Deep Transfer Learning Optimization Techniques for Medical Image Classifcation - A Survey

SUBMISSION DATE / POSTED DATE

15-04-2023 / 19-04-2023

CITATION

Kariuki, Paul; Gikunda, Dr. Patrick Kinyua; Mwangi Wandeto, Dr. John (2023): Deep Transfer Learning Optimization Techniques for Medical Image Classifcation - A Survey. TechRxiv. Preprint. https://doi.org/10.36227/techrxiv.22638937.v1

DOI

10.36227/techrxiv.22638937.v1

Deep Transfer Learning Optimization Techniques for Medical Image Classifcation: A Survey

1st Paul Wahome Kariuki
Department of Computer Science and Informatics
Karatina University
Karatina, Kenya
kariukip@karu.ac.ke

2nd Dr. Patrick Kinyua Gikunda
Department of Computer Science
Dedan Kimathi University of Technology
Nyeri, Kenya
patrick.gikunda@dkut.ac.ke

3rd Dr. John Mwangi Wandeto
Department of Computer Science
Dedan Kimathi University of Technology
Nyeri, Kenya
john.wandeto@dkut.ac.ke

Abstract— **Medical image classification is not only a complex task but also a challenging one due to the heterogeneous nature of medical data. Deep transfer learning has proven to be a viable technique for medical image classification throughout the years, mostly because it is able to leverage knowledge from pre-trained models learned from large-scale datasets, improved performance, minimal training and overcoming the disadvantage of small data sets. This paper offers a succinct review of the cutting-edge deep transfer learning optimization approaches for medical image classification. The paper begins with an overview of convolutional neural networks (CNN) and transfer learning techniques, such as relation-based, feature, parameter and instance-based transfer learning. Then, the study examines classical classifiers, such as Resnet, VGG, Alexnet, Googlenet, and Inception, and compare their performance on medical image classification tasks. The study also presents optimization techniques, including batch normalization, regularization, and weight initialization, data augmentation and the kernel mathematical formulations. Finally, the study unearths various challenges that arise when using deep transfer learning for medical image classification as well as potential future approaches for this field.**

Keywords—Deep Transfer Learning, Optimization Techniques, Medical Images Classification

I. Introduction

Medical image classification is a rapidly growing field of research [1] that aims to develop computational methods and tools [17] that are capable of classifying medical images accurately and efficiently. Medical imaging is a crucial part of Computer-Aided Diagnosis (CAD), which has a growing concern of having fast and accurate annotations and or grading of medical images [18]. Several reasons are attributed to its success in healthcare, including: (i) early detection and diagnosis of disease [19], by analysis of patterns or abnormalities in medical images that are not visible to the naked eye. (ii) Personalized treatment [20] - enabling doctors to customize treatment plans based on patient's specific conditions, reducing chances of side effects and (iii) Precision medicine [21] in which patients with unique healthcare needs are identified or those that respond to treatments differently.

Artificial intelligence (AI) augments the innate intelligence of clinicians by using complex computation and inference to generate insights, allowing medical systems to reason and learn [21]. Deep learning has made significant impact in medical imaging due the many processing and preprocessing capabilities it has. Different levels of abstract features can be extracted from the original data and used for target detection and classification by combining multiple nonlinear processing layers [22] present in deep learning models. One of the advantages of deep learning in medical imaging classification is its ability to learn complex features from images without requiring explicit feature extraction [18]. This has been achieved through the use of convolutional neural networks (CNNs), which are specifically designed to handle image data. CNNs have been used in a wide range of medical imaging applications, including classification of brain tumors [24], breast cancer, and lung cancer [23]. In addition to improving accuracy, deep learning has also reduced the time required for medical image classification. This is because deep learning approaches can analyze entire images in a fraction of the time, making them well-suited for use in clinical settings.

Despite its numerous advantages, there are a number of obstacles associated with the classification of medical imaging using deep learning. The requirement for vast amounts of labeled data is one of the primary obstacles [1]. Creating labeled datasets for medical images can be time-consuming, costly and in some instances may not be possible [17].

Transfer learning, in comparison to other prevalent deep learning methods such as Convolutional Neural Networks, (CNN) is notable for being easy to implement, effective, and cheap to train, thus overcoming the limitations of small data sets that has been a big hurdle for researchers in this domain. Medical image analysis complements both scientific research and clinical diagnosis in significant ways. Classification of medical images is an important task for disease diagnosis, treatment planning, and monitoring [1]. However, the quantity and quality of medical images are frequently limited, making it difficult to train deep neural networks from scratch. Deep transfer learning is an effective technique for circumventing this limitation by leveraging pre-trained models that have been trained on large-scale datasets in order to get improved performance with little or less training data.

This paper discusses the use of deep transfer learning optimization techniques for medical image classification. Specifically, the study explores the potential of transfer learning to improve the performance of deep learning classifiers in medical imaging applications. The study begins by providing an overview of transfer learning and medical imaging modalities. Then, the study discusses various deep transfer learning classifiers that have been proposed for medical image classification, followed by a review of deep transfer learning optimization techniques that can be used to further enhance the performance of these classifiers.

The rest of this paper is organized as follows. In Section 2, the paper provides an overview of deep transfer learning classifiers, including a discussion of commonly used architectures and their applications in medical imaging. In Section 3, the paper reviews deep transfer learning optimization techniques, including batch normalization, regularization, and weight initialization. The paper also discusses how these techniques can be combined to further improve performance. In Section 4, the paper provides a critical analysis of the current state of the field and highlight areas for future research. Finally, in Section 5, the paper concludes with a summary of the findings and discuss the potential impact of deep transfer learning optimization techniques on medical imaging classification.

A. *Transfer Learning*

In the realm of deep learning, transfer learning seeks to utilize the acquired knowledge and representations from a previously trained model and apply it to a different task. Broadly speaking, transfer learning is categorized either as homogenous or heterogeneous. Taxonomically, homogenous TL can be presented as scenario where, $X_t = X_s$ and $Y_t = Y_s$ with the goal being to narrow the discrepancy in the data distributions between the source and target domains, i.e. address, $P(X_t) \neq P(X_s)$ and/or $P(Y_t \,|X_t) \neq P(Y_s|X_s)$. Whereas, in heterogeneous TL, the scenario is such that the source and target domains contain distinct feature spaces, $X_t \neq X_s$ (generally non-overlapping) and/or $Y_t \neq Y_s$, as the source and target domains may share no features and/or labels. There are four categories of homogeneous transfer learning methods: instance-based, feature-based, parameter-based, and relation-based transfer learning. In (i) instance-based transfer learning, the acquired knowledge from the source task is transferred directly to the target task in the form of individual instances or examples. Here, the actual data points or instances from the source task are used to augment the training data for the target task. For (ii) Feature-based transfer learning enhances the performance of a pre-trained model on a new task by keeping the pre-trained weights unchanged. In this method, the model learns features from the source task and transfers them to the target task. The model is trained on the source task to extract pertinent features, which are then employed to train the target model; for (iii) Parameter-based transfer learning involves transferring the acquired parameters of the source model to the target model. The source model is trained, and the weights of its layers serve as the initial point for training the target model. Through updating some or all of the pre-trained weights, parameter-based transfer learning fine-tunes the pre-trained model on the new task; in (iv) relation-based transfer learning, the objective is to learn the connection between the source and target domains and then transfer knowledge accordingly. The relationships between the source and target tasks are explicitly modeled, and the knowledge learned from the source task is transferred to the target task via this relationship.

Convolutional neural networks (CNNs) are extensively used for medical image classification due to their capability to automatically learn hierarchical features [25]. A CNN's overall structure is made up of multiple convolutional layers, pooling layers, activation functions, and a softmax layer for classification. Convolutions enable the network to recognize local patterns, whereas pooling reduces the spatial dimensionality of the feature maps. The network is made nonlinear by activation functions, and the softmax layer outputs the probability distribution over the classes.

Studies suggest that CNN-based methods employ a set of strategies that make them suitable for image classification and adoption in transfer learning [25]. One such strategy is data augmentation, used in [26] on an adversarial neural network together with a CNN for image classification. Authors in [27] used a CNN in transfer learning on ImageNet dataset, then chest X-ray 14 dataset and thereafter fine-tuned on COVID-19 dataset. They demonstrated that CNN-based methods are suitable choice for transfer learning because they can learn highly complex visual features from raw image data and generalize well to new datasets [28, 29].

Deep learning methods however suffer from two key challenges, high dependency on extensive labeled training data and higher training costs [31]. It is argued that when transfer learning in deep learning, called Deep Transfer Learning (DTL), is used, these dependencies are minimized and time required to train drastically reduced. Deep transfer learning involves the process of reducing learning costs by using knowledge gained from another task and dataset (even if it is not closely related to the source task or dataset) [31].

B. *Medical Imaging Modalities*

Medical images are visual representations of the internal or external structures of the human body or other living organisms, produced through various imaging modalities. Multi-parametric Magnetic Resonance Imaging (mpMRI), Computer Tomography (CT), X-Rays and Ultrasound (US) are among the most frequently utilized medical imaging techniques. A CT scan produces high tissue resolution; however, they are heavily reliant on the skill of the doctor in addition to exposing a patient to ionizing radiation, increasing the risk of having cancer over time [30]. For initial medical examinations, X-Rays are a low-cost and convenient option, but like CT scans, they can be harmful to the body, limiting the number of times a patient can undergo this procedure. On the flip side, MRI’s do not use radiation and provide clear images of soft tissue making them more preferred for internal tissues medical examination [4]. However, they are time-consuming, and some patients may find it difficult to remain still for the entire process, especially if the patient uses metallic medical devices like a pacemaker.

Deep transfer learning techniques have been widely employed to: (i) address medical image analysis issues, particularly in detecting and diagnosing diseases that affect the heart, kidney, breast, lungs, brain, and other organs [3]. As such, this naturally has led to more and more researchers in recent times to continue seeking opportunities for (ii) optimization of the classifiers to achieve better performance. Deep transfer learning has been applied in medical imaging classification and segmentation tasks [32].

Medical images differ significantly from natural images in datasets such as ImageNet [43]. Medical images in a specific field often have standardized views, with relevant task features typically having limited texture variations or small patches rather than high-level semantic features [43]. High-resolution is typically important, and images are often grayscale, such as X-ray images. Thus, though transfer learning seems a better option in analyzing and classifying such images, models have to be optimized to ensure that they best fit the scenario at hand [44].

II. Deep Transfer learning classifiers

Deep transfer learning has evolved into a powerful tool for developing classifiers in a variety of machine learning applications. This section looks at the following common [33, 34] deep transfer learning classifiers (i) ResNet, (ii) VGGNet, (iii) AlexNet, and (iv) Inception. These classifiers utilize a pre-trained deep neural network as a foundation for a new task [34], allowing them to benefit from the pre-trained network's feature extraction abilities. This essentially results in a classifier that requires fewer training samples and exhibits faster convergence [5]. The discussion below on these classifiers brings a sense of the current state-of-the-art approaches in the field of deep transfer learning in image classification, setting up a framework for later discussion of deep transfer learning optimization techniques.

These deep transfer learning classifiers have been shown to achieve high accuracy in a variety of medical imaging applications, including lung nodule detection [35,36], breast cancer detection [37], and brain tumour segmentation [38]. However, the effectiveness of these classifiers can be affected by factors such as the size and quality of the medical image dataset, the specific architecture used, and the optimization techniques employed during training [1].

A. ResNet

Residual Network (Resnet),originally introduced by Kaiming et al., in 2015 [13] are special type of neural network that has won numerous machine learning competitions. Since then, ResNet has had many variants, namely implemented as V1 or V2 with 50, 101, or 152 layers. To tackle complex problems, researchers often incorporate additional layers into deep neural networks to enhance accuracy and performance. The rationale behind adding more layers is that these layers gradually learn more intricate features. For instance, in image recognition, the first layer may learn to recognize edges, the second layer may identify textures, and the third layer may detect objects, and so on. However, it has been observed that the traditional Convolutional Neural Network model has a maximum depth limit. This means that adding more layers to a network may diminish its performance. This issue may be due to the optimization function, network initialization, and, notably, the vanishing gradient problem.

Sarwinda et al., [39] used ResNet for image classification in order to detect colorectal cancer. They trained ResNet-18 and ResNet-50 models on colon glands images to distinguish between benign and malignant cancer. The prototypes were assessed on three different testing data sets. The performance of the models was evaluated based on accuracy, sensitivity, and specificity values. The results showed that ResNet-50 provided the most reliable performance compared to ResNet-18 in all three testing data sets. The authors also maintained skip connections inherent in ResNet as a method to optimize this model.

Showcat et al., [40] used a transfer learning approach on ResNet to classify pneumonia cases in CXR images by freezing first few layers of the original ResNet. The model achieved an accuracy of 95% on a GPU accelerated machine. The authors also used batch normalization in robust training, which [41] concludes that it is a prerequisite for achieving convergence and whose overall effect is faster training [42].

B. VGGNet

The Visual Geometry Group Network (VGGNet) is a CNN architecture that was introduced by Karen et al. [2] at the University of Oxford in 2014. Its primary purpose was to explore the impact of CNN depth on the accuracy of deep learning models. The VGGNet is renowned for its simplicity, featuring a consistent design consisting of recurring blocks of convolutional layers with pooling layers, as well as several fully connected layers. This simplicity has made it a popular choice for various transfer learning applications, including object detection, image classification, and semantic segmentation. The VGG architecture is available in two main versions: VGG16 and VGG19. The VGG16 architecture includes 13 convolutional layers and 3 fully connected layers, while the VGG19 architecture includes 19 weight layers with 16 convolutional layers and 3 fully connected layers. Both versions of the VGGNet contain two fully connected layers with 4096 channels each, followed by another fully connected layer with 1000 channels to predict 1000 labels. Finally, the architecture's last fully connected layer employs the softmax layer for classification purposes.

C. AlexNet

In 2012, the architecture of AlexNet, a neural network, won the ImageNet Large Scale Visual Recognition Challenge. Developed by researchers at the University of Toronto, AlexNet was the first deep neural network to achieve significant improvement in image classification tasks when compared to traditional machine learning techniques. It has 8 CNN layers with an image input size of 227-by-227 and ability to classify images into 1000 objective categories. AlexNet is characterized by its use of convolutional layers, pooling layers, and dropout regularization to prevent overfitting, making it popular for transfer learning applications such as medical image classification and object detection in various studies.

D. Inception

Prior to the discovery of Inception and as observed above, many neural networks only exploited the technique of stacking convolutional layers deeper and deeper in the hope of improving the performance of the network. The Inception neural network architectures were created by Google researchers and consist of several versions, including Inception-v1 or GoogleNet, as well as Inception-v2, Inception-v3, and Inception-v4. The original variant, Inception-v1, is considered a deep network with a total of 22 layers, including the pooling layers. It uses global average pooling at the end of the network. The evolutions relied on the original GoogLeNet architecture and incorporated additional features such as batch normalization, factorized convolution, and residual connections to create versions 2,3 and 4. To this day, Inception-v3 and Inception-v4 are considered as some of the most efficient neural network architectures in terms of performance on the ImageNet dataset, and they have become the go-to models for many transfer learning tasks.

III. Deep transfer learning optimization techniques

Optimization is one the most important phenomena in machine learning with a goal state of building models that perform better than those that exist. Medical Image Classification is no exception as deep transfer learning classifiers require careful optimization to achieve optimal performance [16]. During optimization, a model is trained iteratively and the results compared in every iteration through Maxima and Minima functions. This is achieved by changing the hyperparameters in each step until the optimum results are achieved [7]. Optimization techniques such as data augmentation and transfer learning with fine-tuning can be useful in medical imaging classification tasks. Data augmentation can help to increase the size and diversity of the training dataset, which is particularly useful when the dataset is limited. Transfer learning with fine-tuning can be used to adapt pre-trained models to specific medical imaging tasks, which can improve the performance of the model.

Batch normalization has been shown to improve the convergence and stability of deep neural networks, which is particularly useful in medical imaging classification tasks where the dataset is often limited and the images may have varying brightness and contrast levels. For example, in a study by Wang et al., [47] on the classification of breast cancer histopathology images using transfer learning, batch normalization was used to improve the performance of the model [47].

Regularization techniques such as L1 and L2 regularization and dropout can also be useful in medical imaging classification tasks to prevent overfitting and improve generalization performance. In a study by Kim et al. on the classification of breast ultrasound images using transfer learning, dropout regularization was used to improve the performance of the model [48].

In this section, the study introduces the most popular and best-fit optimization techniques [46] for the task of medical image classification using deep learning [45]. As outlined below, these techniques aim to enhance the accuracy of neural networks while also facilitating faster and easier training [8].

A. Batch Normalization Techniques

Sergey Ioffe and Christian Szegedy [14] discovered Batch normalization as a technique to solve the problem of internal covariate shift. In pursuit of tuning and optimization opportunities in a neural network, researchers go deeper and deeper into the structure of the network which causes internal covariate shift. This occurs when there's a change in the distribution of the input to a layer that takes place while the network is being trained. This leads to two main issues. Firstly, the upper layers of the network have to frequently adjust to keep up with the variations in the input network, which causes the activation function to enter the gradient saturation zone, hindering the speed of network convergence.

Let us assume that we have a batch of input data for a specific layer in a neural network

$$X = [X_1, X_2, \dots, X_n] \tag{1}$$

From the above x_i means a sample and n means batch size. First, we can compute the average value of the elements in the mini-batch using the following formula,

$$\varphi_B = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{2}$$

Then, we determine the variance of the mini-batch as shown below,

$$\omega_B^2 = \frac{1}{n}\sum_{i=1}^{n} (x_i - \varphi_B)^2 \tag{3}$$

From the mini batch above, we can then perform normalization on each element,

$$x_i' = \frac{x_i - \varphi_B}{\sqrt{\omega_B^2 + \varepsilon}} \tag{4}$$

Finally, in order to account for the non-linear properties of the network, we can apply a scaling and shifting operation to the original output,

$$y_i = \alpha_i \cdot x_i' + \beta_i \tag{5}$$

The normalization of the input to each layer of the network is an essential part of the batch normalization technique. This normalization is carried out across a smaller batch of examples, which assists in lowering the impact of the data's noise as shown above. After normalization, the output is scaled and shifted by learned parameters, which allows the network to learn non-linear transformations of the input.

The merits of batch normalization lie in the following benefits:

1) Fast Network Convergence: Batch normalization can speed up the convergence of the training process, especially for networks that exploit transfer learning. This is because it reduces the internal covariate shift, which can cause the gradients to vanish or explode.

2) Introduction of Normalization Range: Batch normalization can act as a regularization technique, which helps to reduce overfitting. This is because it adds noise to the input, which makes the network more robust to variations in the input.

3) Improved Generalization: Batch normalization helps to enhance the network's generalization performance by decreasing its sensitivity to weight initialization, which in turn makes the network more adaptable to different parameters. Additionally, it increases the stability of the learning process in the network.

Xu et al., [52] used a U-Net network to perform image segmentation on MRI scan images, with the goal of improving physiological evaluation of the heart. The authors note that while deep learning-based models have improved segmentation accuracy compared to traditional methods, they still face challenges in fully differentiating the left and right ventricles from the myocardium, and training can be complex. To address these challenges, the authors adopted an improved U-Net network in a fully convolutional neural network to perform cardiac segmentation.

In addition, the authors used batch normalization (BN) and different loss functions to enhance the performance of the neural network. Batch normalization can help to reduce internal covariate shift and improve the stability and generalization of the network [53]. The authors also used a combination weighted loss function, which assigns different weights to different regions of the heart based on their importance in the segmentation task.

B. *Regularization Techniques*

In addition to drop out techniques, deep transfer learning classifiers can be optimized using regularization techniques. Regularization is a technique that is frequently utilized in deep learning models to avoid overfitting. When a model memorizes the training data instead of discovering the fundamental patterns in the data, overfitting occurs. Regularization adds a penalty to the loss function with an aim of reducing the complexity of the model [2]. Because of this penalty, the model is prevented from learning complex functions that provide an overly precise fit to the training data.

Deep learning makes use of the following regularization methods, L1 Regularization and L2 Regularization. Other methods to overcome overfitting include data augmentation, early stopping, and others.

1) L1 Regularization: In L1 regularization, a penalty term is added to the loss function. The L1 regularization term is directly proportional to the absolute value of the weights, and it is added to the loss function during training. This encourages the model to learn sparse weights, which can improve the model's interpretability.

$$MSE = \frac{1}{n}\sum_{i=1}^{n} \left(y_{\text{predicted}} - y_{\text{original}}\right)^2 + \lambda \sum_{i=1}^{n} |m_i| \quad (6)$$

The first component is the usual Mean Squared Error (MSE) formula while the second component is the Lasso regression normally referred to as L1 regularization. To calculate the regularization term, the absolute values of the slopes are summed up and multiplied by a constant lambda. Increasing lambda leads to a higher regularization term, which in turn increases the mean squared error. As a result, the slopes become smaller as the error term becomes larger. This approach encourages sparsity in the model, which can enhance interpretability.

2) L2 Regularization: A penalty term that is proportional to the square of the weights is added to the loss function as part of the L2 regularization procedure. This encourages the model to learn small weights, which can prevent overfitting.

$$MSE = \frac{1}{n}\sum_{i=1}^{n} \left(y_{\text{predicted}} - y_{\text{original}}\right)^2 + \lambda \sum_{i=1}^{n} m_i^2 \quad (7)$$

We begin by adding the square of each slope, then multiply that total by the lambda. In the same way as with L1 regularization, selecting a higher lambda value will result in a higher MSE, which in turn will cause slopes to flatten. Additionally, if the values of the slopes are higher, then the MSE will be higher which means a higher penalty will be applied. However, because it takes the square of the slopes, the slope values can never be zero. As a result, the model will suffer no loss in the algorithmic contribution of the features.

The contributions of regularization lie in the following benefits:

a) Improved Model Generalization: Regularization can improve the generalization performance of the model by reducing overfitting

b) Improved Model Inter-predictability: Regularization can improve the interpretability of the model by encouraging sparse or small weight.

c) Model Robustness: Regularization can make the model more robust to variations in the input by encouraging the network to learn redundant representations of the input.

Regularization is an important technique in medical image analysis using deep learning because it helps to prevent overfitting, which is a common problem when working with large and complex datasets [54]. Overfitting occurs when a model is too complex and fits the training data too closely, resulting in poor generalization performance when presented with new, unseen data. Regularization methods impose constraints on the model to prevent it from becoming too complex, which can improve its ability to generalize to new data [54].

Limited datasets in medical imaging often leads to lack of the ability for a model to generalize well to new and unseen data [54]. To address this challenge, the proposed approach in [54] leverages variational encoding to learn a compact and representative feature space that captures the shared information among medical data from different domains. Additionally, a novel linear-dependency regularization term is introduced to ensure that the learned feature space is more discriminative and generalizable. The experiments conducted on two challenging medical imaging classification tasks demonstrate that the proposed approach outperforms state-of-the-art baselines in terms of cross-domain generalization capability. This suggests that the learned feature space is more effective in capturing the underlying structure and variability of medical images [56], allowing the model to generalize better to new and unseen data.

C. *Dropout Techniques*

Dropout is a method used in transfer learning to address the issue of overfitting. When a model is too complex, it may fit too closely to the training data, leading to poor performance when applied to new, unseen data. Dropout helps to mitigate this problem.

This technique works by randomly dropping out (set to zero) a fraction of the neurons in a layer during each training iteration. This compels the remaining neurons to learn how to represent the input data without relying on the dropped-out neurons. Dropout can be seen as an ensemble of miniature neural networks, which are trained on distinct subsets of the original training data. By combining the outcomes of these smaller networks, a final prediction can be made.

Hence, the primary advantage of using dropout is to reduce the correlation among nodes in the hidden layers, penalize influential neurons, and decrease the reliance of the network on those influential neurons that have been penalized. This helps to prevent overfitting by reducing the co-adaptation of neurons and increasing the generalization performance of the network [2]. However, this has a drawback on increased training time since each training iteration requires dropping out neurons and scaling the output of the remaining neurons.

D. *Weight Initialization Techniques*

Weight initialization is a technique employed to establish the initial values for the weights employed in a neural network. In deep transfer learning, the starting weights can greatly impact the network's performance. Notably, Inadequate weight initialization can cause the network to become deeply entrenched in the local optima, which ultimately results into poor performance [8].

There are several weight initialization techniques that are commonly used in deep learning namely: Orthogonal, Positive Unitball Initialization, Lecun Initialization, Truncated Normal, Random Uniform, Zeros & Constant, Random Normal, Identity, Xavier Initialization and He Initialization.

1) Orthogonal: Gradient propagation in deep nonlinear networks benefits from initializing weights with orthogonal matrices. Due to the fact that orthogonal matrices preserve the norm, the input norm is constant over the whole network. Consequently, it assists in addressing the problem of either exploding or vanishing gradients [16]. Another feature of an orthogonal matrix that facilitates the learning of different input information by the weights is that its columns are mutually perpendicular.

2) Random Normal: Random initialization sets the weights of a neural network to arbitrary values. These weights are then assigned values that are randomly chosen through the process of random initialization. However, there are two potential issues that may arise when weights are initialized with random values: vanishing gradients and exploding gradients. If weights are initially set to a small random value, the model may work effectively for a while, but as time passes, the gradient approaches zero during propagation, which may lead to slow learning and vanishing gradients. On the other hand, if weights are initially set to a large random number, this can cause a problem referred to as the exploding gradient during training.

3) Truncated Normal: A significant overlap exists between random normal initialization and truncated normal initialization. The primary difference is that any values that are more than two standard deviations away from the mean are removed and redistributed among other categories. By using the truncated normal distribution, the neurons' saturation can be avoided, which is a common issue.

With a truncated normal distribution, the weights are drawn from a normal distribution with a fixed mean, μ and variance, σ^2.They lie within the interval(a, b), such that

$$a = \mu - 2\sigma$$
$$b = u + 2\sigma \qquad (8)$$

The probability density function f can be expressed as

$$f(w;\mu,\sigma,a,b) = \begin{cases} \frac{1}{\sigma}\frac{\phi\left(\frac{w-\mu}{\sigma}\right)}{\Phi\left(\frac{b-\mu}{\sigma}\right)-\Phi\left(\frac{a-\mu}{\sigma}\right)}, & \text{for } a \le w \le b \\ 0, & \text{for } w < a \text{ or } w > b \end{cases} \qquad (9)$$

Here:

$$\phi(\xi) = \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}\xi^2\right) \qquad (10)$$

Is the description of the probability density function of the standard normal distribution and below is the cumulative distribution function

$$\Phi(x) = \frac{1}{2}\left[1 + \text{erf}\left(\frac{x}{\sqrt{2}}\right)\right] \qquad (11)$$

Error function, donated by erf(x), is defined by

$$\text{erf}(x) = \frac{1}{\pi}\int_{-x}^{x} e^{-t^2}\,dt \qquad (12)$$

For instance, when a sigmoid activation function is employed, the input values that are too small or too large may result in the activation values that are too small or too large, which can lead to saturation of the neuron. When neurons reach the saturation zone, they cease to function and do not update themselves.

Random Uniform: With random uniform initialization, weight values are drawn at random from a uniform distribution within a given interval. All numbers within the range have an equal probability of being chosen. The probability density function at the two boundaries a and b is given by,

$$f(w) = \begin{cases} \frac{w-a}{b-a} & \text{for } a \le w \le b \\ 0 & \text{for } w < a \text{ or } w > b \end{cases} \qquad (13)$$

The cumulative distribution function is given by,

$$F(w) = \begin{cases} 0, \text{ for } w < a \\ \frac{w-a}{b-a}, \text{ for } a \le w \le b \\ 1, \text{ for } w > b \end{cases} \qquad (14)$$

4) Zeros & Constant: The simplest initialization method is Zeros and Constants, where all weight parameters are initialized to either 0 or a constant value. However, this results in all neurons in the network learning the same features. This is because, no matter how many iterations of feed-forward propagation and backpropagation are conducted, the weight values between any two connected hidden layers remain identical and symmetrical. This is not a good initialization technique because it can cause the neurons to learn identical representations of the input, which can slow down the training process.

5) Lecun Initialization: To allow the network to learn the linear portion of the mapping and prevent disappearing or exploding back-propagated gradients, it is crucial to ensure that the weights fall within the linear region of the sigmoid. This is done to avoid hindering learning and impeding the network's progress. Lecun et al. achieved this by standardizing the training set and ensuring that each layer had a constant activation variance of one [15].

The weights are initialized by randomly selecting values from a distribution with a mean of zero and a standard deviation is calculated. This was accomplished by Lecun et al. by standardizing the training set and mandating that each layer have a constant of $\sigma = 1$. The weights are then set to values that are randomly chosen from a distribution with a mean of zero and a standard deviation as follows, s, with n_i the size of layer i,

$$\text{stddev} = \sqrt{\frac{1}{n_i}} \qquad (15)$$

Weights in a Lecun initialization that uses a uniform distribution are drawn as follows, with n_i the size of layer i,

$$W \sim U\left[-\frac{\sqrt{3}}{\sqrt{n_i}}, \frac{\sqrt{3}}{\sqrt{n_i}}\right] \qquad (16)$$

6) Identity: To initialize the weight values, identity matrices are used, which are square tensors with 0's everywhere except for 1's along the diagonal. The identity matrix can be scaled by a multiplicative factor. This method is only used for producing two-dimensional square tensors.

By adding ones to the diagonal, identity weight tensors break the symmetry of the weight vector, which can lead to better performance compared to zero and constant initialization. However, when each layer is activated by a linear function, the activation values will exponentially decrease or increase with the number of layers in the network, resulting in vanishing or exploding gradients [16]. This happens as a result of the activation values being inversely proportional to the network's layer count.

7) Xavier Initialisation: Xavier Normal Initialization aims to ensure that information flows effectively during forward-propagation by keeping the deviations of the output of every two connected layers consistent. This method was developed based on several assumptions, including using a symmetric activation function with a unit derivation of 0, independently initializing weights, maintaining the same input feature variances, and being in a linear regime during initialization. The final initialization distribution of Xavier can be obtained as follows, with n_i the size of layer i,

$$W \sim U\left[-\frac{\sqrt{6}}{\sqrt{n_i+n_{i+1}}}, \frac{\sqrt{6}}{\sqrt{n_i+n_{i+1}}}\right] \qquad (17)$$

Xavier initialization following normal distribution borrows heavily from truncated normal distribution discussed above centered on 0 with standard deviation $= sqrt(\frac{2}{n_i+n_{i+1}})$ where n_i is the total number of input units in the weight and n_{i+1} is the number of output units in the weight.

$$\text{stddev} = \sqrt{\frac{2}{n_i+n_{i+1}}} \qquad (18)$$

8) He Initialization: Xavier initialization assumes that a linear activation function is used in the model, which is not accurate for ReLU activation function [16]. When ReLU is used as the activation function, networks initialized with simple normal distribution or Xavier initialization struggle to converge as the depth of the network increases. To address this issue, He and his team developed a new initialization method that works well with ReLU activation function [9]. When compared to a model that uses Xavier initialization, a model that uses He initialization increases the rate of convergence; however, there is no obvious distinction between the two models in terms of accuracy. In this approach, the values of the weights are distributed according to a zero-mean normal distribution, and the standard deviation is calculated as follows with n_i the size of layer i;

$$\text{stddev} = \sqrt{\frac{2}{n_i}} \qquad (19)$$

Furthermore, He initialization also uses uniform distribution to acquire weight values, with the size of the input layer being the only factor to be taken into account.

$$W \sim U\left[-\frac{\sqrt{6}}{\sqrt{n_i}}, \frac{\sqrt{6}}{\sqrt{n_i}}\right] \qquad (20)$$

Where n_i is the size of layer i

The table below shows a summary of the weight initialization techniques.

E. Data Augmentation Techniques

Data augmentation is a useful technique to overcome model overfitting, as it helps to expand and diversify the training dataset without collecting additional data [50]. Data augmentation techniques have become increasingly important in computer-aided medical classification, as they help to address the limitations of small and imbalanced datasets commonly found in the medical domain.

In medical image analysis, data augmentation is widely used to enhance the quality and diversity of images, including X-rays, MRIs, CT scans, and ultrasound images[43]. Techniques such as rotation, flipping, scaling, cropping, and elastic transformations are applied to create new variations of the existing images, effectively increasing the size of the dataset. These augmented images help the model to learn more robust and invariant features, making it less sensitive to slight changes in image orientation, position, or scale that can occur in real-world clinical settings. Moreover, data augmentation can also be beneficial in addressing class imbalance issues, as generating more samples of underrepresented classes can help to mitigate the bias towards majority classes[52].

Memetic algorithms and Generative Adversarial Networks (GANs) are two distinct approaches to optimization and learning within the field of artificial intelligence. Memetic algorithms are a class of optimization algorithms inspired by both genetic algorithms and the concept of memes (cultural units of information). These algorithms combine the global search capabilities of genetic algorithms with local search techniques to explore the solution space more effectively. Memetic algorithms typically consist of three main components namely Population-based search, Local Search and Evolutionary Operators.

They are particularly useful for solving complex optimization problems, such as combinatorial or multi-objective optimization tasks, where traditional search methods might struggle to find high-quality solutions efficiently [56].

TABLE I. WEIGHT INITIALIZATION TECHNIQUES

Technique	***Normal Distribution***	***Uniform Distribution***	***Random Initialization***
Orthogonal	No	No	Yes
Positive Unitball Initialization	Possible	Possible	Yes
Lecun Initialization	Possible	Possible	Yes
Truncated Normal	Yes	No	Yes
Random Uniform	No	Yes	Yes
Zeros & Constant	No	No	No
Random Normal	Yes	No	Yes
Identity	No	No	No
Xavier Initialization	Possible	Possible	Yes
He Initialization	Possible	Possible	Yes

The suitability of each technique can depend on the architecture of the network and the characteristics of the data, and experimentation is often needed to determine the best choice for a particular task [49]. Some weight initialization techniques may work better for deep neural networks with a large number of layers, while others may work better for networks with fewer layers [50]. Similarly, some techniques may work better for networks with specific activation functions or regularization techniques. The characteristics of the data can also impact the suitability of a weight initialization technique. For instance, if the data has a high degree of variability or complexity, a weight initialization technique that allows for greater variability in the initial weights may be more effective [50].

Different weight initialization techniques can impact the convergence speed and final accuracy of the model [49]. This means that experimentation is often needed to determine the best choice for a particular task, which involves testing the performance of the model with different weight initialization techniques and selecting the one that achieves the best results.

The following table shows various selected studies with an optimization technique used in each, and the performance achieved after its application.

TABLE II. PERFORMANCE ANALYSIS OF OPTIMIZATION TECHNIQUES

Study	*Purpose*	*Technique*	*Performance*
Xiao et al., [53]	Medical Image segmentation (CT and MRI images)	Batch Normalization	91.48%
Xu et al., [52]	Physiological Heart evaluation from medical images (MRI images)	Batch Normalization	Dice index of 96.5%
Selman et al., [54]	Domain Generalization for Medical Imaging Classification with Linear-Dependency Regularization	Linear dependency Regularization	69.78%
Wang et al., [55]	Deep virtual adversarial self-training with consistency regularization for semi-supervised medical image classification	Consistency regularization loss (L1 regularization)	89.3%
You et al., [57]	SimCVD: Simple Contrastive Voxel-Wise Representation Distillation for Semi-Supervised Medical Image Segmentation	Dropout	Dice score of 90.85%
Abbas et al., [58]	A hybrid transfer learning-based architecture for recognition of medical imaging modalities for healthcare experts	Dropout	99%
Sharmay et al., [59]	Understanding Transfer Learning for Histopathology	Weight initialization	90.3% AUC

IV. DISCUSSION

Medical image analysis tasks are still regarded as NP hard problems in computer science [51]. Even with the immense research that has gone into comouter aided image analysis, there are still some challenges and difficulties researchers face, despite the encouraging results that deep transfer learning has produced. One of the main obstacles to overcome in deep transfer learning is the domain shift problem. This issue arises due to the differences in distributions between the source domain and the target domain. Deep transfer learning assumes that the source and target domains have similar statistical characteristics, but this assumption may not hold true for medical imaging.

Again, there is lack of labeled data in medical imaging, which limits the ability of deep transfer learning to learn complex representations from limited data. The heterogeneity of medical images, which includes differences in imaging modalities, resolutions, and acquisition protocols, is another factor that can be problematic for deep transfer learning.

Several approaches have been suggested as possible solutions to these issues in recent times. The aim of domain adaptation methods is to align the source and target domain distributions. This is accomplished by learning a mapping function that can transform the data in the source domain into the format required by the target domain. The objective of semi-supervised learning techniques is to enhance the effectiveness of deep transfer learning by incorporating both labeled and unlabeled data in the learning process. On the other hand, meta-learning techniques aim to gain transferable knowledge that can be applied across various domains and tasks, thereby enhancing the generalization ability of deep transfer learning.

V. CONCLUSION

Deep transfer learning has demonstrated some exemplary results in medical image classification tasks, particularly in situations where there is a shortage of labeled data. This paper has presented a condensed review of the deep transfer learning optimization techniques that have been developed for medical image classification. The study went over the fundamental architecture of CNNs, along with some optimization strategies like batch normalization, regularization, weight initialization and data augmentation techniques. In addition, the study discussed some potential solutions to the challenges that arise when applying deep transfer learning to medical imaging tasks. To sum up, the study has shown that deep transfer learning will maintain its significance in medical image analysis. Nonetheless, further investigation is required to investigate how novel meta learning techniques like few shot learning, one shot learning and zero shot learning can be used to reduce the learning curve of deep transfer learning model.

ACKNOWLEDGMENT

Special thanks Karatina University and Dedan Kimathi University of Technology for the support accorded to the authors. We also extend our gratitude to the anonymous reviewers whose constructive criticism helped improve the quality of this paper.

REFERENCES

[1] P. Kora et al., “Transfer learning techniques for medical image analysis: A review,” Biocybernetics and Biomedical Engineering, vol. 42, no. 1, pp. 79–107, Jan. 2022, doi: 10.1016/j.bbe.2021.11.004.

[2] C. F. G. D. Santos and J. P. Papa, “Avoiding Overfitting: A Survey on Regularization Methods for Convolutional Neural Networks,” ACM Computing Surveys, vol. 54, no. 10s, pp. 1–25, Jan. 2022, doi: 10.1145/3510413.

[3] S. M. Abdullah et al., “Deep Transfer Learning Based Parkinson’s Disease Detection Using Optimized Feature Selection,” IEEE Access, vol. 11, pp. 3511–3524, 2023, doi: 10.1109/access.2023.3233969.

[4] A. S. Alatrany, W. Khan, A. J. Hussain, J. Mustafina, and D. Al-Jumeily, “Transfer Learning for Classification of Alzheimer’s Disease Based on Genome Wide Data,” IEEE/ACM Transactions on Computational Biology and Bioinformatics, pp. 1–12, 2023, doi: 10.1109/tcbb.2022.3233869.

[5] S. Sharma and K. Guleria, “Deep Learning Models for Image Classification: Comparison and Applications,” in 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Apr. 2022. Accessed: Mar. 18, 2023. [Online]. Available: http://dx.doi.org/10.1109/icacite53722.2022.9823516

[6] S. R. Zeebaree, O. Ahmed, and K. Obid, “CSAERNet: An Efficient Deep Learning Architecture for Image Classification,” in 2020 3rd International Conference on Engineering Technology and its Applications (IICETA), Sep. 2020. Accessed: Mar. 18, 2023. [Online]. Available: http://dx.doi.org/10.1109/iiceta50496.2020.9318859

[7] Y. Arun and G. S. Viknesh, “Leaf Classification for Plant Recognition Using EfficientNet Architecture,” in 2022 IEEE Fourth International Conference on Advances in Electronics, Computers and Communications (ICAECC), Jan. 2022. Accessed: Mar. 18, 2023. [Online]. Available: http://dx.doi.org/10.1109/icaecc54045.2022.9716637

[8] A. Anaya-Isaza and L. Mera-Jimenez, “Data Augmentation and Transfer Learning for Brain Tumor Detection in Magnetic Resonance Imaging,” IEEE Access, vol. 10, pp. 23217–23233, 2022, doi: 10.1109/access.2022.3154061.

[9] J. Liu, M. Xing, H. Yu, and G. Sun, “EFTL: Complex Convolutional Networks With Electromagnetic Feature Transfer Learning for SAR Target Recognition,” IEEE Transactions on Geoscience and Remote Sensing, vol. 60, pp. 1–11, 2022, doi: 10.1109/tgrs.2021.3083261.

[10] L. Datta, “A Survey on Activation Functions and their relation with Xavier and He Normal Initialization,” arXiv.org, Mar. 18, 2020. https://arxiv.org/abs/2004.06632

[11] M. V. Narkhede, P. P. Bartakke, and M. S. Sutaone, “A review on weight initialization strategies for neural networks,” Artificial Intelligence Review, vol. 55, no. 1, pp. 291–322, Jun. 2021, doi: 10.1007/s10462-021-10033-z.

[12] S. K. Kumar, “On weight initialization in deep neural networks,” arXiv.org, Apr. 28, 2017. https://arxiv.org/abs/1704.08863

[13] F. He, T. Liu, and D. Tao, “Why ResNet Works? Residuals Generalize,” IEEE Transactions on Neural Networks and Learning Systems, vol. 31, no. 12, pp. 5349–5362, Dec. 2020, doi: 10.1109/tnnls.2020.2966319.

[14] S. Ioffe and C. Szegedy, “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift,” arXiv.org, Feb. 11, 2015. https://arxiv.org/abs/1502.03167

[15] A. Kamath, M. Singh, Y. LeCun, G. Synnaeve, I. Misra, and N. Carion, “MDETR - Modulated Detection for End-to-End Multi-Modal Understanding,” in 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Oct. 2021. Accessed: Mar. 18, 2023. [Online]. Available: http://dx.doi.org/10.1109/iccv48922.2021.00180

[16] Wang, Zhu, Wang, and Zhang, “A Review of Deep Learning on Medical Image Analysis,” Mobile Networks and Applications, vol. 26, no. 1, pp. 351–380, Nov. 2020, doi: 10.1007/s11036-020-01672-7.

[17] Kim, Cosa-Linan, Santhanam, Jannesari, Maros, and Ganslandt, “Transfer learning for medical image classification: A literature review,” BMC Medical Imaging, vol. 22, no. 1, pp. 1–13, Apr. 2022, doi: 10.1186/s12880-022-00793-7.

[18] Z. Lai and H. Deng, “Medical Image Classification Based on Deep Features Extracted by Deep Model and Statistic Feature Fusion with Multilayer Perceptron,” *Computational Intelligence and Neuroscience*, vol. 2018, pp. 1–13, Sep. 2018, doi: 10.1155/2018/2061516.

[19] D. S. Kermany *et al.*, “Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning,” *Cell*, vol. 172, no. 5, pp. 1122-1131.e9, Feb. 2018, doi: 10.1016/j.cell.2018.02.010.

[20] “Medical imaging in personalised medicine: a white paper of the research committee of the European Society of Radiology (ESR),” *Insights into Imaging*, vol. 6, no. 2, pp. 141–155, Mar. 2015, doi: 10.1007/s13244-015-0394-0.

[21] K. B. Johnson *et al.*, “Precision Medicine, AI, and the Future of Personalized Health Care,” *Clinical and Translational Science*, vol. 14, no. 1, pp. 86–93, Oct. 2020, doi: 10.1111/cts.12884.

[22] L. Cai, J. Gao, and D. Zhao, “A review of the application of deep learning in medical image classification and segmentation,” *Annals of Translational Medicine*, vol. 8, no. 11, pp. 713–713, Jun. 2020, doi: 10.21037/atm.2020.02.44.

[23] D. Ardila *et al.*, “End-to-end lung cancer screening with three-dimensional deep learning on low-dose chest computed tomography,” *Nature Medicine*, vol. 25, no. 6, pp. 954–961, May 2019, doi: 10.1038/s41591-019-0447-x.

[24] J. Zhou and O. G. Troyanskaya, “Predicting effects of noncoding variants with deep learning–based sequence model,” *Nature methods*, vol. 12, no. 10, Oct. 2015, doi: 10.1038/nmeth.3547.

[25] Yadav and Jadhav, “Deep convolutional neural network based medical image classification for disease diagnosis,” *Journal of Big Data*, vol. 6, no. 1, pp. 1–18, Dec. 2019, doi: 10.1186/s40537-019-0276-2.

[26] Z. Li and Y. Wu, “The Effectiveness of Image Augmentation in Breast Cancer Type Classification Using Deep Learning,” in *2021 3rd International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI)*, Dec. 2021. Accessed: Apr. 05, 2023. [Online]. Available: http://dx.doi.org/10.1109/mlbdbi54094.2021.00134

[27] P. R. A. S. Bassi and R. Attux, “A deep convolutional neural network for COVID-19 detection using chest X-rays,” *Research on Biomedical Engineering*, vol. 38, no. 1, pp. 139–148, Apr. 2021, doi: 10.1007/s42600-021-00132-9.

[28] Sarvamangala and Kulkarni, “Convolutional neural networks in medical image understanding: a survey,” *Evolutionary Intelligence*, vol. 15, no. 1, pp. 1–22, Jan. 2021, doi: 10.1007/s12065-020-00540-3.

[29] “Convolutional neural networks for medical image analysis: State-of-the-art, comparisons, improvement and perspectives,” *Neurocomputing*, vol. 444, pp. 92–110, doi: 10.1016/j.neucom.2020.04.157.

[30] S. P. Power, F. Moloney, M. Twomey, K. James, O. J. O’Connor, and M. M. Maher, “Computed tomography and patient risk: Facts, perceptions and uncertainties,” *World Journal of Radiology*, vol. 8, no. 12, Dec. 2016, doi: 10.4329/wjr.v8.i12.902.

[31] M. Iman, H. R. Arabnia, and K. Rasheed, “A Review of Deep Transfer Learning and Recent Advancements,” *Technologies*, vol. 11, no. 2, p. 40, Mar. 2023, doi: 10.3390/technologies11020040.

[32] “Deep Transfer Learning Based Classification Model for COVID-19 Disease,” *IRBM*, vol. 43, no. 2, pp. 87–92, doi: 10.1016/j.irbm.2020.05.003.

[33] Z. Huang, C. O. Dumitru, Z. Pan, B. Lei, and M. Datcu, “Classification of Large-Scale High-Resolution SAR Images With Deep Transfer Learning,” *IEEE Geoscience and Remote Sensing Letters*, vol. 18, no. 1, pp. 107–111, Jan. 2021, doi: 10.1109/lgrs.2020.2965558.

[34] F. Zhuang *et al.*, “A Comprehensive Survey on Transfer Learning,” *Proceedings of the IEEE*, vol. 109, no. 1, pp. 43–76, Jan. 2021, doi: 10.1109/jproc.2020.3004555.

[35] J. Gao, Q. Jiang, B. Zhou, and D. Chen, “Lung Nodule Detection using Convolutional Neural Networks with Transfer Learning on CT Images,” *Combinatorial Chemistry &amp; High Throughput Screening*, vol. 24, no. 6, pp. 814–824, Jul. 2021, doi: 10.2174/1386207323666200714002459.

[36] I. D. Apostolopoulos *et al.*, “Automatic classification of solitary pulmonary nodules in PET/CT imaging employing transfer learning techniques,” *Medical &amp; Biological Engineering &amp; Computing*, vol. 59, no. 6, pp. 1299–1310, May 2021, doi: 10.1007/s11517-021-02378-y.

[37] S. Arooj *et al.*, “Breast Cancer Detection and Classification Empowered With Transfer Learning,” *Frontiers in Public Health*, vol. 10, Jul. 2022, doi: 10.3389/fpubh.2022.924432.

[38] M. Ghaffari *et al.*, “Automated post-operative brain tumour segmentation: A deep learning model based on transfer learning from

pre-operative images," *Magnetic Resonance Imaging*, vol. 86, pp. 28–36, Feb. 2022, doi: 10.1016/j.mri.2021.10.012.

[39] D. Sarwinda, R. H. Paradisa, A. Bustamam, and P. Anggia, "Deep Learning in Image Classification using Residual Network (ResNet) Variants for Detection of Colorectal Cancer," *Procedia Computer Science*, vol. 179, pp. 423–431, 2021, doi: 10.1016/j.procs.2021.01.025.

[40] S. Showkat and S. Qureshi, "Efficacy of Transfer Learning-based ResNet models in Chest X-ray image classification for detecting COVID-19 Pneumonia," *Chemometrics and Intelligent Laboratory Systems*, vol. 224, p. 104534, May 2022, doi: 10.1016/j.chemolab.2022.104534.

[41] M. Amthor, E. Rodner, and J. Denzler, "Impatient DNNs - Deep Neural Networks with Dynamic Time Budgets," in *Procedings of the British Machine Vision Conference 2016*, 2016. Accessed: Apr. 06, 2023. [Online]. Available: http://dx.doi.org/10.5244/c.30.116

[42] S. Santurkar, D. Tsipras, A. Ilyas, and A. Madry, "How Does Batch Normalization Help Optimization?," *Advances in Neural Information Processing Systems*, vol. 31.

[43] L. Alzubaidi *et al.*, "Novel Transfer Learning Approach for Medical Imaging with Limited Labeled Data," *Cancers*, vol. 13, no. 7, p. 1590, Mar. 2021, doi: 10.3390/cancers13071590.

[44] L. Alzubaidi *et al.*, "Towards a Better Understanding of Transfer Learning for Medical Imaging: A Case Study," *Applied Sciences*, vol. 10, no. 13, p. 4523, Jun. 2020, doi: 10.3390/app10134523.

[45] Y. Chen, X. Qin, J. Wang, C. Yu, and W. Gao, "FedHealth: A Federated Transfer Learning Framework for Wearable Healthcare," *IEEE Intelligent Systems*, vol. 35, no. 4, pp. 83–93, Jul. 2020, doi: 10.1109/mis.2020.2988604.

[46] M. Awais, Md. T. Bin Iqbal, and S.-H. Bae, "Revisiting Internal Covariate Shift for Batch Normalization," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 11, pp. 5082–5092, Nov. 2021, doi: 10.1109/tnnls.2020.3026784.

[47] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers, "ChestX-Ray8: Hospital-Scale Chest X-Ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017. Accessed: Apr. 06, 2023. [Online]. Available: http://dx.doi.org/10.1109/cvpr.2017.369

[48] M. Byra *et al.*, "Breast mass classification in sonography with transfer learning using a deep convolutional neural network and color conversion," *Medical Physics*, vol. 46, no. 2, pp. 746–755, Jan. 2019, doi: 10.1002/mp.13361.

[49] K. He, X. Zhang, S. Ren, and J. Sun, "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification," *arXiv.org*, Feb. 06, 2015. https://arxiv.org/abs/1502.01852

[50] A. Ghatak, "Initialization of Network Parameters," in *Deep Learning with R*, Singapore: Springer Singapore, 2019, pp. 87–102. Accessed: Apr. 06, 2023. [Online]. Available: http://dx.doi.org/10.1007/978-981-13-5850-0_4

[51] L. Rundo, C. Militello, S. Vitabile, G. Russo, E. Sala, and M. C. Gilardi, "A Survey on Nature-Inspired Medical Image Analysis: A Step Further in Biomedical Data Integration," *Fundamenta Informaticae*, vol. 171, no. 1–4, pp. 345–365, Oct. 2019, doi: 10.3233/fi-2020-1887.

[52] W. Xu *et al.*, "Deep learning-based image segmentation model using an MRI-based convolutional neural network for physiological evaluation of the heart," *Frontiers in Physiology*, vol. 14, Mar. 2023, doi: 10.3389/fphys.2023.1148717.

[53] J. Xiao *et al.*, "CateNorm: Categorical Normalization for Robust Medical Image Segmentation," in *Domain Adaptation and Representation Transfer*, Cham: Springer Nature Switzerland, 2022, pp. 129–146. Accessed: Apr. 13, 2023. [Online]. Available: http://dx.doi.org/10.1007/978-3-031-16852-9_13

[54] S. Vesal, N. Ravikumar, and A. Maier, "Dilated Convolutions in Neural Networks for Left Atrial Segmentation in 3D Gadolinium Enhanced-MRI," in *Statistical Atlases and Computational Models of the Heart. Atrial Segmentation and LV Quantification Challenges*, Cham: Springer International Publishing, 2019, pp. 319–328. Accessed: Apr. 13, 2023. [Online]. Available: http://dx.doi.org/10.1007/978-3-030-12029-0_35

[55] [40]X. Wang, H. Chen, H. Xiang, H. Lin, X. Lin, and P.-A. Heng, "Deep virtual adversarial self-training with consistency regularization for semi-supervised medical image classification," *Medical Image Analysis*, vol. 70, p. 102010, May 2021, doi: 10.1016/j.media.2021.102010.

[56] Y. Jiang, X. Sui, Y. Ding, W. Xiao, Y. Zheng, and Y. Zhang, "A semi-supervised learning approach with consistency regularization for tumor histopathological images analysis," *Frontiers in Oncology*, vol. 12, Jan. 2023, doi: 10.3389/fonc.2022.1044026.

[57] C. You, Y. Zhou, R. Zhao, L. Staib, and J. S. Duncan, "SimCVD: Simple Contrastive Voxel-Wise Representation Distillation for Semi-Supervised Medical Image Segmentation," *IEEE Transactions on Medical Imaging*, vol. 41, no. 9, pp. 2228–2237, Sep. 2022, doi: 10.1109/tmi.2022.3161829.

[58] Q. Abbas, "A hybrid transfer learning-based architecture for recognition of medical imaging modalities for healthcare experts - IOS Press," *Journal of Intelligent & Fuzzy Systems*, vol. 43, no. 5, pp. 5471–5486, doi: 10.3233/JIFS-212171.

[59] Y. Sharmay, L. Ehsany, S. Syed, and D. E. Brown, "HistoTransfer: Understanding Transfer Learning for Histopathology," in *2021 IEEE EMBS International Conference on Biomedical and Health Informatics (BHI)*, Jul. 2021. Accessed: Apr. 13, 2023. [Online]. Available: http://dx.doi.org/10.1109/bhi50953.2021.9508542.