Weight Rotation as a Regularization Strategy in Convolutional Neural Networks

Eduardo Castro[1,2] Jose Costa Pereira[1,2,3] Jaime S. Cardoso[1,2]

***Abstract*— Convolutional Neural Networks (CNN) have become the gold standard in many visual recognition tasks including medical applications. Due to their high variance, however, these models are prone to over-fit the data they are trained on. To mitigate this problem, one of the most common strategies, is to perform data augmentation. Rotation, scaling and translation are common operations. In this work we propose an alternative method to rotation-based data augmentation where the rotation transformation is performed inside the CNN architecture. In each training batch the weights of all convolutional layers are rotated by the same random angle. We validate our proposed method empirically showing its usefulness under different scenarios.**

I. Introduction

Over the last few years Convolutional Neural Networks (CNNs) have gained importance in the field of visual recognition [1]. When enough data is available and the complexity of the task makes it difficult to use traditional methods, CNNs have shown remarkable results. Also in medical imaging these models have been increasingly used [2].

A central aspect of CNNs is their ability to represent virtually any function by changing their parameterization (*i.e.* they are universal approximators). From the bias-variance trade-off point of view it is easy to see that CNNs are models with very low bias and high variance [3]. Because of this, they are prone to learn the idiosyncrasies of the training data; a well known behaviour called over-fitting. For most applications this behaviour is undesired as it compromises the learning of features that generalize to unseen data in real world scenarios, where the model is likely to be used. In small datasets – which are typical in medical imaging applications –, over-fit is therefore more likely to occur.

A common strategy to mitigate this problem is through data augmentation [4]. This technique aims at expanding the original training set with transformed inputs, which are also present in the theoretical population where the data is sampled from. One of the most common transformations for data augmentation in medical imaging are rotations.

In this work we propose a method in which rotations are applied to the model's parameters instead of the input. Briefly, in each training iteration the weights of all convolutional layers are rotated by the same random angle. Through this, we can generate internal representations which are approximately equal to those obtained by input rotation, but with the benefit of not having to perform this operation on data. Experimental evaluation shows that the proposed method can substitute data rotation and in some cases lead to better generalization. Additionally, it is faster to do weight rather than input rotation making the former more appealing.

[1] Faculty of Engineering, University of Porto, Porto, Portugal
[2] Instituto de Engenharia de Sistemas e Computadores, Tecnologia e Ciência, Porto, Portugal
[3] Huawei Noah's Ark Lab, London, UK

II. Related Work

Rotation transformations for data augmentation are common in image recognition problems, including medical applications [5]. This process can be done either online where transformations are calculated during training, or offline where data manipulation takes place before optimization. Usually, the justification behind using a set of transformations for data augmentation is that they occur naturally in the data; *i.e.*, under reasonable transformation constraints, if x is a sample taken from an unknown population, a transformed version of x must also be part of that population. Due to this, angles in $[0°, 360°]$ are sampled for rotation invariant problems [4] while a smaller range is selected for rotation variant ones [6]. For angles not multiple of $90°$, rotation leads to the occlusion of some parts of the image (see Fig. 1).

Some works deal with rotation by designing adequate model architectures. Examples of these include Group Equivariant Networks [7] where convolutional layers are made equivariant to specific families of transformations and Harmonic Networks [8] where circular harmonics replace regular CNN filters and each neuron returns a maximal response and orientation.

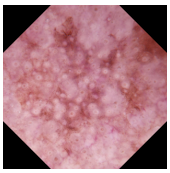

Fig. 1: Image rotation can lead to occlusion.

III. Methods

A. Convolutional Neural Netwoks

Convolutional Neural Networks are feed-forward deep neural networks that make use of convolutional layers. Different from fully-connected ones, convolutional layers share weights across spatial dimensions. Weight sharing across spatial dimensions leads to important properties in CNNs which explain their good performance when modeling natural data [1]. In this work we propose a new method which manipulates convolutional weights in order to promote rotation invariance at the model's output.

Fig. 2: Comparison between the traditional 90° rotation-based data augmentation (left) and the proposed method (right). While in the first case the objects are rotated, our proposed method rotates the model "observing" the objects.

B. Weight Rotation in Convolutional Layers

Given a 2D image, I, and a convolutional filter W we denote the convolution operation as:

$$(I * W)(x) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} I(\tau)W(x-\tau)d\tau_1 d\tau_2 \quad (1)$$

A rotation transformation, T_α, applied to I is defined as $(T_\alpha.I)(x) = I(x')$ where x' is obtained by multiplying x by the rotation matrix R_α:

$$\begin{bmatrix} x'_1 \\ x'_2 \end{bmatrix} = R_\alpha x = \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad (2)$$

Our first observation is that the result of the convolution operation for a rotated image can be obtained by rotating the filter in the opposite direction, performing the convolution and then rotating back the result. To prove this we start by writing the convolution operation for a rotated I.

$$((T_\alpha I) * W)(x) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} (T_\alpha I)(\tau)W(x-\tau)d\tau_1 d\tau_2 \quad (3)$$

Let $\tau' = R_\alpha \tau$ the integral is rewritten as:

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} I(\tau')W(x-\tau)d\tau_1 d\tau_2 = \quad (4)$$

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} I(\tau')(T_{-\alpha}W)(x'-\tau')d\tau'_1 d\tau'_2 = \quad (5)$$

$$(I * (T_{-\alpha}W))(x') = (T_\alpha(I * (T_{-\alpha}W)))(x) \quad (6)$$

Convolutional layers work by performing many 2D convolutions and summing the outputs in an ordered way. The layer has an input with multiple channels, $\mathbf{I}$, and a weight tensor, $\mathbf{W}$, with many filters, each with the same number of channels as $\mathbf{I}$. We denote the operation of a convolutional layer as $f(\mathbf{I}, \mathbf{W})$. Because we are only performing convolutions and sums, the equivalence previously demonstrated, between filter rotation and image rotation, applies to convolutional layers also as long as all channels and filters are rotated by the same amount and discretization and edge effects are not considered:

$$f(\mathbf{T}_\alpha.\mathbf{I}, \mathbf{W}) = \mathbf{T}_\alpha.f(\mathbf{I}, \mathbf{T}_{-\alpha}.\mathbf{W}) \quad (7)$$

Equivalently,

$$\mathbf{T}_{-\alpha}.f(\mathbf{T}_\alpha.\mathbf{I}, \mathbf{W}) = f(\mathbf{I}, \mathbf{T}_{-\alpha}.\mathbf{W}) \quad (8)$$

Effectively, we can implement this weight rotation operation on convolutional layers as shown in section III-D. Composing two convolutional layers with this rotation property, yields:

$$\begin{aligned} &f(f(\mathbf{I}, \mathbf{T}_{-\alpha}.\mathbf{W}_1), \mathbf{T}_{-\alpha}.\mathbf{W}_2) = \\ &\mathbf{T}_{-\alpha}.f(\mathbf{T}_\alpha.\mathbf{T}_{-\alpha}.f(\mathbf{T}_\alpha.\mathbf{I}, \mathbf{W}_1), \mathbf{W}_2) = \\ &\mathbf{T}_{-\alpha}.f(f(\mathbf{T}_\alpha.\mathbf{I}, \mathbf{W}_1), \mathbf{W}_2) \end{aligned} \quad (9)$$

This simply shows that the relation expressed in eq.(8) can be extended to compositions of convolutional layers as long as all of them operate with weights rotated by $-\alpha$.

Although the previous argument disregards discretization and edge effects, they are present when we deal with digital images. With both filter rotation and input rotation there are interpolation errors for all $\alpha \notin [0°, 90°, 180°, 270°]$. In the case of weight rotation these errors can significantly change the distribution of weights, leading the intermediate representations of a CNN to be angle-dependent. In this work we reduced this dependence by normalizing each filter so that it has the same mean and standard deviation as the original filter, where α is zero.

C. Rotation-based Weight Regularization

In CNNs, each filter captures some feature in the data, and these features are often directional. In rotation-invariant problems, features in data appear with the same frequency in all orientations. As such, a model which generalizes to unseen data should be able to capture the same high-level information regardless of input orientation.

To encourage this, we propose to rotate all filters of the network by the same random angle, before each forward pass in training. As shown, if we disregard discretization and edge occlusion effects, input and filter rotation are equivalent. This can also be thought of as a generalization of normal training, where the weights are optimized with a *single orientation*. When using rotation-based weight regularization the weights are trained with a *random orientation*, which varies in each iteration.

There are some key differences between the proposed method and rotation-based data augmentation. The network is explicitly encouraged to be invariant to filter orientation rather than input orientation. Additionally, due to image occlusion and interpolation errors, the numerical results are

equal only for $\alpha \in [0°, 90°, 180°, 270°]$. While in data augmentation the interpolation error is only present on the input, for rotation-based weight regularization, this error is introduced in all layers of the network. These differences explain different outputs for the same model when using the same α, and can lead to different performance.

Based on this differences weight rotation can be advantageous in some real world scenarios: (i) rotating big images is costly; (ii) the response for multiple rotations of the image can be computed while transferring only the original data to the GPU; and (iii) image rotation can lead to occlusion.

D. Weight Rotation Implementation

When rotating a digital image by angles not multiple of $90°$, the integer coordinates of the result lie on non-integer positions of the original image, requiring an interpolation method. In this work we used bi-linear interpolation to obtain the coefficients necessary to rotate the weights. These coefficients were stored in a tensor, $\mathbf{C}$, which is used by the model to compute the rotated weights. Each element of this tensor, $c_{i,j,k,l}$, holds the coefficient of element (i, j) of the new filter which is to be multiplied by the element (k, l) of the reference filter. The weight-rotation operation can be summarized in Einstein notation:

$$\mathbf{U}^{i,j}{}_{f_i,f_o} = \mathbf{C}^{i,j}{}_{k,l}\mathbf{W}^{k,l}{}_{f_i,f_o} \tag{10}$$

Where $\mathbf{U}$ is the rotated weight tensor and $\mathbf{W}$ the original one. Points outside the regular grid are handled by repeating edge values.

IV. Experiments

In this section weight rotation is evaluated in multiple settings. First the similarities and differences between input rotation and weight rotation are discussed and experimentally validated on the well-known MNIST dataset. We then proceed to show the effectiveness of weight rotation regularization on one rotation variant problem and three medical imaging ones, which are invariant to rotation. A demonstration of the time-efficiency of weight rotation compared to data augmentation concludes this section.

A. Similarity between Weight Rotation and Input Rotation

MNIST is a well-known digit recognition dataset. Although recent years' advances have trivialized this problem, we use it to illustrate our proposed filter rotation method.

The first observation is that the proposed method is able to simulate input rotation. For this, we take into consideration the digits *6* and *9*. When rotated by $180°$, the digit *6* resembles a *9* (Fig. 3), and the converse is also true. We trained a small CNN to classify these two (handwritten) digits. We then verify the effect of image rotation and weight rotation on test set accuracy, as shown on Fig. 3.

Both methods gradually lead the model to confuse between the two classes. When the angle of rotation is $180°$ the accuracy almost reaches zero, meaning most *6's* are being classified as *9's* and, conversely, most *9's* are being classified as *6's*. This percentage is expected as the appearance of the

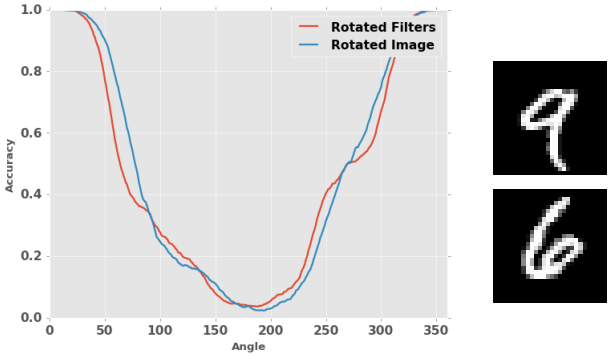

Fig. 3: Effect on the test set accuracy of image rotation vs weight rotation. Both methods lead to confusion between *6's* and *9's*

images belonging to each class, when turned around, closely resemble the appearance of the images of the other class. This experiment demonstrates the similarity between weight rotation and input rotation.

B. Differences between Weight Rotation and Input Rotation

On this experiment we trained a CNN on a variation of the MNIST dataset, where all samples are rotated by a random angle. Notice that, although there is some confusion among some classes – namely *6's* and *9's* –, the problem becomes rotation invariant, as the class of each sample becomes independent of its orientation.

In this experiment, we use online rotation-based data augmentation. Two models are trained, the first one, N_S, where filters have always the same orientation ($\alpha = 0$) and the second, N_M, where filter orientation is random for each batch ($\alpha = \mathcal{U}(0, 360)$). The test-set accuracy for different rotations of the input and the weights is shown on Fig. 4.

For a model trained with single orientation weights, N_S, changing weight orientation during inference leads to a much lower test set accuracy, if interpolation is required. For angles that are multiple of $90°$, where no interpolation is required, the accuracy is equal to that of image rotation. As for N_M, changing filter orientation leads to negligible changes in accuracy. Although the N_S model has a higher accuracy when no weight rotation is used, if we average the predictions of N_M for 16 orientations the test set accuracy surpasses that of N_S (98.19% against 97.68%). Notice that averaging the predictions of N_S for different weight orientations leads to a worse test set accuracy. If we also aggregate the predictions for different image orientations the models compare very similarly (98.34% for the single orientation model against 98.35% for the multiple orientation one). This experiment shows that weight rotation and input rotation are not always interchangeable, as they produce different numerical results for angles not multiple of $90°$.

C. Regularization on Rotation Variant Problems

In this section we demonstrate the usability of rotation-based weight regularization on problems where the data is not independent of orientation. For this we used the

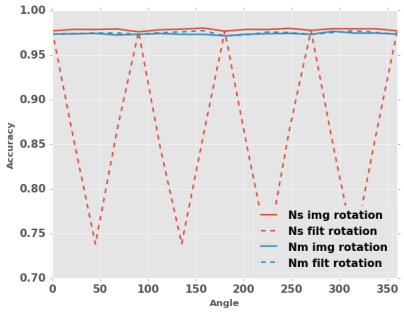

Fig. 4: Test set accuracy for a rotation-invariant variation of MNIST, as a function of angle of rotation, α, of the input and of the weights. N_S is a model trained with *single orientation* weights and N_M with *random orientation* ones.

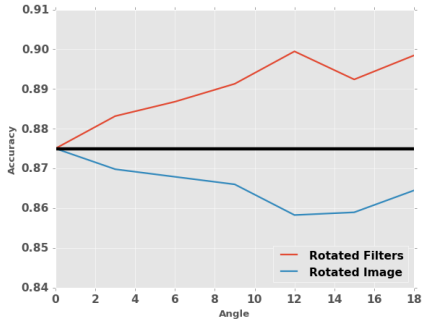

Fig. 5: Effect of rotation during training on the test set accuracy, when applied to the input and to the weights.

Small NORB dataset [9], which is composed of photos of 50 toys equally divided in 5 categories under different lighting conditions, elevations and azimuths, with no color or background and with a standard train/test split.

Small NORB images are squared and have side length of 96. For training we took only a central patch with size 64 to ensure that, if the input was rotated, the resulting image would be contained in the original photo. We adapted the Resnet-34 [10] model to compensate for the smaller input size, by removing the initial convolution and max-pooling layers, each with stride 2. Two models were trained, one with rotation-based data augmentation and the other with weight regularization, for different intervals of $\alpha \in [-\alpha_{max}, \alpha_{max}]$. Each model was trained for 75 epochs. The test set accuracy for different values of α_{max} is shown on Fig. 5.

The results show that, for small values of α_{max}, increasing the angle of rotation leads to more accurate models when weight rotation is used, and to a decrease in accuracy if rotation is intead performed on the input. Differently from rotated MNIST, an image generated by rotation is not part of the theoretical distribution where the data is sampled from, in the case of Small NORB. Additionally, input rotation can occlude some parts of the objects during training. These two factors can explain the good performance of rotation-based weight regularization against data augmentation.

D. Regularization on Medical Imaging Data

Three publicly available datasets were used to validate the proposed method on medical images.

TABLE I: Balanced test set accuracies for Medical datasets.

Rotation	None	Input	Weights	Both
INbreast	54.87%	62.09%	**67.30%**	66.67%
ISIC 2017	77.67%	78.70%	**80.00%**	78.96%
CBIS-DDSM	55.09%	**61.26%**	60.24%	57.44%

INbreast [11] is a mammographic database with precise lesion annotation. To evaluate the performance of the proposed method patches centered in the annotated masses are taken, including their surrounding regions. Lesions are considered positive if they belong to mammograms with a BIRADS (the standard reporting scale in mammography) score higher than 2. A total of 116 patches were taken. The reported accuracy was obtained by averaging over 5 splits

CBIS-DDSM [12], [13] is a scanned film mammography dataset with 2620 images, a standard train/test split and local lesion annotations. To evaluate our method, a patch centered in each lesion was taken. This yielded a classification problem of 3568 images separated in four classes: benign masses, malignant masses, benign calcifications and malignant calcifications.

Finally, the 2017 ISIC challenge data [14], [15] was used. Three classes are available in the dataset: nevus, seborrheic keratosis and melanoma, but to simplify the problem, which is highly unbalanced, we considered only the first two. In total, we used 1626 images for training and 600 for test. Contrary to the previous two datasets, where patches were taken, in this dataset image rotation leads to occlusion.

Our baselines were obtained using Resnet-34 [10], Resnet-18 and Vgg16 [16] for the INbreast, CBIS-DDSM and ISIC, respectively. The number of filters in each model was reduced, as the number of available images is much smaller when compared to datasets like ImageNet, where these models are typically used. All models were trained from scratch using stochastic gradient descent with momentum. Class weights are used, along with balanced accuracy as an evaluation metric, since all datasets are unbalaced. Results are shown in Table I.

Using the proposed regularization method leads to increased accuracy on the test set, demonstrating weight-rotation is an effective way of increasing model robustness on rotation-invariant problems. When compared to rotation-based data augmentation, weight-based regularization performed better on INbreast and ISIC, over 8% and 1.6% respectively. The performance was slightly lower on the CBIS-DDSM. The different margins of gains when comparing rotation-based weight regularization with data augmentation, suggest that dataset idiosyncrasies and model architectures may have an impact on the final performance value. Interestingly, when rotation-based weight regularization is present, adding data augmentation leads to worse test set accuracy. This is due to the fact that we increase the variability of the training data, without adding any valuable information about rotation-invariance.

E. Time Efficiency in Multiple Orientation Inference

As mentioned before, weight rotation is computationally cheaper than image rotation which, for some applications,

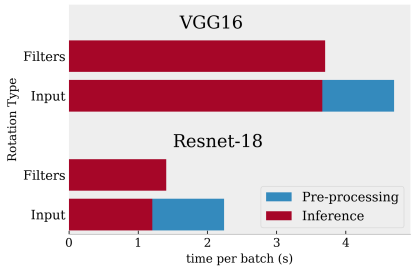

Fig. 6: Time required to evaluate one batch of 120 images when using weight rotation and input rotation.

can be a considerable advantage. To demonstrate this we consider the common case where, at inference, the input is rotated multiple times and the outputs of all orientations combined for a more robust classification. In this section, the models previously trained on ISIC and CBIS-DDSM were used.

We verified that averaging the prediction for multiple orientations leads to an increase in accuracy for both methods, as long as the rotation method used for inference is the same as that of training. For the ISIC dataset, weight rotation leads to an increase from 80.00% to 82.54%, while rotation on the image has a smaller effect, from 78.70% to 80.24%. Similar results were obtained on CBIS-DDSM, with an increase from 60.24% to 62.29% for weight rotation, against 61.26% to 62.70% for image rotation. Combining the two methods of rotation did not lead to higher accuracy in any model.

Regarding the computational cost, Fig. 6 shows the time required for each model to perform inference on a set of 120 images with 16 orientations. The results shown were obtained by averaging over 100 runs. Using weight rotation instead of image rotation leads to a reduction of 21.2% of the time required, for the Vgg16 model, and 37.3% for the Resnet-18 model. Although weight rotation increases the time required to do model inference, this increase is small when compared to the time necessary for input rotation. The reduced time at inference is highly dependent on the model used, image size and hardware. In this section we demonstrate this difference for average-sized images and commonly used models. In this work a GTX 1080 GPU along with an i7-6700k CPU were used.

V. Conclusion

In this work we propose a regularization method based on weight-rotation, which aims at increasing the robustness of convolutional neural networks to changes in orientation of the objects on the image. The method is well-suited for rotation-invariant problems, but can also be useful for rotation-variant ones. We also provide a detailed explanation on the differences and similarities between the proposed method and traditional rotation-based data augmentation, along with empirical evaluation. As conclusion, rotation-based weight regularization is a competitive alternative to rotation-based data augmentation, which can be preferred in some situations.

VI. Acknowledgements

The authors would like to thank NVIDIA by their generous donation of a Titan Xp GPU. This work is co-financed by the ERDF - European Regional Development Fund through the Norte Portugal Regional Operational Programme (NORTE 2020), and the LISBOA2020 under the PORTUGAL 2020 Partnership Agreement, through the Portuguese National Innovation Agency (ANI) as a part of project BCCT.plan: NORTE-01-0247-FEDER-017688 and by National Funds through the Portuguese funding agency, FCT - Fundação para a Ciência e a Tecnologia within PhD grant number *SFRH/BD/136274/2018*.

References

[1] Yann Lecun, Yoshua Bengio, and Geoffrey Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 5 2015.

[2] T. Kooi, G. J. S. Litjens, B. van Ginneken, A. Gubern-Mérida, C. I. Sánchez, R. Mann, A. den Heeten, and N. Karssemeijer, "Large scale deep learning for computer aided detection of mammographic lesions," *Medical image analysis*, vol. 35, pp. 303–312, 2017.

[3] S. Geman, E. Bienenstock, and R. Doursat, "Neural networks and the bias/variance dilemma," *Neural Computation*, vol. 4, no. 1, pp. 1–58, Jan. 1992.

[4] E. Castro, J. S. Cardoso, and J. C. Pereira, "Elastic deformations for data augmentation in breast cancer mass detection," in *2018 IEEE EMBS International Conference on Biomedical Health Informatics (BHI)*, March 2018, pp. 230–234.

[5] Hao Chen, Qi Dou, Xi Wang, Jing Qin, and Pheng-Ann Heng, "Mitosis detection in breast cancer histology images via deep cascaded networks," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*. 2016, AAAI'16, pp. 1160–1166, AAAI Press.

[6] Yoshihiro Shima, "Image augmentation for object image classification based on combination of pre-trained cnn and svm," *Journal of Physics: Conference Series*, vol. 1004, pp. 012001, 2018.

[7] Taco S. Cohen and Max Welling, "Group equivariant convolutional networks," in *Proceedings of the 33rd ICML - Volume 48*. 2016, ICML'16, pp. 2990–2999, JMLR.org.

[8] D. E. Worrall, S. J. Garbin, D. Turmukhambetov, and G. J. Brostow, "Harmonic networks: Deep translation and rotation equivariance," in *CVPR 2017*, 2017, pp. 7168–7177.

[9] Y. LeCun, Fu Jie Huang, and L. Bottou, "Learning methods for generic object recognition with invariance to pose and lighting," in *Proceedings of CVPR 2004*, June 2004, vol. 2, pp. II–104 Vol.2.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," *CVPR*, 2016.

[11] I. C. Moreira, I. F. A. Amaral, I. Domingues, A. R. Cardoso, M. J. Cardoso, and J. S. Cardoso, "Inbreast: toward a full-field digital mammographic database.," *Academic radiology*, vol. 19(2), pp. 236–48, 2012.

[12] R. S. Lee, F. Gimenez, A. Hoogi, K. K. Miyake, M. Gorovoy, and D. L. Rubin, "A curated mammography data set for use in computer-aided detection and diagnosis research," *Scientific Data*, vol. 4, pp. 170177, 2017.

[13] K. Clark, B. Vendt, K. Smith, J. Freymann, J. Kirby, P. Koppel, S. Moore, S. Phillips, D. Maffitt, M. Pringle, L. Tarbox, and F. Prior, "The cancer imaging archive (tcia): Maintaining and operating a public information repository," *Journal of Digital Imaging*, vol. 26, no. 6, pp. 1045–1057, Dec 2013.

[14] Philipp Tschandl, Clifford Rosendahl, and Harald Kittler, "The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions," in *Scientific data*, 2018.

[15] N. C. F. Codella, D. Gutman, M. E. Celebi, B. Helba, M. A. Marchetti, S. W. Dusza, A. Kalloo, K. Liopyris, N. K. Mishra, H. Kittler, and A. Halpern, "Skin lesion analysis toward melanoma detection: A challenge at the 2017 international symposium on biomedical imaging (isbi), hosted by the international skin imaging collaboration (isic)," *ISBI*, pp. 168–172, 2018.

[16] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.