Transforming view of medical images using deep learning

Article *in* Neural Computing and Applications · September 2020

DOI: 10.1007/s00521-020-04857-z

Nitesh Pradhan
Manipal University Jaipur
21 PUBLICATIONS **171** CITATIONS
SEE PROFILE

Vijaypal Singh Dhaka
Manipal Academy of Higher Education
80 PUBLICATIONS **643** CITATIONS
SEE PROFILE

Geeta Rani
Manipal University Jaipur
49 PUBLICATIONS **354** CITATIONS
SEE PROFILE

Himanshu Chaudhary
Manipal University Jaipur
24 PUBLICATIONS **132** CITATIONS
SEE PROFILE

Transforming view of medical images using deep learning

Nitesh Pradhan, Vijaypal Singh Dhaka, Geeta Rani & Himanshu Chaudhary

ONLINE FIRST

Neural Computing and Applications

ISSN 0941-0643

Neural Comput & Applic
DOI 10.1007/s00521-020-04857-z

Springer

Neural Computing and Applications
https://doi.org/10.1007/s00521-020-04857-z

ORIGINAL ARTICLE

Transforming view of medical images using deep learning

Nitesh Pradhan[1] · Vijaypal Singh Dhaka[2] · Geeta Rani[2] · Himanshu Chaudhary[3]

Received: 3 July 2019 / Accepted: 14 March 2020

Abstract

Since the last decade, there is a significant change in the procedure of medical diagnosis and treatment. Specifically, when internal tissues, organs such as heart, lungs, brain, kidneys and bones are the target regions, a doctor recommends 'computerized tomography' scan and/or magnetic resonance imaging to get a clear picture of the damaged portion of an organ or a bone. This is important for correct examination of the medical deformities such as bone fracture, arthritis, and brain tumor. It ensures prescription of the best possible treatment. But 'computerized tomography' scan exposes a patient to high ionizing radiation. These rays make a person more prone to cancer. Magnetic resonance imaging requires a strong magnetic field. Thus, it becomes impractical for patients with implants in their body. Moreover, the high cost makes the above-stated techniques unaffordable for low economy class of society. The above-mentioned challenges of 'computerized tomography' scan and magnetic resonance imaging motivate researchers to focus on developing a technique for conversion of 2-dimensional view of medical images into their corresponding multiple views. In this manuscript, the authors design and develop a deep learning model that makes an effective use of conditional generative adversarial network, an extension of generative adversarial network for the transformation of 2-dimensional views of human bone into the corresponding multiple views at different angles. The model will prove useful for both doctors and patients.

Keywords Deep learning · Conditional generative adversarial network · CT scan · MRI · 2-Dimensional · 3-Dimensional

1 Introduction

In the past few years, there is a boom in the demand of medical imaging techniques. It is due to the effectiveness of these techniques in the correct diagnosis of a disease/defect/deformity. Imaging techniques lie under two categories: 2-dimensional (2-D) and 3-dimensional (3-D).

X-ray is the most common 2-D imaging technique. It distinguishes bone from soft tissues around it. This is advantageous in the case of weight-bearing imaging and dynamic imaging of loin motion (fluoroscopy). Low cost of X-ray machines makes them more available at government hospitals, private hospitals and small as well as large health centers. Moreover, X-ray imaging exposes a patient to less harmful and non-ionizing X-rays. The above-said advantages of 2-D imaging techniques increase their use in medical diagnosis. But these techniques do not give multiple views of an organ or bone. Thus, a doctor gets insufficient information for the diagnosis of a defect/disease. The introduction of 3-D imaging techniques brings revolution in the field of medical imaging. Medical imaging techniques such as computerized tomography scan (CT scan), magnetic resonance imaging (MRI), ultrasound, thermography, positron emission tomography (PET), mammography, electron microscopy and single-photon emission computed tomography (SPECT) are the true

✉ Geeta Rani
geeta.rani@jaipur.manipal.edu

Nitesh Pradhan
nitesh.pradhan@jaipur.manipal.edu

Vijaypal Singh Dhaka
vijaypalsingh.dhaka@jaipur.manipal.edu

Himanshu Chaudhary
himanshu.chaudhary@jaipur.manipal.edu

[1] Department of Computer Science and Engineering, Manipal University Jaipur, Jaipur, Rajasthan, India

[2] Department of Computer and Communication Engineering, Manipal University Jaipur, Jaipur, Rajasthan, India

[3] Department of Electronics and Communication Engineering, Manipal University Jaipur, Jaipur, Rajasthan, India

Published online: 24 March 2020

assistants of a doctor. The images obtained by these techniques give a 3-D view of an organ. The details of these imaging techniques are given in [1].

Images obtained using CT scan and MRI give a 3-D view of anatomy and physiology of an organ. Thus, these techniques help a doctor to diagnose a deformity or a disease. In osteology, a patient-specific 3-D model of a bone is useful for extracting the information about the exact orientation and configuration of a bone. For a bone with complex structure and/or multiple fractures, 3-D reconstruction of its image helps in visualizing chronic bone loss, for example 'glenoid defect' in recurrent shoulder dislocation. 3-D reconstruction is also useful for detecting the degree of osteophytes in an arthritic joint.

In post-modern societies, medical doctors prefer 3-D imaging techniques such as CT scan and MRI rather than 2-D imaging techniques. CT scan exposes a patient to highly ionizing X-rays. These rays make a person more prone to cancer [2]. MRI produces a high intensity of magnetic field in its surroundings; therefore, imaging becomes infeasible for patients having iron and/or steel implants in their bodies. Both the above-stated techniques have a high cost. Thus, these are unaffordable for people with low wages. As per the report of World Bank Group published in 2016 [3], 20.7% of the total population in developing countries like India is economically weak. Moreover, in hospitals at remote locations, CT scan and MRI machines are not available. In public hospitals, the availability of limited resources and heavy demand lead to a waiting time of 3 to 4 months for the scanning of a patient. Therefore, there is an urgent requirement to develop a cost-effective technique that can give multiple views of an anatomy of a body without causing adverse effects on human health. In this manuscript, the authors propose the deep learning-based model for the transformation of 2-D images into their corresponding multiple views. The multiple views can be combined to give a 3-D view of an image. The model uses the generative adversarial network (GAN) and conditional generative adversarial network (CGAN) for the transformation. It receives 2-D X-ray or dual-energy X-ray absorptiometry (DXA) images of a bone as input and gives multiple views of the input images as outputs. The images obtained as outputs resembles CT scan and MRI images.

The remaining manuscript is organized as follows: Sect. 2 includes the survey of related work. It presents a comparative analysis of the existing techniques. Section 3 describes the background of the proposed model. Section 4 demonstrates the CGAN-based model. It presents the architecture, training parameters, training procedure and working of the model. In Sect. 5, the authors present the experimental results. They conclude the work in Sect. 6.

2 Related work

An extensive review of the literature teaches about the use of techniques such as 'direct linear transformation' (DLT) [4], 'free from deformation'(FFD) [5], 'statistical shape model' [6, 7], 'non-stereo radiography corresponding points algorithm' [8, 9], 'deep convolutional neural network' (DCNN) [10], 'Laplacian surface deformation' [11], 'iterative closest point algorithm' [12] and 'partial least square regression' [13] for converting 2-D images into 3-D images. All the above-said techniques take high computation time even when applied on a small dataset.

Authors in [14] proposed an architecture that minimizes the gaps between supervised and unsupervised learning. The architecture makes use of convolutional neural networks (CNNs) and replaces a pooling layer with strided convolutions, i.e., discriminator (*D*) and fractional-strided convolutions, i.e., generator (*G*). In this architecture, authors train GANs on a set of images. They use discriminator and generator networks for feature extraction in supervised learning. They applied GAN and CGAN for image colorization, a transformation of text into an image, image-to-image translation and face generation. It gives an insight to extend the applications of GAN and CGAN in the transformation of medical images.

Chen et al. proposed an 'information theoretic version of generative adversarial network' (InfoGAN) in [15]. InfoGAN follows an unsupervised learning technique. This is a special type of GAN that links a tiny subset of latent observations with corresponding variables. It tries to maximize the complementary information such as mutual information between generated images and latent codes and variational mutual information. The complementary information gives the salient structured semantic features of data distribution. Performance evaluation of InfoGAN proves its effectiveness in learning hairstyles, eyeglasses, emotions, writing styles, lighting pose of 3-D images and predicting background digits. InfoGAN needs a large storage space, to accommodate an extra hyperparameter for latent code.

Zhang et al. [16] applies 'stacked generative adversarial networks' (SGAN) for the transformation of text description into good quality images. SGAN adequately uses an illustrative data from a pre-prepared discriminative network. Thus, it has less computation complexity in comparison with InfoGAN.

Ledig et al. proposed the 'super resolution generative adversarial network' (SRGAN) model in [17]. This model uses a loss function as a combination of content loss and adversarial loss. This model does 4X upscaling of an image quality. The value 4X is decided based on two parameters, namely peak signal-to-noise ratio (PSNR) and structured similarity index (SSIM). SRGAN takes a low-resolution image as an input and produces its corresponding high-resolution image.

In [18], Zhu Yan et al. proposed 'cycle-consistent adversarial networks' (CCAN), for image-to-image translation. This translation is important in application areas such as season transfer, object transfiguration, photo-enhancement and collection style transfer. Images obtained on translation are clearer than input images. This is due to the fact that input images captured by smartphones have low resolution, whereas translated images are generated with shallow depth of field. The depth of field is the region of adequate sharpness inside a photograph that will appear in focus. CCAN is preferable over GAN, as it does not require a paired dataset.

Zhang et al. proposed an image colorization technique in [10]. Image colorization is useful in automatic coloring of old pictures, correct identification of a thief from a sketch image, object recognition from an image, information retrieval from a medical image, etc. This technique uses DCNN approach to convert a grayscale image into its corresponding realistic and vibrant colored image. DCNN works well for non-noisy data. But it is ineffective for noisy data.

Nazeri et al. used deep convolutional generative adversarial network (DCGAN) in [19], for image colorization. DCGAN is advantageous over DCNN as it is effective in handling noisy data. The model uses L*a*b* color space format. Here, L is the lightness or luminance channel for a grayscale version of an image, a is the chrominance for red-green, and b is the chrominance for blue–yellow. L*a*b* color space uses a determination of two extra color channels, i.e., 'a' and 'b.' But in RGB, it uses three channels red, green and blue.

Antipov et al. [20] introduced CGAN as an extension of GAN. They designed age conditional generative adversarial network (ACGAN) to generate face images of a specific age group. Researchers apply a latent vector optimization approach that allows ACGAN to reconstruct an input face image. Its performance evaluation claims that the generated faces are visually good, but there is a loss of identity in 50% of cases. To enhance the identity preservation, Mirza et al. [21] applied CGAN on MNIST dataset. On this dataset, CGAN uses the class labels as conditions and produces digits as an output.

Gauthier in [22] proposed a conditional adversarial network (CAN) for face generation. The author presents a comparative study of CGAN and vanilla GAN for face generation on publically available wild dataset [23]. The author claim that the likelihood of the test set using CGAN and vanilla GAN is roughly the same. But CGAN is advantageous over the vanilla GAN because it easily accepts a multimodal embedding as a condition input y.

The study of the literature uncovers the application areas, advantages and disadvantages of the existing techniques. Thus, it provides a clear idea about the extension of the application area of the existing work.

Table 1 presents a comparative analysis of the existing techniques. Column 1 records the year of publication, column 2 shows the names of authors, column 3 gives the name(s) of technique(s), column 4 displays the application area, column 5 includes the advantages, and the last column highlights the drawbacks of the discussed techniques.

Table 2 presents a comparison of the existing techniques in the field of reconstruction of medical images. These techniques convert 2-D views of medical images into multiple views. Its first column records the year of publication, the second column shows names of authors, the third column gives the name(s) of technique(s), the fourth column includes the advantages, and the last column highlights the drawbacks of the discussed techniques.

The comparison given in Table 1 shows the application areas of GANs. It gives an insight to extend the applications of GANs in the arena of reconstruction of medical images.

The comparison given in Table 2 shows the existing techniques for the conversion of 2-D views of medical images into multiple views. These techniques face the challenges such as ineffectiveness for complex regions or cavities of bone [11], inability to recognize ribs [24], low suitability for real-life applications [25], less effective in dealing with noisy data [13] and high computation time [6]. These challenges need immediate attention.

3 Background

3.1 Generative adversarial network (GAN)

Generative adversarial network is a machine learning model that consists of two different sets of neural networks namely '*generator*' and '*discriminator*.' Both sets are contestants of each other. These learn in an unsupervised mode and become efficient in dealing with noisy data [26]. GAN is efficient in generating samples of photorealistic images for visualizing new design [27].

A generator (G) generates false data items, similar to a training dataset, and tends to wrongly train a model. Its task is to mislead a discriminator (D) with its disguise. In case it generates an image/data item which is unsuccessful in befooling a discriminator, then the model compares a generated image with its corresponding real image. The difference between the two images is named as a loss. This loss is a learning factor for improving a generator. The comparison, loss calculation and learning of generator

Table 1 Techniques showing applications of GANs

Year	Author(s)	Technique used	Application(s)	Advantage(s)	Disadvantage(s)
2014 [21]	Mehdi Mirza, Simon Osindero	CGAN	Digit generation	Uses stochastic gradient descent to solve optimization problem which is faster than gradient descent and batch gradient descent	Use of each tag individually is time-consuming
2014 [22]	Jon Gauthier	CGAN	Face generation	Accuracy is higher than auto-encoder and Boltzmann machines	Requires large training dataset
2016 [10]	Richard Zhang, Phillip Isola, Alexei A. Efros	DCNN	Image colorization	Capable of learning relevant feature(s) from an image	In case of small dataset size, DCNN faces the problem of overfitting
2016 [14]	Alec Radford, Luke Metz, Soumith Chintala	DCGAN	Image classification	Its generators have interesting vector arithmetic properties which allow easy manipulation of semantic qualities of samples generated	Ineffective for audio and video stream
2016 [15]	Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, Pieter Abbeel	InfoGAN	Extraction of writing styles from digits' shapes and pose from lighting of 3D rendered images	Able to learn disentangled representations in completely unsupervised mode	Requires large space because it uses an extra hyperparameter for recording discrete latent codes
2017 [16]	Han Zhang et al.	SGAN	Text to image generation	Resolution of output images is higher than energy-based GAN and conditional pixel CNN	In case all images are virtually identical then model collapses
2017 [17]	Christian Ledig et al.	GANs	Image super resolution	Does not require the prior and posterior probability calculations for generating an image	High computational complexity for training
2017 [20]	Grigory Antipov, Moez Baccouche, Jean-Luc Dugelay	CGAN	Face aging	Does not require labeled data for training, as they learn the internal representations of data	In case of small sample size, it faces the problem of overfitting
2018 [18]	Jun-Yan Zhu et al.	CGAN	Unpaired image to image translation	Useful even though paired dataset is not available	Experiences failure due to distribution characteristic of a training dataset
2018 [19]	Kamyar Nazeri, Eric Ng, Mehran Ebrahimi	GAN	Image colorization	Yields realistic visual content	High computation cost of training

continue until it achieves perfection in generating an image close to its corresponding image available in a training set.

A discriminator identifies a false data item generated by generator. Its task is to distinguish an original training image from a misleading image given by generator. In case the generator misleads the discriminator, the model calculates loss. This loss is used as a learning parameter for discriminator. The same procedure is iterated until the discriminator is detective enough to identify a befooling image/data item.

In this way, both generator and discriminator neural networks compete with each other in a min–max game and thus enhance the learning of each other simultaneously. Both neural networks follow the min–max game with a value function $V(G,\ D)$ as defined in Eq. (1). Generator uses Eq. (1) to minimize the function V. Discriminator uses the same equation to maximize the function V. Here, E_x is an entropy of sample image, $pdata(x)$ is a real data distribution, x is a sample image from real dataset, $D(x)$ is a discriminator network, E_z is an entropy of noisy image, $p_{z(Z)}$ is the probability distribution of generator, z is the sample image from probability distribution, and $G(z)$ is the generator network:

$$\min_{G}\max_{D} V(G,D) = \Big(E_{x\sim pdata(x)}[\log D(x)] + E_{z\sim p_{z(Z)}}[\log(1-D(G(z)))]\Big). \qquad (1)$$

3.2 Conditional generative adversarial network (CGAN)

GANs are ineffective to control the outcome(s) of the generator. Let the authors train the GANs model on publicly available MNIST dataset [28], then the generator generates images of all categories possible in a dataset. It does not generate an image from a specific category. This

Table 2 Comparative analysis of the existing techniques for image reconstruction

Year	Title	Techniques	Advantage	Disadvantage
2015 [11]	Karade and Ravi	Laplacian mesh deformation	Lower computation time than free from deformation (FFD) and thin plate spine (TPS) techniques. Effective for simple region (a structure where either one bone or joint of two bones is observed, for example femur)	Give inaccurate results for a complex region (a structure where multiple bones join together by joints, for example, knee, ankle, elbow) of a bone. Use of only 5 test cases for evaluation of model
2014 [24]	Rizque et al.	Point extraction	Can work in combination with other techniques for disease diagnosis. Cost-effective technique	The designed prototype fails to recognize ribs and pulmonary vessels
2013 [25]	Zhang et al.	Hough transformation	Generalized Hough transform (GHT) covers whole contour rather than corner points; thus, it is effective to deal with noisy data	The designed model is not suitable for real applications such as surgical navigation. Optimal searching strategies are required to avoid matching all projections of the whole contour
2013 [13]	Zheng	Partial least squares regression	Use of 2-dimensional X-ray images for processing reduces the computational time. No requirement for generation of digitally reconstructed radiographs (DDR)	Yields unpredictable results for noisy data
2011 [6]	Baka et al.	Statistical shape models	Training of model on complete dataset is required at the initial phase. After initial phase, it combines 3-dimensional distance-based objective function with automatic edge selection on a Canny edge map for automatic reconstruction	High computational time of approximately 5 min. Computational time is higher than Laplacian mesh deformation, FFD and TPC techniques

can be exemplified as: the generator generates images of all digits from 0 to 9, possible in a dataset. If there is a need for an image of only one digit, say 6, then GANs are ineffective. To solve real-world problems, training of a model with a specific category is a key requirement. Thus, an extension of GAN to conditional generative adversarial networks (CGANs) becomes popular.

An extension of GAN is defined in Eq. (2). Here, E_x is the entropy of sample image, $pdata(x)$ is the real data distribution, $D(x, y)$ is the probability calculated by discriminator for sample image x based on given condition y, E_z is the entropy of noisy image, and $p_{z(Z)}$ is the probability distribution function for a sample image z. $G(z, y)$ is the modeling of the distribution of data:

$$\min_G \max_D \Big(E_{x \sim pdata(x)}[\log D(x, y)] + E_{z \sim p_{z(Z)}}[\log(1 - D(G(z, y)), y)] \Big). \tag{2}$$

CGAN assigns a condition 'y' to all images of a dataset based on their category. Generator of CGAN model calculates the loss '$L1$' based on condition 'y.' Loss '$L1$' is the difference yielded on comparing pixel values of real and its corresponding generated images. It enables the model to generate category-specific images.

CGAN uses a function for generator, as given in Equation (3). The function accepts noisy data(z) and embedding space (y) as input. The condition y includes the 90°/180°/270° view(s) of an image. The condition y given to the generative model is based on external information, drawn from the training data. It produces x as an output image of the generator model. It can be exemplified as: if $y = 90°$ is given as a condition, then the trained model is expected to generate 90° views of an image as output.

Function for discriminator is given in Equation (4). The discriminator model accepts an image x and condition y as input. It predicts the probability of the generated image between 0 and 1.

$$(z \times y) \rightarrow x \tag{3}$$

$$(x \times y) \rightarrow [0, 1] \tag{4}$$

In CGAN, both generator and discriminator receive an individual condition 'y' for generating output at a particular angle. The condition is a training parameter for generator and discriminator to train the model for 'n' number of epochs. It can be exemplified as: if authors want a 90° view of an image for an input image at 0° view, then they pass 90° image as a condition to discriminator. Based on this condition, the discriminator assigns a probability value of the image generated by generator. The same is applicable for 180°, 270° or any other angle.

4 CGAN-based model

In this section, the authors present the architecture, training parameters and training methodology of the proposed model. The model extends the applicability of CGAN in the field of orthopedics.

The model applies GAN with a condition on the generator. Generator generates a random image based on a given condition '*y*.' The discriminator compares the output image of generator with its corresponding image available in real dataset. Based on comparison, it identifies whether the generated image resembles its real image or not. The model converts 2-D X-ray images into their corresponding multiple views. These views are similar to CT scan images. The model effectively deals with noisy data images due to competitive learning of generator and discriminator.

4.1 Architecture of generator and discriminator

The proposed model follows two different architectures. Figure 1 demonstrates the architecture of generator (*G*), and Fig. 2 shows the architecture of discriminator (*D*).

In this model, deep CNN consists of three layers, namely convolutional layer, pooling layer and flatten layer [5]. The convolutional layer works as a filter. It is applied on an image to get its feature map. The feature map is smaller in size than an original image. Now, pooling is applied on this feature map to get pooled feature map. Pooled feature map is smaller in dimension than feature map. The type of pooling to be applied is dependent on its application. The model can choose any one of the max pooling, min pooling or average pooling based on the application. At the next step, the model performs flattening on the pooled feature map. Flattening is an important feature of an image and treated as an input layer neuron of an artificial neural network (ANN) [29]. All the three layers extract useful information from an image.

The generator receives an input image of size 1 × 128 × 128. It applies scaling on an input image. First-half of generator applies downscaling. It reduces original size of an input image from 128 × 128 to 2 × 2. The remaining part of the generator applies upscaling. It magnifies an image from 2 × 2 to its original size 128 × 128. Some filters such as 128, 256 and 512 perform reduction and magnification of an image. Filters are useful in extracting all possible features from an image and hence uncover relevant information recorded in an image. The discriminator receives an image from real data set and its corresponding output image yielded by the generator. It predicts the probability of matching of an image generated by generator with its real image.

4.2 Training parameters

The model is trained using the following parameters.

(a) *Optimizer* An optimizer updates the weights of a network. In deep learning, optimizers such as root mean square propagation (RMSProp), adaptive gradient (AdaGrad), stochastic gradient descent (SGD) and adaptive moment estimation (Adam) are in common use. RMSProp focuses on sign of a gradient. It resolves the problem, if there is a large variation in values of gradients. It is successful in training of full batch. AdaGrad is fruitful to use for sparse data as claimed in [30]. It increases the learning rate for sparse parameters. Simultaneously, it decreases the learning rate for less sparse parameters. SGD chooses a random sample rather than whole batch for one iteration in the training phase. Thus, it reduces the time complexity of training. The model proposed in this manuscript applies Adam optimizer. This optimizer uses the advantages of AdaGrad and RMSProp optimizers [30]. It determines the individual learning rate for each gradient. It is capable of dealing with sparse parameters. Thus, it achieves better results.

Adam stores an exponentially decaying average (m_t) of past gradient. The definition of m_t is given in Eq. (5). Here, β_1 is a forgetting factor for mean gradients. The authors experimentally determined the best possible values of β_1 as 0.5 for the proposed model. g_t is a loss function in the current training iteration *t*.

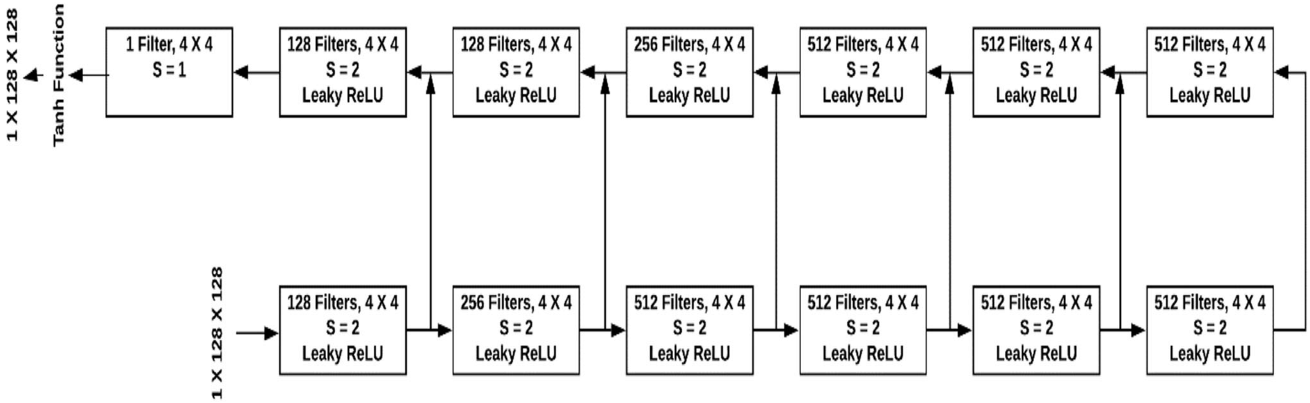

Fig. 1 Architecture of generator of CGAN

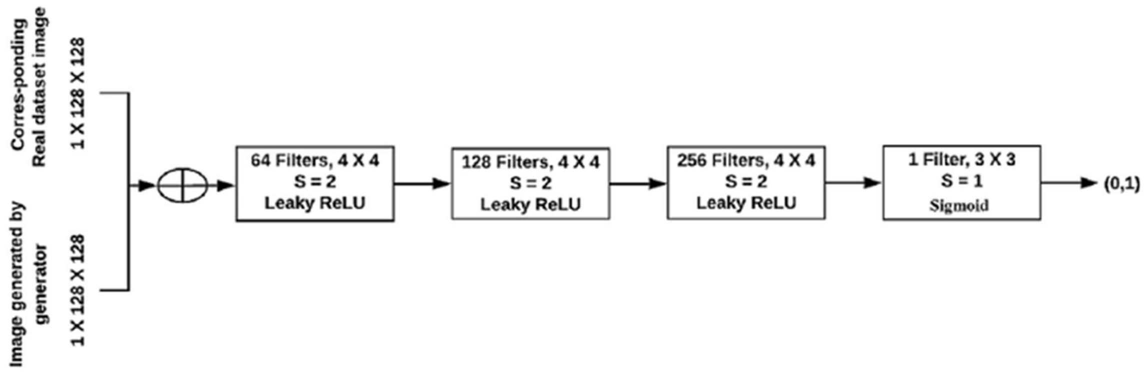

Fig. 2 Architecture of discriminator of CGAN

$$m_t = \beta_1 m_{t-1} + (1-\beta_1)g_t \tag{5}$$

Adam makes use of past square gradients (v_t). Its definition is given in Eq. (6). Here, β_2 is a forgetting factor for the non-centered variance of gradients. On performing a set of experiments, the authors set its value as 0.999 for the proposed model. As per discussion given in [30], the value of β_2 gives more weightage to the previous gradient and less weightage to the current gradient. Thus, it avoids rapid changes which cause instability. In case the value of β_2 is set 1, the second term of Eq. (6) will become zero. Thus, it provides no weightage to the current gradient.

$$v_t = \beta_2 v_{t-1} + (1-\beta_2)g_t^2 \tag{6}$$

(b) *Loss function* In the CGAN-based model, this function calculates the following two types of losses: $L1$ loss—$L1$ loss is the least absolute deviations calculated for the generator. It is the difference that arises on a pixel-to-pixel comparison of a real image with its corresponding generated image. Equation (7) defines $L1$ loss. Here, y_i is the actual/target pixels, w_{ij} is the weight assigned to a neuron, and x_{ij} is an input pixel.

$$L1\text{Loss} = \sum_{i=0}^{N}\left(y_i - \sum_{j=0}^{M} x_{ij}w_{ij}\right)^2 \tag{7}$$

Binary cross entropy loss Binary cross entropy (BCE) loss is calculated for discriminator as well as generator. It uses the negative of the logarithm function as shown in Eq. (8) for binary classification. Here, y_i is the actual label (0 or 1). P is the predicted probability for a class, and N is the total number of samples:

$$\text{BCE Loss} = -\frac{1}{N}\sum_{i=1}^{n}(y_i \log(p) + (1-y_i)\log(1-p)). \tag{8}$$

(c) *Activation function* The role of the activation function in machine learning is to introduce nonlinearity into an output of a neuron. This is important to enhance the learning of a machine.

In the proposed model, the authors use three different categories of activation functions, namely sigmoid, LeakyReLU and tanh.

The discriminator employs the sigmoid activation function. Scaling of its input is done between 0 and 1. Its sigmoid center is 0.5. It resolves the problems such as vanishing of a gradient for a small input and explosion of a gradient for a large value of the input. Sigmoid function yields binary output.

The generator employs the tanh activation function in its architecture. The input values of tanh are scaled from − 1 to 1. This activation function is zero centered. It makes optimization easier. LeakyReLU is applicable for both generator and discriminator. It allows a nonzero constant gradient $\alpha = 0.01$ instead of zero gradients. Thus, it resolves the dying ReLU problem, where the gradient becomes zero.

(d) *Weight initialization* The proposed model uses the backpropagation technique for updating weights. The current weights are dependent on previous weights. This refrains from initializing the weights with zero value. Random initialization of weights is not a good approach because there are fair chances of gradients explosion. There are different ranges for different outputs. So, we took a normal distribution of weights, i.e., mean is 0 and standard deviation is 1.

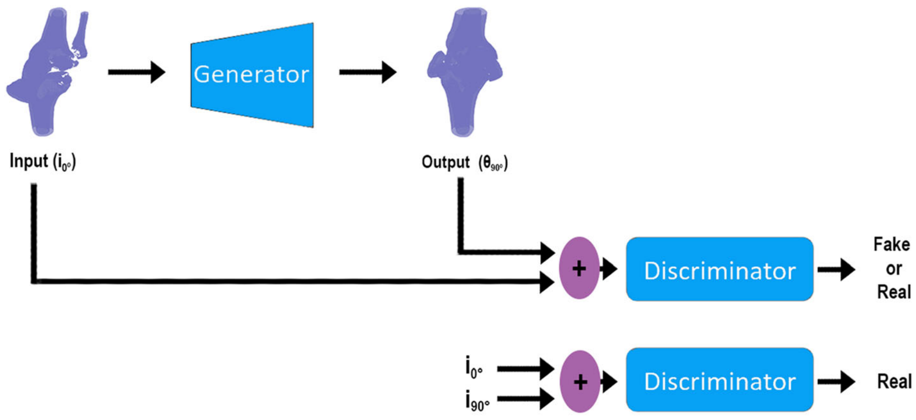

Fig. 3 Working architecture of CGAN-based model

4.3 Working architecture of CGAN-based model

Figure 3 demonstrates the actual operation of the CGAN-based model. The generator takes view of an image as input, say view at an angle of 0°. It generates a random corresponding image, say a view at an angle of 90°. The discriminator takes two input images, an original image and an output image of a generator. It discriminates between two fed images and determines whether the image produced by the generator is real or fake.

Both the generator and discriminator are contestants of each other. These continuously train each other and improve the accuracy of the model. The discriminator is fed with real images as input. It also receives the output of generator as an input. The discriminator compares the real images with images generated by the generator. Based on the comparison, it identifies whether the generator produces fake or real images as output. In case the generator generates fake images as output, the updation in its weights is done. In Fig. 3, '+' sign shows that more than one image is sent as input to the discriminator. The upper '+' sign represents that an input image given to and an output image obtained from Generator are given as input to the discriminator. The lower '+' sign represents that two real images are given as input to the discriminator. One image is an input, and another image is the corresponding output of the input image.

4.4 Training of CGAN-based model

The objective of the proposed model is to visualize 3-D and multiple views of an input image at any angle between 0° and 360°. To achieve this objective, there is a need to train the model in such a way that its generator can produce an image which matches to a real image available in real dataset and its discriminator must be effective in identifying fake image generated by the generator. Both generator and discriminator undergo training simultaneously. The model completes its training in the following 16 steps given in Algorithm 1. All the 16 steps are executed in each iteration for one batch of a dataset. The next iteration works for a new batch of the dataset.

Definitions of terminology used

- Target image: This is an image view that is required as an output. For example, if the authors want to rotate a 0° input image view to a 90° output image view, then 90° view is a target image.
- Scaler probability: The probability is in the range of 0 to 1.
- Score 0: This score is fed to the discriminator to indicate that image generated by the generator is not available in real dataset.
- Score 1: This score is fed to the discriminator to indicate that image generated by the generator is same as the image available in real dataset.
- Hyperparameter (α): α is a hyperparameter that increases the effect of regularization on a model. Regularization is designed to penalize the complexity of a model. An increase in the value lowers the complexity of a model and reduces the error caused due to overfitting. But, a very high value of α increases the error due to underfitting [31]. Therefore, it is necessary to select an optimal value of α. In the model proposed in this manuscript, the authors choose 100 as the optimal value of α. On decreasing the value of α, from 100 to 0, the problem of overfitting increases. On the other hand, increasing the value of α from 100 to infinity, the impact of underfitting increases. The values just below 100 does not cause a strong impact of overfitting. But the impact increases gradually with a decrease in value of α. Similarly, the value of α just above 100 does not cause a strong impact of underfitting. But the underfitting increases gradually with an increase in the value of α [19].

Algorithm 1 Training Procedure of CGAN based Model.

```
Input: An image view at a specific angle, say $0^0$.
Output: Set of images at different angles of view for each input image. Let image view at $90^0$, $180^0$ and
        $270^0$.

 1. Concatenate an input and target image.
 2. Discriminator receives a concatenated image as an input.
 3. Compute the value of BCE loss(D)_real for real image available in dataset. Here target values
    are all 1s.
 4. Back-propagate the BCE loss(D)_real computed in step 3.
 5. Generator generates fake image from an input image.
 6. Concatenate output image of generator with an input image.
 7. Feed concatenated image to discriminator.
 8. Re-compute the value of BCE loss(D)_fake for an image not available in real dataset.  Here
    target values are all 0s.
 9. Calculate Total BCE loss as Sum BCE loss(D)_real and BCE loss(D)_fake.
10. Back-propagate the sum calculated in step 9.
11. Apply Adam optimizer to update the parameters (weights and bias) of the discriminator.
12. Calculate BCE loss (G)_real for the generator using Equation 8. Here target values are all 1s,
    indicating that generated image matches to image available in real dataset.
13. Compute the Loss L1(G)_fake for the generator using Equation 7. Here target values are all
    0s, indicating that generated image is not available in real dataset.
14. Compute the Total Loss(G) as, $Total\ Loss(G) = BCE\ loss(G)\_real + \alpha * L1\ Loss(G)\_fake$
15. Back propagate the Total Loss(G) calculated in step 14.
16. Apply Adam optimizer to update the weights of a generator.

    Repeat steps 1 to 16 for each batch of dataset until the values of Total BCE loss and Total Loss
    L1 becomes negligible. The negligible values indicate that the model is trained to work
    effectively.
```

5 Experiments

For training the proposed model, the authors use Google Colab [32]. This is a freely available online training platform. It provides Tesla K80 GPU and 12 GB RAM. It provides the flexibility of continuous use for 12 h without any halt. The authors use a small batch size of 64 images for performing experiments due to limited RAM size of 12 GB. Here, the batch size represents the number of images in a batch. The authors divide the dataset of 33,820 images into 529 batches. Small batch size provides an option to start training the model before making it familiar with the whole dataset. Thus, it minimizes the problem of generalization and does effective training of the model.

To evaluate the proposed model, the authors collected the real dataset from 'Sir Chhotu Ram Dana Shivam Hospital, Jaipur.' The dataset is a collection of 33,820 X-ray images of bones. The dataset contains images of three different categories of bones. It includes 11,650 images of knee bones, 11,363 images of ankle bones and 10,807 images of lower limb bones.

The authors divide the gathered dataset into training and testing sets in the ratio of 75:25, respectively. Table 3 shows the number of images in training and testing datasets in each of the three categories.

Table 3 Dataset description

Categories	Training images	Testing images	Total
Knee	8737	2913	11,650
Lower limb	8522	2841	11,363
Ankle	8105	2702	10,807
Total	25,364	8456	33,820

5.1 Experimental results

To evaluate the performance of the proposed model, the authors use $L1$ loss for the generator and BCE loss for the discriminator as evaluation parameters. The generator and the discriminator are trained using the $L1$ loss and BCE loss. A variation in the values of $L1$ loss and BCE loss as the number of epochs increases determines the effectiveness of the model. Table 4 presents the values of $L1$ loss at 20, 40, 60, 80 and 100 epochs. Its column 1 contains the number of epochs (number of data entities for experiment), column 2 shows the values of $L1$ loss at an angle of 90°, column 3 contains the values of $L1$ loss at 180°, and column 4 includes the values of $L1$ loss at 270°.

By performing the set of experiments, the authors observed that values of $L1$ loss for the generator module at angles 90°, 180° and 270° vary with the change in the number of epochs. Table 4 and Fig. 4 demonstrate the trends that the values of $L1$ loss sharply decrease with an increase in the number of epochs from 0 to 20. There is a regular and smooth decrease in its values when the number of epochs increases from 20 to 100.

Table 5 presents the values of BCE loss. Column 1 shows the number of epochs, column 2 displays the values of BCE loss at an angle of 90°, column 3 includes the values of BCE loss at 180°, and column 4 shows the values of BCE loss at 270° for 20, 40, 60, 80 and 100 epochs.

By performing the set of experiments, it is observed that the values of BCE loss at angles 90°, 180° and 270° vary with the changes in the number of epochs. Table 5 and Fig. 5 demonstrate the trends for the values of BCE loss with an increase in the number of epochs. It is clear from Table 5 and Fig. 5 that the values of BCE loss randomly fluctuate with the change in the number of epochs at different angles of view. This is due to the fact that the values of BCE loss are not only dependent on the number of epochs but also on the outputs of the generator and the discriminator.

For 90° view, it shows a decrease in values of BCE loss when the number of epochs increases from 20 to 60. The values increase abruptly when the number of epochs changes from 60 to 100.

Table 4 L1 loss of Generator

Epochs	Error_G_90	Error_G_180	Error_G_270
20	0.076344741	0.0791355843	0.0840184341
40	0.0503620976	0.0548225876	0.0505365174
60	0.0504090522	0.0489202047	0.0497219803
80	0.047478915	0.0424315613	0.0451875681
100	0.0392845255	0.0508289727	0.0450121215

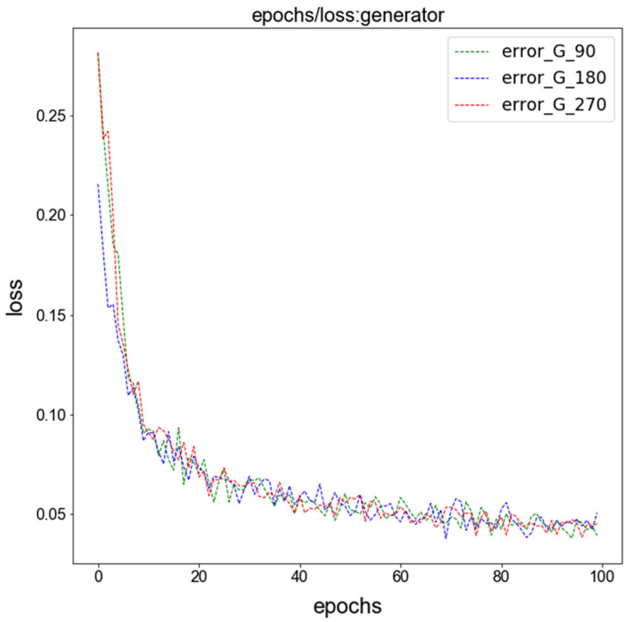

Fig. 4 Trend $L1$ loss versus number of epochs

Table 5 BCE loss of discriminator

Epochs	Error_D_90	Error_D_180	Error_D_270
20	1.4439532897	1.4041874094	1.3840672854
40	1.4297566606	1.4304836812	1.4330456268
60	1.4013079973	1.3913566502	1.4022396201
80	1.4516418288	1.4559470579	1.3426549561
100	1.4588383655	1.3833025032	1.3854327947

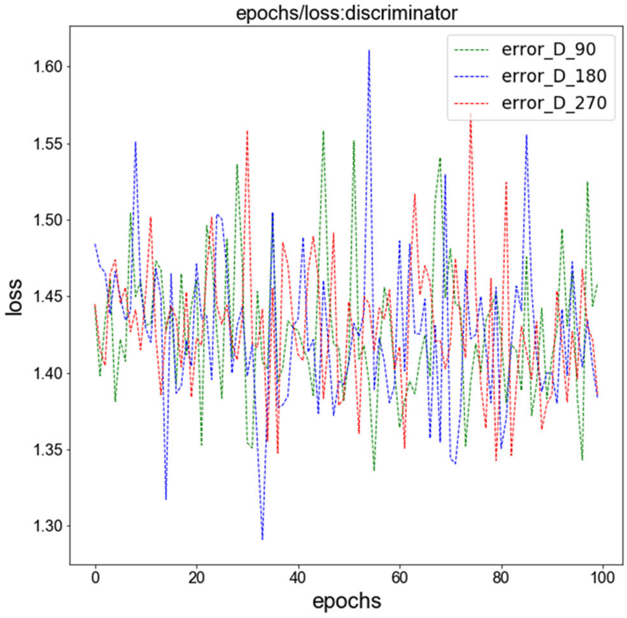

Fig. 5 Trend BCE loss versus number of epochs

For 180° view, there is a random variation in values of BCE loss with a change in the number of epochs. The values increase feebly on changing the number of epochs from 20 to 40. There is a decrease in values of BCE loss on increasing the number of epochs from 40 to 60. The values again show an increase when the number of epochs changes from 60 to 80. The values of BCE loss show a decrease on increasing the number of epochs from 80 to 100.

For 270° view, there is a random variation in values of BCE loss on changing the number of epochs. The values increase in changing the number of epochs from 20 to 40. There is a decrease in values of BCE loss on increasing the number of epochs from 40 to 80. The values show a feeble increase when the number of epochs changes from 80 to 100.

The model receives an image at only one angle of view and predicts three best possible angles. Thus, it presents multiple views of the input image. The model uses the front view at an angle 0° for all the three categories of bone images: knee, lower limb and ankle. It predicts three new angles of view 90°, 180° and 270° for the fed images.

The experimental results on sample test images as given in Fig. 6 prove that multiple views of generated images match to its corresponding real 3-D image available in the dataset. Thus, the model is effective in the reconstruction of multiple views of an image at different angles from its 2-D view at an angle.

6 Discussion and conclusion

Economic and effective treatment is the prime objective of the medical field. To achieve this objective, a correct diagnosis and visualization of diseases/defects/deformities are required. The CGAN-based deep learning model proposed in this manuscript is a contribution in the above-mentioned direction. It receives 2-D X-ray images as input and gives multiple views of input images as outputs. The experimental results shown in Fig. 6 prove that the multiple views of images generated by the proposed model are indistinguishable from real images captured by CT scan or MRI. Thus, the model reduces the requirement of 3-D imaging techniques such as CT scan and MRI. This reduces the cost of medical imaging by taking X-ray images as input and increases the accessibility of 3-D imaging techniques for poor populations and remote locations. The model is also a good contribution toward minimizing the exposure of a patient to the high ionizing and hazardous radiations. Moreover, it effectively overcomes the drawbacks of 2-D imaging techniques such as X-ray which is ineffective in identifying the exact deformity.

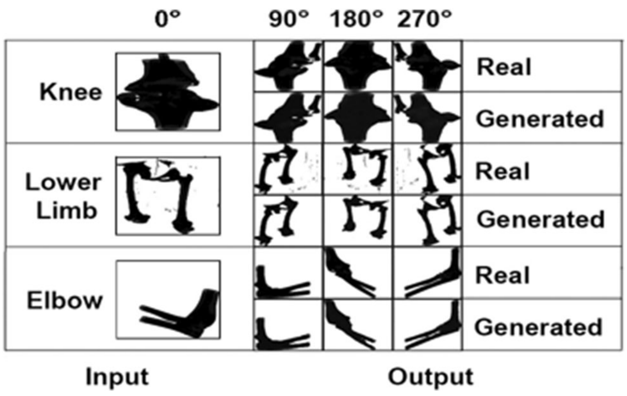

Fig. 6 Comparison of real and generated images of knee, lower limb and elbow

The qualitative comparison with the existing techniques shows that the model is effective in dealing with noisy data. It generates multiple views of an image. Thus, it is effective in detecting a defect or deformity from both simple and complex regions. Low cost, low computation time and applicability for simple as well as complex bone structures make the model suitable for real-life applications.

This technique is applied to visualize a bone's image. Its applications can be extended to visualize the complete anatomy of the body.

Acknowledgements We would like to thank Dr. Nand Kishore Poonia, Managing Director, Sir Chhotu Ram Dana Shivam Hospital, Jaipur, for providing the dataset.

Compliance with ethical standards

Conflict of interest Authors declare that they have no conflict of interest.

References

1. Wani N, Raza K (2018) Multiple kernel-learning approach for medical image analysis. In: Dey N, Ashour AS, Shi F, Balas VE (eds) Soft computing based medical image analysis. Academic Press, London, pp 31–47
2. Watson M, Holman DM, Maguire-Eisen M (2016) Ultraviolet radiation exposure and its impact on skin cancer risk. In: Seminars in oncology nursing, vol 32, no 3. WB Saunders, pp 241–254
3. The World Bank (2016) India's Poverty Profile. https://www.worldbank.org/en/news/infographic/2016/05/27/india-s-poverty-profile. Accessed 17 Feb 2020
4. Zhang B, Sun S, Chi Z et al (2010) 3D reconstruction method from bi-planar radiography using DLT algorithm: application to the Femur. In: 1st international conference on pervasive computing, signal processing and applications. IEEE, pp 251–254. https://doi.org/10.1109/PCSPA.2010.68
5. Koh K, Yo KH, Kim K et al (2011) Reconstruction of patient-specific femurs using X-ray and sparse CT images. Comput Biol Med 41:421–426. https://doi.org/10.1016/j.compbiomed.2011.03.016

6. Baka N, Kaptein LB, Bruijne M et al (2011) 2D–3D shape reconstruction of the distal femur from stereo X-ray imaging using statistical shape models. Med Image Anal 15:840–850
7. Zheng G, Schumann S, Dong X et al (2009) A 2D/3D correspondence building method for reconstruction of a patient-specific 3D bone surface model using point distribution models and calibrated X-ray images. Med Image Anal 13:883–899
8. Le Bras A, Laporte S, Bousson V et al (2003) Personalised 3D-reconstruction of proximal femur from low-dose digital bi-planar radiographs. Int Congr Ser 1256:214–219. https://doi.org/10.1016/S0531-5131(03)00285-1
9. Laporte S, Skalli J, Lavaste F et al (2003) A biplanar reconstruction method based on 2D and 3D contours: application to the distal femur. Comput Methods Biomech Biomed Eng 6:1–6
10. Zhang R, Isola P, Efros A (2016) Colorful image colorization. In: European conference on computer vision, pp 649–666
11. Karade V, Ravi B (2014) 3D femur model reconstruction from bi-plane X-ray Images: a novel method based on Laplacian surface deformation. Int J Comput Assist Radiol Surg 10:473–485. https://doi.org/10.1007/s11548-014-1097-6
12. Akkoul S, Hafine A, Leconge R et al (2014) 3D reconstruction method of the proximal femur and shape correction. In: 4th international conference on image processing theory, tools and application. IEEE, pp 1–6. https://doi.org/10.1109/IPTA.2014.7001939
13. Zhang G (2013) 3D Volumetric intensity reconstruction from 2D X-ray images using partial least-squares regression. In: IEEE 10th international symposium on biomedical imaging, pp 1268–1271
14. Radford A, Metz L, Chintala S (2016) Unsupervised representation learning with deep convolutional generative adversarial networks. In: International conference on learning representations. arXiv:1511.06434
15. Chen X, Duan Y, Houthooft R et al. (2016) InfoGAN: interpretable representation learning by information maximizing generative adversarial nets. arXiv:1606.03657v1
16. Zhang H, Xu T, Li H et al. (2017) StackGAN: text to photo-realistic image synthesis with stacked generative adversarial networks. In: IEEE international conference on computer vision. arXiv:1612.03242
17. Ledig C, Theis L, Huszár F et al (2017) Photo-realistic single image super-resolution using a generative adversarial network. In: IEEE conference on computer vision and pattern recognition. arXiv:1609.04802
18. Zhu Yan J, Park T, Isola P et al (2018) Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceeding of the IEEE international conference on computer vision, pp 2223–2232
19. Nazeri K, Eric N, Ebrahimi M et al (2018) Image colorization using generative adversarial network. arXiv:1803.05400v5
20. Antipov G, Baccouche M, Dugelay Luc J (2017) Face aging with conditional generative adversarial network. In: IEEE international conference on image processing, pp 2089–2093. https://doi.org/10.1109/ICIP.2017.8296650
21. Mirza M, Osindero S (2014) Conditional generative adversarial nets. arXiv:1411.1784
22. Gauthier J (2014) Conditional generative adversarial nets for convolutional face generation. Class Project for Stanford CS231N: Convolutional Neural Networks for Visual Recognition, Winter Semester 2014(5):2.
23. Huang B G, Ramesh M, Berg T et al. (2007) Labeled faces in the wild: a database for studying face recognition in unconstrained environments. Technical Report 07–49, University of Massachusetts, Amherst
24. Rizqie Q, Pahl C, Dewi DE, Ayob MA, Maolana I, Hermawan R, Soetkino RD, Suprivanto E (2014) 3D coordinate reconstruction from 2D X-ray images for guided lung biopsy. WSEAS Trans Biol Biomed, EISSN, pp 2224–2902
25. Zhang J et al (2013) 3-D reconstruction of the spine from biplanar radiographs based on contour matching using the Hough transform. IEEE Trans Biomed Eng 60(7):1954–1964
26. Goodfellow I, Abadie P J, Mirza M et al (2014) Generative adversarial nets. Adv Neural Inf Process Syst 2:2672–2680.
27. Raza K, Singh NK (2018) A tour of unsupervised deep learning for medical image analysis. arXiv preprint arXiv:1812.07715
28. Ernst K, Baidyk T (2004) Improved method of handwritten digit recognition tested on MNIST database. Image Vis Comput 22:971–981. https://doi.org/10.1016/j.imavis.2004.03.008
29. Sharma MK, Dhaka VP (2016) Segmentation of English offline handwritten cursive scripts using a feedforward neural network. Neural Comput Appl 27(5):1369–1379
30. Ruder S (2017) An overview of gradient descent optimization algorithms. arXiv:1609.04747v2
31. The scikit-yb developers. Alpha Selection. https://www.scikit-yb.org/en/latest/api/regressor/alphas.html. Accessed Feb 2020
32. Pessoa et al (2018) Performance analysis of google colaboratory as a tool for accelerating deep learning applications. IEEE Access. https://doi.org/10.1109/ACCESS.2018.2874767

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.