality of the generated images is equally



0° Degree

45° Degree

90° Degree

135° Degree

180° Degree

225° Degree

270° Degree

315° Degree

360° Degree

FIGURE 6. 3-D view of bones of the lower limb at different angles.

TABLE 4. Trends of loss function reported by CGAN during different epochs of training.

Epoch Number	Loss of Discriminator	Loss of Generator
15	0.0022553926	1.4292654991
30	0.0030624993	0.9195939898
45	0.0026638618	1.2375315428
60	0.0016441641	1.5464997292
75	0.0014240577	1.4429641962
90	0.001654461	1.5576143265
105	0.0016032617	1.5418564081
120	0.0014812108	1.4896333218
135	0.0014094849	1.5114090443
150	0.0019811528	1.4078457355

important for adopting the proposed model in clinical applications. Further, the generated images must be potent in preserving the information available in the input image. Therefore, the authors evaluated the visual quality of the generated images by calculating the Entropy, and Peak Signal to Noise Ratio (PSNR), Mean Square Error (MSE), and Structural Similarity Index Method (SSIM).

A. ENTROPY

This is the measure of the variation recorded in an image [33] as defined in equation 6. In this equation E represents the entropy, P(X_K) is the value of probability distribution, and L is the total number of different intensity values present in an image.

$$E\left(X_K\right) = \sum_{K=0}^{L-1} P\left(X_K\right) * log_2 P\left(X_K\right) bits/pixel \qquad (6)$$

The entropy works on the concept of probability. For example, in a black and white image, the value '0' represents the black pixel, and '1' represents the white pixel. While scanning an image from left to right and from top to bottom, the changes in values from 0 to 1 and 1 to 0 are the measure of the entropy of the image. The low value of entropy ensures a small variation between the original image and the generated image.

In this research, the values of entropy for the input images of the knee, elbow, and lower limb bones are reported as 3.55, 3.53, and 3.65 respectively. The authors also calculated the values of entropy for the generated images at different angles of view as shown in column 3 of Table 5. It is clear from table 5, that the values of entropy of the generated images at different angles of view are nearly the same as values of entropy of the input images. There is a negligible difference of 0.026 in the entropy reported for the knee bone, 0.01 for the elbow bone, and 0.0 for the lower limb bone at 360° views of the generated image and its corresponding input image. These values prove the efficacy of the proposed model in generating images that are similar to the input images in quality. Also, the generated images are effective in preserving the information.

B. MEAN SQUARE ERROR

MSE is used to measure the number of squared errors between the original image and the image generated by the generator [34]. Its definition is given in equation 7. In this equation, $\hat{f}\left(x, y\right)$ is the original image, and f(x, y) is the generated image. Here, M and N are the heights and widths of the original and generated images respectively. In equation 7, x and y are the values of pixels of the images. The minimum value of MSE indicates that the generated image resembles the original image.

$$MSE = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \left(\hat{f}\left(x, y\right) - f\left(x, y\right)\right)^2 \qquad (7)$$

TABLE 5. Evaluation of the quality of the generated image.

Bone Category	*Angles of View*	*R_En*	*G_En*	*MSE*	*PSNR*	*SSIM*	*Contrast*
Knee	45°	3.55	3.08	58.48	30.46	0.71	41.55
	90°	3.55	3.66	54.97	30.73	0.78	41.51
	135°	3.55	3.39	59.47	30.39	0.82	41.59
	180°	3.55	3.49	47.23	31.39	0.74	41.55
	225°	3.55	3.12	58.93	30.43	0.72	41.58
	270°	3.55	3.62	57.28	30.55	0.79	41.55
	315°	3.55	3.36	52.59	30.92	0.75	41.58
	360°	3.55	3.54	45.29	31.57	0.89	41.48
Elbow	45°	3.53	3.24	50.48	31.09	0.71	41.58
	90°	3.53	3.49	49.98	31.14	0.71	41.39
	135°	3.53	3.31	50.62	31.08	0.74	41.43
	180°	3.53	3.37	52.27	30.95	0.89	41.35
	225°	3.53	3.15	52.41	30.93	0.72	41.49
	270°	3.53	3.44	50.1	31.13	0.71	41.29
	315°	3.53	3.35	41.91	31.91	0.77	41.49
	360°	3.53	3.54	38.11	32.32	0.76	41.41
Lower Limb	45°	3.65	3.37	56.92	30.58	0.75	41.61
	90°	3.65	3.71	53.99	30.81	0.74	41.17
	135°	3.65	3.49	49.68	31.17	0.74	41.58
	180°	3.65	3.76	56.12	30.64	0.71	41.38
	225°	3.65	3.54	55.67	30.67	0.75	41.59
	270°	3.65	3.75	53.31	30.86	0.74	41.32
	315°	3.65	3.67	57.52	30.53	0.72	41.56
	360°	3.65	3.65	32.76	32.85	0.75	41.31

The values of MSE calculated for the generated images are given in column 5 of Table 5. The 360° views of the generated images reported the minimum values of 31.57, 32.32, and 32.85 for the knee, elbow, and lower limb bones respectively. These values show that there are marginal errors between the input images and the generated images. This proves the efficacy of the proposed model.

C. PEAK SIGNAL TO NOISE RATIO

PSNR is the measure of the quality of the reconstructed image. It is the ratio between the signal and noise of an image [35]. This matric is dependent upon the MSE as shown in equation 8. Its higher value indicates the better quality of the generated image.

$$PSNR = 10\ log_{10}\frac{(L-1)^2}{MSE} \tag{8}$$

The value of PSNR calculated for the images generated by the proposed model is shown in column 5 of Table 5. The generated image reported the highest values of 59.47, 52.27, and 56.92 for the knee, elbow, and lower limb bones respectively. These values clearly prove that the quality of the reconstructed images is similar to the input images. Therefore, the proposed model is reliable for generating 3-D images and images at a different angle of view from the 2-D input images.

D. STRUCTURAL SIMILARITY INDEX METHOD

MSE works on the individual pixels of an image whereas SSIM works on the groups of pixels. It is the measure of the similarity index between the original and generated image. Its value lies in the range from −1 to +1 [34] where -1 indicates that there is no matching between the input and generated image. On the other hand, +1 indicates that the generated image is the same as the input image. The values of SSIM calculated for the generated images for knee, elbow, and lower limb bones are presented in column 6 of Table 5. The values 0.89, 0.89, and 0.75 obtained for the knee, elbow, and lower limb bones respectively indicate that the generated images have a close match to the input images. Therefore, the model is effective in maintaining the visual quality and preserving the information available in the image.

E. CONTRAST

The contrast of an image is defined as a measurement of average intensities and their deviation about a center pixel as defined in equation 9. In this equation, r is the width and c is the height of an image. l_{enh} is the intensity of a pixel at position (i, j). The contrast of an image is measured in DB for representing the large range of numbers by a convenient and small number. Its definition is given in equation 10. The contrast in DB is important for clearly visualizing the changes.

$$C_{contrast} = \frac{1}{rc}\sum_{i=1}^{r}\sum_{j=1}^{c} l_{enh}\,(i,j)^2 - \left|\frac{1}{rc}\sum_{i=1}^{r}\sum_{j=1}^{c} l_{enh}\,(i,j)\right|^2 \tag{9}$$

$$C^{*}_{contrast} = 10log_{10}C_{contrast} \tag{10}$$

VI. DISCUSSION

The work proposed in this manuscript met the objective of converting a 2-D X-ray image to its corresponding 3-D view. The proposed system receives a 2-D image at 5°, 10°, 15°…360°. It assumes the missing information such as views of an image at 1°, 2°, 3°, 4°, 11°, 12° …359° for predicting the complete 360° view of an image. The tailored architecture of the CGAN is found effective in generating a view of the 2-D images at any angle from 0° to 360°. Also, it is equally efficient in generating a complete 3-D view of a 2-D image.

The works proposed in the literature for conversion of 2-D to 3-D images left unaddressed challenges. For example, the authors in [9] used two different cameras to capture different views of an image. They employed Direct Linear Transformation (DLT), for the conversion of a 2-D image to 3-D. The DLT technique is more expensive than the system proposed in this manuscript. Also, it is challenging to synchronize both cameras. Further, the authors in [12] employed the Non-Stereo corresponding contour (NSCP) method for the conversion of X-rays into 3-D form. The efficacy of this technique is dependent on the expertise of the operator in identifying exact points. Also, the technique requires about 2 to 4 hours for the reconstruction of an image. Thus, its' implementation in real life is impractical. One more technique Statistical Shape Model (SSM) [11] was applied for the conversion of 2-D to 3-D that requires one-to-one mapping for training the model. Thus, the model requires the actual dimension of the dataset for training. This becomes time-intensive when input images have large dimensions. Also, a huge dataset is required to achieve high accuracy. Similarly, the Deep Convolutional Neural Networks (DCNN) based technique employed in reference [13] requires a huge dataset for training. Also, this technique lacks in encoding the position and orientation of an object. Therefore, it was found ineffective for spatially invariant of input data. Moreover, the techniques proposed so far fails to handle the noisy data. The CGAN-based model proposed in this manuscript addressed the aforementioned challenges. It does not require any camera and reduces the cost of conversion of 2-D to 3-D form. Also, the model is based on the concept of Artificial Intelligence. Therefore, its' efficacy is independent of the expertise of the operator. Furthermore, the proposed model is efficient in converting a 2-D image to 3-D in real time. Thus, it can be easily adopted as a technical assistant for medical experts.

We also evaluated the visual quality of generated images by calculating the entropy, MSE, PSNR, SSIM, and contrast for the knee, elbow, and lower limb bones. It is evident from the results shown in Table 5 that the generated 3-D images have high values of entropy, PSNR, SSIM, and the low value of MSE. The value of the contrast of generated images is approximately equal to the contrast of the input images. These results prove that the model retains the quality of the generated images and preserves the information available in an input image.

The proposed model can be integrated with web applications as well as mobile applications for making it a handy diagnostic and training tool. The users can upload the X-ray image as input and receive a 3-D view of the image. Converting 2-D bone X-ray images to 3-D can provide additional depth information. This assists radiologists and orthopedic specialists to visualize a more comprehensive view of the bone structure. This enhanced visualization helps in accurate diagnosis. Additionally, generating 3-D views from X-ray images may be a boon in telemedicine. Access to a web app or mobile app can be shared remotely for expert consultations. The 3-D view facilitates more accurate diagnoses. Also, the tool may prove a game-changer in surgical planning. The 3-D view of a bone can assist surgeons to evaluate different surgical approaches and implant sizes. This may improve surgical outcomes. Moreover, the tool can be used by medical students for a better understanding of complex bone structures and pathology. This facilitates interactive learning and training.

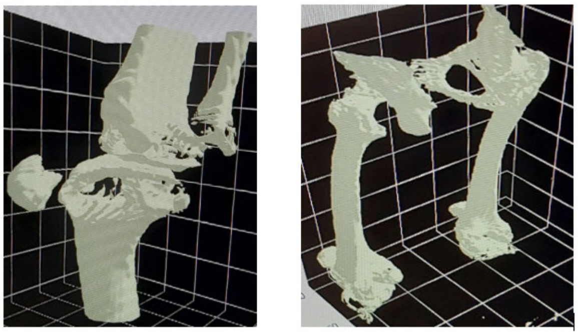


FIGURE 7. Sample of noisy image.

VII. CONCLUSION

The authors in this manuscript proposed the CGAN based model and extended its application for the conversion of a 2-D X-ray image into a 3-D view of bones. They applied the pre-processing techniques to deal with the noisy and sparse dataset. The model provides an option to rotate the obtained 3-D view at different angles from 0° to 360°. The 3-D view gives a clear vision of the joints and bones at all angles. Therefore, it is useful in the diagnosis of a fracture or deformity in bones at a low cost. It is evident from the results shown in Table 5 that the model is effective in maintaining the visual quality of generated images and preserving the information available in the input images. Also, it is clear from the results shown in figures 4, 5, 6 and 7, that it is efficient for the noisy as well as a non-noisy dataset. Moreover, it has the potential to convert a 2-D image to 3-D in real time. Thus, the model provides a low-cost and quick technological solution to visualize the 3-D view of a bone. As the model can generate a 3-D view similar to a CT scan and MRI corresponding to a 2-D X-ray image. Therefore, it helps in minimizing the exposure of carcinogenic radiation directed to the patients during a CT scan and also provides a 3-D view of an organ for patients having implantation where MRI imaging is not feasible. Thus, it may prove useful in developing the assisting tool for doctors in visualizing the different views of the bone from an X-ray image. The clear view of a bone at all possible angles from 0° to 360° gives an option to find the severity of the disease or disorder [36]. The model can be associated with a web or mobile application where a health expert can upload the X-ray image of a patient and visualize the 3-D view of the bone. The web application can be deployed on a cloud server and accessed across the globe. Even, the patient can log in, upload the X-ray image and view the 3-D image of a bone.

Future Scope: The work proposed in this research can be extended to obtain 3-D views of different parts of the body such as lungs, throat, and kidneys which in turn will be useful for the diagnosis of the disease and visualization of the infected region.

DECLARATION OF COMPETING INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

ACKNOWLEDGMENT

The authors would like to thank RG Stone Urology and Laparoscopy Hospital, Delhi, for providing the dataset for this research. The authors declare that this work was supported by the Department of Informatics, Modeling, Electronics and Systems (DIMES), University of Calabria [Grant/Award Number: SIMPATICO ZUMPANO]. *(All authors contributed equally to this work.)*

REFERENCES

[1] E. P. Balogh, B. T. Miller, and J. R. Ball, "Overview of diagnostic error in health care," Improving Diagnosis Health Care, Nat. Academies Press, Washington, DC, USA, Tech. Rep. 3, 2015.

[2] M. Agarwal, G. Rani, and V. S. Dhaka, "Optimized contrast enhancement for tumor detection," *Int. J. Imag. Syst. Technol.*, vol. 30, no. 3, pp. 687–703, Sep. 2020.

[3] T. T. W. Wong, R. Zhang, C. Zhang, H.-C. Hsu, K. I. Maslov, L. Wang, J. Shi, R. Chen, K. K. Shung, Q. Zhou, and L. V. Wang, "Label-free automated three-dimensional imaging of whole organs by microtomy-assisted photoacoustic microscopy," *Nature Commun.*, vol. 8, no. 1, pp. 1–8, Nov. 2017.

[4] D. C. Sullivan, L. H. Schwartz, and B. Zhao, "The imaging viewpoint: how imaging affects determination of progression-free survival," *Clin. Cancer Res.*, vol. 19, no. 10, pp. 2621–2628, May 2013, doi: 10.1158/1078-0432.CCR-12-2936.

[5] Y. Arlachov and R. H. Ganatra, "Sedation/anaesthesia in paediatric radiology," *Brit. J. Radiol.*, vol. 85, no. 1019, pp. e1018–e1031, Nov. 2012.

[6] E. J. van Beek, C. Kuhl, Y. Anzai, P. Desmond, R. L. Ehman, Q. Gong, G. Gold, V. Gulani, M. Hall-Craggs, T. Leiner, C. C. T. Lim, J. G. Pipe, S. Reeder, C. Reinhold, M. Smits, D. K. Sodickson, C. Tempany, H. A. Vargas, and M. Wang, "Value of MRI in medicine: More than just another test?" *J. Magn. Reson. Imag.*, vol. 49, no. 7, pp. e14–e25, 2019.

[7] N. Pradhan, V. S. Dhaka, G. Rani, and H. Chaudhary, "Transforming view of medical images using deep learning," *Neural Comput. Appl.*, vol. 32, no. 18, pp. 15043–15054, Sep. 2020.

[8] A. P. de Andrade da Costa e Silva, J. L. F. Antunes, and M. G. P. Cavalcanti, "Interpretation of mandibular condyle fractures using 2D- and 3D-computed tomography," *Brazilian Dental J.*, vol. 14, no. 3, pp. 203–208, 2003.

[9] B. Zhang, S. Sun, J. Sun, Z. Chi, and C. Xi, "3D reconstruction method from biplanar radiography using DLT algorithm: Application to the femur," in *Proc. 1st Int. Conf. Pervasive Comput., Signal Process. Appl.*, Sep. 2010, pp. 251–254.

[10] K. Koh, Y. H. Kim, K. Kim, and W. M. Park, "Reconstruction of patient-specific femurs using X-ray and sparse CT images," *Comput. Biol. Med.*, vol. 41, no. 7, pp. 421–426, Jul. 2011.

[11] N. Baka, B. L. Kaptein, M. de Bruijne, T. van Walsum, J. E. Giphart, W. J. Niessen, and B. P. F. Lelieveldt, "2D–3D shape reconstruction of the distal femur from stereo X-ray imaging using statistical shape models," *Med. Image Anal.*, vol. 15, no. 6, pp. 840–850, Dec. 2011.

[12] S. Hosseinian and H. Arefi, "3D reconstruction from multi-view medical X-ray images–review and evaluation of existing methods," *Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. 40, pp. 319–326, Dec. 2015.

[13] X.-F. Han, H. Laga, and M. Bennamoun, "Image-based 3D object reconstruction: State-of-the-art and trends in the deep learning era," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 5, pp. 1578–1604, May 2021.

[14] V. Karade and B. Ravi, "3D femur model reconstruction from biplane X-ray images: A novel method based on Laplacian surface deformation," *Int. J. Comput. Assist. Radiol. Surgery*, vol. 10, no. 4, pp. 473–485, Apr. 2015.

[15] A. Vokhmintcev, A. Melnikov, S. Pachganov, and V. Burlutskii, "The new combined closed-solution for 3D reconstruction of environment based on iterative closest point algorithm," in *Proc. 7th Sci. Conf. Inf. Technol. Intell. Decis. Making Support (ITIDS)*, 2019, pp. 23–27.

[16] G. Zheng, "3D volumetric intensity reconsturction from 2D X-ray images using partial least squares regression," in *Proc. IEEE 10th Int. Symp. Biomed. Imag.*, Apr. 2013, pp. 1268–1271.

[17] W. Wei, G. Wang, and H. Chen, "3D reconstruction of a femur shaft using a model and two 2D X-ray images," in *Proc. 4th Int. Conf. Comput. Sci. Educ.*, Jul. 2009, pp. 720–722.

[18] A. Le Bras, S. Laporte, V. Bousson, D. Mitton, J. A. De Guise, J. D. Laredo, and W. Skalli, "Personalised 3D reconstruction of proximal femur from low-dose digital biplanar radiographs," in *Proc. Int. Congr. Ser.*, vol. 1256. Amsterdam, The Netherlands: Elsevier, 2003, pp. 214–219.

[19] S. Akkoul, A. Hafiane, R. Leconge, K. Harrar, E. Lespessailles, and R. Jennane, "3D reconstruction method of the proximal femur and shape correction," in *Proc. 4th Int. Conf. Image Process. Theory, Tools Appl. (IPTA)*, Oct. 2014, pp. 1–6.

[20] P. Gamage, S. Q. Xie, P. Delmas, and P. Xu, "3D reconstruction of patient specific bone models from 2D radiographs for image guided orthopedic surgery," in *Proc. Digit. Image Comput., Techn. Appl.*, 2009, pp. 212–216.

[21] M. K. Lee, S. H. Lee, A. Kim, I. Youn, T. S. Lee, N. Hur, and K. Choi, "The study of femoral 3D reconstruction process based on anatomical parameters using a numerical method," *J. Biomech. Sci. Eng.*, vol. 3, no. 3, pp. 443–451, 2008.

[22] S. Laporte, W. Skalli, J. A. De Guise, F. Lavaste, and D. Mitton, "A biplanar reconstruction method based on 2D and 3D contours: Application to the distal femur," *Comput. Methods Biomech. Biomed. Eng.*, vol. 6, no. 1, pp. 1–6, Feb. 2003.

[23] S. Kolta, A. Le Bras, D. Mitton, V. Bousson, J. A. de Guise, J. Fechtenbaum, J. D. Laredo, C. Roux, and W. Skalli, "Three-dimensional X-ray absorptiometry (3D-XA): A method for reconstruction of human bones using a dual X-ray absorptiometry device," *Osteoporosis Int.*, vol. 16, no. 8, pp. 969–976, Aug. 2005.

[24] A. Goli, E. B. Tirkolaee, and N. S. Aydin, "Fuzzy integrated cell formation and production scheduling considering automated guided vehicles and human factors," *IEEE Trans. Fuzzy Syst.*, vol. 29, no. 12, pp. 3686–3695, Dec. 2021.

[25] T. D. DenOtter and J. Schubert, "Hounsfield unit," in *StatPearls [Internet]*. Treasure Island, FL, USA: StatPearls Publishing, Mar. 2023. [Online]. Available: https://www.ncbi.nlm.nih.gov/books/NBK547721/

[26] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014, *arXiv:1411.1784*.

[27] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2015, pp. 234–241.

[28] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, "Activation functions: Comparison of trends in practice and research for deep learning," 2018, *arXiv:1811.03378*.

[29] D. Wu, Y. Wang, S.-T. Xia, J. Bailey, and X. Ma, "Skip connections matter: On the transferability of adversarial examples generated with ResNets," 2020, *arXiv:2002.05990*.

[30] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[31] S. Ruder, "An overview of gradient descent optimization algorithms," 2016, *arXiv:1609.04747*.

[32] J. Jin, M. Li, and L. Jin, "Data normalization to accelerate training for linear neural net to predict tropical cyclone tracks," *Math. Problems Eng.*, vol. 2015, pp. 1–8, Jul. 2015.

[33] N. Pradhan, V. Singh Dhaka, G. Rani, and H. Chaudhary, "Machine learning model for multi-view visualization of medical images," *Comput. J.*, vol. 65, no. 4, pp. 805–817, Apr. 2022, doi: 10.1093/comjnl/bxaa111.

[34] U. Sara, M. Akter, and M. S. Uddin, "Image quality assessment through FSIM, SSIM, MSE and PSNR—A comparative study," *J. Comput. Commun.*, vol. 7, no. 3, pp. 8–18, 2019.

[35] M. Agarwal, G. Rani, S. Agarwal, and V. S. Dhaka, "Sequential model for digital image contrast enhancement," *Recent Adv. Comput. Sci. Commun.*, vol. 14, no. 9, pp. 2772–2784, Dec. 2021.

[36] K. Han, Y. Wang, H. Chen, X. Chen, J. Guo, Z. Liu, Y. Tang, A. Xiao, C. Xu, Y. Xu, Z. Yang, Y. Zhang, and D. Tao, "A survey on vision transformer," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 1, pp. 87–110, Jan. 2023.