J Med Syst (2018) 42:231
https://doi.org/10.1007/s10916-018-1072-9

IMAGE & SIGNAL PROCESSING

Generative Adversarial Network for Medical Images (MI-GAN)

Talha Iqbal[1] · **Hazrat Ali**[1]

Received: 31 May 2018 / Accepted: 17 September 2018

Abstract

Deep learning algorithms produces state-of-the-art results for different machine learning and computer vision tasks. To perform well on a given task, these algorithms require large dataset for training. However, deep learning algorithms lack generalization and suffer from over-fitting whenever trained on small dataset, especially when one is dealing with medical images. For supervised image analysis in medical imaging, having image data along with their corresponding annotated ground-truths is costly as well as time consuming since annotations of the data is done by medical experts manually. In this paper, we propose a new Generative Adversarial Network for Medical Imaging (MI-GAN). The MI-GAN generates synthetic medical images and their segmented masks, which can then be used for the application of supervised analysis of medical images. Particularly, we present MI-GAN for synthesis of retinal images. The proposed method generates precise segmented images better than the existing techniques. The proposed model achieves a dice coefficient of 0.837 on STARE dataset and 0.832 on DRIVE dataset which is state-of-the-art performance on both the datasets.

Keywords GAN · Medical imaging · Style transfer · Deep learning · Retinal images

Introduction

In recent times, strong interest has emerged in the use of computer-aided medical diagnosis [1–10]. Computer aided diagnosis relies on advanced machine learning and computer vision techniques [11]. Today, majority of the medical professionals use computer-aided medical images for diagnosis purposes. Retinal vessel network analysis gives us information about the status of general system and conditions of the eyes. Ophthalmologists can diagnose early sign of vascular burden due to hypertension and diabetes as well as vision threatening retinal diseases like Retinal Artery Occlusion (RAO) and Retinal Vein Occlusion (RVO) from abnormality in vascular structure [12]. To aid this kind of analysis, automatic vessels segmentation methods have been extensively studied. Recently, deep learning methods have shown potential to produce promising results with higher accuracy, occasionally better than medical specialist in the field of medical imaging [13]. Deep learning also improves efficiency of analyzing data due to its computational and automated nature but most of the medical images are often 3 dimensional (e.g. MRI and CT) and it is difficult as well as inefficient to produce manually annotated images. In general, medical images are inadequate, expensive and offer restricted use due to legal issues (patient privacy). Moreover, the datasets of medical images available publicly often lack consistency in size and annotation. This makes them less useful for training of neural networks, which are data-hungry. This directly limits the development of medical diagnosis systems. So, generation of synthetic images along with their segmented images will help in medical image analysis and provide better diagnosis systems.

Recent work in the domain of medical imaging has shown possibility of improved performance even on small datasets. This has became possible through provision of some prior knowledge in a deep neural network [14]. U-net [13] architecture is popular for segmentation of bio-medical images, which shows how strongly an augmented data can be utilized to cope with low amount of training data available to train deep networks. Data augmentation is easy to implement and gives good results but it is only able to give fixed variations for any given dataset and requires the

This article is part of the Topical Collection on *Advanced Computational Intelligence and Soft Computing in Medical Imaging*

✉ Hazrat Ali
hazratali@ciit.net.pk

[1] Department of Electrical Engineering, COMSATS University Islamabad, Abbottabad Campus, Islamabad, Pakistan

Published online: 12 October 2018

augmentation to fit in the given dataset. Impressive results are achieved by Gatys et al. [15] by application of deep learning algorithm. Similar approach with modifications has been used by [16, 17], reducing the computational complexity. More traditional approaches for segmentation of filamentary structured images have been reported in [18] and [19].

Generative Adversarial Networks (GANs) are useful for many applications like unsupervised representation learning [20] or image-to-image translation [21]. Typically, vessel segmentation task is considered as image translation problem where segmented vessel map is produced at output using fundoscopic image as an input to the model. We can have clearer and sharper vessel segmented masks, if we constrain our output to resemble the annotation done by human experts. For image generation, Generative Adversarial Networks (GANs) [22] provide a different approach. GANs are divided into two networks i.e. Generator and Discriminator. Both are trained to compete with each other like min-max game. Goal of discriminator is to classify the input image as real or synthetic image while generator goal is to generate images that are close to real so that discriminator gets fooled by it. To deal with over-fitting, generator is never shown the training dataset and is only fed with the gradient of discriminator decision. The training process is highly affected by the values of hyper-parameters. The major problem in GANs is to find Nash Equilibrium to stop the training process of generator and discriminator, which can otherwise lead to training instability.

Number of GANs like [23–26] have been developed. DCGAN [23] introduced set of constraints which stabilized the training of the model. CGAN [24] trained the model and generated output conditioned to some auxiliary information. LAPGAN [25] uses cascade formation of convolutional neural networks within framework of Laplacian pyramid for the generation of the new images. InfoGAN [26] learns disentangled representations in unsupervised manner. GANs have performed well on small medical image datasets as discussed in [27]. The authors in [27] have used GANs for unsupervised adaptation of the multi-model medical images.

In this paper, we propose a new approach for generation of **retinal vessel images** as well as their segmented masks using generative adversarial networks. The closest to our work is that of [28]. The method proposed in [28] is limited to generation of fixed output for a given input. On the contrary, our method can produce unlimited number of synthetic images from same input. Moreover, unlike [28] that uses hundred to millions of training examples, our approach works on only tens of training images. Our method not only extracts sharp and clearer vessels having less false positives as compared to existing methods but also achieve state-of-the-art performance on two publicly available datasets i.e. STARE [1]and DRIVE [2]. Our model, when trained on the generated datasets, gives comparable results with the network trained on real data images. The major contributions of this work are:

- We propose a GAN which is able to generate realistic looking retinal images from only tens of examples, unlike [28], which requires hundreds of training examples. The proposed GAN has a re-defined set of loss functions to get better generation and discrimination ability.
- Generally, style transfer algorithm aims to represent output image as generic representation. We propose a variant of the style transfer based on particular style representation , instead of generic representation, provided by additional input.
- Generally, GANs are computational expensive. Unlike the traditional training of GANs, we propose a new technique. In each iteration, we update generator twice than discriminator to get quicker convergence. Thus, the overall training time is reduced significantly.

The rest of the paper is organized as follows: We explain Generative Adversarial Network and the proposed design of our model in Section II. We have discussed experimental setup and results in Section III. Finally, the paper is concluded in Section IV.

Generative adversarial network for medical imaging (MI-GAN)

We generate segmented images using ground truth segmented images of each dataset. To produce realistic filamentary structured output, we imitate image formation process G_θ i.e. image generation function. Input to this function is segmented binary image y and normally distributed noise z. Our goals are:

1. Learn G_θ function from very small training set.
2. Explore conditional probability of image formation distribution $p(X|y)$. Here X is random variable used to show feasible image realization conditioning for any particular realization y. In simple words, by varying noise vector z, we should get plausible as well as distinct RGB image from same segmented input y.
3. Add these synthesized images to training set and improve the overall performance of the supervised segmentation.
4. Interesting thing about our method is that a specific image style learned from an additional input x_s is

[1] http://cecas.clemson.edu/~ahoover/stare/

[2] https://www.isi.uu.nl/Research/Databases/DRIVE/

directly transferred to output image $\hat{x}$. Note that the style of the x_s can be different from original image x. Similarly, their corresponding segmented images y_s and y are also unrelated.

The achievement of these goals is challenging as image generation process is complex process and G_θ is a sophisticated function. Nonetheless, using a powerful deep learning methodology i.e. GANs, an end to end machine learning algorithm is proposed in this work. Figure 1 shows the overall flow of our proposed approach.

Along with generator G_θ, we have discriminator D_γ which gives output [0 *or* 1] depending on the input. Discriminator function is to classify synthetic image as 0 or synthetic and real image as 1 or real. Mathematically: $X := \hat{x}$ i.e. generated image then (d $\rightarrow$ 0) and if $X := x$ i.e. real image from dataset then (d $\rightarrow$ 1) (see Fig. 1). Here d is discriminator output.

The training mechanism of GANs can be considered as two players competing against each other in a min-max game. Each player wants to get better than other and ultimately become the winner. Based on this analogue we define the optimization problem characterizing the G and D interplay, as:

$$\begin{aligned}\min_{\theta}\max_{\gamma} L\ (G_\theta, D_\gamma) \ = \ & E_{x,y\Rightarrow p(x,y)}\ [log D_\gamma(x,y)] \\ & + E_{y\Rightarrow p(y),z\Rightarrow p(z)} \\ & \times \left[log\left(1 - D_\gamma\ (G_\theta(y,z),y)\right)\right] \\ & + \lambda L_{DEV}(G_\theta) \end{aligned} \quad (1)$$

First term $E_{x,y\Rightarrow p(x,y)}\ [log D_\gamma(x,y)]$ says, input image is real along with its segmentation mask so discriminator will identify it as 1 (real). Second term $E_{y\Rightarrow p(y),z\Rightarrow p(z)}\ [log(1 - D_\gamma\ (G_\theta(y,z),y))]$ says, the input segmented image is with generated image and so discriminator will tend to identify it as 0 (fake). This last term is introduced to make sure that the synthetic image produced by the generator is not too much deviated from the real image. Here λ is a trade-off constant and $\lambda > 0$ while $L_{DEV}\ (G_\theta)$ can be considered as simple L1 loss function, denoted as:

$$L_{DEV}\ (G_\theta) = E_{x,y\Rightarrow p(x,y)}\ [\|x - G_\theta(y,z)\|_1] \quad (2)$$

During the training, generator tries to generate realistic looking synthesized images so that it may fool the discriminator and let the discriminator classify these generated images as real ones. The generator achieves this by minimizing Eq. 1, which is our objective function. Practically, by using the approximation scheme as in [29], this can be done by minimizing $-log\ (D_\gamma(G_\theta(y,z)))$, which is a simpler form than original $log\ (1 - D_\gamma\ (G_\theta(y,z)))$. Overall generator loss can be defined as:

$$\begin{aligned} L_G(G_\theta) \ = & -\sum_i log\ D_\gamma(G_\theta(y_i,z_i),y_i) \\ & + \lambda\ \|x_i\ - G_\theta\ (y_i,z_i)\|_1 \end{aligned} \quad (3)$$

On the other side, discriminator D tries to properly classify and separate synthesized images from the real images by maximizing the objective function (see Eq. 1). The discriminator loss is determined by:

$$\begin{aligned} L_D(D_\gamma) \ = & \sum_i log\ D_\gamma(x_i,y_i) \\ & + log\ (1 - D_\gamma\ (G_\theta(y_i,z_i),y_i)). \end{aligned} \quad (4)$$

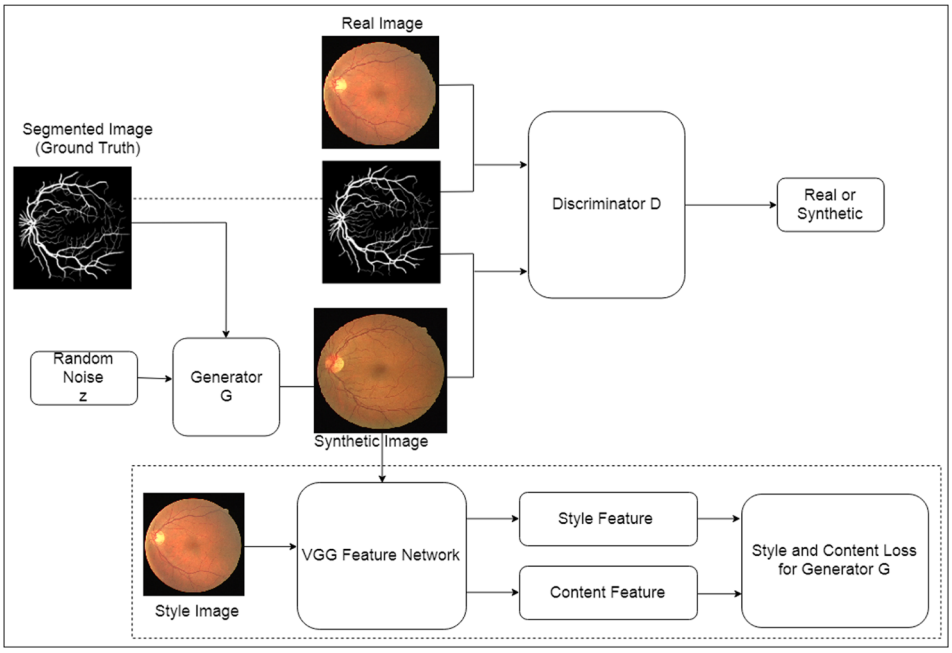

Fig. 1 Flowchart of our method

The empirical summation is used to approximate the expectation value. The training is done by alternating the optimization operation between the generator and discriminator objective function. This is same as adopted by different GANs [23, 29, 30]. Unfortunately, these GANs do not provide a formal guarantee that this optimization process will converge and we will be able to reach at Nash Equilibrium point. Different tricks are available which guarantee convergence of GANs training process and produces reasonable realistic looking synthesized images at output [23, 29, 30]. Figure 1 illustrates overview of the work flow of proposed GAN model, excluding the dashed box. Next we discuss specific neural network architecture of our Generator G and Discriminator D in detail.

Generator and discriminator architecture in MI-GAN

Explained in [28, 31, 32] and [33], commonly used technique of encoder-decoder is adopted here. This allows us to introduce noise code in natural manner. Encoder acts as feature extractor. It is a multiple layered neural network which captures local data representation in first few layers and goes on to capture more global representation as we move deep inside the neural network. A 400 dimensional random noise code z is fully connected to first layer of the network (see Fig. 2). This noise code is then reshaped. One thing to note is that for all the layers of G and D, we use kernel with fixed size and there are two strides with no pooling layers. Meanwhile in our case, it is important for the generator to respect morphology of input segmented image while generating output images. To do so, the 'skip connections' of U-Net [34] are taken into consideration. In skip connections approach the previous layer is mirrored and then duplicated by appending it to the current layer. Odd numbered layers are skipped and the center coding is considered as origin. Note that if we have small image size and a deep neural network, the encoder-decoder framework does work well even without using skip connections. However, we are working with 512 × 512 sized images (which is a large size) and our network is relatively deep.

Training such a model is challenging. The main challenge one may face while using deeper network is of vanishing gradient over a long path during error back-propagation. 'Skip connections', used similarly as in residual nets [21], allows us to pass the error gradients directly from decoder layer to its corresponding encoder layer. This facilitates the memorization of local and global shapes representation as well as their corresponding textures encountered in training dataset, thus we are able to generate better results. We use the basic architecture of the network proposed in [23] to build layers of our generator having multiple convolution layers, Batch Normalization and Leaky ReLU components as shown in Fig. 2. The activation function used to squash the output of the final layer is $tanh$. This function limits the output value between 1 and -1. With our generator, the discriminator network is also built by convolution layers, Batch Normalization and Leaky ReLU, as shown in Fig. 3. The activation function used at output layer is $sigmoid$ instead of $tanh$. After every convolution the feature map size is halved. For example, as we have input image of 512 × 512 so after one convolution layer image size will be decreased to 256 × 256. The number of feature maps (filters) are doubled from 32 through 512 as we move from first to last layer.

Uptill here, we have explained how our proposed approach learns the generic representation from a small training dataset and use it to employ generation of synthesized segmented images. Next, we discuss the segmentation process and a variant of style transfer technique.

Segmentation technique

For segmentation, we utilize gold standard segmented images. We add a loss function that penalizes the distance between gold standard images and output segmented images. This loss is defined as binary cross-entropy i.e.

$$L_{SEG}\,(G_\theta) = E_{x,y\Rightarrow p(x,y)} - y\,log G_\theta(x) - (1-y)\,log(1-G_\theta(x)). \quad (5)$$

The objective function can be formulated by summing the GAN objective function and segmentation loss. So, the new objective function is as follows:

$$L_G(G_\theta) = -\sum_i log D_\gamma\,(G_\theta(y_i, z)) + \lambda L_{SEG}(G_\theta). \quad (6)$$

λ is used to balance both the objective functions.

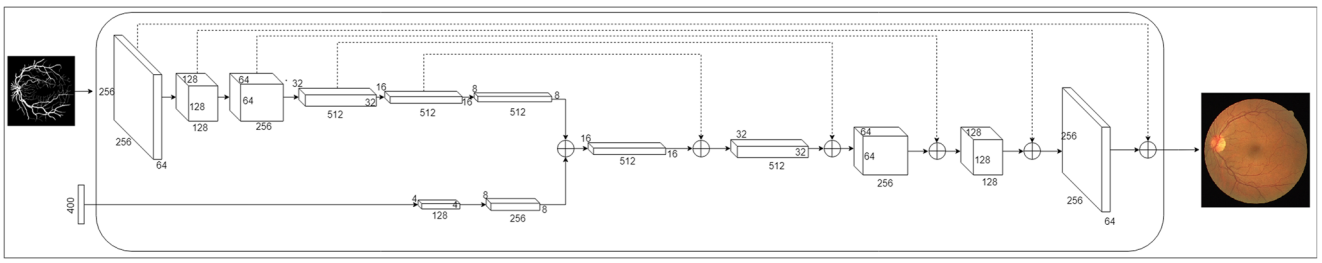

Fig. 2 Generator structure

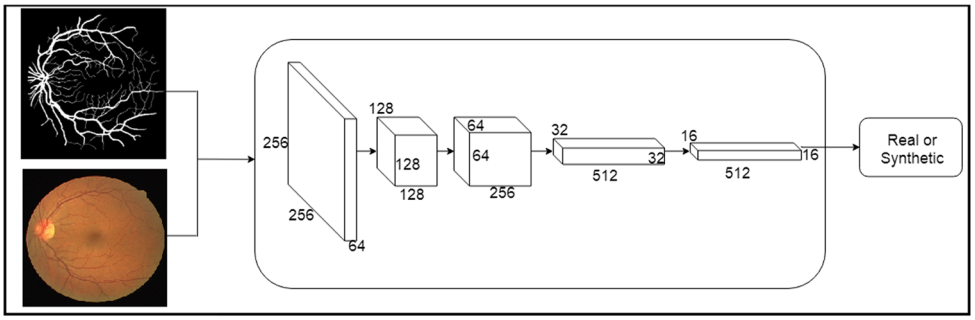

Fig. 3 Discriminator structure

Style transfer variant for MI-GAN

Recent advancement in image style transfer, such as [35], inspired us to use this technique in the field of medical imaging. Here given an input segmentation image y which delineates content of its filamentary structure, we expect that the generated image $\hat{x}$ possess the unique texture (referred as style) of the input x_s which is our target, while still adhering the content of y presented during the training. The difference of our style transfer approach from original style transfer is that instead of generic representation, our synthesized image is based on a particular style representation provided by x_s. The procedure we follow is that we introduce a style image as an additional input along with training input i.e. a new segmented image x_s is introduced having different style and texture. Note that in general x_s has its own filamentary structure (segmentation), which is different from other input y. Nonetheless, this does not affect the performance of generating synthesized images using our method. It is worth noticing that the proposed methodology is practically implementable in biomedical imaging field. On one hand we have very less annotated images available while on the other hand there are a lots of unannotated images available on world wide web which can be used as potential style inputs.

The overall training and testing methodology of this new algorithm is same as we have described in our approach. The training is carried out in form of batches for all n annotated examples in training set. The generator and discriminator is same as mentioned before but the only difference is that in objective function (see Eq. 1), a new cost term $L_{ST}(G_\theta)$ is introduced, which replaces $\lambda L_{DEV}(G_\theta)$ in Eq. 1. We follow style transfer idea proposed in [16, 17] to use the Convolutional Neural Network (CNN) of VGG-19 [36] for extraction of the features from this multi-layered network. VGG-19 network architecture is basically a series of five CNN blocks of VGG net. Each block further consists of two to four consecutive CNN layers of same size. Let us define some notations for convenience. Let Γ be the index for a set of CNN blocks and γ is the index of a particular block where $\gamma \in \Gamma$. Set of layers be represented by $\Lambda(\gamma)$ or Λ. Here layer index is λ such that $\lambda \in \Lambda$. Now the segmented image X is denoted as $\phi_\gamma^\lambda(X)$, irrespective of real image x or generated $\hat{x}$. VGG-19 network is obtained by training the *ImageNet omega classification* problem, which is explained in detail in [36]. Optimization problem for this style transfer algorithm is explicitly incorporated with two perceptual losses i.e. style loss and content loss of [15], as well as total variational loss.

Style loss This loss is used to minimize total textural deviation between target style x_s and generated image $\hat{x}$. To calculate this loss, consider Γ_s showing set of CNN blocks and for each block $\gamma_s \in \Gamma_s$. The set of layer is represented by Λ_s. λ_s-th layer of γ_s block is defined as $\phi_{\gamma_s}^{\lambda_s}(X)$. Here, $X = \hat{x}$ or $X = x_s$. Total number of interest feature maps inside current layer λ_s is denoted by $|\lambda_s|$. Let i and j be index of interest feature map and k be index of an element of current feature map. Information of the corresponding feature is characterized using Gram matrix $G_{\gamma_s}^{\lambda_s}(X)$ which belongs to $R^{|\lambda_s|\times|\lambda_s|}$. Each element $G_{\gamma_s,ij}^{\lambda_s}(X)$ is defining an inner product of i^{th} and j^{th} interest feature maps in λ_s^{th} layer of block γ_s. Mathematically,

$$G_{\gamma_s,ij}^{\lambda_s} = \sum_k \phi_{\gamma_s,ik}^{\lambda_s} \; \phi_{\gamma_s,jk}^{\lambda_s}. \tag{7}$$

The style loss of x_s and $\hat{x}$ during training is defined as:

$$l_{sty}(G_\theta) = \sum_{\gamma_s\in\Gamma_s,\lambda_s\in\Lambda_s} \frac{\varpi_{\gamma_s}}{W_{\gamma_s} H_{\gamma_s}} \times \left\| G_{\gamma_s}^{\lambda_s}(x_s) - G_{\gamma_s}^{\lambda_s}(\hat{x}) \right\|_F^2. \tag{8}$$

Here $\|.\|_F$ is matrix Frobenius norm, ϖ_{γ_s} represents weight of γ_s-th block Gram matrix. Note that by definition $\hat{x} = G_\theta(y, z)$.

Content loss Following notations are considered for content loss: Γ_c is index of set of convolution neural network blocks while each block index is as $\gamma_c \in \Gamma_c$. Set of layers is represented as Λ_c. We expect the synthesized output $\hat{x}$ will abide the segmentation pattern of the real image (input image) x. To make sure this happens, we encourage output image to minimize the Frobenius norm

of the difference between input and output CNN features. Mathematically,

$$l_{cont}(G_\theta) = \sum_{\gamma_c \in \Gamma_c, \lambda_c \in \Lambda_c} \frac{1}{W_{\gamma_c} H_{\gamma_c}} \left\| \phi^{\lambda_c}_{\gamma_c}(x) - \phi^{\lambda_c}_{\gamma_c}(\hat{x}) \right\|^2_F. \quad (9)$$

Total variational loss Total variational loss is incorporated using following equation for spatial smoothness of the generated images.

$$l_{tv}(G_\theta) = \sum_{w,h} (\left\| \hat{x}_{w,h+1} - \hat{x}_{w,h} \right\|^2_2 + \left\| \hat{x}_{w+1,h} - \hat{x}_{w,h} \right\|^2_2). \quad (10)$$

Here $\hat{x}_{w,h}$ denotes pixel value of location in generated image $\hat{x}$, where $w, h \in W, H$ respectively. Summarizing all the three loss functions combined together gives us *Style Loss* $L_{ST}(G_\theta)$,

$$L_{ST}(G_\theta) = \omega_{cont} l_{cont} + \omega_{sty} l_{sty} + \omega_{tv} l_{tv}. \quad (11)$$

So, now we modify L_{DEV} in equation 1 by this style transfer loss L_{ST}. The new objective function for generator G becomes:

$$L_G(G_\theta) = -\sum_i log D_\gamma(G_\theta(y_i, z)) + L_{SEG}(G_\theta) + L_{ST}(G_\theta). \quad (12)$$

Discriminator objective function remains unchanged (see Eq. 4). Style transfer from input style x_s is obtained using back-propagation optimization of the above objective function.

Fig. 4 (**From Top to Bottom**) DRIVE Dataset Images with their ground truth and STARE Dataset Images with their ground truth

Table 1 Comparison of different models having different discriminators

Model	DRIVE		STARE	
	ROC	PR	ROC	PR
U-Net [36]	0.970	0.886	0.973	0.902
Pixel GAN [12]	0.971	0.889	0.967	0.897
Patch GAN-1 (10x10) [28]	0.970	0.889	0.976	0.903
Patch GAN-2 (80x80) [28]	0.972	0.893	0.977	0.908
Image GAN [39]	**0.980**	**0.914**	**0.983**	**0.916**

The bold text is meant to highlight the results obtained by the method

Experimental setup

Datasets preparation

For evaluation of the proposed approach, we use two benchmark datasets. The first is DRIVE dataset and the second is STARE dataset. These both datasets include a broad spectrum of vascular structured retinal images. The image sizes and number of training examples are different in each dataset. In DRIVE dataset there are 20 training examples with image size of 584×565 while STARE dataset has 10 training images with image size of 700 × 605. The images in both the datasets are roughly similar. In pre-processing stage all the images are re-sized to 512 × 512. Images in DRIVE dataset contain large size background area thus they are cropped into 565 × 565 sized sub-image centered to the original one to make sure all the fore-ground pixels are still contained in the new image. Then this image is again re-sized to 512 × 512 using bi-cubic interpolation. Images in STARE dataset has rather small background margins (area outside fore-ground mask) so they are directly converted to 512 × 512 using bi-cubic interpolation. Pixel values of all the input signals are scaled down in-between -1 and 1 so that our generator should learn to generate synthetic image of size 512 × 512. In the last stage these images are again up-sampled to there original sizes. The final result is obtained by applying circular mask to the segmented image so that only inside pixels are retained as a fore-ground. Figure 4 shows few fundoscopic images from DRIVE (upper row) and STARE (lower row) along with their ground truths.

Parameters of proposed model

The 3D boxes in generator as well as in discriminator (in Figs. 2 and 3) shows CNN layer with its number of features size. Edges of the boxes show the convolutional or de-convolutional operation having filter size of $w_f \times h_f \times l_f$. Here we have considered $4 \times 4 \times l_f$, where l_f is self-manifested by third dimension of consecutive layer. The number in the Figs. 2 and 3 specify intrinsic parameters of the networks. For example, length of the noise vector is 400 and size of first layer is 256 × 256 × 64. In generator G, the sign $\oplus$ along with two directed edges pointing inward shows concatenation operation. Let us see the first $\oplus$; here concatenation operation takes place between 8 × 8 × 256 tensor and 8 × 8 × 512 tensor to produce 8 × 8 × 786 tensor. This concatenation operation is followed by deconvolution operation using filter size of 4 × 4 × 512 which in-return produces 3-D box with size of 16 × 16 × 512.

We update generator G twice and then update discriminator D during the learning iteration to balance the overall learning process of generator and discriminator. Noise is sampled element-wise from zero mean Gaussian having standard deviation of 0.001 during training. Standard deviation is changed to 1 and sampling is done in same manner as above, when we evaluate our algorithm. Based on observation, this change in standard deviation is useful to maintain proper level of diversity as we have very small-size data. To get better training of generator and discriminator in our model, batch normalization [37] is used right after every convolutional layer.

The VGG-19 network is used to produce feature descriptor for style transfer algorithms. Output of this network is

Table 2 Comparison of proposed method with other existing techniques on basis of AUC ROC and PR and Dice Score

Method	DRIVE			STARE		
	Dice score	AUC ROC	AUC PR	Dice score	AUC ROC	AUC PR
Our method	**0.832**	**0.984**	**0.916**	**0.838**	**0.985**	**0.922**
Kernel boost [11]	0.800	0.931	0.846	–	–	–
N^4 - fields [13]	0.805	0.968	0.885	–	–	–
DRIU [40]	0.822	0.979	0.906	0.831	0.972	0.910
Wavelets [41]	0.762	0.943	0.814	0.774	0.969	0.843
HED [42]	0.796	0.969	0.877	0.805	0.976	0.888
Human expert	0.791	–	–	0.76	–	–

The bold text is meant to highlight the results obtained by the method

style and content features. Values of the parameters used in this network are: $\Gamma_s = 1, 2, 3, 4, 5$ and $\Gamma_c = 4$. $\Lambda_s = 1$ for style loss and $\Lambda_c = 0$ for content loss. ϖ_{γ_s} is kept fixed for all blocks and is set to 0.2. The weights of three loss functions are as follow: $\omega_{cont} = 1$, $\omega_{sty} = 10$ and $\omega_{tv} = 100$.

Images are augmented by rotation and left-right flip and then normalization is done on each image to get z-score for each channel. These augmented images are then divided in train and validation images with ratio of 19 to 1. Generator having least validation loss is selected from the models. The generator and discriminator are trained for n epochs until convergence. For optimization of the objective function we use Adam optimizer. The learning rate is set to $2e^{-4}$ and trade-off co-efficient $\lambda = 10$.

All the experimentation is carried out using standard PC with Intel Core i5 CPU and GeForce GTX 1080 GPU with 8 GB memory. We evaluate our technique with Area Under Curve for Precision and Recall (AUC-PR), Dice co-efficient (F1-score) and Area Under Curve for Receive Operating Characteristics (AUC-ROC). The probability map is threshold using Ostu thresholding [38], which is mostly used to separate fore-ground from background for calculation of dice co-efficient. Pixels inside the Field Of View (FOV) is counted when we are computing the measures, for fair measurement.

Results and discussions

In Table 1, we have compared performance of different models with different discriminators. There is no discriminator in U-Net so it shows inferior performance as compare to patch GAN and Image GAN. Patch GAN and

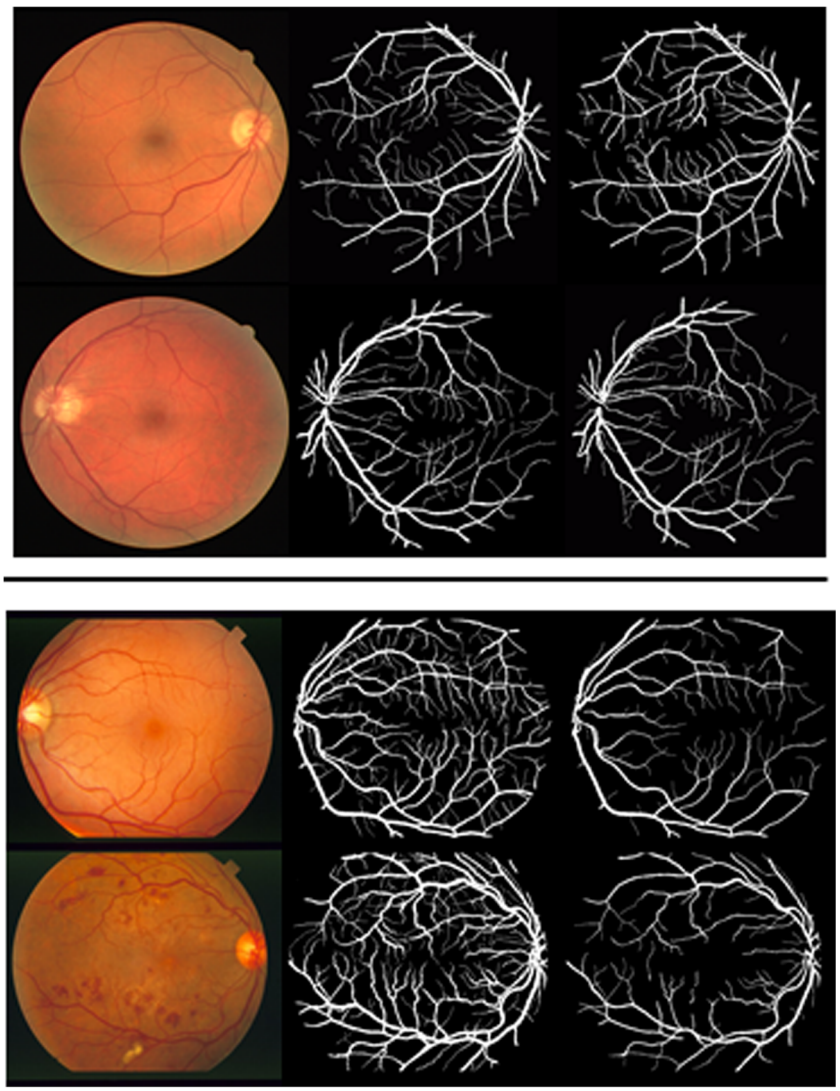

Fig. 5 Fundoscopic images (first column), Probability Map of DRIU (second column) and Probability Map of Our Method (third column). Top image is DRIVE dataset and Bottom image is STARE dataset

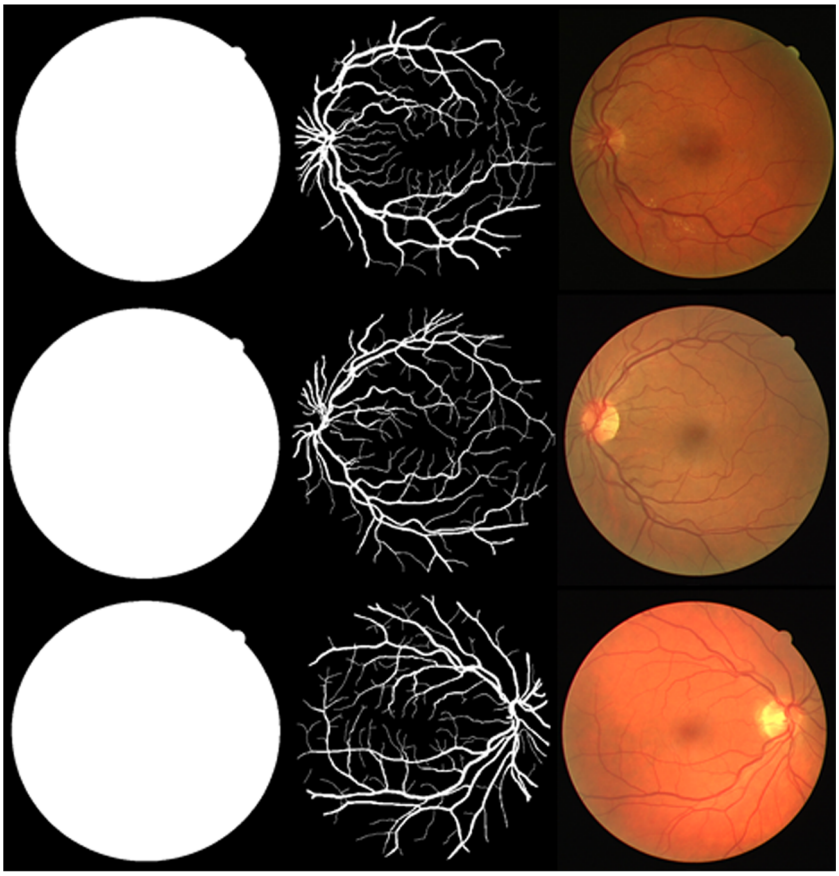

Fig. 6 (**From left to right**) Masks, filamentary structures and Output retinal images

Image GAN have shown improvement in overall segmentation quality but Image GAN, which has most powerful discriminator framework, out-performs all the other networks. This result is enough to claim that a powerful discriminatory framework is key for successful training of the networks with GANs [39],[23].

Table 2 summarizes dice coefficients (F1-score), AUC for ROC and AUC for PR for our proposed method in comparison with other methods. Our method outperformed all the existing methods and shows better dice coefficient and AUC values. Our method also surpasses human's annotating ability on DRIVE dataset.

Qualitative comparison of segmentation using our method and best existing method DRIU (Deep Retinal Image Understanding [40]) is illustrated in Fig. 5. Our proposed method generates concordant probability values to the gold standard while DRIU gives overconfident probability on boundaries between vessels and background, as well as on fine vessels. This may cause over-segmentation of retinal image, resulting in high false positive values. In contrast, the proposed technique allows more false negatives near the edges and terminal end of the vessels because it has tendency to give low probability to the pixels which falls in uncertain region. This is same as human annotators would do.

In Fig. 6, we have shown the generated masks (outer boundary), filamentary structured image and generated output images. We can see that these generated output images are visually close to real ones.

Conclusion

In this paper, we have introduced a new Generative Adversarial Network for Medical Imaging (MI-GAN) framework which focuses on retinal vessels image segmentation and generation. These synthesized images are realistic looking. When used as additional training dataset, the framework helps to enhance the image segmentation performance. The proposed model is capable of learning useful features from a small training set. In our case the training set consisted of only 10 examples from each dataset namely DRIVE and STARE. Our model outperformed other existing models in terms of AUC ROC, AUC PR and Dice co-efficient. Our method had less false positive rate at fine vessels and have drawn more clearer lines, as compared to other methods. Future work involves investigation into datasets of different bio-medical images for interplay of synthesized images, domain adaptation tasks and segmentation of the medical images.

Compliance with Ethical Standards

Funding No funding declared.

Conflict of interests Talha Iqbal declares that he has no conflict of interest. Hazrat Ali declares that he has no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Informed consent Not applicable.

References

1. de Vos, B. D., Wolterink, J. M., de Jong, P. A., Viergever, M. A., and Išgum, I., 2D image classification for 3D anatomy localization: employing deep convolutional neural networks. In: *Medical imaging 2016: Image processing, vol. 9784, international society for optics and photonics*, 2016.
2. Cai, Y., Landis, M., Laidley, D. T., Kornecki, A., Lum, A., and Li, S., Multi-modal vertebrae recognition using transformed deep convolution network. *Comput. Med. Imaging Graph.* 51:11–19, 2016.
3. Chen, H., Ni, D., Qin, J., Li, S., Yang, X., Wang, T., and Heng, P. A., Standard plane localization in fetal ultrasound via domain transferred deep neural networks. *IEEE J. of Biomedical and Health Informatics* 19(5):1627–1636, 2015.
4. Kumar, A., Sridar, P., Quinton, A., Kumar, R. K., Feng, D., Nanan, R., and Kim, J., Plane identification in fetal ultrasound images using saliency maps and convolutional neural networks. In: *Biomedical Imaging (ISBI), 2016 IEEE 13th International Symposium on, IEEE*, pp. 791–794, 2016.
5. Ghesu, F. C., Georgescu, B., Mansi, T., Neumann, D., Hornegger, J., and Comaniciu, D., An artificial agent for anatomical landmark detection in medical images. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer*, pp. 229–237, 2016.
6. Baumgartner, C. F., Kamnitsas, K., Matthew, J., Smith, S., Kainz, B., and Rueckert, D., Real-time standard scan plane detection and localisation in fetal ultrasound using fully convolutional neural networks. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer*, pp. 203–211, 2016.
7. Kong, B., Zhan, Y., Shin, M., Denny, T., and Zhang, S., Recognizing end-diastole and end-systole frames via deep temporal regression network. In: *International conference on medical image computing and computer-assisted intervention, Springer*, pp. 264–272, 2016.
8. Barbu, A., Lu, L., Roth, H., Seff, A., and Summers, R. M., An analysis of robust cost functions for cnn in computer-aided diagnosis. *Comput. Methods Biomechanics Biomedical Eng.: Imaging & Vis.* 6(3):253–258, 2018.
9. Roth, H. R., Lu, L., Liu, J., Yao, J., Seff, A., Cherry, K., Kim, L., and Summers, R. M., Improving computer-aided detection using convolutional neural networks and random view aggregation. *IEEE Trans. Med. Imaging* 35(5):1170–1181, 2016.
10. Teramoto, A., Fujita, H., Yamamuro, O., and Tamaki, T., Automated detection of pulmonary nodules in PET/CT images: Ensemble false-positive reduction using a convolutional neural network technique. *Med. Phys.* 43(6Part1):2821–2827, 2016.
11. Becker, C., Rigamonti, R., Lepetit, V., and Fua, P., Supervised feature learning for curvilinear structure segmentation. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer*, pp. 526–533, 2013.
12. Makhzani, A., and Frey, B. J., PixelGAN autoencoders. In: *Advances in Neural Information Processing Systems*, pp. 1972–1982, 2017.
13. Ganin, Y., and Lempitsky, V., N^4-Fields: Neural Network Nearest Neighbor Fields for Image Transforms. In: *Asian Conference on Computer Vision, Springer*, pp. 536–551, 2014.
14. Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., van der Laak, J. A., van Ginneken, B., and Sánchez, C. I., A survey on deep learning in medical image analysis. *Med. Image Anal.* 42:60–88, 2017.
15. Gatys, L. A., Ecker, A. S., and Bethge, M., A neural algorithm of artistic style, arXiv:1508.06576.
16. Ulyanov, D., Lebedev, V., Vedaldi, A., and Lempitsky, V. S. In: *ICML*, pp. 1349–1357, 2016.
17. Johnson, J., Alahi, A., and Fei-Fei, L., Perceptual losses for real-time style transfer and super-resolution. In: *European Conference on Computer Vision, Springer*, pp. 694–711, 2016.
18. Kirbas, C., and Quek, F., A review of vessel extraction techniques and algorithms. *ACM Comput. Surveys (CSUR)* 36(2):81–121, 2004.
19. Lesage, D., Angelini, E. D., Bloch, I., and Funka-Lea, G., A review of 3D vessel lumen segmentation techniques: models, features and extraction schemes. *Med. Image Anal.* 13(6):819–845, 2009.
20. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y., Generative adversarial nets. In: *Advances in neural information processing systems*, pp. 2672–2680, 2014.
21. Szegedy, C., Ioffe, S., Vanhoucke, V., and Alemi, A. A., Inception-v4, inception-resnet and the impact of residual connections on learning. In: *AAAI, Vol. 4*, p. 12, 2017.
22. Ding, C., Xia, Y., and Li, Y., Supervised segmentation of vasculature in retinal images using neural networks. In: *Orange technologies (ICOT), 2014 IEEE International Conference on, IEEE*, pp. 49-52, 2014.
23. Radford, A., Metz, L., and Chintala, S., Unsupervised representation learning with deep convolutional generative adversarial networks, arXiv:1511.06434.
24. Mirza, M., and Osindero, S., Conditional generative adversarial nets, arXiv:1411.1784.
25. Denton, E. L., Chintala, S., Fergus, R. et al., Deep generative image models using a Laplacian pyramid of adversarial networks. In: *Advances in Neural Information Processing Systems*, pp. 1486–1494, 2015.
26. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. C., Improved training of Wasserstein GANs. In: *Advances in Neural Information Processing Systems*, pp. 5769–5779, 2017.
27. Peng, H., Hawrylycz, M., Roskams, J., Hill, S., Spruston, N., Meijering, E., and Ascoli, G. A., Bigneuron: large-scale 3d neuron reconstruction from optical microscopy images. *Neuron* 87(2):252–256, 2015.
28. Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A., Image-to-image translation with conditional adversarial networks. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5967–5976, 2017.
29. Zhang, J., Chen, L., Zhuo, L., Liang, X., and Li, J., An Efficient Hyperspectral Image Retrieval Method: Deep Spectral-Spatial Feature Extraction with DCGAN and Dimensionality Reduction Using t-SNE-based NM Hashing. *Remote Sens.* 10(2):271, 2018.
30. Wan, L., Zeiler, M., Zhang, S., Le Cun, Y., and Fergus, R., Regularization of neural networks using dropconnect. In: *International Conference on Machine Learning*, pp. 1058–1066, 2013.

31. Wang, X., and Gupta, A., Generative image modeling using style and structure adversarial networks. In: *European Conference on Computer Vision, Springer*, pp. 318–335, 2016.
32. Mao, X., Shen, C., and Yang, Y.-B., Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. In: *Advances in neural information processing systems*, pp. 2802–2810, 2016.
33. Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A., Unpaired image-to-image translation using cycle-consistent adversarial networks. In: *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2242–2251, 2017.
34. Ronneberger, O., Fischer, P., and Brox, T., U-NET: Convolutional networks for biomedical image segmentation. In: *International Conference on Medical image computing and computer-assisted intervention, Springer*, pp. 234–241, 2015.
35. Zhao, H., Li, H., and Cheng, L., Synthesizing Filamentary Structured Images with GANs, arXiv:1706.02185.
36. Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L., Deeplab: Semantic image segmentation with deep convolutional nets, Atrous Convolution, and fully connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* 40(4):834–848, 2018.
37. Li, Y., Wang, N., Shi, J., Liu, J., and Hou, X., Revisiting batch normalization for practical domain adaptation. *Pattern Recogn.* 80:109–117, 2018.
38. Zhang, W., Li, W., Yan, J., Yu, L., and Pan, C., Adaptive threshold selection for background removal in fringe projection profilometry. *Opt. Lasers Eng.* 90:209–216, 2017.
39. Son, J., Park, S. J., and Jung, K.-H., Retinal vessel segmentation in fundoscopic images with generative adversarial networks, arXiv:1706.09318.
40. Maninis, K.-K., Pont-Tuset, J., Arbeláez, P., and Van Gool, L., Deep retinal image understanding. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer*, pp. 140–148, 2016.
41. Farokhian, F., Yang, C., Demirel, H., Wu, S., and Beheshti, I., Automatic parameters selection of Gabor filters with the imperialism competitive algorithm with application to retinal vessel segmentation. *Biocybernetics Biomedical Eng.* 37(1):246–254, 2017.
42. Xie, S., and Tu, Z., Holistically-nested edge detection. In: *Proceedings of the IEEE international conference on computer vision*, pp. 1395–1403, 2015.