Artificial Intelligence In Medicine 141 (2023) 102556

Contents lists available at ScienceDirect

Artificial Intelligence In Medicine

journal homepage: www.elsevier.com/locate/artmed

Research paper

Progressive growing of Generative Adversarial Networks for improving data augmentation and skin cancer diagnosis

Eduardo Pérez [a,b], Sebastián Ventura [a,b,*]

[a] Andalusian Research Institute in Data Science and Computational Intelligence (DaSCI). University of Córdoba, Córdoba, Spain
[b] Maimónides Biomedical Research Institute of Córdoba (IMIBIC). University of Córdoba, Córdoba, Spain

ARTICLE INFO

Keywords:
Melanoma diagnosis
Generative Adversarial Networks
Residual connections
Transfer learning

ABSTRACT

Early melanoma diagnosis is the most important factor in the treatment of skin cancer and can effectively reduce mortality rates. Recently, Generative Adversarial Networks have been used to augment data, prevent overfitting and improve the diagnostic capacity of models. However, its application remains a challenging task due to the high levels of inter and intra-class variance seen in skin images, limited amounts of data, and model instability. We present a more robust Progressive Growing of Adversarial Networks based on residual learning, which is highly recommended to ease the training of deep networks. The stability of the training process was increased by receiving additional inputs from preceding blocks. The architecture is able to produce plausible photorealistic synthetic 512 × 512 skin images, even with small dermoscopic and non-dermoscopic skin image datasets as problem domains. In this manner, we tackle the lack of data and the imbalance problems. Additionally, the proposed approach leverages a skin lesion boundary segmentation algorithm and transfer learning to enhance the diagnosis of melanoma. Inception score and Matthews Correlation Coefficient were used to measure the performance of the models. The architecture was evaluated qualitatively and quantitatively through the use of an extensive experimental study on sixteen datasets, illustrating its effectiveness in the diagnosis of melanoma. Finally, four state-of-the-art data augmentation techniques applied in five convolutional neural network models were significantly outperformed. The results indicated that a bigger number of trainable parameters will not necessarily obtain a better performance in melanoma diagnosis.

1. Introduction

Melanoma is one of the most deadly types of skin cancer and begins in cells known as melanocytes. According to the American Cancer Society in the latest edition of its publication "Cancer Facts and Figures 2021", just in the United States, more than 100,000 new cases of melanoma and 7000 deaths were expected in 2021 [1]. However, the early diagnosis can increase the chance of survival, achieving a 98% five-year survival rate. The first step in the diagnosis of a skin lesion by a dermatologist is an initial clinical examination. If any doubt remains, a dermoscopic analysis, biopsy and histopathological examination are performed [2]. With regard to clinical diagnosis, dermatologists have an accuracy rate between 65%–80% and up to 75%–84% through the use of dermoscopic images [3,4]. Despite the expertise of dermatologists, early diagnosis of melanoma remains a daunting task as it presents in many different shapes, sizes and colors — even between samples in the same category.

Nowadays, advanced computational techniques are used in the diagnosis of melanoma in order to make the diagnosis easier and aid dermatologists in their decision making. For example, those based on descriptors [5] and Convolutional Neural Networks (CNNs) [6]. Descriptor-based methods require the previous extraction of handcrafted features, which involves the expertise of dermatologists. However, this task is time-consuming and prone to errors [7]. In order to solve such limitations, CNN models have been applied to learn high-level features from raw images without the involvement of experts [2]. Several authors have corroborated that CNN models can overcome handcrafted feature-based methods [8] and can even rival the diagnostic accuracy of dermatologists [9].

CNN models have proven effective in solving complex problems [10]. However, these models present several disadvantages, especially when applied to the diagnosis of skin lesions. They are prone to overfitting on datasets with a small number of training examples per category and, as a result, attaining a poor generalization capacity. Also, CNNs require large datasets in order to learn accurately, which is a major issue in public melanoma datasets. On top of that, most

* Corresponding author at: Andalusian Research Institute in Data Science and Computational Intelligence (DaSCI). University of Córdoba, Córdoba, Spain.
E-mail address: sventura@uco.es (S. Ventura).

https://doi.org/10.1016/j.artmed.2023.102556
Received 12 October 2022; Received in revised form 6 April 2023; Accepted 14 April 2023
Available online 28 April 2023

datasets are unbalanced, and the minority category is often melanoma. On the other hand, CNN models are sensitive to some characteristics in data, such as large inter-class similarities and intra-class variances, variations in viewpoints, changes in lighting conditions, occlusions and background clutter [11].

CNN models seem to work better with high-quality and standardized conditions, such as dermoscopic images. However, keeping in mind the growing tendency to collect images taken with common digital cameras, we made the effort to include as many non-dermoscopic datasets as possible [12]. In this way, it is possible to reduce the number of invasive treatments and required economic resources in addition to boosting the development of modern and inexpensive tools. Finally, CNN models are approximately invariant with respect to small translations to the input but are not rotation, color or lighting-invariant [13]; invariance to a transformation is an important concept in the realm of image recognition. If you take the input and transform it, the resulting representation is the same as the representation of the original.

There are several techniques for overcoming the problems of invariance and limited training datasets. The simplest ones include basic data augmentation [14], transfer learning [2] and ensemble learning [15]. In the past few years, other techniques such as advanced data augmentation through the use of multi-task learning (MTL) [16] and Generative Adversarial Networks (GANs) [17], have emerged. Also, new architectures have been proposed based on representing and preserving properties of an object such as position, size, texture, and hierarchical spatial relationships [18]. So far, basic techniques have been widely applied in the diagnosis of melanoma, significantly improving the performance of CNN models. However, advanced techniques like GANs have been used with discretion in the diagnosis of melanoma, which is further explained in Section 2. Consequently, in this work, a new GAN-based approach for diagnosing melanoma from images and a deep analysis of its core components are carried out.

Firstly, an improvement on progressively growing GANs is proposed. The proposal applies the residual connections extensively, as well as an active equalization of the intermediate layer outputs. As a result, photorealistic synthetic skin lesion images and significantly better predictive performance compared to other state-of-the-art data augmentation methods could be obtained. Generally speaking, GANs consist of two models that are trained together in an adversarial zero-sum game: a generative model that captures the data distribution and a discriminative model that predicts whether a sample is fake or not. However, GANs are limited to small image sizes due to model stability. We propose a customized Progressive Growing GAN architecture (RGAN), which is a stable approach for training GANs models to generate large high-quality images. This involves incrementally increasing the size of the model during training and requires a highly complex training process. In addition, bearing in mind the restrictions in our problem, where only few images are commonly available and high-resolution images are needed, two residual connections were used to progressively train the models, instead of one. This approach is inspired by Residual Learning Framework (ResNet) [19] and Dense Convolutional Network (DenseNet) [20], where each building block receives additional inputs from preceding blocks. By doing this, the stability of the training process was further increased, particularly in small datasets. As such, the generative model is capable of generating photorealistic synthetic 512×512 skin images that are almost indistinguishable from the real ones. As a result, it is possible to accurately train deep CNN models to overcome the lack of data present in skin datasets. Additionally, data augmentation could improve the generalization capacity of CNN models and can be used as a regularization method in order to prevent overfitting [21]. It is interesting to compare basic and advanced data augmentation techniques across a large number of datasets. Secondly, a lesion segmentation method based on Chan-Vese segmentation algorithm is applied [15]. It is well understood that a preprocessing phase can improve the quality of any biomedical data analysis [22]. The algorithm is capable of effectively obtaining reliable segmentation masks without prior knowledge. Thirdly, the proposed double residual architecture was assessed qualitatively and quantitatively using state-of-the-art metrics. In this manner, we double-checked for realism and high-quality images. Then, an extensive experimental study was conducted on sixteen skin image datasets in order to evaluate the augmented images. Five CNN models were trained with several state-of-the-art data augmentation techniques, and that performances were compared when training with the augmented images. The results showed that the proposed approach attained suitable results and outperformed the rest of techniques significantly.

The following is a non-exhaustive list of the contributions from this manuscript:

- A study of Progressive Generative Adversarial Networks and its inclusion within different architectures and training settings.
- An improvement regarding GANs for the diagnosis of melanoma. The proposal is inspired by residual learning, where each building block receives additional regulated inputs from preceding blocks. Initially, the biggest weight is given to the largest scaled-up version of the image, and it ended gradually assigning weight to the main output layer.
- An extensive experimental study, which includes: (i) sixteen skin image datasets, (ii) four state-of-the-art data augmentation techniques, (iii) five CNN models, (iv) dermoscopic and non-dermoscopic images, (v) segmented and non-segmented images.

The rest of this work is organized as follows: Section 2 briefly presents the state-of-the-art in solving melanoma diagnosis problem mainly by using CNN models; Section 3 describes the proposal architecture to augment images and how the CNN models were trained; the analysis and discussion of the results are portrayed in Section 4; finally, the concluding remarks are presented in Section 5.

2. Related works

Since 2016, *The International Skin Imaging Collaboration*[1] (ISIC) project organizes annually a challenge in which more than 180 teams have already participated. Most submissions are based on CNN models, and it is very common that authors applied additional techniques to even improve the performance, such as transfer learning [2], lesion segmentation [23], data augmentation [14] and in recent years advanced data augmentation by using GANs [24].

Transfer learning is an effective method that tries to transfer and reuse the knowledge that was extracted from a source task, on a target task. This helps to alleviate the fact that it is required an enormous collection of data in order to build accurate CNN models. For example, Esteva et al. [2] used Google's InceptionV3 architecture pretrained on ImageNet, they remove the final classification layer and then they re-trained with 129,450 skin lesion images. The CNN achieved a great performance as was mentioned before. In addition, Liang and Zheng [25] applied a custom transfer learning method for pediatric pneumonia diagnosis, which involved 112,120 chest X-ray images labeled with 14 different chest diseases. The authors explored the problem of low image resolution, partial occlusion, and the importance of transfer learning. The results showed that the proposal achieved the top diagnostic performance in the classification task of children pneumonia.

On the other hand, skin lesion segmentation is able to isolate the lesion and it plays an important role improving the performance of CNN models. This is a complex task and it is very important because some areas not related with the lesion can lead CNN models to misclassify samples. From ISIC-2016 to ISIC-2018 there was a special task related

[1] https://www.isic-archive.com

to lesion segmentation. In ISIC-2016, considering the top three average results, there was a slight improvement when using segmentation.

Data augmentation is employed to obtain new data through transformations of existing images [6] or generating new ones based in such images [24]. The above helps reducing overfitting and obtaining transformation-invariant models [13]. In the case of melanoma diagnosis, most of the datasets available lack of balance between the categories, so data augmentation is commonly used to tackle imbalance [14]. For example, Esteva et al. [2] showed the suitability of CNN models as a powerful tool for melanoma diagnosis. The authors augmented the images by a factor of 720 using basic random transformations, such as rotation, flip and crop. Also, they compared the performance of a CNN model versus 21 board-certified dermatologists on biopsy-proven clinical images with two critical binary classification use cases: keratinocyte carcinomas versus benign seborrheic keratoses; and malignant melanomas versus benign nevi. The results showed that the CNN model achieved a performance on par with experts in both tasks. In addition, Lenc and Vedald [13] trained CNN models by applying random data augmentation, where the benefits were more noticeable in deeper models.

Furthermore, advanced data augmentation techniques have been proposed, such as GANs, Random Erasing Data Augmentation (RE) [26] and Unsupervised Data Augmentation for Consistency Training (UDA) [27]. GANs are fairly new models to augment data. GANs represent a way of training two models: the generator model that it is trained to generate new examples, and the discriminator model that tries to classify examples as real or fake. It is commonly used to generate new images that plausibly could have been drawn from the original dataset. Baur et al. [17] applied GANs to generate realistically looking high resolution images of skin lesions. The authors proposed a new GANs model based on two classical GANs architectures, namely Deep Convolutional GANs [28] and Laplacian Pyramid of Adversarial Networks [29], and they showed that this type of method are able to mimic the data distribution with diverse and realistic samples, even when the training dataset is very small.

Recent approaches regarding GANs and data augmentation for melanoma diagnosis have been conducted by using self-attention technique, transfer learning, and custom GAN-network architectures. Qin et al. [30] proposed a custom style-based GAN architecture in order to generate 256×256 images, where the same weights were injected directly in the growing layers. The dataset of the International Skin Imaging Collaboration Challenge (2018) was used as source data. The authors concluded that the synthetic images helped the diagnostic model to achieve a better classification performance. On the other hand, Abdelhalim et al. [31] researched about self-attention and a custom Conditional Progressive Generative Adversarial Network to generate 128×128 and 256×256 skin images. However, this would mean adding to the input extra information about the class of each image, which increases the complexity of the training process and could hamper the performance. Finally, Pollastri et al. [32] proposed a customized Deep Convolutional GAN and a Laplacian GAN in order to produce both synthetic 192×256 images and their corresponding segmentation masks. Although the previous works have been well designed, the proposals were evaluated by using only one dataset and one CNN model. Also, those proposals do not analyze non-dermoscopic images, which are taken mostly by using mobile devices. These images could broad the public access to a modern, cheaper, and enhanced melanoma diagnosis. As a result, invasive treatments and economical resources could be reduced. In addition, there are missing important details about the training process in the first two, such as the number of epochs that the architectures required to achieve plausible results.

Nevertheless, we believe that there is still room for further study of GAN architectures in the diagnosis of melanoma. For example, although there are some studies, they are limited to corroborating their proposal on only one dataset, which can be a constraint when applying a proposal on another dataset with different characteristics. In addition, to the best of our knowledge, there is not evidence of the analysis of GAN in non-dermoscopic images.

On the other hand, Random Erasing generates images with various levels of occlusion by randomly erasing pixels in an image, which reduces the risk of overfitting. This technique do not have parameter and is easy to implement. The method obtained good performance in several datasets, such as CIFAR10, CIFAR100, and Fashion-MNIST. The method even achieved a reasonable improvement on object detection and person re-identification. In Unsupervised Augmentation, the authors investigated the noise injection during training, and concluded that RandAugment [33] and back-translation [34] methods achieved competitive performance compared to other techniques.

Despite the advantages that some techniques offer for a better training of CNN models, such as ensemble and multi-task learning methods, it should be stressed that important limitations arise when they are applied. For example, the training of ensemble commonly requires the assessment of a high number of possibilities that hamper the process, and the way to combine the learned representations and partial predictions yielded by each member of an ensemble. On the other hand, the main limitation when applying MTL is the lack of public datasets that contain heterogeneous information, e.g image and clinical data of each patient.

To sum up, in this work our main aim is to explore how to generate plausible high resolution skin images by designing a customized RGAN architecture. The proposal applies two residual connections in order to progressively train the models, which is a valid approach to increase the stability of the training process. Also, we evaluate the performance of CNN models with augmented data by applying basic techniques versus advanced ones in an extensive experimental study conducted on sixteen image datasets. So far, there is no evidence of previous research addressing the increase of residual connections and conducting such an in-depth evaluation. In addition, transfer learning and segmentation to improve the performance of CNN models are applied. In the next section, the proposed architecture is described.

3. Material and methods

GANs were first described in 2014 by Ian Goodfellow [35]. The proposal was composed by two sub-models, a generator used to generate new believable examples from a domain dataset and a discriminator used to classify each one as real or fake. The generator tries to fool the discriminator by generating plausible images. Then, the parameters of the generator are updated by using the feedback from the discriminator. Commonly, when the discriminator is fooled about half the time, means the generator is ready. The architecture was unstable and hard to train at the beginning. However, Radford et al. [28] proposed a stable approach called Deep Convolutional Generative Adversarial Networks (DCGAN), and nowadays most GANs architectures are based on it. Consequently, GANs have started to be widely applied in tasks related with images, especially to augment data when training datasets are limited.

Traditional GANs models work well in low resolution datasets, typically comprising images that are less than 100 pixels square, which can be problematic when dealing with medium-resolution images in the problem domain, such as dermoscopic or non-dermoscopic skin images. Generating high-quality images is one of the challenges for GANs, as the generator must learn how to output both large structures and small details. High-resolution images make it easier to detect issues in the fake images for the discriminator, thus, the training process may fail. Also, large images require more GPU memory, which is a disadvantage if you are using enthusiast-grade GPU cards. Therefore, the batch size must be reduced in order to fit into memory and update the model weights each training iteration. The above introduces instability into the training process. Karras et al. [36] proposed a solution for high-resolution generative models, which consisted in progressively increase the number of layers during the training process. This approach is called Progressive

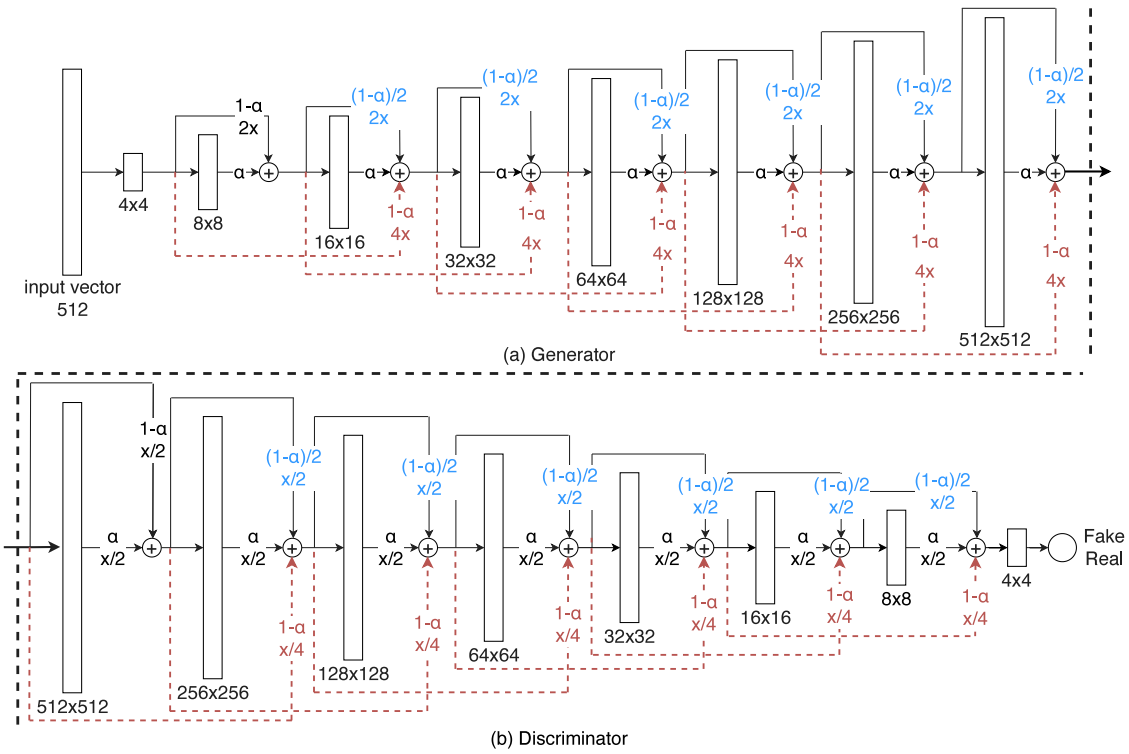


Fig. 1. Transition between each layer of the proposed generator and discriminator models. The input vector is formed by sampling from a standard Gaussian distribution; 2× and 4× mean doubling and quadrupling the image resolution in the generator, and $x/2$ and $x/4$ represent the opposite, respectively. The colors represent the introduced changes. Red color represents the new injected outputs and weights, and blue color represents the modified weights. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Growing GAN. The incremental addition of layers allows the models to discover large-scale structure of the image distribution and then it focuses on to increasingly finer scale detail, instead of learning all scales at the same time. In this work, we generate realistically looking high resolution skin images from small and medium dataset sizes.

Fig. 1 shows the steps during the transition between each block of layers in the proposed architecture for melanoma diagnosis. Firstly, the generator requires points in a specific latent space to generate new output images. The generator will give meaning to the latent points and at the end of training, the latent space represents a compressed representation of the output space. The more recent best practice is to sample from a standard Gaussian distribution, meaning that the shape of the latent space is a hypersphere, with a mean of zero and a standard deviation of one [37]. There is not a specific dimension for the latent space, but it is recommended to use values between 100 and 512.

Secondly, the process of growing GANs architectures requires adding layers to the generator and discriminator during the training process, specifically blocks of layers. The blocks are phased in the addition of the blocks of layers rather than adding them directly. In this work, the output of each layer is modified by the output of the previous ones, by using a double residual connection. This approach is inspired by ResNet and DenseNet, where each building block receives additional inputs from preceding blocks (Fig. 2). Residual learning is highly recommended for easing the training of deep networks, which is considered in this work [19]. By doing this, the stability of the training process was further increased.

In addition, bearing in mind that most of the existing melanoma datasets only encompass a few hundred of images, a second residual connection was used. The main output is weighted by α, the intermediate one by $\frac{1-\alpha}{2}$ and the most scaled-up output by $1-\alpha$; α is small initially, giving first the biggest weight to the largest scaled-up version of the image, although slowly transitions to giving more weight and then all weights to the new main output layers over training iterations.

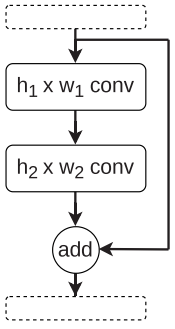


Fig. 2. Example of a residual block.

The generator starts with a very low resolution image, just about 4×4 and ends up with 512×512, which is the final image resolution (see Fig. 1a). We use leaky ReLU in all layers of generator and discriminator, except for the last layer that uses linear activation. Nearest neighbor filtering was used for increasing the image resolution. The resolution is chosen taking into account that some well-known models, such as InceptionV3, require image quality higher than 256×256. The purpose is not to lose image quality in any case. On the other hand, the discriminator takes as input an image from the generator and outputs *fake* or *real*. The discriminator executes the opposite process of the generator; now the goal is to downsample the image progressively until a resolution of 4×4 is attained (see Fig. 1b). Average pooling was applied when downsampling the image. The downsampled versions of the input are progressively combined in a weighted manner, in a similar way as the generator. In this work, eight blocks of layers for both generator and discriminator models were used.

The classical GANs architecture is trained by using a minimax GANs loss; minimax loss means the minimization of the generator and the

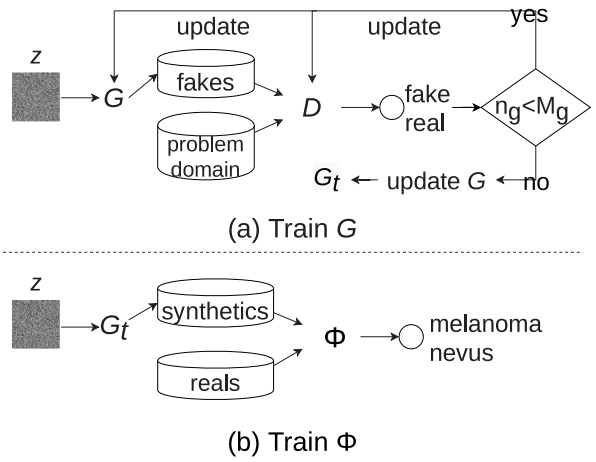


Fig. 3. Proposed pipeline for melanoma diagnosis; G, D and G_t mean the generator, the discriminator and the trained generator, respectively; "fakes" and "reals" are the images created by the generator and the original ones, respectively; Φ is a CNN model trained for identifying if an image is melanoma or nevus; n_g and M_g are the number of epochs trained so far and the maximum number of epochs, respectively.

maximization of the discriminator's loss. The loss produced by both models can be calculated as follows,

$$\mathcal{L}_D^{GAN} = -\mathbb{E}_{x\sim\mathbb{P}_d}[log(D(x))] - \mathbb{E}_{\hat{x}\sim\mathbb{P}_g}[log(1 - D(\hat{x}))], \quad (1)$$

$$\mathcal{L}_G^{GAN} = \mathbb{E}_{\hat{x}\sim\mathbb{P}_g}[log(1 - D(\hat{x}))], \quad (2)$$

where $\mathbb{P}_d$ is the data distribution and $\mathbb{P}_g$ is the model distribution; x is a real image and $\hat{x} = G(z)$ where z is a standard Gaussian distribution and $\hat{x}$ is a synthetic image; D and G mean the discriminator and generator, respectively. In some scenarios it was found that if the generator cannot learn as quickly as the discriminator, the generator's loss saturates and the discriminator wins [35].

Consequently, nowadays there are other loss formulations designed to overcome the saturation problem. The least squares (LSGAN) [38] and Wasserstein loss (WGAN) [39] functions are commonly used in modern GANs architectures. Gulrajani et al. [40] demonstrated that both loss functions obtained good performance, however, Wasserstein loss was better compared to least squares for progressive growing architectures. These losses can be calculated as

$$\mathcal{L}_D^{LSGAN} = -\mathbb{E}_{x\sim\mathbb{P}_d}[(D(x) - 1)^2] + \mathbb{E}_{\hat{x}\sim\mathbb{P}_g}[D(\hat{x})^2], \quad (3)$$

$$\mathcal{L}_G^{LSGAN} = -\mathbb{E}_{\hat{x}\sim\mathbb{P}_g}[(D(\hat{x} - 1))^2], \quad (4)$$

$$\mathcal{L}_D^{WGAN} = -\mathbb{E}_{x\sim\mathbb{P}_d}[D(x)] + \mathbb{E}_{\hat{x}\sim\mathbb{P}_g}[D(\hat{x})], \quad (5)$$

$$\mathcal{L}_G^{WGAN} = -\mathbb{E}_{\hat{x}\sim\mathbb{P}_g}[D(\hat{x})]. \quad (6)$$

Wasserstein loss provides a useful gradient almost everywhere, allowing for the continued training of the models. Also, a lower Wasserstein loss correlates with a better generator image quality, so we are seeking for a minimization in the generator loss. The above loss function was the first one showing this property. To sum up, Wasserstein loss shows the greatest advantages and it is applied in the proposed RGAN architecture for melanoma diagnosis. Next, it is shown how to combine the RGAN components mentioned so far.

Fig. 3a shows how the proposed architecture is trained. The algorithm involves training both generator and discriminator in parallel. In addition, the training process requires a problem domain image dataset, from which synthetic images will be generated. Commonly, skin image datasets are unbalanced at the expense of the melanoma category. As a result, melanoma images were taken as problem domain in almost all datasets when augmenting the images to feed the CNN models. However, both normal and melanoma images were assessed when evaluating the generative architecture in Section 4.5.1. The aim is to increase the minority category in order to balance the training set while keeping low the number of images and the required computational resources. First, the latent space z is generated following a standard Gaussian distribution. Then, all pixel values from images are transformed in the range $[-1, 1]$ before passing to the models [28]. Second, $G(z)$ outputs fake images. Third, D is trained with fake and real images, which is recommended to do it with separated batches; first by using real images and then by using only the fake ones [37]. Four, the weights are updated. The last three steps are repeated until a stop condition is reached. In our case, the number of epochs and the Inception Score were selected as stop criterion. Finally, the generator associated to the problem domain is obtained. Generally speaking, the generator is capable of generating suitable images between epochs 100 and 300, and also between 300 and 450 as well. In this work, we noticed that training between 190 and 260 epochs was enough to obtain an acceptable performance in all skin image datasets. Next, the above generator is used to create synthetic images, which are used to train all CNN models.

Fig. 3b shows how a deep CNN model is trained in order to predict whether an image is melanoma or not. Let us say T is a dataset of synthetic (s) and real (r) images, where x_i represents the ith image and y_i its label. In this work two classes were considered, melanoma ($y = 1$) and nevus ($y = 0$). Let be Φ a model that follows a CNN architecture, which learns the representations from the feature space and yields a prediction. Φ extracts representations from the dataset and finally, Φ predicts the label of the sample i ($\hat{y}_i$). Once the prediction for a given training image is computed, the loss obtained by applying Φ on the ith training image ($\mathcal{L}(i)$) is computed by means of a binary cross entropy.

Finally, in this work the CNN models were trained using mini-batch gradient descent. This method splits the training dataset into small batches that are used to calculate the model error. The above method have several advantages, such as the model update frequency is higher than batch gradient descent, which allows for a more robust convergence, avoiding local minima. In addition, batch-based updates provide a computationally more efficient process than stochastic gradient descent; and the split in small batches allows the efficiency of not having all training data in memory. All images in the batch are processed in parallel using GPU memory, increasing significantly the training speed; also, small batches can serve as regularizing effect.

4. Experimental study

This section describes the experimental study carried out in this work. First, the datasets and the experimental protocol are portrayed, and then, the experimental results and a discussion of them are presented.

4.1. Datasets

To validate the proposal, non-dermoscopic and dermoscopic images were obtained from several reputable sources, and can be consulted at the KDIS Research Group web page.[2] Table 1 shows a summary of the benchmark datasets: 13 dermoscopic and 3 non-dermoscopic. Although the majority of melanoma datasets comprise dermoscopic images validated and labeled by expert dermatologists, bear in mind that nowadays there is a growing tendency to collect images shot with common digital cameras [41]. As a result, we aimed to validate the proposal in both types of images. Only the images labeled as melanoma and nevus were considered, being in total 36,703 images. Most datasets

[2] http://www.uco.es/kdis/skin-diagnosis-pggan/

Table 1
Summary of the benchmark datasets. Last three datasets represent non-dermoscopic data.

Dataset	Img	ImbR	IntraC	InterC	DistR	Silho
BCN20000	17,393	2.848	9014	10,107	0.892	0.153
DERM-LIB	407	4.355	7171	9163	0.783	0.270
DERM7PT-D	827	2.282	15,971	16,866	0.947	0.087
HAM10000	7818	6.024	8705	9770	0.891	0.213
ISBI2016	1273	4.092	10,553	10,992	0.960	0.101
ISBI2017	2745	4.259	9280	9674	0.959	0.089
MSK-1	1088	2.615	11,753	14,068	0.835	0.173
MSK-2	1522	3.299	9288	9418	0.986	0.062
MSK-3	225	10.842	8075	8074	1.000	0.112
MSK-4	943	3.366	6930	7162	0.968	0.065
PH2	200	4.000	12,688	14,928	0.850	0.210
UDA-1	557	2.503	11,730	12,243	0.958	0.083
UDA-2	60	1.609	11,297	11,601	0.974	0.020
DERM7PT-C	827	2.282	15,442	16,318	0.946	0.086
MED-NODE	170	1.429	9029	9513	0.949	0.068
SDC-198	648	4.735	14,054	14,840	0.947	0.116

Table 2
Basic configuration used.

Parameter	Value
Segmentation threshold	40%
Number of epochs (U-Net, R2U-Net)	150
M_g	400
Rotations	[1°,270°]
Flip	Vertical and horizontal
Translations in X and Y	[−30%,30%]
Crop	[10%,30%]
Number of epochs	150
Mini-batch size	8
Learning rate (α)	SGD=0.01

present high imbalance ratio (ImbR), up to 10 in the case of MSK-3, which hampers the learning process. On the other hand, several metrics were calculated at pixel level, such as the intra-class (IntraC) and inter-class (InterC) scores, which show the average distances between images belonging to different classes, as well as between images belonging to the same class. Both metrics were computed using the Euclidean function distance; each image i was represented as a vector. Then, the ratio (DistR) between these metrics showed that both distances are similar, which commonly indicates a high degree of overlapping between classes. Finally, the silhouette score (Silho) [42] was calculated, representing how similar an image is to its own cluster compared to other clusters. The results indicated that images were not well matched to their own cluster, and even samples belonging to different clusters are close in the feature space.

4.2. Experimental settings

Firstly, we trained the proposed architecture over each problem domain for 400 epochs. The goal was to obtain generators capable of creating high-quality and realistic images. As mentioned before, generator and discriminator models are trained to maintain an equilibrium. In consequence, there are not many objective loss functions used to effectively train both generator and discriminator. In this regard, Wasserstein loss was used to train both discriminator and generator architectures. The above method obtains a training process more stable and less sensitive to the architecture and hyperparameters [39]. Also, this loss is related with the quality of the images, showing properties of convergence. Although this means that it is not necessary to evaluate the generated samples looking for failures, we manually compare them with the real ones in order to assess the quality of the generator [43], which is a common practice. In addition, the proposed architecture is objectively evaluated by applying the Inception Score (IS), which was originally proposed as an alternative to Amazon Mechanical Turk. The above metric seeks to assess the image quality and diversity, and correlates well with the subjective evaluation from human annotators [44]. IS has been used in different scenarios, such as Civil Engineering [45], synthesizing images from text descriptions [46], and generating videos from text [47]. This is perhaps the most widely adopted score for GAN evaluation.

On the other hand, a linear activation function was used in the output layer of the discriminator model, instead of sigmoid. Also, Karras et al. [36] recommended using Adam as optimization algorithm with a small learning rate ($\alpha = 0.001$, $\beta_1 = 0$, $\beta_2 = 0.99$), and as well as low momentum.

Secondly, InceptionV3, DenseNet, MobileNet, Xception and NASNetMobile convolutional architectures were assessed by using the synthetic images generated with the proposed architecture. These results were compared to the same CNN models, but training with other classical and modern data augmentation techniques, such as random data augmentation, RE, UDA and a baseline Progressive Generative Adversarial Networks. These techniques have been used before in the diagnosis of melanoma [48]. In this manner, we quantitatively evaluate which data augmentation technique obtains the higher predictive performance in the diagnosis of melanoma, which is the main aim of this work. Random Erasing selects a rectangle region in an image and erases its pixels. Training images with various levels of occlusion reduces the risk of overfitting and makes the model robust. The above can be integrated with most of the CNN-based recognition models, such as InceptionV3. On the other hand, Unsupervised Data Augmentation represents a new perspective on how to effectively noise unlabeled examples and supports that advanced data augmentation methods plays a crucial role in semi-supervised learning. This method consists in substituting simple noising operations with advanced data augmentation methods such as RandAugment [33] and back-translation [34].

Thirdly, each tuple CNN and data augmentation technique was evaluated by using non-segmented and segmented images in order to discover if the preprocessing step is suitable. To be fair, transfer learning was applied in all cases. Table 2 shows the configuration used to train all the models: the learning rate (α) was equal to 0.01 and it was reduced by a factor of 0.2 if an improvement in predictive performance was not observed during 10 epochs; a batch of size 8 was used due the medium size of the datasets; and the models were trained along 150 epochs. *Stochastic Gradient Descent* (SGD) [49] was used for training the models. SGD is one of the most used optimizers for training CNN models and despite its simplicity, it performs well across a variety of applications [50] and has been successfully applied for training networks in melanoma diagnosis [6,14]. Regarding the tuning of the hyper-parameters of SGD, it is noteworthy that finding the optimal set of the hyper-parameter values is a task that commonly requires expensive and arduous work due to the many possible combinations [51]. In this work, a tuning process was not carried out and so the results could not be conferred to an over-adjustment. The datasets utilized in this work correspond to binary classification problems, so the cost function used for training the models was defined as the average of the binary cross entropy along all training samples.

On the other hand, despite CNNs have shown to be useful in several scenarios, they still present issues in melanoma diagnosis. Firstly, CNNs can fit on a wide diversity of non-linear data points, which can be particularly problematic when the training data per class is small. As a consequence, data augmentation aims at improving the performance of CNNs. Data augmentation techniques were applied only to the training data, while the test data were left untouched. For each dataset, the number of images in the minority category was increased by generating new images until both categories were equal. For example, there are 160 and 40 benign and malignant images in PH2 dataset, respectively. In each fold, approximately 144 and 36 images for each category were taken for training. After that, 108 new malignant images were generated in order to balance both categories in the training data. The generated training images were considered as independent from the original ones. The same procedure was repeated for each of the sixteen datasets and five data augmentation techniques, considering their own characteristics.

4.3. Evaluation process

Regarding the evaluation process of the GANs architectures, a manual inspection of the generated images was performed and the IS values were assessed. The IS has a lowest value of 1 and a highest value of the number of classes supported by the classification model; in our case, the highest score is 2. The higher the IS, the better the image quality. The IS is computed as the Kullback–Leibler (KL) divergence summed over all images and averaged over the two classes:

$$\text{IS} = exp(E_x KL(p(y|x) \parallel p(y))). \tag{7}$$

In order to assess the CNN models when training with the augmented data, a 3-times 10-fold cross validation process was performed individually on each of the sixteen data sets, and the results were averaged across all fold executions. It is worth noting that in each fold execution all data augmentation techniques were performed, as specified in the previous section.

In addition, *Matthews Correlation Coefficient* (MCC) was used to measure the predictive performance of the models. It is noteworthy that MCC is widely used in Bioinformatics as performance metric [6,15,52] and has the property to summarize well the performance of the classifiers on complex data [53–55]. Also, it is specially designed to analyze the predictive performance on unbalanced data, which is common in skin lesion datasets (Table 1). On the other hand, it should be noted that MCC is statistically consistent with the Area Under Curve (AUC) in both balanced and unbalanced datasets [56]. MCC is computed as:

$$\text{MCC} = \frac{t_p \times t_n - f_p \times f_n}{\sqrt{(t_p + f_p)(t_p + f_n)(t_n + f_p)(t_n + f_n)}}, \tag{8}$$

where t_p, t_n, f_p, and f_n are the number of true positives, true negative, false positives, and false negatives, respectively. MCC value is always in range $[-1, 1]$, where 1 represents a perfect prediction, 0 indicates a performance similar to a random prediction, and -1 an inverse prediction.

Finally, non-parametric statistical tests were used to detect whether there was any significant difference in predictive performance. Wilcoxon Signed–Rank [57] was performed when only two methods were compared. Friedman's test [58] was conducted in cases where a multiple comparison was carried out. After that, Hommel's test [59] was applied to detect significant differences with a control algorithm. All hypothesis testing were conducted at 95% confidence.

4.4. Software and hardware

The experimental study was executed with Ubuntu 18.04, four GPUs NVIDIA Geforce RTX 2080-Ti and four GPUs NVIDIA Geforce RTX 1080-Ti. Python v3.6, Keras and TensorFlow as backend were used to implement the proposal, and it is available on Github.[3] On the other hand, the original implementations of InceptionV3,[4] DenseNet,[5] MobileNet,[6] Xception[7] and NASNetMobile[8] were employed. In addition, the authors' implementations regarding data augmentation were employed: a baseline Progressive Growing GANs,[9] Random Erasing Data Augmentation[10] and Unsupervised Data Augmentation for Consistency Training.[11]

[3] https://bit.ly/3lopzOx
[4] https://keras.io/api/applications/inceptionv3/
[5] https://bit.ly/3Yr7Lkq
[6] https://keras.io/api/applications/mobilenet/
[7] https://keras.io/api/applications/xception/
[8] https://keras.io/api/applications/nasnet/#nasnetmobile-function
[9] https://bit.ly/3E2BFmX
[10] https://github.com/zhunzhong07/Random-Erasing
[11] https://github.com/google-research/uda

Table 3
Average IS values of the baseline progressive growing of GAN and the proposal RGAN; "%" represents the percent of improvement after comparing the proposed model with the baseline. Wilcoxon's test rejected the null hypothesis with a p-value equal to 2.189E−4.

Dataset	GAN	RGAN	%
BCN20000	1.885	**1.992**	6
DERM-LIB	1.876	**1.986**	6
DERM7PT-C	1.866	**1.993**	7
DERM7PT-D	1.873	**1.993**	6
HAM10000	1.882	**1.996**	6
ISBI2016	1.758	**1.948**	11
ISBI2017	1.546	**1.700**	10
MED-NODE	1.841	**1.961**	7
MSK-1	1.873	**1.990**	6
MSK-2	1.871	**1.989**	6
MSK-3	1.750	**1.947**	11
MSK-4	1.856	**1.978**	7
PH2	1.856	**1.980**	7
SDC-198	1.867	**1.988**	7
UDA-1	1.857	**1.990**	7
UDA-2	1.782	**1.897**	6

4.5. Results and discussions

In this section the results are presented. First, we analyzed qualitatively and quantitatively the images generated by the proposal. Second, the performance when training five CNN models with the generated images was assessed. Finally, the proposal was compared to several state-of-the-art data augmentation techniques.

4.5.1. Evaluating generative adversarial networks

Fig. 4 shows images generated by the proposal during several epochs. In this scenario, it took more than 200 epochs to notice some real progress in the synthetic images. We qualitatively assessed the images augmented by our proposal, where an increase in the stability of the models was observed, particularly in small datasets. In addition, plausible images were found in a lower number of epochs in some datasets (≈200 epochs). It is hard to point out differences between Fig. 4g and 4h regarding quality. In overall, after a high number of epochs, the generators mimic the real data very well, obtaining high-quality images.

On the other hand, Figs. 5 and 6 show a quantitative progressive evaluation using the IS. The IS required square images of about 300 × 300 pixels and an equal number of images in each category. The above requirements were fulfilled since 512 × 512 images were used and the same number of images peer category for evaluating the proposal. The attention was focused during 200–250 epochs, where a suitable performance was achieved in overall. It is noteworthy that the learning curves of the proposal were more stable compared to the standard GANs for diagnosing melanoma. HAM10000, DERM7PT-C and DERM7PT-D were the more feasible data sources, where the proposal achieved the top average performance. The biggest improvements were detected in ISBI2016 and MSK-3, which could be related with better predictive performance when training the CNN models in the next phase. Table 3 summarizes the top performance in each dataset, where our proposal achieved the highest result all the time, confirming the benefit and effectiveness of using the proposed double residual architecture for generating plausible images.

On the other hand, it is worth noting that an exact and objective metric for evaluating the proposal remains as an open issue [43]. The above is the main downside of GAN-based proposals nowadays. Efforts have been made, but they are inconclusive [44,60,61]. Although the IS indicated high-quality generated images, it is necessary to double-check this through the diagnostic capability of CNN models that are trained with the generated images. As a result, in the next section our proposal and state-of-the-art data augmentation techniques are evaluated by using several CNN models.

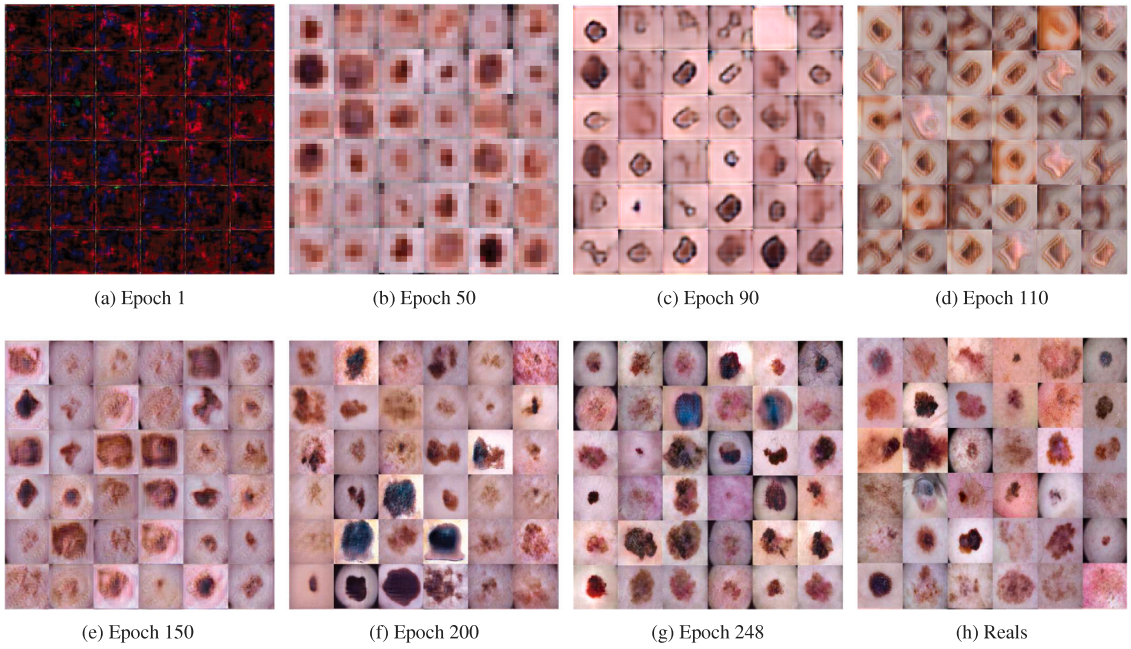


Fig. 4. Manual inspection of the images generated by using the proposed double residual architecture.

Table 4
Average MCC values obtained by using NASNet. "RDA" means random data augmentation; "RE" and "UA" represent Random Erasing and Unsupervised Data Augmentation, respectively; "GAN" and "RGAN" mean the default progressive growing architecture and our proposal, respectively; "Avg" means the average performance of all the baseline methods and the "%" columns represent the percent of improvement when comparing the proposed model with the "Avg" column; "Ranking" means the average ranking computed by Friedman's test and Hommel's *p*-values are showed in the last row. The Friedman's test rejected the null hypothesis with a *p*-value equal to 2.870E−6 when not using segmented images; Friedman's statistic was equal to 31.138 with four degrees of freedom. The Friedman's test rejected the null hypothesis with a *p*-value equal to 6.954E−6 when using segmented images; Friedman's statistic was equal to 29.25 with four degrees of freedom.

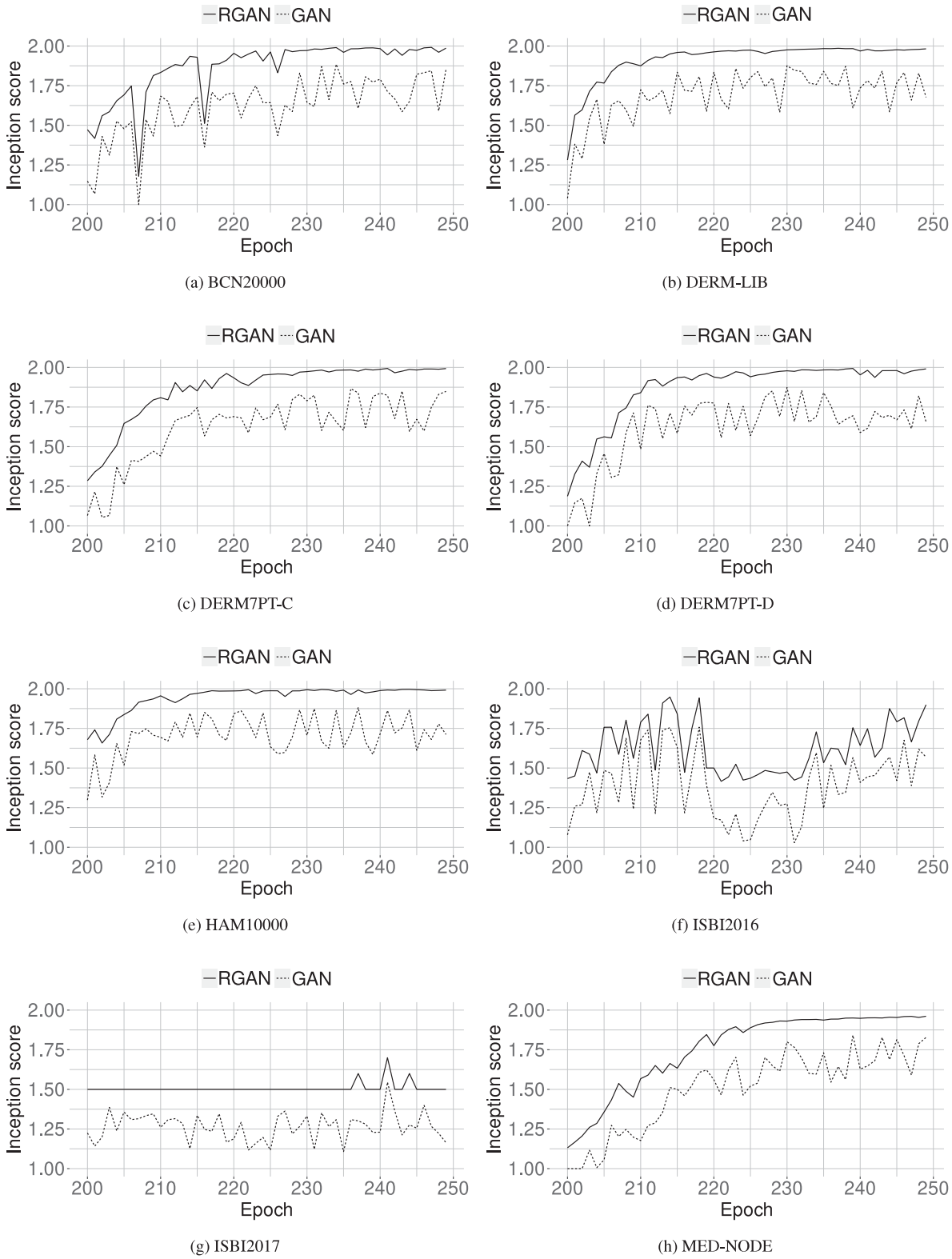
Dataset	Non-segmented							Segmented						
	RDA	RE	UA	GAN	Avg	RGAN	%	RDA	RE	UA	GAN	Avg	RGAN	%
BCN20000	**0.749**	0.659	0.664	0.658	0.682	0.747	9	0.754	0.659	0.700	0.688	0.700	**0.780**	11
DERM-LIB	0.886	0.900	0.900	0.922	0.902	**0.986**	9	0.975	0.930	0.906	0.942	0.938	**1.000**	7
DERM7PT-C	0.471	0.330	0.474	0.419	0.424	**0.529**	25	0.500	0.350	0.485	0.459	0.448	**0.545**	22
DERM7PT-D	0.617	0.552	0.551	0.511	0.558	**0.698**	25	0.657	0.590	0.570	0.531	0.587	**0.736**	25
HAM10000	0.702	0.696	0.702	0.712	0.703	**0.781**	11	0.759	0.707	0.734	0.762	0.740	**0.781**	5
ISBI2016	0.429	0.285	0.254	0.349	0.329	**0.443**	35	0.471	0.325	0.274	0.379	0.362	**0.488**	35
ISBI2017	0.447	0.421	0.409	0.415	0.423	**0.505**	19	0.487	0.432	0.417	0.455	0.448	**0.512**	14
MED-NODE	0.633	0.787	0.887	0.700	0.752	**0.997**	33	0.673	0.826	0.925	0.710	0.784	**1.000**	28
MSK-1	0.669	0.701	0.716	0.750	0.709	**0.827**	17	0.704	0.707	0.737	0.760	0.727	**0.838**	15
MSK-2	0.423	0.439	0.463	0.446	0.443	**0.550**	24	0.459	0.475	0.482	0.476	0.473	**0.558**	18
MSK-3	0.232	0.225	0.452	0.000	0.227	**0.685**	201	0.498	0.238	0.481	0.040	0.314	**0.691**	120
MSK-4	0.481	0.593	0.618	0.552	0.561	**0.675**	20	0.490	0.626	0.627	0.592	0.584	**0.684**	17
PH2	0.741	0.900	0.900	0.900	0.860	**0.988**	15	0.856	0.915	0.925	0.940	0.909	**1.000**	10
SDC-198	0.603	0.617	**0.788**	0.598	0.652	0.649	0	0.614	0.654	**0.822**	0.618	0.677	0.651	−4
UDA-1	0.539	0.537	0.584	0.495	0.539	**0.667**	24	0.540	0.556	0.611	0.545	0.563	**0.682**	21
UDA-2	0.472	0.900	0.900	0.900	0.793	**0.956**	21	0.459	0.920	0.908	0.950	0.809	**1.000**	24
Ranking	3.656	3.656	2.875	3.688		**1.125**		3.563	3.750	3.188	3.375		**1.125**	
p-values	1.191E−5	1.191E−5	1.745E−3	9.126E−6		–		3.896E−5	1.063E−5	2.247E−4	1.140E−4		–	

4.5.2. Comparing with state-of-the-art techniques

Tables 4–8 show each CNN model trained using basic and advanced data augmentation techniques and the proposed method. The best MCC value by dataset was highlighted in bold typeface. As can be seen, the segmentation method helped the CNN models to achieve better performance, e.g. InceptionV3, RDA and segmented images outperformed its baseline by 12% in PH2 dataset.

Table 4 shows the results of NASNet model, where the proposal achieved the best performance all the time, except in BCN20000 and SDC-198 with non-segmented images, and only in SDC-198 when using segmented images. It is noteworthy that the proposal achieved 201% and 120% better performance compared to the average state-of-the-art techniques in MSK-3. The Friedman's test rejected the null hypothesis with a *p*-value equal to 2.870E−6; Friedman's statistic was equal to 31.138 with four degrees of freedom. The Friedman's ranking shows that the proposal obtained the first position, indicating that this model in average achieved a better performance than the rest of methods. Afterwards, the Hommel's post-hoc test was conducted by considering the proposal as the control method, and the results showed the proposal significantly outperformed the rest of the state-of-the-art techniques. Fig. 7 summarizes such performance, where the CNNs trained with the proposed data augmentation technique obtained the highest MCC mean value, as well as a low standard deviation. RE and the baseline GAN were the most unstable techniques when analyzing non-segmented and segmented images, respectively.

Table 5 shows the results of DenseNet201, where the proposal achieved the best performance the 81% and 75% of the time using non-segmented and segmented images, respectively. It is noteworthy that the proposal achieved 156% and 123% better performance compared to the average state-of-the-art techniques in MSK-3. The Friedman's test



(a) BCN20000

(b) DERM-LIB

(c) DERM7PT-C

(d) DERM7PT-D

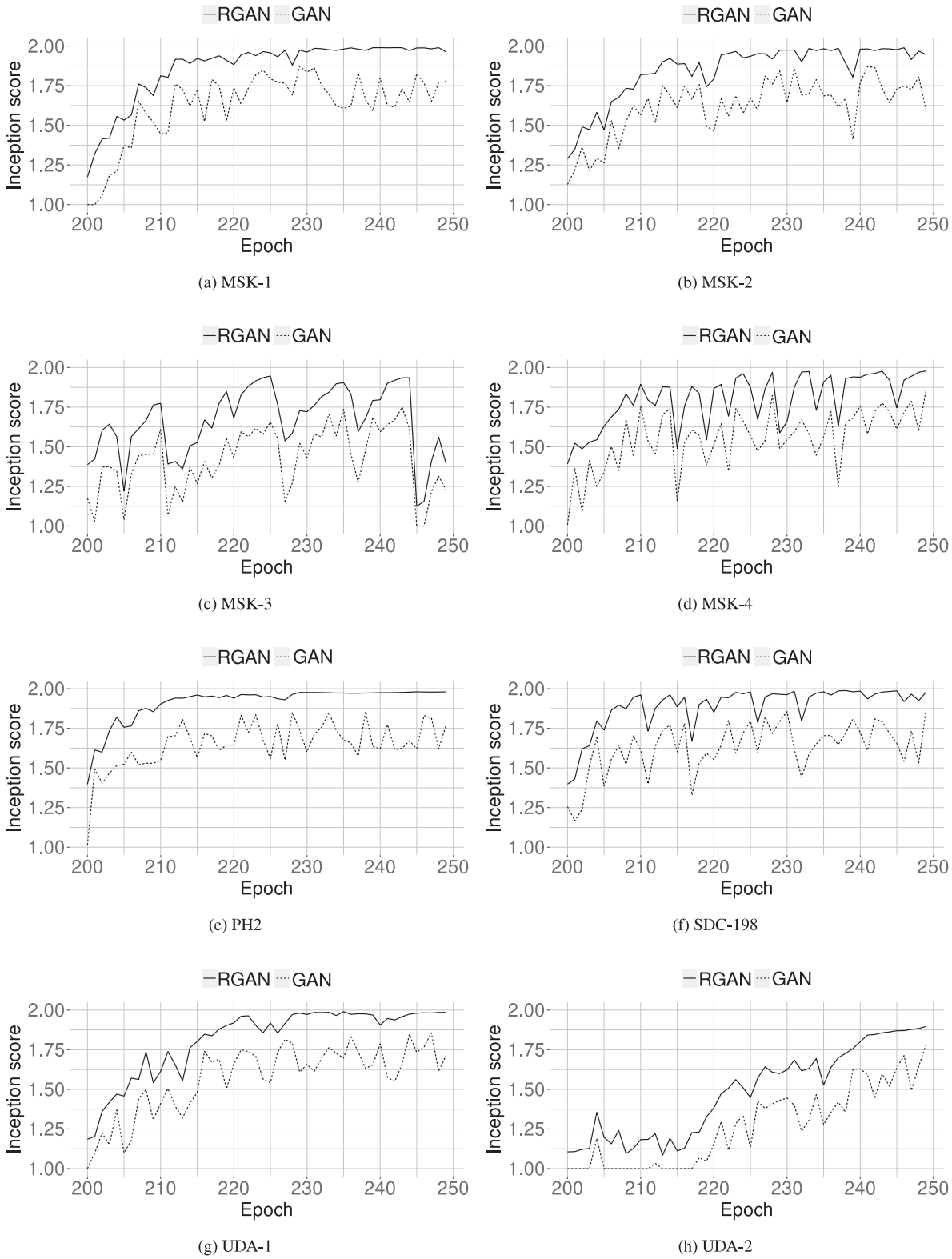
(e) HAM10000

(f) ISBI2016

(g) ISBI2017

(h) MED-NODE

Fig. 5. Inception score. "GAN" and "RGAN" mean the baseline Progressive Growing of Generative Adversarial Networks and the proposed architecture, respectively.



(a) MSK-1

(b) MSK-2

(c) MSK-3

(d) MSK-4

(e) PH2

(f) SDC-198

(g) UDA-1

(h) UDA-2

Fig. 6. Inception score. "GAN" and "RGAN" mean the baseline Progressive Growing of Generative Adversarial Networks and the proposed architecture, respectively.

Table 5
Average MCC values obtained by using DenseNet201. The Friedman's test rejected the null hypothesis with a *p*-value equal to 1.023E−4 when not using segmented images; Friedman's statistic was equal to 23.463 with four degrees of freedom. The Friedman's test rejected the null hypothesis with a *p*-value equal to 7.904E−4 when using segmented images; Friedman's statistic was equal to 18.988 with four degrees of freedom.

Dataset	Non-segmented							Segmented						
	RDA	RE	UA	GAN	Avg	RGAN	%	RDA	RE	UA	GAN	Avg	RGAN	%
BCN20000	0.771	0.687	0.690	0.698	0.712	**0.786**	10	0.792	0.716	0.701	0.708	0.729	**0.795**	9
DERM-LIB	0.966	0.900	0.900	0.900	0.916	**0.967**	6	0.992	0.937	0.923	0.910	0.941	**1.000**	6
DERM7PT-C	0.535	0.517	0.400	0.513	0.491	**0.590**	20	0.550	0.532	0.439	0.523	0.511	**0.624**	22
DERM7PT-D	0.655	0.533	0.689	0.547	0.606	**0.767**	27	0.636	0.564	0.723	0.547	0.618	**0.767**	24
HAM10000	0.747	0.726	0.708	0.707	0.722	**0.748**	4	0.698	0.738	0.729	0.737	0.726	**0.756**	4
ISBI2016	0.437	0.344	0.356	0.350	0.372	**0.441**	19	0.447	0.357	0.384	0.350	0.385	**0.450**	17
ISBI2017	0.447	0.428	0.429	0.393	0.424	**0.517**	22	0.419	0.437	0.449	0.413	0.430	**0.517**	20
MED-NODE	0.694	0.689	0.700	0.787	0.718	**0.878**	22	0.743	0.708	0.732	0.787	0.742	**0.883**	19
MSK-1	0.696	0.762	**0.807**	0.741	0.752	0.758	1	0.760	0.770	**0.841**	0.771	0.786	0.791	1
MSK-2	0.442	0.486	0.434	**0.502**	0.466	0.453	−3	0.436	0.521	0.472	**0.552**	0.495	0.443	−11
MSK-3	0.232	0.000	0.591	0.222	0.261	**0.670**	156	0.390	0.006	0.622	0.222	0.310	**0.691**	123
MSK-4	0.564	**0.625**	0.590	0.555	0.584	0.570	−2	0.533	**0.654**	0.620	0.605	0.603	0.565	−6
PH2	0.815	0.900	0.900	0.900	0.879	**0.981**	12	0.913	0.921	0.918	0.940	0.923	**1.000**	8
SDC-198	0.662	0.480	0.607	0.605	0.588	**0.699**	19	0.690	0.495	0.611	0.605	0.600	**0.707**	18
UDA-1	0.555	0.637	0.637	0.592	0.605	**0.644**	6	0.594	0.653	**0.668**	0.632	0.637	0.653	3
UDA-2	0.424	0.347	0.707	0.607	0.521	**0.974**	87	0.646	0.369	0.723	0.627	0.591	**1.000**	69
Ranking	3.188	3.719	3.031	3.688		**1.375**		3.500	3.406	2.938	3.625		**1.531**	
p-values	2.371E−3	8.272E−5	3.049E−3	1.057E−4		–		1.194E−3	1.592E−3	1.188E−2	7.204E−4		–	

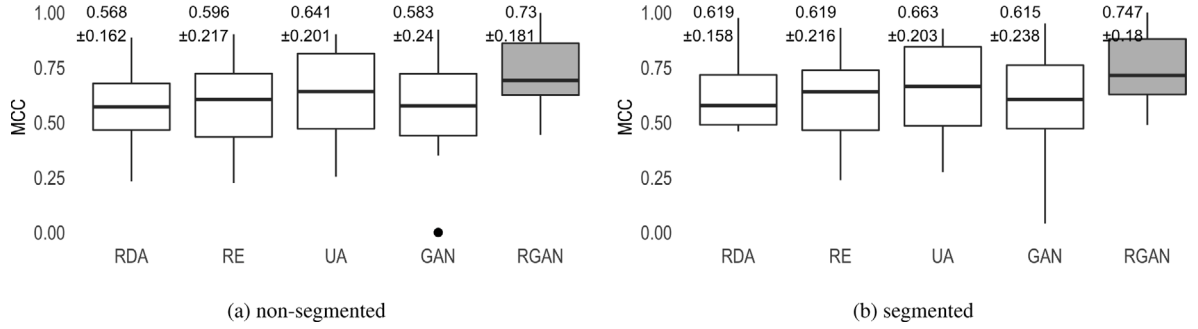


Fig. 7. Average MCC values obtained by using NASNet. Top labels represent the mean and standard deviation, respectively.

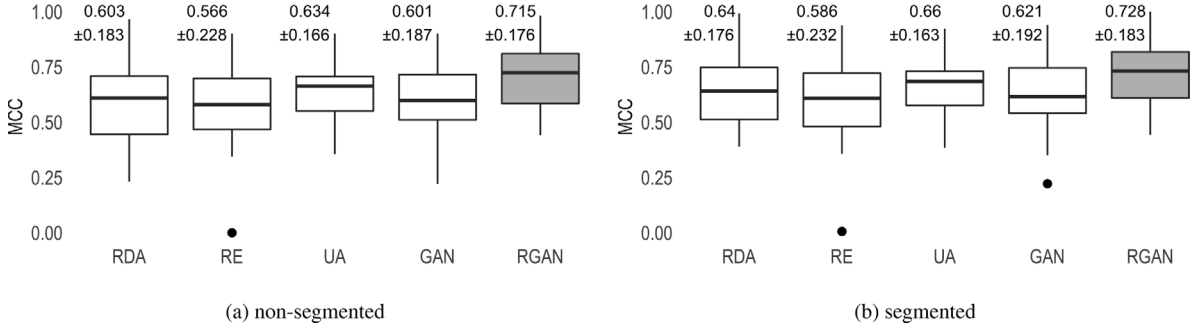


Fig. 8. Average MCC values obtained by using DenseNet201. Top labels represent the mean and standard deviation, respectively.

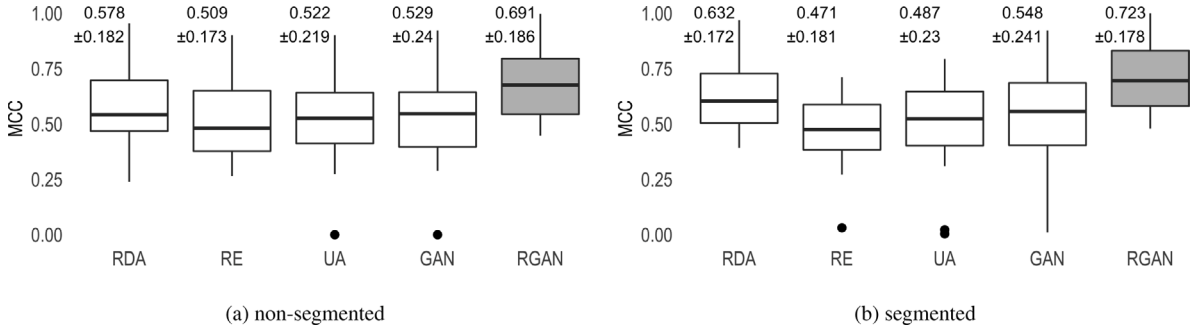


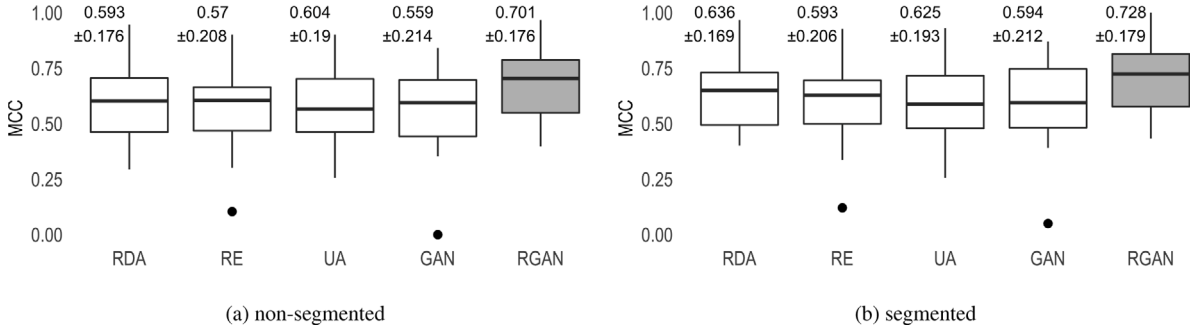
Fig. 9. Average MCC values obtained by using InceptionV3. Top labels represent the mean and standard deviation, respectively.

Table 6
Average MCC values obtained by using InceptionV3. The Friedman's test rejected the null hypothesis with a *p*-value equal to 7.288E−6 when not using segmented images; Friedman's statistic was equal to 29.15 with four degrees of freedom. The Friedman's test rejected the null hypothesis with a *p*-value equal to 4.785E−7 when using segmented images; Friedman's statistic was equal to 34.938 with four degrees of freedom.

Dataset	Non-segmented							Segmented						
	RDA	RE	UA	GAN	Avg	RGAN	%	RDA	RE	UA	GAN	Avg	RGAN	%
BCN20000	0.745	0.678	0.667	0.668	0.690	**0.758**	10	0.785	0.699	0.684	0.718	0.722	**0.788**	9
DERM-LIB	0.954	0.900	0.900	0.922	0.919	**0.975**	6	0.968	0.031	0.004	0.922	0.481	**1.000**	108
DERM7PT-C	0.499	0.349	0.415	0.418	0.420	**0.548**	30	0.507	0.362	0.446	0.418	0.433	**0.584**	35
DERM7PT-D	**0.622**	0.478	0.489	0.534	0.531	0.594	12	0.636	0.494	0.528	0.534	0.548	**0.641**	17
HAM10000	0.684	0.675	0.632	0.634	0.656	**0.691**	5	0.651	0.711	0.655	0.674	0.673	**0.739**	10
ISBI2016	0.452	0.372	0.404	0.288	0.379	**0.485**	28	0.464	0.391	0.433	0.328	0.404	**0.517**	28
ISBI2017	0.416	0.290	0.294	0.302	0.326	**0.447**	37	0.392	0.329	0.311	0.312	0.336	**0.479**	43
MED-NODE	0.732	0.549	0.789	0.797	0.717	**0.842**	17	0.836	0.569	0.793	0.817	0.754	**0.887**	18
MSK-1	0.682	0.641	0.604	0.618	0.636	**0.779**	22	0.708	0.643	0.624	0.638	0.653	**0.812**	24
MSK-2	0.473	0.379	**0.487**	0.331	0.418	0.462	11	0.440	0.414	**0.517**	0.361	0.433	0.510	18
MSK-3	0.239	0.265	0.000	0.000	0.126	**0.659**	423	0.577	0.271	0.022	0.010	0.220	**0.691**	214
MSK-4	0.493	0.403	0.535	**0.559**	0.498	0.531	7	0.531	0.413	0.562	**0.579**	0.521	0.571	10
PH2	0.803	0.688	0.740	0.900	0.783	**0.996**	27	0.925	0.705	0.741	0.910	0.820	**1.000**	22
SDC-198	0.584	0.540	0.517	**0.604**	0.561	0.602	7	0.630	0.543	0.518	**0.654**	0.586	0.651	11
UDA-1	0.496	0.483	0.274	0.437	0.423	**0.695**	64	0.561	0.508	0.310	0.437	0.454	**0.700**	54
UDA-2	0.381	0.447	0.607	0.447	0.471	**0.993**	111	0.499	0.457	0.643	0.457	0.514	**1.000**	95
Ranking	2.563	4.000	3.813	3.250		**1.375**		2.500	3.969	3.875	3.469		**1.188**	
p-values	3.365E−2	1.063E−5	3.896E−5	1.592E−3		–		1.888E−2	2.607E−6	4.584E−6	8.975E−5		–	

Table 7
Average MCC values obtained by using MobileNet. The Friedman's test rejected the null hypothesis with a *p*-value equal to 3.940E−6 when not using segmented images; Friedman's statistic was equal to 30.463 with four degrees of freedom. The Friedman's test rejected the null hypothesis with a *p*-value equal to 1.741E−6 when using segmented images; Friedman's statistic was equal to 32.2 with four degrees of freedom.

Dataset	Non-segmented							Segmented						
	RDA	RE	UA	GAN	Avg	RGAN	%	RDA	RE	UA	GAN	Avg	RGAN	%
BCN20000	**0.732**	0.654	0.664	0.626	0.669	**0.732**	9	0.744	0.661	0.702	0.666	0.693	**0.759**	9
DERM-LIB	0.945	0.900	0.900	0.822	0.892	**0.959**	8	0.967	0.915	0.928	0.852	0.916	**1.000**	9
DERM7PT-C	0.490	0.413	0.517	0.445	0.466	**0.534**	15	0.546	0.437	0.533	0.485	0.500	**0.579**	16
DERM7PT-D	0.657	0.608	0.543	0.511	0.580	**0.682**	18	0.642	0.620	0.549	0.551	0.590	**0.690**	17
HAM10000	0.701	0.662	0.685	0.660	0.677	**0.726**	7	0.672	0.694	0.696	0.680	0.686	**0.756**	10
ISBI2016	0.422	0.300	0.256	0.361	0.335	**0.481**	44	0.454	0.336	0.256	0.391	0.359	**0.494**	38
ISBI2017	**0.431**	0.350	0.392	0.352	0.381	0.397	4	0.430	0.389	0.429	0.392	0.410	**0.433**	6
MED-NODE	0.717	0.672	0.449	0.789	0.657	**0.873**	33	0.768	0.697	0.477	0.839	0.695	**0.887**	28
MSK-1	0.700	**0.746**	0.716	0.693	0.714	0.743	4	0.725	0.764	0.735	0.743	0.742	**0.789**	6
MSK-2	0.472	0.486	0.466	0.433	0.464	**0.527**	14	0.401	0.520	0.477	0.473	0.468	**0.551**	18
MSK-3	0.294	0.104	**0.697**	0.000	0.274	0.661	141	0.468	0.121	**0.710**	0.050	0.337	0.691	105
MSK-4	0.525	0.525	0.446	0.520	0.504	**0.553**	10	0.502	0.528	0.479	0.570	0.520	**0.572**	10
PH2	0.818	0.900	0.900	0.840	0.864	**0.967**	12	0.937	0.926	0.930	0.870	0.916	**1.000**	9
SDC-198	0.653	0.601	0.540	0.604	0.600	**0.757**	26	0.667	0.636	0.570	0.604	0.619	**0.770**	24
UDA-1	0.551	0.592	0.589	0.584	0.579	**0.665**	15	0.599	0.600	0.605	0.584	0.597	**0.682**	14
UDA-2	0.375	0.607	0.900	0.707	0.647	**0.961**	48	0.658	0.642	0.923	0.757	0.745	**1.000**	34
Ranking	2.875	3.406	3.375	4.125		**1.219**		3.063	3.688	3.375	3.813		**1.063**	
p-values	3.049E−3	1.822E−4	2.294E−4	8.021E−7		–		3.466E−4	7.969E−6	7.046E−5	3.473E−6		–	



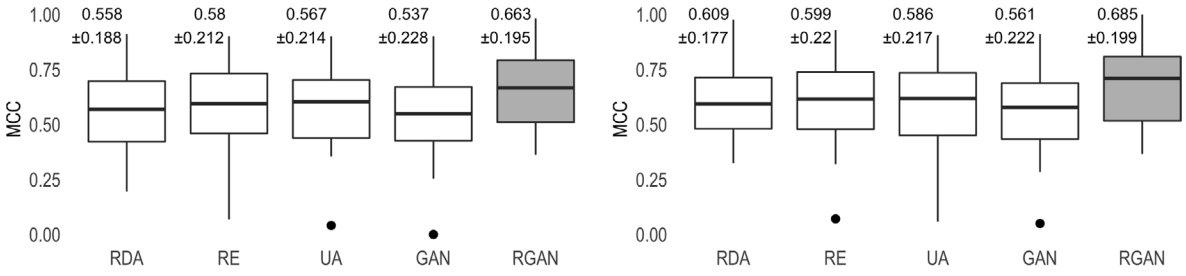
(a) non-segmented (b) segmented

Fig. 10. Average MCC values obtained by using MobileNet. Top labels represent the mean and standard deviation, respectively.

rejected the null hypothesis with a *p*-value equal to 1.023E−4; Friedman's statistic was equal to 23.463 with four degrees of freedom. The Friedman's ranking shows that the proposal obtained the first position, indicating that this model in average achieved a better performance than the rest of methods. Unsupervised data augmentation technique achieved the second best performance. Afterwards, the Hommel's post-hoc test was conducted by considering the proposal as the control method, and the results showed the proposal significantly outperformed

Table 8
Average MCC values obtained by using Xception. The Friedman's test rejected the null hypothesis with a p-value equal to 5.854E−4 when not using segmented images; Friedman's statistic was equal to 19.65 with four degrees of freedom. The Friedman's test rejected the null hypothesis with a p-value equal to 1.421E−4 when using segmented images; Friedman's statistic was equal to 22.75 with four degrees of freedom.

Dataset	Non-segmented							Segmented						
	RDA	RE	UA	GAN	Avg	RGAN	%	RDA	RE	UA	GAN	Avg	RGAN	%
BCN20000	0.752	0.663	0.653	0.707	0.694	**0.766**	10	0.764	0.681	0.670	0.717	0.708	**0.785**	11
DERM-LIB	0.910	0.822	0.822	0.900	0.863	**0.953**	10	0.976	0.860	0.836	0.910	0.896	**1.000**	12
DERM7PT-C	0.484	0.419	0.392	0.451	0.436	**0.513**	18	0.500	0.423	0.413	0.451	0.447	**0.517**	16
DERM7PT-D	0.634	0.580	0.605	0.584	0.601	**0.753**	25	0.689	0.615	0.642	0.584	0.632	**0.767**	21
HAM10000	**0.691**	0.650	0.609	0.653	0.651	0.667	2	0.707	0.688	0.615	0.663	0.668	**0.713**	7
ISBI2016	0.421	0.411	0.354	0.254	0.360	**0.500**	39	0.488	0.424	0.358	0.284	0.388	**0.514**	32
ISBI2017	**0.402**	0.306	0.371	0.352	0.358	0.397	11	0.437	0.319	0.376	0.382	0.379	**0.439**	16
MED-NODE	0.717	0.789	0.789	0.789	0.771	**0.884**	15	0.734	0.820	0.822	0.839	0.804	**0.887**	10
MSK-1	0.665	0.721	0.700	0.659	0.686	**0.781**	14	0.699	0.721	0.732	0.679	0.708	**0.788**	11
MSK-2	0.421	0.473	0.493	**0.564**	0.488	0.455	−7	0.443	0.497	0.518	**0.604**	0.516	0.455	−12
MSK-3	0.195	0.069	0.041	0.000	0.076	**0.362**	375	0.324	0.071	0.058	0.050	0.126	**0.365**	190
MSK-4	0.459	0.525	0.453	0.507	0.486	**0.552**	14	0.487	0.527	0.463	0.537	0.504	**0.581**	15
PH2	0.727	0.766	**0.900**	0.800	0.798	0.821	3	0.854	0.790	**0.906**	0.810	0.840	0.866	3
SDC-198	0.640	0.607	0.601	0.532	0.595	**0.667**	12	0.661	0.616	0.620	0.572	0.617	**0.707**	15
UDA-1	0.504	**0.582**	**0.582**	0.503	0.543	0.557	3	0.525	0.605	**0.607**	0.553	0.572	0.580	1
UDA-2	0.307	0.900	0.707	0.333	0.562	**0.982**	75	0.461	0.928	0.743	0.343	0.619	**1.000**	62
Ranking	3.000	3.313	3.500	3.688		**1.500**		3.000	3.500	3.438	3.688		**1.375**	
p-values	7.290E−3	2.371E−3	1.040E−3	3.644E−4		–		3.650E−3	3.370E−4	4.494E−4	1.409E−4		–	



(a) non-segmented (b) segmented

Fig. 11. Average MCC values obtained by using Xception. Top labels represent the mean and standard deviation, respectively.

the rest of the state-of-the-art techniques. Fig. 8 summarizes such performance, where again our proposal achieved the best predictive performance and a low standard deviation. RE was the most unstable data augmentation technique for non-segmented and segmented images.

Table 6 shows the results of InceptionV3, where the proposal achieved the best performance the 75% and 81% of the time using non-segmented and segmented images, respectively. Once again, the proposal was slightly surpassed in MSK-2, MSK-4 and SDC-198, which stated these datasets as the most difficult to the proposal. It is noteworthy that the proposal achieved 423% and 214% better performance compared to the average state-of-the-art techniques in MSK-3. The Friedman's test rejected the null hypothesis with a p-value equal to 7.288E−6; Friedman's statistic was equal to 29.15 with four degrees of freedom. The Friedman's ranking shows that the proposal obtained the first position, indicating that this model in average achieved a better performance than the rest of methods. Random data augmentation technique achieved the second best performance. Afterwards, the Hommel's post-hoc test was conducted by considering the proposal as the control method, and the results showed the proposal significantly outperformed the rest of the state-of-the-art techniques. Fig. 9 shows that RE and UA decreased their predictive performance with segmented images. On the other hand, baseline GAN and UA were the most unstable methods.

Table 7 shows the results of MobileNet, where the proposal achieved the best performance all the time when using segmented images, except in MSK-3. The proposal was only slightly surpassed in 2.7% by UDA method. However, it is noteworthy that the proposal achieved 141% and 105% better performance compared to the average state-of-the-art techniques in MSK-3. The Friedman's test rejected the null hypothesis with a p-value equal to 3.940E−6; Friedman's statistic was equal to 30.463 with four degrees of freedom. The Friedman's ranking shows that the proposal obtained the first position, indicating that this model in average achieved a better performance than the rest of methods. Random data augmentation technique achieved the second best performance. Afterwards, the Hommel's post-hoc test was conducted by considering the proposal as the control method, and the results showed the proposal significantly outperformed the rest of the state-of-the-art techniques. Fig. 10 shows that in overall all data augmentation techniques perform more stable compared to the previous CNN models. The baseline GAN was again the most changeable.

Table 8 shows the results of Xception, where the proposal did not achieve the best performance in HAM10000, ISBI2017, MSK-2, PH2 and UDA-1. However, it is noteworthy that the proposal achieved 375% and 190% better performance compared to the average state-of-the-art techniques in MSK-3. The Friedman's test rejected the null hypothesis with a p-value equal to 5.854E−4; Friedman's statistic was equal to 19.65 with four degrees of freedom. The Friedman's ranking shows that the proposal obtained the first position, indicating that this model in average achieved a better performance than the rest of methods. Random data augmentation technique achieved the second best performance. Afterwards, the Hommel's post-hoc test was conducted by considering the proposal as the control method, and the results showed the proposal significantly outperformed the rest of the state-of-the-art techniques. Fig. 11 shows that UA and the baseline GAN are the more unstable. In opposite, RDA is the more stable. The proposal achieves the best predictive performance along the experimental study.

Table 9
Average MCC values and parameters of the used CNN models. The models are sorted by average predictive performance.

Model	MCC	Parameters
NASNetMobile	**0.747**	5,326,716
MobileNet	0.728	**4,253,864**
DenseNet201	0.728	20,242,984
InceptionV3	0.723	23,851,784
Xception	0.685	22,910,480

Overall, the most suitable datasets were DERM-LIB, PH2 and MED-NODE with 89%, 88% and 77% MCC, respectively. On the other hand, the most complex datasets were MSK-3, ISBI2016 and ISBI2017 with 30%, 39% and 41% MCC, respectively. However, the proposal surpassed the average performance of its competitors by 190% when using segmented images and 375% when not in MSK-3. In addition, the proposal achieved always a 9% better performance compared to the average results in BCN20000, which is the largest dataset publicly available. Furthermore, the CNN models achieved the best average performance when training with the images generated by the proposal. The proposed RGAN achieved the best performance the 82% of the time.

To sum up, Table 9 shows the average performance of each CNN model and their number of trainable parameters, where NASNet achieved the top average performance and Xception ended last. These results indicate that a bigger number of trainable parameters will not necessarily obtain a better performance in melanoma diagnosis.

5. Conclusions

In this work, the diagnosis of melanoma was addressed via a series of contributions. Firstly, a double residual architecture was designed and applied for melanoma diagnosis in order to generate plausible synthetic skin images. The architecture was evaluated qualitatively and quantitatively by using a manual inspection and the IS score, respectively. Results showed stable learning even with a low number of samples. Then, an extensive experimental study was performed on sixteen skin image datasets. Overall, results showed that the proposed architecture significantly surpassed several state-of-the-art data augmentation techniques in five CNN models. The above corroborated the hypothesis that complex data augmentation techniques were suitable to train CNN models, even in small datasets with complex properties. In addition, to preprocess data, a segmentation method was applied. The results showed that all CNN models improved their average performance by using segmented data. The performance was increased by applying transfer learning from pre-trained ImageNet models. Bear in mind that transfer learning alleviates the requirement for a large number of training data.

Future works will conduct more extensive experiments to validate the full potential of the proposed architecture, for example by considering a wide set of hyperparameters to be tuned. In addition, we look forward to developing new evaluation metrics in order to maintain an equilibrium during the training process, as well as objectively evaluate GAN-based proposal. Furthermore, it will be interesting to receive the feedback from dermatologists regarding realism. Finally, it is noteworthy that the proposed approach is not strictly restricted to melanoma diagnosis problem, and based on the results it could be applied in the future on other complex real-world problems where data is limited.

Declaration of competing interest

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

Acknowledgments

This research was funded by the Spanish Ministry of Science and Innovation, the University of Cordoba, Spain, and the European Regional Development Fund, projects UCO-FEDER 18 REF.1263116 MOD and PID2020-115832GB-I00. It was also funded by the i-PFIS, Spain contract no. IFI17/00015 granted by the Health Institute Carlos III of Spain. Funding for open access charge: Universidad de Córdoba / CBUA.

References

[1] American Cancer Society. Cancer facts and figures. 2021, Consulted on June 22, 2021 URL https://bit.ly/3gNDBVr.
[2] Esteva A, Kuprel B, Novoa R, Ko J, Swetter S, Blau H, et al. Dermatologist-level classification of skin cancer with deep neural networks. Nature 2017;542(7639):115–8.
[3] Kittler H, Pehamberger H, Wolff K, Binder M. Diagnostic accuracy of dermoscopy. Lancet Oncol 2002;3(3):159–65.
[4] Ali A-R, Deserno TM. A systematic review of automated melanoma detection in dermatoscopic images and its ground truth data. In: Progress in biomedical optics and imaging - Proceedings of SPIE, vol. 8318. 2012.
[5] Sánchez-Monedero J, Pérez-Ortiz M, Sáez A, Gutiérrez PA, Hervás-Martínez C. Partial order label decomposition approaches for melanoma diagnosis. Appl Soft Comput 2018;64:341–55.
[6] Pérez E, Reyes O, Ventura S. Convolutional neural networks for the automatic diagnosis of melanoma: An extensive experimental study. Med Image Anal 2021;67.
[7] Binder M, Schwarz M, Winkler A, Steiner A, Kaider A, Wolff K, et al. Epiluminescence microscopy: a useful tool for the diagnosis of pigmented skin lesions for formally trained dermatologists. Arch Dermatol 1995;131(3):286–91.
[8] Nasr-Esfahani E, Samavi S, Karimi N, Soroushmehr S, Jafari M, Ward K, et al. Melanoma detection by analysis of clinical images using convolutional neural network. In: Proceedings of the annual international conference of the IEEE engineering in medicine and biology society. 2016, p. 1373–6.
[9] Brinker TJ, Hekler A, Enk AH, Klode J, Hauschild A. A convolutional neural network trained with dermoscopic images performed on par with 145 dermatologists in a clinical melanoma image classification task. Eur J Cancer 2019;111:148–54.
[10] Khan AI, Shah JL, Bhat MM. CoroNet: A deep neural network for detection and diagnosis of COVID-19 from chest x-ray images. Comput Methods Programs Biomed 2020;196. http://dx.doi.org/10.1016/j.cmpb.2020.105581.
[11] Asif U, Bennamoun M, Sohel F. A multi-modal, discriminative and spatially invariant CNN for RGB-D object labeling. IEEE Trans Pattern Anal Mach Intell 2018;40(9):2051–65.
[12] Ericsson. On the pulse of the networked society. Tech. Rep., Ericsson; 2015, URL https://apo.org.au/node/59109.
[13] Lenc K, Vedaldi A. Understanding image representations by measuring their equivariance and equivalence. Int J Comput Vis 2019;127(5):456–76.
[14] Perez F, Vasconcelos C, Avila S, Valle E. Data augmentation for skin lesion analysis. In: OR 2.0 context-aware operating theaters, computer assisted robotic endoscopy, clinical image-based procedures, and skin image analysis. Granada, Spain: Springer; 2018, p. 303–11.
[15] Pérez E, Ventura S. An ensemble-based convolutional neural network model powered by a genetic algorithm for melanoma diagnosis. Neural Comput Appl 2021. http://dx.doi.org/10.1007/s00521-021-06655-7.
[16] Kawahara J, Daneshvar S, Argenziano G, Hamarneh G. Seven-point checklist and skin lesion classification using multitask multimodal neural nets. IEEE J Biomed Health Inf 2019;23(2):538–46.
[17] Baur C, Albarqouni S, Navab N. MelanoGANs: high resolution skin lesion synthesis with GANs. 2018, arXiv preprint arXiv:1804.04338.
[18] Pérez E, Ventura S. Melanoma recognition by fusing convolutional blocks and dynamic routing between capsules. Cancers 2021;13(19). http://dx.doi.org/10.3390/cancers13194974.
[19] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition. 2016, p. 770–8.
[20] Huang G, Liu Z, Van Der Maaten L, Weinberger K. Densely connected convolutional networks. In: Proceedings - 30th IEEE conference on computer vision and pattern recognition. 2017.
[21] Ciresan DC, Meier U, Gambardella LM, Schmidhuber J. Deep, big, simple neural nets for handwritten digit recognition. Neural Comput 2010;22(12):3207–20.
[22] Van den Berg RA, et al. Centering, scaling, and transformations: improving the biological information content of metabolomics data. BMC Genomics 2006;7(1):142.
[23] Al-masni M, Kim D-H, Kim T-S. Multiple skin lesions diagnostics via integrated deep convolutional networks for segmentation and classification. Comput Methods Programs Biomed 2020;190. http://dx.doi.org/10.1016/j.cmpb.2020.105351.

[24] Rubin M, Stein O, Turko NA, Nygate Y, Roitshtain D, Karako L, et al. TOP-GAN: Stain-free cancer cell classification using deep learning with a small training set. Med Image Anal 2019;57:176–85.
[25] Liang G, Zheng L. A transfer learning method with deep residual network for pediatric pneumonia diagnosis. Comput Methods Programs Biomed 2020;187. http://dx.doi.org/10.1016/j.cmpb.2019.06.023.
[26] Zhong Z, Zheng L, Kang G, Li S, Yang Y. Random erasing data augmentation. 2017, URL https://arxiv.org/abs/1708.04896.
[27] Xie Q, Dai Z, Hovy E, Luong M-T, Le QV. Unsupervised data augmentation for consistency training. 2019, URL https://arxiv.org/abs/1904.12848.
[28] Radford A, Metz L, Chintala S. Unsupervised representation learning with deep convolutional generative adversarial networks. 2015, URL https://arxiv.org/abs/1511.06434.
[29] Denton E, Chintala S, Szlam A, Fergus R. Deep generative image models using a laplacian pyramid of adversarial networks. In: Advances in neural information processing systems, vol. 2015-Janua. Montreal, Canada; 2015.
[30] Qin Z, Liu Z, Zhu P, Xue Y. A GAN-based image synthesis method for skin lesion classification. Comput Methods Programs Biomed 2020;195. http://dx.doi.org/10.1016/j.cmpb.2020.105568.
[31] Abdelhalim ISA, Mohamed MF, Mahdy YB. Data augmentation for skin lesion using self-attention based progressive generative adversarial network. Expert Syst Appl 2021;165:113922. http://dx.doi.org/10.1016/j.eswa.2020.113922, URL https://www.sciencedirect.com/science/article/pii/S0957417420307132.
[32] Pollastri F, Bolelli F, Paredes R, Grana C. Augmenting data with GANs to segment melanoma skin lesions. Multimedia Tools Appl 2020;79(21–22):15575–92. http://dx.doi.org/10.1007/s11042-019-7717-y, cited By 36. URL https://www.scopus.com/inward/record.uri?eid=2-s2.0-8506606484&doi=10.1007%2fs11042-019-7717-y&partnerID=40&md5=53eb2268b1cb3a49085aa185a41c904d.
[33] Cubuk ED, Zoph B, Shlens J, Le QVR. Practical data augmentation with no separate search. 2019, arXiv preprint arXiv:1909.13719.
[34] Sennrich R, Haddow B, Birch A. Improving neural machine translation models with monolingual data. 2015, arXiv preprint arXiv:1511.06709.
[35] Goodfellow IJ, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, et al. Generative adversarial nets. In: Advances in neural information processing systems, vol. 3, no. January. Montreal, Quebec, Canada; 2014, p. 2672–80.
[36] Karras T, Aila T, Laine S, Lehtinen J. Progressive growing of GANs for improved quality, stability, and variation. In: 6th International conference on learning representations. 2018.
[37] Chintala S, Denton E, Arjovsky M, Mathieu M. How to train a GAN? Tips and tricks to make GANs work. 2016.
[38] Mao X, Li Q, Xie H, Lau RYK, Wang Z, Paul Smolley S. Least squares generative adversarial networks. In: Proceedings of the IEEE international conference on computer vision. 2017, p. 2794–802.
[39] Arjovsky M, Chintala S, Bottou L. Wasserstein gan. 2017, arXiv preprint arXiv:1701.07875.
[40] Gulrajani I, Ahmed F, Arjovsky M, Dumoulin V, Courville AC. Improved training of wasserstein gans. In: Advances in neural information processing systems. California, USA; 2017, p. 5767–77.
[41] Cerwall P, Jonsson P, Möller R, Bävertoft S, Carson S, Godor I. On the pulse of the networked society. Ericsson Mobil Rep 2015.
[42] Rousseeuw PJ. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. J Comput Appl Math 1987;20:53–65.
[43] Borji A. Pros and cons of GAN evaluation measures. Comput Vis Image Underst 2019;179:41–65.
[44] Salimans T, Goodfellow I, Zaremba W, Cheung V, Radford A, Chen X. Improved techniques for training gans. Adv Neural Inf Process Syst 2016;29:2234–42.
[45] Gao Y, Kong B, Mosalam KM. Deep leaf-bootstrapping generative adversarial network for structural image data augmentation. Comput-Aided Civ Infrastruct Eng 2019;34(9):755–73. http://dx.doi.org/10.1111/mice.12458.
[46] Li W, Zhang P, Zhang L, Huang Q, He X, Lyu S, et al. Object-driven text-to-image synthesis via adversarial training. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition, vol. 2019-June. 2019, p. 12166–74. http://dx.doi.org/10.1109/CVPR.2019.01245.
[47] Li Y, Min MR, Shen D, Carlson D, Carin L. Video generation from text. In: 32nd AAAI conference on artificial intelligence. 2018, p. 7065–72.
[48] Wei M, Wu Q, Ji H, Wang J, Lyu T, Liu J, et al. A skin disease classification model based on DenseNet and ConvNeXt fusion. Electronics (Switzerland) 2023;12(2). http://dx.doi.org/10.3390/electronics12020438.
[49] Goodfellow I, Bengio Y, Courville A, Bengio Y. Deep learning, vol. 1. MIT press Cambridge; 2016.
[50] Pontoriero A, et al. Automated data quality control in FDOPA brain PET imaging using deep learning. Comput Methods Programs Biomed 2021;208. http://dx.doi.org/10.1016/j.cmpb.2021.106239.
[51] Bergstra J, Bengio Y. Random search for hyper-parameter optimization. J Mach Learn Res 2012;13(Feb):281–305.
[52] Boughorbel S, Jarray F, El-Anbari M. Optimal classifier for imbalanced data using matthews correlation coefficient metric. PLoS One 2017;12(6):e0177678.
[53] Chicco D, Jurman G. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. BMC Genomics 2020;21:1–13.
[54] Chicco D, Tötsch N, Jurman G. The matthews correlation coefficient (MCC) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation. BioData Min 2021;14(1):1–22.
[55] Alzahrani S, Al-Bander B, Al-Nuaimy W. A comprehensive evaluation and benchmarking of convolutional neural networks for melanoma diagnosis. Cancers 2021;13(17):4494.
[56] Halimu C, Kasem A, Newaz SS. Empirical comparison of area under ROC curve (AUC) and Mathew correlation coefficient (MCC) for evaluating machine learning algorithms on imbalanced datasets for binary classification. In: Proceedings of the 3rd international conference on machine learning and soft computing. 2019, p. 1–6.
[57] Wilcoxon F. Individual comparisons by ranking methods. Biometrics 1945;1(6):80–3.
[58] Friedman M. A comparison of alternative tests of significance for the problem of m rankings. Ann Math Stat 1940;11(1):86–92.
[59] Hommel G. A stagewise rejective multiple test procedure based on a modified bonferroni test. Biometrika 1988;75(2):383–6.
[60] Heusel M, Ramsauer H, Unterthiner T, Nessler B, Hochreiter S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. Adv Neural Inf Process Syst 2017;30.
[61] Shmelkov K, Schmid C, Alahari K. How good is my GAN? In: Proceedings of the European conference on computer vision. 2018, p. 213–29.