Genetically Optimized Heterogeneous Ensemble for Histological Image Classification

Eid Alkhaldi[1] **Ezatollah Salari**[2]
Eid.Alkhaldi@gmail.com Ezzatollah.Salari@utoledo.edu
The University of Toledo
Dept. Electrical Engineering and Computer Science
OH, Toledo, USA

Abstract— **Learning an accurate mapping between the complex feature representations of histology images and their labels can magnify the reliability of automated diagnostic systems. Hence, breast histology Image classification is a significant problem in medical image processing. CNNs showed superiority in extracting beneficial features and became the most widely used image classification approach. However, the current state-of-the-art CNN techniques, such as Transfer Learning, fail to generalize for small datasets with complex texture and high resolution due to insufficient domain-adaptation. Furthermore, most ensembles of transfer learning models need manual parameter selection of the ensemble policy based solely on the accuracy of each classifier, without a guarantee of heterogeneity. To solve these problems, we propose a fully automated multi-stage training of horizontally stacked ensemble of CNNs, which consists of two stages. First, we optimize the task-specific layers and the ensemble hyper-parameters on low-resolution images using the Genetic Algorithm during the meta-training stage. Secondly, we thoroughly train the ensemble using the optimal parameters on high-resolution images. Our model achieved 1% more accuracy than the ICIAR 2018 Challenge winning approach. The results of the proposed method have been compared to previously published methods and exceeded many of the state-of-art techniques by a substantial margin.**

Keywords*— *The University of Toledo

I. Introduction

Histological image classification is one of the most crucial problems of medical image processing, which demands extensive work and specialized expertise [1–3]. Hence, An automated intelligent approach is needed to compensate for the time delay of analysis caused by the insufficient number of pathologists [4]. Early diagnosis of breast malignancy through histology patches is of prominent clinical significance for detection, prognosis, therapy, and reducing healthcare costs [5, 6]. Breast histology image classification aims to identify abnormalities in the specimens' structures and specify their carcinogenic level [7]. Deep Learning (DL) showed remarkable competence in tackling image classification over the past years due to the accelerated developments in computational resources [3, 5, 6, 8].

Deep Learning techniques extract very complex features for images, which expands class separability considerably better than the conventional machine learning methods [9]. Various DL approaches have been applied to categorize histology images. CNNs are the most reliable DL models employed for Histology image labeling [10]. However, for CNNs to be accurate, they demand a high quantity of labeled images. The more parameters a CNN holds, the larger the dataset needs to be, and the more training time is required, which is a limitation acknowledged in the DL research as the curse of dimensionality [11]. Consequently, end-to-end training of CNNs suffered significantly, due to the scarcity of annotated hematoxylin and eosin (H&E) stained histology images.

The trade-off between the scale of the model and the computational demands is often one of the issues hindering the performance of the state-of-art techniques. The extent of the model depends on the depth of the network and the resolution of the input images. The size of the input images is particularly crucial to histological image classification as a result of the enigmatic characteristics of these high-resolution images. However, in many cases, computational capacity limitations constrain researchers to resize the patches to a shallower resolution, which negatively influences the accuracy of the model [3].

Several preprocessing methods were applied to improve the performance of CNNs such as image enhancement, thresholding, normalization, and spatial transformation [6, 9]. Although preprocessing methods achieved a slight improvement, they failed to address the curse of dimensionality bottleneck.

Transfer Learning is the process of reutilization of models' weights that were trained on benchmark datasets such as ImageNet [12, 13]. Transfer Learning proved to be a more reliable alternative to end-to-end training because it improves accuracy and robustness and reduces the training time significantly, particularly for small datasets [13]. Utilizing pre-trained models to extract features and fine-tuning the last densely connected layer remained the most two Transfer Learning approaches applied for image classification. Although using pre-trained weights as initial weights performed better than end-to-end training, they overlooked domain transferability and determining the task-specific layers of the pre-trained model in histology images datasets [14].

Another attempt to improve the accuracy of models for small histology image dataset was by employing ensembles. Ensembles are sets of CNNs whose predictions are joined in a particular fashion to produce a final prediction [15]. Bayesian, majority voting, average voting are the most popular DL ensemble methods[15, 16]. Although ensembles achieve higher accuracy than individual networks, they are computationally expensive due to the immense search space of hyper-parameters. Besides, the extracted features of each classifier need to be distinct for an ensemble to generalize well on new unseen data [15, 16]. Modern ensemble methods focus solely on the validation accuracy of the ensemble networks without approaching the heterogeneity of their features and variance of their errors, which frequently leads to over-fitting.

In this paper, we propose a method that leverages the Genetic Algorithm to ascertain the task-specific layers of pre-trained networks on low-resolution images for optimal domain adaptation. We use the confusion matrix of the models over the validation data to define the heterogeneity of CNNs. Subsequently, we thoroughly train the most accurate models and with the most diverse errors on high-resolution images by applying the learned parameters. Finally, we learn the ensemble rule that combines the horizontally stacked prediction vector.

The remaining sections of the manuscript are organized in the following way. Section II introduces the proposed method with a thorough explanation of the Genetic Algorithm and hyper-parameter optimization. Section III discusses in detail the dataset, training, experimental setup, results, and evaluation metrics employed. Section IV presents a conclusion and suggestions for prospective research.

II. Proposed Method

A. Overview

The proposed method consists of meta-training and full training stages. The meta-training phase, as shown in Figure 1, begins with statistical image preprocessing, augmentation, and low-resolution cropping. Several image spatial transformations, such as flipping, rotation, and shifting, lead to effective dataset enlargement, which contributes to lowering overfitting [17]. The augmented images are resized to a lower resolution to expedite the meta-training. The resultant crops are standardized by utilizing the ImageNet statistic to reduce the distribution disparity between the image sources of the two domains. Image whitening demonstrated efficacy in hastening convergence [18]. All of the preprocessing procedures described above occur in real-time right before the mini-batches are fed to the networks to be trained.

Gaussian Dropout is another technique that we implemented in our solution. Adding the gaussian noise to the hidden layer connecting the global averaging and the output layers is an added variation of stochastic regularization. The dropout layer restricts the classifier from learning the irrelevant particularities of an image [19]. Dropout involves blocking several units from firing during the feed-forward and the backpropagation steps of training. The portion of the dropout units is governed by the gaussian distribution, whose standard deviation is expressed in equation (1).

$$\sigma_(gaussian) = \sqrt{rate/(1 - rate)} \qquad (1)$$

Where ***rate*** denotes the user control parameter and $\sigma_(gaussian)$ refers the dropout gaussian standard deviation. Setting the constant rate to 0.2 throughout the trials yielded an excellent performance.

Incrementally raising the learning rate at every batch accelerates the fine-tuning process by estimating the best base learning rate, as explained in subsection II-B [20]. During fine-tuning, all layers are frozen except the batch normalization and the last fully connected layer. Training the batch normalization weights reduces the needed training epochs substantially due to its capacity to lower the internal covariant shift during mini-batch training[21]. The validation loss of the fine-tuned model is the baseline used to compare the achievement of the Genetic Algorithm (GA) best solutions.

In this study, we propose GA to obtain the fittest estimation of the domain-specific layer, as illustrated thoroughly in subsection II-C. The Cyclical learning rate and GA cooperatively evade the inconvenience of extravagantly experimenting with numerous freeze-layers and base learning rates. The intuition behind the proposed strategy relies on the fact that the domain-adaptation maximization is separable. The foreknowledge, as mentioned above, remarkably narrows the possible combinations of hyperparameters.

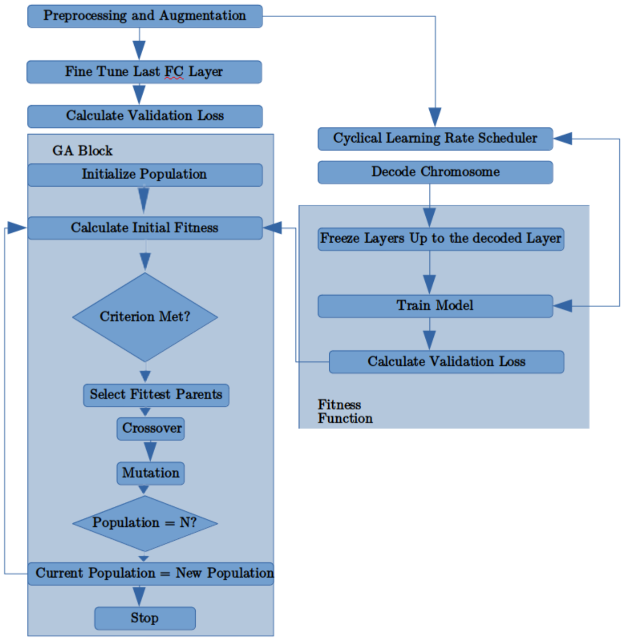

Preprocessing and Augmentation
Fine Tune Last FC Layer
Calculate Validation Loss
GA Block
Initialize Population
Calculate Initial Fitness
Criterion Met?
Select Fittest Parents
Crossover
Mutation
Population = N?
Current Population = New Population
Stop
Cyclical Learning Rate Scheduler
Decode Chromosome
Freeze Layers Up to the decoded Layer
Train Model
Calculate Validation Loss
Fitness Function

Fig. 1: Flowchart of the meta-training phase

B. Learning Rate Schedulers

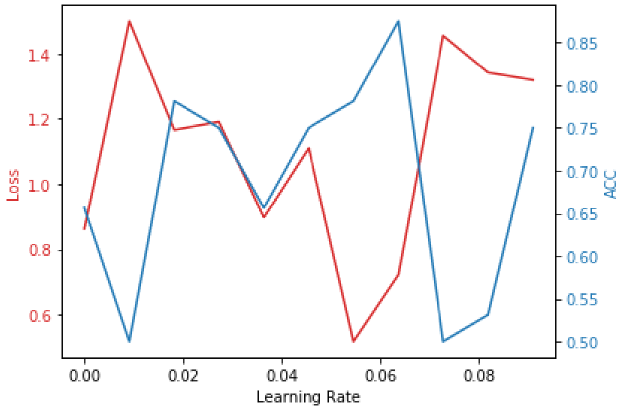


Fig. 2: High spikes of validiation loss and accuracy curves over exponential cyclical learning rate in a single epoch in ResNet50 with freeze layer = 141 and a batch size = 32. The learning rate with the lowest validation loss at lr = 0.05 is chosen as the base lr for the full training.

Learning rate scheduling methods aim to minimize the cost function. It is unlikely to get trapped at local minima at high dimensional cost functions. Nevertheless, saddle points are more likely to occur. Similar to local minima, saddle points slow down the training significantly, particularly if the LR is small[20]. Increasing the LR in this scenario could help the optimizer to escape the saddle point. However, over-increasing the LR will cause fluctuations with high spikes. Hence, a cyclical LR scheduler facilitates picking the optimal LR by defining the scaling function, the upper, and the lower bounds [20].

$$updatedLR = baseLR + (maxLR - baseLR) \times \frac{\arg\max(0, (1 - LR)) \times 1}{2^{LR-1}} \quad (2)$$

C. Genetic Algorithm and Ensemble Optimization

The objective of implementing the genetic algorithm is to evolve the solutions to the task-specific layer problem by simulating the natural selection though mating and mutation [22].

Each possible solution to the domain-specific layer is represented by a chromosome. Any layer in the search space of the network is encoded into fixed-length nine binary numbers representing 512 possible solutions. After calculating the initial fitnesses, the fittest layers are chosen to exchange genes at a predefined crossover probability P_c to produce better offsprings in the next generation.

We freeze all layers up to the decoded chromosome. Then, a cyclical learning scheduler learns the best base learning rate, as explained in the previous subsection. The model is then trained for more epochs, and the validation loss is used as the fitness function parameter to be minimized.

After the completion of the meta-training, models that achieved the most reliable in terms of validation accuracy and loss are analyzed based on heterogeneity, as displayed in Figure **??**. The confusion matrix facilitates a valuable representation of the kind of misclassifications that the model makes. The models that produce similar misclassifications are discarded.

The predictions of the top-performing heterogeneous networks are then concatenated horizontally to form a pre-classification layer. We further train the weights of this layer while freezing the whole model to make the ensemble training efficient and more accurate than simple ensembling policies such as the majority and average voting. Adding the horizontal layer notably improved the generalizability of the model, as demonstrated in Table IV and Table V.

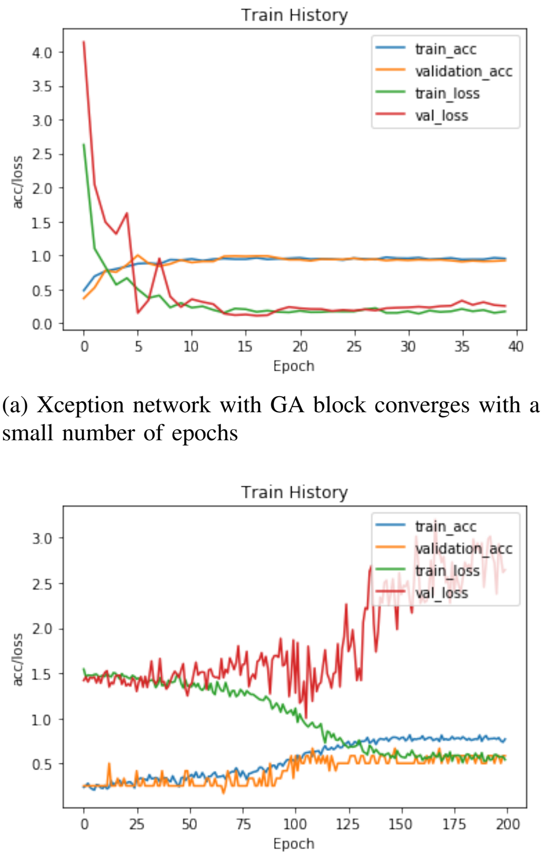
III. Experimental Setup

A. Dataset

The histology patches dataset consist of 400 labeled training data and 100 unlabeled test images, each with a 2048 x 1536 pixels. The distribution of cancerous classes of the images is uniform, as shown in Table I. Highly experienced medical specialists classified the images into four categories. The Classes are Normal, Benign, In-Situ, and Invasive. The resolution of the images is 0.42 μm x μm per pixel.

TABLE I: The Distribution of the ICIAR Dataset in our Study.

Type	Normal	Benign	InSitu	Invasive
Train	90	90	90	90
validation	5	5	5	5
Held-out test	5	5	5	5
Test	25	25	25	25



(a) Xception network with GA block converges with a small number of epochs

(b) Xception network bottom-top training overfits due to the lack of proper domain adaptation

Fig. 3: The Effect of The GA Block on Convergence

B. Evaluation Metrics

We employed the standard evaluation metrics to compare the performance of the proposed method to other existing techniques. The used evaluation metrics are as follows:

- Accuracy: the rate of correct predictions to the total number of samples [23].

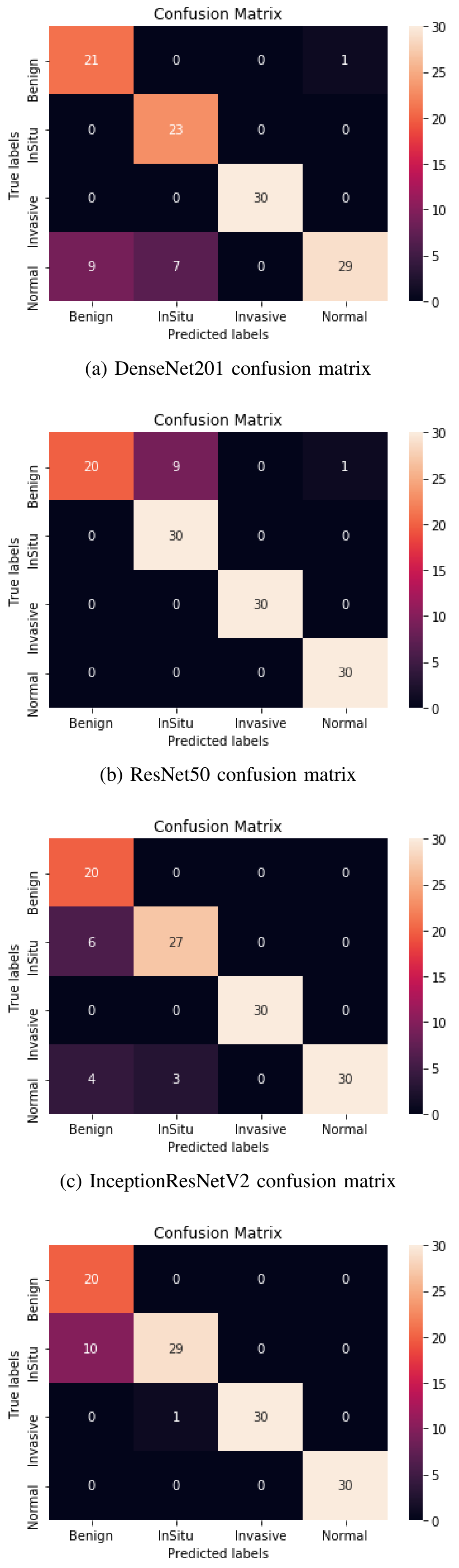
$$acc = \sum TP + \sum TN / \sum \#samples \quad (3)$$

- Precision: the rate of correct class predictions to the total number of samples belonging to that class [23].

$$Precision = \sum TP + \sum TN \quad (4)$$

- Recall: the true positive rate [23]

$$TPR = \sum TP / \#positive\ samples \quad (5)$$



(a) DenseNet201 confusion matrix

(b) ResNet50 confusion matrix

(c) InceptionResNetV2 confusion matrix

(d) Xception confusion matrix

Fig. 4: Error Type Visualization to Aim Choose Hetrougenous Models by Using the Confusion Matrix

-

$$F1score = 2 \times \frac{Precision \times Recall}{(Precision + Recall)} \quad (6)$$

- log-loss: the cross-entropy loss, which is the negative log-likelihood of the class labels given predictions [23].
- The area under the ROC curve

C. Results and Discussion

Domain adaptation is decisive for transfer learning. The Genetic Algorithm demonstrated supremacy in determining the transferable layers, as shown in Figure 3a. As an example, the GA solution of the freeze layers of the Xception enabled the model to transfer swiftly and converge with few epochs. In contrast, the bottom-top strategy of fine-tuning failed to converge even with substantially more iterations. As the number of epochs increased, the training accuracy increased, but the validation loss increased as well. This phenomenon is referred to as overfitting since the model is not learning the characteristics that increase the class separability margin. Alternatively, the model is learning the dataset noise, which is profoundly undesirable. Hence, implementing GA for specifying the transferable features of the network was vital for convergence.

Besides, the effectiveness of the base learning rate is highly correlated to the level of transferability of layers. The more transferable the layer is, the higher the base learning rate needs to be and vice versa. The occasional divergence of the bottom-top training is a result of setting a high base learning rate for layers with low transferability. Figure **??** shows how quickly cyclical learning rate picks the most appropriate base LR with minimal time cost.

Our proposed approach shows notable improvements over earlier published schemes for ensemble learning. The optimized horizontally stacked predictions' vector exhibit increased classification accuracy and additional DL evaluation measures, as confirmed in Table V.

TABLE II: Majority Voting Classification Report

Class. Report	precision	recall	f1-score
Benign	1.00	0.37	0.54
InSitu	0.60	1.00	0.75
Invasive	1.00	0.93	0.97
Normal	0.97	1.00	0.98
micro avg	0.82	0.82	0.82
macro avg	0.89	0.82	0.81
weighted avg	0.89	0.82	0.81
samples avg	0.82	0.82	0.82

TABLE III: Average Voting Classification Report

Class. Report	precision	recall	f1-score
Benign	1.00	0.70	0.82
InSitu	0.97	1.00	0.98
Invasive	1.00	1.00	1.00
Normal	0.79	1.00	0.88
micro avg	0.93	0.93	0.93
macro avg	0.94	0.93	0.92
weighted avg	0.94	0.93	0.92
samples avg	0.93	0.93	0.93

TABLE IV: Proposed Voting Classification Report

Class. Report	precision	recall	f1-score
Benign	1.00	0.73	0.85
In Situ	0.97	1.00	0.98
Invasive	1.00	1.00	1.00
Normal	0.81	1.00	0.90
micro avg	0.93	0.93	0.93
macro avg	0.94	0.93	0.93
weighted avg	0.94	0.93	0.93
samples avg	0.93	0.93	0.93

TABLE V: Evaluation of The Proposed Method Compared to Mavority and Average voting

Eval. /method	Proposed	Maj.	Avg.
P. auc roc	0.9994	1.0	0.9934
L. auc roc	0.95556	0.8833	0.95
L. accuracy	0.933	0.825	0.925
P. precision	0.9985	1.0	0.988
L. precision	0.8946	0.760	0.883
L. log loss	2.302	6.0442	2.590
L. coverage error	1.2	1.525	1.225
P. coverage error	1.116	1.25	1.133
L. LRAP	0.95	0.868	0.943
P. LRAP	0.9583	0.899	0.9527
ranking loss	0.066	0.175	0.075
ICIAR acc.	**88%**	85%	87%

IV. Conclusion

The current state-of-the-art transfer learning methods for microscopy image classification lack the systematic separation between task-dependent layers and transferrable ones, resulting in overfitting when training over a limited amount of samples. Moreover, their ensembles have a massive number of hyper-parameters, making it challenging and time-consuming to choose the optimal ones manually.

This paper presents the utilization of evolutionary algorithms and the cyclical learning rate scheduler to automate the selection of task-specific layers and its appropriate base learning rate during meta-training. The meta-training shrinks the hyper-parameters search space considerably, yielding an accelerated convergence during the comprehensive training stage. Also, it exhibits an intuitive measure of heterogeneity and automatic optimization of the horizontally stacked prediction vector's wights. The experimental outcomes confirm the competence of the proposed method in terms of robustness and training time efficiency.

We strongly recommend further investigation toward learning rate schedulers and other search methods of hyper-parameters. Hence, future research should place a particular emphasis on quantum search methods, such as Grover's algorithm, which provides a precise calculation of the iterations' upper bound that assures an optimal solution.

References

[1] Yan He, Wei Jin Ma, and Ji Ping Zhang. “The Parameters Selection of PSO Algorithm influencing On performance of Fault Diagnosis”. In: *MATEC Web of Conferences* 63.2016 (2016), p. 02019. DOI: 10.1051/matecconf/20166302019.

[2] Hafiz Mughees Ahmad, Sajid Ghuffar, and Khurram Khurshid. “Classification of Breast Cancer Histology Images Using Transfer Learning”. In: *Proceedings of 2019 16th International Bhurban Conference on Applied Sciences and Technology, IBCAST 2019* (2019), pp. 328–332. DOI: 10.1109/IBCAST.2019.8667221. arXiv: 1802.09424.

[3] Artem Pimkin et al. “Ensembling Neural Networks for Digital Pathology Images Classification and Segmentation”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 10882 LNCS (2018), pp. 877–886. ISSN: 16113349. DOI: 10.1007/978-3-319-93000-8_100. arXiv: arXiv:1802.00947v1.

[4] Hongliu Cao et al. “Improve the Performance of Transfer Learning Without Fine-Tuning Using Dissimilarity-Based Multi-view Learning for Breast Cancer Histology Images”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 10882 LNCS (2018), pp. 779–787. ISSN: 16113349. DOI: 10.1007/978-3-319-93000-8_88. arXiv: arXiv:1803.11241v1.

[5] Nadia Brancati et al. “A Deep Learning Approach for Breast Invasive Ductal Carcinoma Detection and Lymphoma Multi-Classification in Histological Images”. In: *IEEE Access* 7 (2019), pp. 44709–44720. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2019.2908724. URL: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp={\&}arnumber=8678759https://ieeexplore.ieee.org/document/8678759/.

[6] Thaína A. Azevedo Tosta, Leandro A. Neves, and Marcelo Z. do Nascimento. “Segmentation methods of H&E-stained histological images of lymphoma: A review”. In: *Informatics in Medicine Unlocked* 9.February (2017), pp. 35–43. ISSN: 23529148. DOI: 10.1016/j.imu.2017.05.009. URL: https://doi.org/10.1016/j.imu.2017.05.009.

[7] GR. He, L.; Long, LR; Antani, S. and Thoma. “Computer Assisted Diagnosis in Histopathology.” In: 3 (2009), pp. 272–287.

[8] Afshine Amidi et al. “EnzyNet: Enzyme classification using 3D convolutional neural networks on spatial representation”. In: *PeerJ* 2018.5 (2018), pp. 1–11. ISSN: 21678359. DOI: 10.7717/peerj.4750. arXiv: 1707.06017.

[9] aban Öztürk and Bayram Akdemir. “Effects of Histopathological Image Pre-processing on Convolutional Neural Networks”. In: *Procedia Computer Science* 132.June (2018), pp. 396–403. ISSN: 18770509. DOI: 10.1016/j.procs.2018.05.166. URL: https://doi.org/10.1016/j.procs.2018.05.166.

[10] Guilherme Aresta et al. “BACH: Grand challenge on breast cancer histology images”. In: *Medical Image Analysis* 56 (2019), pp. 122–139. ISSN: 13618423. DOI: 10.1016/j.media.2019.05.010. arXiv: arXiv:1808.04277v1.

[11] Hessam Zakerzadeh, Charu C. Aggrawal, and Ken Barker. “Towards Breaking the Curse of Dimensionality for High-Dimensional Privacy: An Extended Version”. In: (2014). arXiv: 1401.1174. URL: http://arxiv.org/abs/1401.1174.

[12] Jia Deng et al. “ImageNet : A Large-Scale Hierarchical Image Database”. In: (), pp. 2–9.

[13] Dan Hendrycks, Kimin Lee, and Mantas Mazeika. “Using Pre-Training Can Improve Model Robustness and Uncertainty”. In: 2018 (2019). arXiv: 1901.09960. URL: http://arxiv.org/abs/1901.09960.

[14] Jason Yosinski et al. “How transferable are features in deep neural networks?” In: *Advances in Neural Information Processing Systems* 4.January (2014), pp. 3320–3328. ISSN: 10495258. arXiv: arXiv:1411.1792v1.

[15] Thomas G. Dietterich. “Ensemble methods in machine learning”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 1857 LNCS (2000), pp. 1–15. ISSN: 03029743.

[16] Loris Nanni, Stefano Ghidoni, and Sheryl Brahnam. “Ensemble of convolutional neural networks for bioimage classification”. In: *Applied Computing and Informatics* (2018). ISSN: 22108327. DOI: 10.1016/j.aci.2018.06.002.

[17] Luis Perez and Jason Wang. “The Effectiveness of Data Augmentation in Image Classification using Deep Learning”. In: (2017). arXiv: 1712.04621. URL: http://arxiv.org/abs/1712.04621.

[18] Yann Lecun et al. “LeNet”. In: *Proceedings of the IEEE* November (1998), pp. 1–46. ISSN: 00189219. DOI: 10.1109/5.726791. arXiv: 1102.0183.

[19] Nitish Srivastava et al. “Dropout: A simple way to prevent neural networks from overfitting”. In: *Journal of Machine Learning Research* 15 (2014), pp. 1929–1958. ISSN: 15337928.

[20] Leslie N. Smith. “Cyclical learning rates for training neural networks”. In: *Proceedings - 2017 IEEE Winter Conference on Applications of Computer Vision, WACV 2017* April (2017), pp. 464–472. DOI: 10.1109/WACV.2017.58. arXiv: 1506.01186.

[21] Sergey Ioffe and Christian Szegedy. “Batch normalization: Accelerating deep network training by reducing internal covariate shift”. In: *32nd International Conference on Machine Learning, ICML 2015* 1 (2015), pp. 448–456. arXiv: arXiv:1502.03167v3.

[22] Rajiv K. Singh. *Fundamentals of natural representation.* Vol. 9. 7. 2018. ISBN: 9781420011449. DOI: 10.3390/info9070168.

[23] Nigel Williams, Sebastian Zander, and Grenville Armitage. “A preliminary performance comparison of five machine learning algorithms for practical IP traffic flow classification”. In: *Computer Communication Review* 36.5 (2006), pp. 7–15. ISSN: 01464833. DOI: 10.1145/1163593.1163596.