Comparative Performance Analysis of CNN-Based Feature Extraction and Classification Techniques for Histopathological Lung Cancer Image Classification

Carlo Dino
Stanford University
cdino@stanford.edu

Nicole Garcia
Stanford University
nicolejg@stanford.edu

Yu Han Daisy Wang
Stanford University
daisywyh@stanford.edu

Abstract

Lung cancer is one of the deadliest types of cancer, yet in the status quo, detection is usually done by hand via examination of histopathological imaging. The goal of this project is to compare various architectures to identify what models best identify instances of lung cancer from medical imaging. We aim to produce multiple architectures for computer aided diagnosis (CAD) models to identify instances of lung cancer from different medical images taken such that we reduce the burden placed upon medical professionals to diagnose lung cancer at any stage. To do this, we compared four different feature extractors (AlexNet, EffientNetB0, ConvNeXt, and our custom architecture DGW-Net), which we then feed into three different image classifiers (Softmax, SVM, SVM + PCA). Our results demonstrate that deeper models perform better for feature extraction (EfficientNetB0, ConvNeXt), and classifiers that are better capture variance across all features and consequently perform better (Softmax, SVM). Although we do not intend to replace the role of a medical professional to identify instances of lung cancer, we hope to create CAD models that make this process more efficient for both medical providers and patients alike.

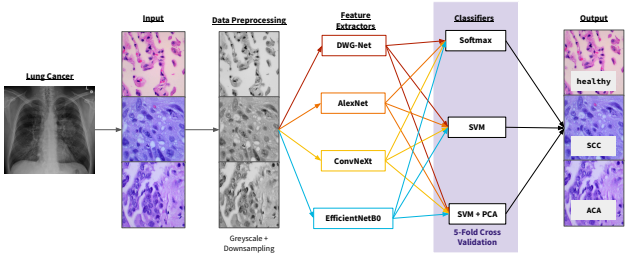

Figure 1. Graphical representation of our abstract.

1. Introduction

Cancer is one of the leading causes of death worldwide, and amongst all cancers, lung cancer is one of the deadliest types. Current diagnosis and treatment of lung cancer relies heavily on histopathological imaging, thus making the task of comprehending the information present in histopathological images especially important. Traditional interpretation of these images is done manually by trained staff, but this process is very time intensive. In order to make this process more efficient, we will be implementing three different deep learning pre-trained models as image classifiers in order to perform computer-aided diagnosis (CAD) of cancer.

1.1. Problem Statement

Our input is various lung cancer histopathology images which are either of healthy lungs, squamous cell carcinoma (SCC), or adenocarcinoma (ACA). We aim to adapt three different pre-trained models (AlexNet, EfficientNet, and ConvNeXt), as well as our own custom CNN architecture (DWG-Net), as feature extractors for lung cancer histopathological images. Using these extracted features, we will use Softmax, SVM, and an SVM + PCA approach as classifiers, from which we will get our final output, which is the predicted label for each image. Despite potentially flagging more images than necessary, our preliminary goal of eliminating false negatives in our models highlights our fundamental commitment to ensuring no patient goes undiagnosed due to misclassification.

2. Literature Review

2.1. Main Approach

Our main approach for this project, as well as our baseline, is taken from the paper "Classification of PCA based Reduced Deep Features by SVM for Diagnosing Lung and Colon Cancer". In this paper, the authors compare three pre-trained CNN model architectures – specifically

AlexNet, SqueezeNet, ShuffleNet, as well as three different classifiers models – specifically Softmax, SVM, and PCA + SVM, and compared the accuracy of each approach over the dataset that we will be using [1].

We wanted to see if we could recreate the results shown in the paper. In the paper, they were able to achieve accuracies of 93.12% just using AlexNet and SoftMax [1], which we find to be an incredibly high number, especially considering that AlexNet was only able to acheive roughly 84.7% on ImageNet on its debut [5]. As such, we want to see if we can reliably replicate the results shown by Al-Ofary and Ilhan, thus we have decided to recreate their implementation of AlexNet.

This paper demonstrated that for lung cancer, an approach that employed using ShuffleNet for feature extraction, applied PCA over the extracted features, and then fed these results in SVM resulted in the highest accuracy out of all the approaches they tested. As SqueezeNet and ShuffleNet are no longer considered state-of-the-art architectures, we wanted to recreate the approach of this paper using state-of-the-art models.

In order to review Al-Ofary and Ilhan's results, as well expand to more state-of-the-art models, we have chosen to test the following models: AlexNet, EfficientNet, and ConvNeXt.

2.2. AlexNet For Medical Image Classification

AlexNet was perhaps one of the first convolutional neural networks to show great success with the task of image classification, having achieved at the time record breaking accuracy on the ImageNet Large Scale Visual Recognition Challenge. AlexNet itself is a CNN with five convolutional layers with ReLU as the activation function, followed by three fully connected layers, and one last SoftMax layer [5].

Usage of AlexNet for medical image classification has had historical precedent with commendable results, not just in the work by Al-Ofary and Ilhan. For example, Hosny et al. were able to acheive success in classifying skin lesions using an AlexNet based approach, acheiving 97.93% accuracy on their best run. More importantly, Hosny et al. were able to show that image augmentation was significant in being able to improve training accuracy. Specifically, they rotated each image 72 times, each time by 5 degrees, which lead to as much as a 26.45% increase in accuracy [3].

2.3. EfficientNet For Medical Image Classification

EfficientNet serves as an early application of ConvNet scaling. Its usage for medical image classification purposes is well established, with many teams proposing various EfficientNet based approaches for various medical image classification problems, such as interpreting CT scans. EfficientNet's architecture consists of an initial convolution layer followed by 7 mobile inverted bottleneck convolution layers, whose output is then passed to the final convolution, pooling, and affine layers [9]. Using this architecture, EfficientNet employs the scaling of its network's width, depth, and resolution to achieve high accuracy. Specifically, it uses the following relationships to determine the applicable dimensionalities:

$$\begin{gathered}\text{depth: } d = \alpha^\phi \\ \text{width: } w = \beta^\phi \\ \text{resolution: } r = \gamma^\phi \\ \text{such that } \alpha \cdot \beta^2 \cdot \gamma^2 \approx 2 \text{ and } \alpha, \beta, \gamma \geq 1\end{gathered}$$

where ϕ is a compound coefficient that uniformly scales the network's width, depth, and resolution, and α, β, γ are constants that can be determined by a small grid search.

The usage of EfficientNet for medical imaging classification purposes, specifically histopathology images, has been shown to have success by Kallipolitis et. al. in their paper "Ensembling EfficientNets for the Classification and Interpretation of Histopathology Images". In this paper, Kallipolitis et. al. were able to achieve high accuracy on classifying histopathology images, while still maintaining relatively explainable models. Specifically, they were able to achieve accuracy rates of at best 98.35% for binary classificaiton using EfficientNet B0-2, with a 40-60 training-validation split. On multi-class classification, they were able to achieve accuracy rates of 92.64% using EfficientNet B1-3. Furthermore, Kallipolitis et. al. were also able to demonstrate that EfficientNets were effective at classification with various scales of magnification, ranging from the original images, to almost 400x magnification scales [4].

2.4. ConvNeXt For Medical Image Classification

The last model we will consider is ConvNeXt, which is made entirely from standard ConvNet modules and achieves competitive performance to transformers in terms of accuracy and scalability. Maintaining the simplicity and efficiency of standard ConvNets, ConvNeXt follows the same design changes to reach the architecture of a hierarchical visual Transformer without the use of any attention-based modules [6].

Similar to the other two architectures, there has been success with the use of ConvNeXt for medical image classification, specifically in histopathology images. In the paper "From modern CNNs to vision transformers: Assessing the performance, robustness, and classification strategies of deep learning models in histopathology", Springenberg et. al. demonstrate that ConvNeXt type architecture, specifically ConvNeXt-L, achieved great success on various histopathology datasets. Notably, Springenberg et. al. used colour shifts of images in their image augmentation process, hoping to reflect the discrepancies between staining procedures and camera lighting between labs. However, this gave

mixed results, and is why we have chosen not to pursue this type of augmentation for our project [8].

3. Methodology

3.1. General Overivew

Our method can be divided into four parts: data preprocessing, feature extraction, classification, then prediction.

Our project works off of the architecture demonstrated in [1], but replaces their pre-trained CNN models with our aforementioned pre-trained models in 2.

The process begins with grayscaling the dataset, while maintaining the three RGB channels. This is done because our pre-trained models assume an image with the standard three RGB channels as input. We accomplish this by grayscaling the input image into one channel, but simply duplicating this channel two more times to match the original shape.

Afterwards, we begin by splitting the down-scaled dataset, partitioned from the original dataset described in Section 4, into a train/test split. We then feed the training set into our models, including our baseline and pre-trained CNN's in 2, and split the training data into five separate folds to perform 5-fold cross-validation during classifier training.

We then pump each output into each of our classifiers: Softmax, SVM, and SVM + PCA. Every one of these classifiers are newly trained for each of the models. This results in a totl computation of twelve different classifiers being produced under our methodology.

We then measure output via the four metrics described in 3.5 by running classifier predictions on the test set that was segmented out earlier in the process. This is done by evaluating the difference between each classifier's predictions against the true labels of each image.

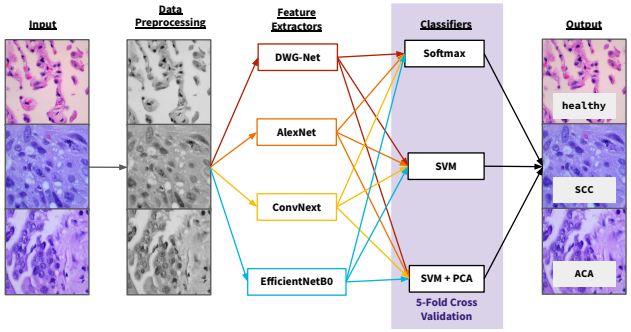

Figure 2. Flowchart of our proposed methodology.

3.2. CNNs as Feature Extractors

For feature extraction, we decided to test out four different CNNs: our custom CNN architecture DGW-Net, as well as three CNNs pretrained on ImageNet – AlexNet, ConvNeXt, and EfficientNet.

3.2.1 Proposed Baseline: DGW-Net

As a baseline, we propose DGW-Net, a simple CNN architecture that consists of two Convolution-ReLU-Pool cycles that feed into an FC layer for classification. A detailed look at our model can be found in figure 3

DGW-Net has a very basic architecture that pales in the number of parameters compared to these pretrained models, and as such we expect its results to serve as a bare minimum for these other models to achieve. This gives us a reference point for evaluating the performances of the more complex models, helping us understand whether not the additional complexity is justified by its performance gains.

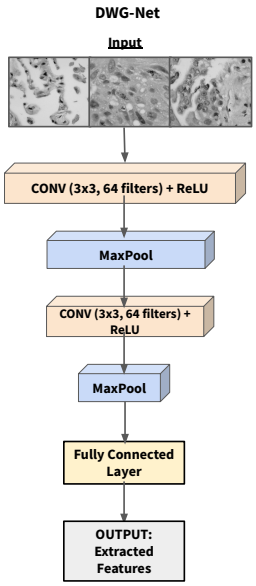

Figure 3. Flowchart depiction of our custom CNN architecture, DGW-Net.

3.3. Transfer Learning: AlexNet, ConvNeXt, EfficientNet

For the three more complex models, namely AlexNet, ConvNeXt, and EfficientNet, we utilize a transfer learning approach, using the weights from when the models were trained on ImageNet-1K. For each pretrained model, we removed the last layer, which would have been the "classification layer", in order to use each model as a feature extractor. We then froze all remaining layers, so that we wouldn't change the pretrained weights. By utilizing this transfer learning approach, we're able to significantly decrease the overall training time and resources used, while still maintaining reasonably high accuracy and performance. Additionally, this allowed us to further evaluate the performance

between state-of-the-art CNNs between each other and our proposed baseline model in medical image classification, which itself is an important subquestion for our project.

Given the limited funding and resources for our project, we decided to use the base models of each of these listed models. For this reason, we trained using a base model of ConvNeXt, EfficientNetB0, and a base model of AlexNet. These models have less pretrained weights that their more modern counterparts, allowing us to run our tests within reasonable execution times and smaller memory loads.

3.4. Classifiers

For classifiers, we decided to replicate the approach taken by our reference paper, so we tested Softmax, SVM, and SVM + Principle Component Analysis (PCA) as our approaches. When training our classifiers, we used 5-fold cross validation. By using 5-fold cross-validation, we were able to get a much more reliable estimate of the performances for the classifiers, especially when compared to a single train-test split. When doing a single train-test fold, the model is only evaluated once on one dataset, which means that the results may vary significantly based on the distribution of data across the train and validation sets. However, with k-fold cross-validation, one is forced to test the performance of the model across multiple datasets, thus reducing variance as we average the performance of the model across multiple validation sets, making our performance estimation much more reliable.

We aim to train our classifiers to correctly classify images into the following three classes: None (N), Adenocarcinoma (ACA), and Squamous Cell Carcinoma (SCC). We chose on implementing the following classifiers in order to replicate the approach shown in Al-Ofary and Ilhan's paper.

3.4.1 Softmax

The Softmax classifier is a single fully-connected layer of weights that we train on the features extracted from each of our models. The classification process works by first computing the raw scores of each class, a matrix-matrix product between the classifier's weights and the input, and then squashes these scores into normalized class probabilities. The model classifies the input as the class with the highest calculated probability, which can interpreted as being the class it has the most "confidence" in.

The Softmax classifier utilizes cross-entropy loss during its training process to calibrate its weights. As taken from Stanford's CS231N course website, the loss follows the following form, in terms of the i'th input:

$$L_i = -f_{y_i} + \log \sum_j e^{f_j} \tag{1}$$

where f_j is the j-th element of the vector f of class scores produced by the matrix-matrix product, and f_{y_i} is the score of the true label of the input i.

3.4.2 Support Vector Machine

The other classifier that we train in this paper is a Support Vector Machine (SVM). Unlike it's Softmax counterpart, the SVM classifier solely computes the raw class scores for each input, computed by the same matrix-matrix product between the classifier's weights and the input. The model classifies the input as the class with the highest computed raw score.

The SVM classifier utilizes Multiclass SVM loss during its training process to calibrate its weights. As taken from Stanford's CS231N course website, the loss follows the following form, in terms of the i'th input:

$$L_i = \sum_{j \neq y_i} \max(0, f_j - f_{y_i}) \tag{2}$$

where f_j is the j-th element of the vector f of class scores produced by the matrix-matrix product, and f_{y_i} is the score of the true label of the input i.

3.4.3 SVM with Principle Component Analysis

This classifier serves as an extension to our original SVM classifier, but modifies the process by incorporating Principal Component Analysis (PCA) for dimensionality reduction of the extracted features from each of our models. This process was very successful in Al Ofary and Ilhan [1], and we sought to recreate their results through the use of this classifier in our experiments.

PCA transforms the input data into a set of orthogonal vectors, known as "principal components", which aim to capture the maximum variance in the data. In theory, PCA should capture all of the important aspects of the dataset and disregard the rest, creating effective dimensionality reduction. Through dimensionality reduction, we're able to reduce the complexity of the feature space, making our SVM training more efficient, more robust to overfitting, as well as improved generalization.

In our current implementation, we decided to reduce the dimensionality of our extracted features to the top twenty components. As Al Ofary and Ilhan [1] do not publish the max number of commponents utilized in their experiment, this number was decided solely from consideration for our limited resources and short timeline.

3.5. Metrics

The metrics that we will utilize on our experiment follow from the metrics utilized in [1]. These values are derived from the values of True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN). From this, we

derive the following metrics: Accuracy, Precision, Recall, and Harmonic Mean (F1). The Harmonic Mean of Precision and Recall blunts the strength of outliers impacting our metrics, making its a more significant metric to calculate [1]. The metrics are calculated via the equations below:

$$\text{Accuracy} = \frac{\text{TF} + \text{TN}}{\text{TF} + \text{TN} + \text{FP} + \text{FN}} \tag{3}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{4}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{5}$$

$$\text{F1} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{6}$$

A number of similar medical image classification projects serve as a comparison to our lung cancer CAD models. Specifically, in the realm of melanoma skin lesions, accuracy rates ranging from 75.1% to 96.86% [3] were achieved across the different models of comparable types. Similarly, for prostate cancer detection, the reported accuracy ranged from 71% to 97% when using CAD models for medical classification [7]. We aim to use these similar models used for other types of cancer detection to gauge the performance of our own three chosen models.

4. Dataset

To finetune our models, we selected a dataset [2] consisting of 25,000 histopathological images of lung and colon cancer with 5 classes. Each image is generated from an original sample of HIPAA compliant and validated sources, consisting of 750 total images of lung tissue (250 benign lung tissue, 250 lung adenocarcinomas, and 250 lung squamous cell carcinomas) and 500 total images of colon tissue (250 benign colon tissue and 250 colon adenocarcinomas) and augmented to 25,000 using the Augmentor package.

4.1. Scaling Down the Dataset

For the purposes of our project, we have decided to downscale and only utilize the lung cancer portion of the original dataset. This is due to both funding and time limitations. A full scale rendition of our work would perform the same work in colon cancer detection for this dataset as well.

4.2. Data Preprocessing

There is no universally standardized staining protocal for histopathological imaging. As such, each lab has their own process, which leads to varying colour grading across labs. This was evident in our data, where the diseased samples had a clear purple hue, while the healthy samples had a clear pink hue. In order to increase the generalizability of our model, and also prevent it from associating certain hues with certain conditions.

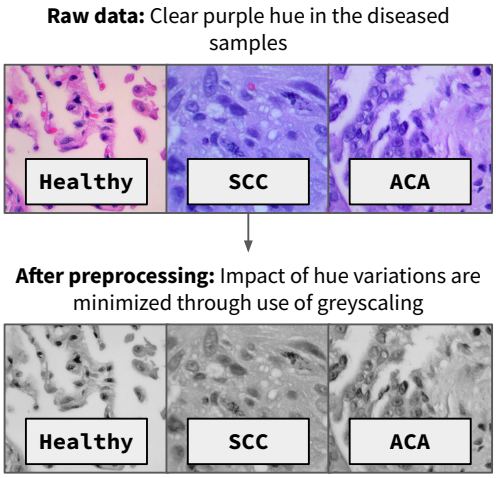

Figure 4. Before data preprocessing, it is very clear that the healthy samples have strong pink hue, while the diseased samples have a strong purple hue. After greyscaling, the impact of hue variations has been mitigated.

We also further downsampled the images to 224x224, in accordance to the image sizes that AlexNet was trained on [5], which allowed us to reduce our computational complexity, while still retaining a large amount of detail from the original images.

4.3. Data Augmentation

The dataset comes pre-augmented, reducing the original images from size 1024x768 pixels to 768x768 pixels. Through augmentations, the dataset is expanded to 25,000 images from the original 1,250 images by applying left and right rotations (up to 25 degrees, 1.0 probability) as well as horizontal and vertical flips (0.5 probability).

5. Results and Discussion

In this section, we briefly discuss our experimental setup, review the final results from our experiments, and discuss their significance via quantitative and qualitative metrics.

5.1. Experimental Setup

5.1.1 Hyperparameters and Optimizers

For DGW-Net, we trained our classifier using Pytorch Optimizer's implementation of Stochastic Gradient Descent. We achieved our current best results with a learning rate of 5e-4 for SVM, 2.5e-4 for Softmax, and 5e-3 for SVM+PCA. All classifiers utilized momentum in their optimizers, with a momentum value of 0.9 each.

For each of our pre-trained models, we utilized Scikit's pipeline 'fit' function to train our classifiers. We utilized the

default values for each of these pipelines.

5.1.2 Batch Size

In all of our classifiers, we utilized a mini-batch size of 32 during each training process. This number was partly chosen because 32 is a power of two, which is a standard convention in the field. Another reason is because our limited computational resources, matched with the fact that each image is originally 768x768 pixels before downscaling, forced us to limit the number of loaded images in each mini-batch. This was especially important as we transfer our images from CPU into GPU for computation, as GPUs suffer from smaller memory sizes.

5.2. Quantitative Results

As described in our Methodology 3.1 section, we ultimately trained and tested twelve different classifiers throughout our experiments. The final metrics, as described in our Metrics 3.5 section, calculated from each classifier's performance on the test set are described in the table shown in figure 5.

Feature Extractor	Classifier	Average Validation Accuracy	Test Accuracy	Precision	Recall	F1 Score
AlexNet	SVM	0.8422	0.8388	0.8380	0.8380	0.8383
	Softmax	0.8398	0.8423	0.8423	0.8423	0.8418
	PCA + SVM	0.7929	0.7807	0.7796	0.7807	0.7799
EfficientNet B0	SVM	0.9186	0.9303	0.9260	0.9260	0.9259
	Softmax	0.9249	0.924	0.9343	0.9343	0.9343
	PCA + SVM	0.9009	0.9003	0.9080	0.9080	0.9079
ConvNeXt	SVM	0.9594	0.9593	0.9593	0.9593	0.9593
	Softmax	0.9665	0.9663	0.9663	0.9663	0.9663
	PCA + SVM	0.9327	0.9247	0.9243	0.9247	0.9247
DGW-Net	SVM	0.8468	0.8497	0.8481	0.8497	0.8483
	Softmax	0.8125	0.8270	0.8271	0.8271	0.8212
	PCA + SVM	0.8301	0.8212	0.5771	0.8212	0.5681

Figure 5. A table of the various metrics for our models.

When analyzing our models, AlexNet and DGW Net did not perform as well when compared to the other two models being tested. In addition, the SVM + PCA classifier generally performed worse among all feature extractor models, indicating that this specific classifier may not be best suited for lung cancer detection.

In analyzing our models and classifiers, we found that the ConvNeXt model and Softmax classifier performed best among all the model-classifier combinations. This could be in part due to the efficiency of the ConvNeXt model as well as its ability to capture the complexity of the feature space.

5.3. Discussion

5.3.1 Depth is essential to the success of a vision model

Out of all of our models, the ones that performed the worse over all classifiers was our DWG-Net model. However, we can see that AlexNet was not too far behind. These results surprised us, as AlexNet has nearly sixty billion learnable parameters compared to our simple DGW-Net baseline.

In comparison, we notice that the ConvNeXt and EfficientNet models vastly outperform DGW-Net and AlexNet across all metrics over each of their respective classifiers. Specifically, both models achieve performance of over 90% over all of their respective classifiers over all of our metrics.

The difference between these outcomes are stark, but can be explained by the architecture of the inidividual models. Both ConvNeXt and EfficientNet are deep models, whereas DGW-Net and AlexNet are more shallow. For context, DGW-Net's architecture results in a total count of 8 layers, matching the total number of layers employed by AlexNet. Meanwhile, EfficientNetB0 totals up to 237 layers and ConvNeXt contains 7 layers in just a single ConvNeXt residual block.

Our metrics show that deeper architectures result in a significant improvement in classification performance on our dataset of lung cancer cell images. This conclusion comes at complete odds with the results described in Ilhan and Ofary [1], where the authors describe the success of AlexNet over both SqueezeNet (eighteen layers) and ShuffleNet (fifty layers).

This result was very interesting, as our experiments show that deeper architectures do perform vastly better than shallower ones. We will further discuss this difference between our results and Ilhan and Ofary's in Section 5.3.4.

5.3.2 Dimensionality reduction and its drawbacks

In terms of the classifier architectures, all of our models with PCA + SVM struggled the most. This surprised us, as Al-Ofary and Ilhan had their PCA + SVM models as their most succesful models. We suspect that this discrepancy comes from two reasons: too little primary components/too much dimensionality reduction through PCA in our approach, and the lack of image normalisation in Al-Ofary and Ilhan's approach.

In our approach, due to technical constraints, we were only able to produce 24 principal components for our PCA analysis, while we suspect that Al-Ofary and Ilhan might have produced more principal components, though it is unclear from their paper. As such, we suspect that we may have been unable to capture the full variance in the feature space, resulting in the SVM classifer having to classify with access to lack of information.

Furthermore, when we examine the respective confusion

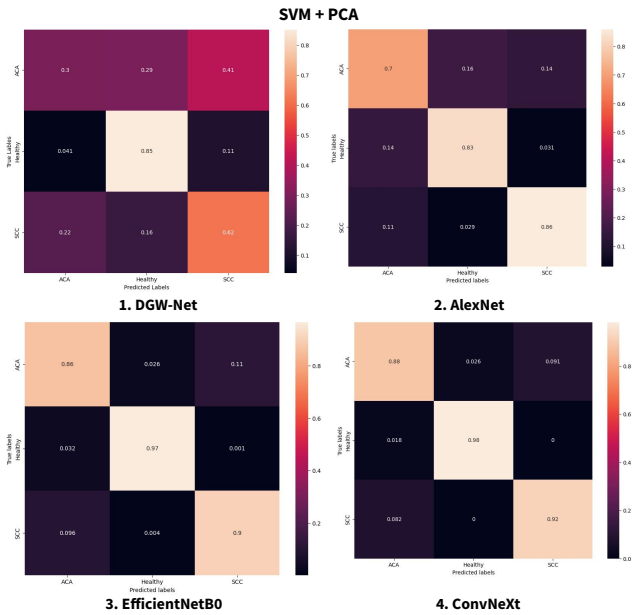

Figure 6. Confusion matrices for each feature extractor model that used a PCA + SVM architecture as the classifier. The remaining confusion matrices can be found in the appendix.

diagrams for each model that used the PCA + SVM setup, we can see that overall there was "confusion" for this specific setup. In particular, the DGW-Net model performed exceptionally bad when compared to other models. This is reflected as well in the exceptionally low precision and F1 scores for the DGW-Net model with PCA + SVM, being at 0.5771 and 0.5681 respectively. This shows that the proportion of true positives to positives as a whole is very low, which is reflected in the confusion matrix. In the confusion matrix, we see that the most popular classification for adenocarcinoma images was squamous cell carcinoma, and notably not adenocarcinoma, again reflecting the poor performance of DGW-Net when using PCA + SVM. We hypothesize that this is due to the simplicity of the DGW-Net model. As the DGW-Net model has relatively few parameters, it has a reduced dimensionality compared to the full feature space. As such, by performing PCA on this model, we actually "remove" too much of the already relatively sparsely captured feature space, resulting in the poor performance of the model.

5.3.3 Minimizing false negatives

At the beginning, we outlined that our goal was to create an architecture that would have no false negatives, given the nature of our system as a potential *CAD*, the minimization of false negatives serves as a bare minimum for practical use in the field. Although our metrics do not capture the raw number of false negatives from each classifier, we instead reference each model's performance in *accuracy* and *recall* to measure this goal.

Our best feature extractor and classifier pair, ConvNeXt-Softmax, achieves accuracy of 0.9665 and recall of 0.9663. These high levels imply that the total number of false negatives produced by ConvNeXt-Softmax is exceedingly low. Although not currently perfect by any means, we believe that our architectures can reach even stronger metrics given longer training times and the utilization of deeper versions of ConvNeXt and/or EfficientNet.

5.3.4 Comparison to Al-Ofary and Ilhan

As discussed above in section 2.2, we wanted to see if we could replicate the results obtained by Al-Ofary and Ilhan, specifically in their AlexNet setup. Without greyscaling our images, our AlexNet setup was able to achieve similar results to Al-Ofary and Ilhan, reaching a peak accuracy of 91.8% with the Softmax classifer on a trial run of 2-fold cross-validation. This result is comparable to the 93.12% that Al-Ofary and Ilhan achieved using their AlexNet and Softmax setup.

However, after greyscaling our images, our AlexNet model obtained much lower results across all classifiers, achieving a peak test accuracy of 84.23% with Softmax. The significantly lower accuracies post-greyscaling put the results of Al-Ofary and Ilhan into question and indicate that their classifier models may be trained on the hue of the images as opposed to actual signs of lung cancer. Furthermore, we question the validity of the results of Al-Ofary and Ilhan as a whole. Despite running less powerful models, such as AlexNet, SqueezeNet, and ShuffleNet, they were able to achieve near perfect accuracies, which we were unable to achieve. Here, we propose that due to the lack of image normalization, their models have not actually learned to recognise the relevant structures, but instead rely on hue changes between the images to classify whether not something is diseased, which is greatly

6. Conclusion and Future Work

In this paper, we compared the usage of four CNN architectures as feature extractors, as well as the use of three classifiers, producing 12 models, which we then evaluated over histopathology images of lung cancer. Shallower models proved to be less successful, while deeper models proved to be more successful, though none were able to achieve the goal of zero false negatives.

The key takeaways from this paper are as follows:

- Depth is one of the main limiting factors in a vision model.
- Too much dimensionality reduction will lead to a model not being able to learn all features of the data,

and perform poorly in testing.

- In the specific case of histopathology images, it is important to account for the variances in staining processes across labs.

For future work, we plan to explore different feature extractors and classifiers. Due to resource constraints, we were only able to deploy relatively smaller models of model families such as EfficientNetB0 and ConvNeXt, but in the future, we would like to investigate if there are any performance gains to be recorded from using EfficientNetB7 and ConvNeXt V2, which serve as the more complex versions of each of our respective feature extractors. Furthermore, we would like to see how transformer based architecture performs for feature extraction, and if there are different strengths and weaknesses to using it. We would also like to try more complex classifiers, perhaps even using CNN-type architecture as the classifiers, as opposed to our currently relatively simply classifier structures.

7. Acknowledgements

All authors contributed equally to this project and made significant efforts on the code and manuscript. Specifically, Daisy Wang worked on the EfficientNetB0 model and classifiers, Nicole Garcia worked on the AlexNet model and classifiers, and Carlo Dino worked on the DGW-Net and ConvNeXt models and classifiers. All members contributed equally to the architecture development and written report. All code can be found at https://github.com/devcaelo/CS231N-Final-Project.

References

[1] S. Al-Ofary and H. O. Ilhan. Classification of pca based reduced deep features by svm for diagnosing lung and colon cancer. 06 2023.

[2] A. A. Borkowski, M. M. Bui, L. B. Thomas, C. P. Wilson, L. A. DeLand, and S. M. Mastorides. Lung and colon cancer histopathological image dataset (lc25000). *arXiv (Cornell University)*, 12 2019.

[3] K. M. Hosny, M. A. Kassem, and M. M. Foaud. Classification of skin lesions using transfer learning and augmentation with alex-net. *PLOS ONE*, 14:e0217293, 05 2019.

[4] A. Kallipolitis, K. Revelos, and I. Maglogiannis. Ensembling efficientnets for the classification and interpretation of histopathology images. *Algorithms*, 14:278, 09 2021.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60:84–90, 05 2012.

[6] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie. A convnet for the 2020s. *arXiv:2201.03545 [cs]*, 01 2022.

[7] R. Llobet, J. C. Pérez-Cortés, A. H. Toselli, and A. Juan. Computer-aided detection of prostate cancer. *International Journal of Medical Informatics*, 76(7):547–556, 2007.

[8] M. Springenberg, A. Frommholz, M. Wenzel, E. Weicken, J. Ma, and N. Strodthoff. From modern cnns to vision transformers: Assessing the performance, robustness, and classification strategies of deep learning models in histopathology. *Medical image analysis*, 87:102809–102809, 07 2023.

[9] M. Tan and Q. V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks, 09 2020.

8. Appendix

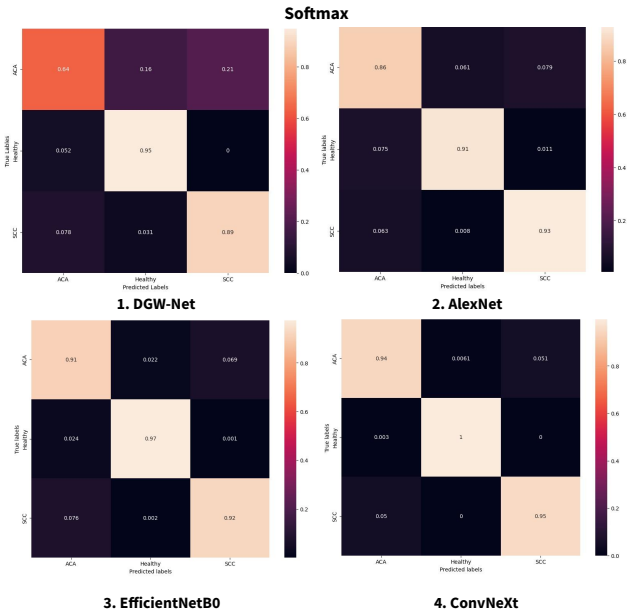

Figure 7. Confusion matrices for each feature extractor model that used a Softmax architecture as the classifier.

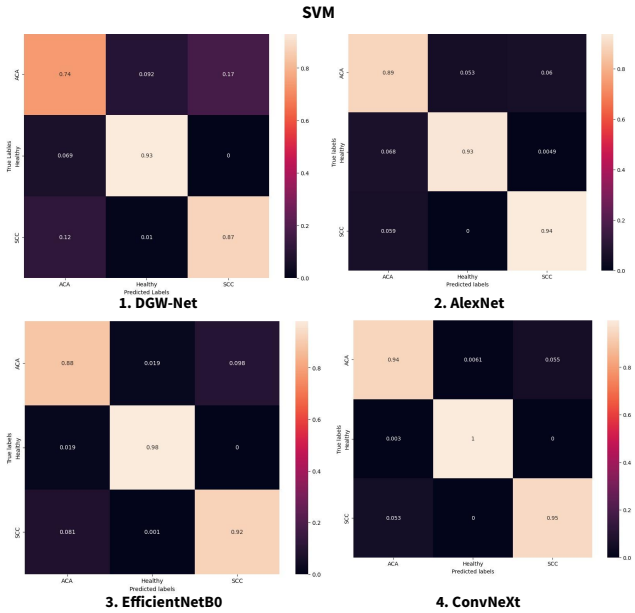

Figure 8. Confusion matrices for each feature extractor model that used a SVM architecture as the classifier.