Machine Learning Methods for Lung Tumour Diagnosis

Carl Viggo Nilsson Gravenhorst-Lövenstierne
carlviggo@icloud.com

February 05, 2024

Abstract

Lung cancer stands as one of the deadliest forms of cancer. In 2020 alone, the disease claimed the lives of more than 1.8 million individuals globally [1]. Advancements in medical imaging technology, such as low-dose computed tomography, have resulted in a global accumulation of high-quality image data. However, the traditional approach of identifying and diagnosing cancer still relies on manual evaluation of such data. This workflow is time-consuming and prone to errors.

To assist clinicians, computer models specifically designed for tumour classification have recently been developed. This study aims to enhance and evaluate machine learning algorithms, specifically random forests and convolutional neural networks, in diagnosing lung cancer through the analysis of computed tomography scan images.

This study reveals that based on the analysis of the Kaggle Data Science Bowl 2017 data set [2], the random forest model with feature extraction surpasses the fine-tuned VGG16 and InceptionV3 models in validation accuracy with statistical significance, underscoring its superior performance.

In addition, we can conclude that the random forest model with feature extraction from the PyRadiomics library, combined with the synthetic minority over-sampling technique (SMOTE), proved more efficient on limited datasets than the fine-tuned VGG16 model trained on data augmented by a generative adversarial network (GAN). Nonetheless, if the dataset had been larger and denser in terms of information, then the deep learning approach might have yielded improved performance.

Acknowledgements

I want to thank Dr. Mehdi Astaraki at Stockholm University, Department of Medical Radiation Physics, for his continuous guidance throughout this project. I am also grateful for the many inspiring discussions with Daria Chamoun and for the proofreading provided by Aleksander Purik. Furthermore, I would like to express my gratitude to the Research Academy for Young Scientists (RAYS), Stiftelsen Hierta-Retzius Stipendiefond, and Stiftelsen Oscar och Maria Ekmans Donationsfond for their support in making this project possible.

Contents

1 Introduction

In 2020, cancer caused more than 10 million deaths worldwide [1]. As our lifestyles change and the global population ages and grows, this number is expected to almost double by 2040 [3]. Among the various types of cancer, lung carcinoma, i.e. cancer in the respiratory region, is the most deadly [1]. If the disease is detected early, the chances of survival increase significantly [1].

Improvements in medical imaging technology have made it possible to capture vast quantities of high-resolution medical images from patients. This has led to an accumulation of cancer image data worldwide. Conventional methods of assessing abnormalities in medical images include uncertain, labour-intensive methods such as manual image observation. In the image data, the presence and type of cancer is not always easily detectable, which increases the risk of misdiagnosis. This uncertainty could prohibit patients from being diagnosed early in the disease course, and thereby impact their survival chances negatively.

When cancer is detected, the malignancy of any observed tumour is classified. The next step is to identify the tumour characteristics. For instance, there exists a complex micro-environment within cancerous regions which represents different levels of aggressiveness [4]. In some cases, such characteristics can be studied by conducting an invasive surgery known as a biopsy. When a tumour is incorrectly classified and its malignancy goes undetected, it poses additional risks to the patient.

Computer-based models have recently been developed to assist clinicians with their daily work. This approach relies upon novel imaging processing techniques and machine learning models. In this project, conventional machine learning algorithms such as random forests and convolutional neural networks, including generative adversarial networks, have been applied to classify benign and malignant tumours in the lung region.

Background

A list of prerequisites and their corresponding explanations can be found in this section.

1.1 Quantitative Imaging Biomarkers

Thanks to recent improvements in medical imaging technology, for example, magnetic resonance imaging or computed tomography scans, the information content within medical images has undergone substantial growth in recent years [5]. This enhanced level of information can be utilized as a quantifiable indicator of potential disease.

Medical images provide 2D or 3D quantitative imaging data from the inside of human organs. Such information can be further processed for diagnosis, prognosis and prediction purposes in a process known as QIB, quantitative image biomarking. A subfield of QIB is radiomics, in which particular image features are extracted, processed and then analyzed. [6]

These features must be chosen with respect to the set of data and the purpose of the study; in this study – lung tumour classification. Examples of extractable features are tissue density, image entropy — a metric of how diverse the pixel values are; and image energy — a measure of variance in pixel intensity. The intensity corresponds to the tissue density. One method of feature extraction is the appliance of filters. The sequential feature selection filter, SFS, is an effective technique to find and keep only the most informative features and remove less predictive ones [7]. Another filter is the least absolute shrinkage and selection operator (lasso) which assigns additional weights to relevant features based on their predictive powers. Lastly, the principal component analysis (PCA) reduces the dimensionality of a dataset by transforming it into a set of variables called principal components. This process retains the most

significant features of the data, simplifying analysis while preserving essential information [8]. Out of the thousands of features that exist, deep learning and machine learning models can be trained to extract the most relevant features with respect to the characteristics of the dataset [9].

1.2 Random Forests

Random forests consist of decision trees, which are supervised machine learning algorithms used for regression and classification tasks. They consist of a series of conditional statements which split and thereby sort the data. The following schematic flowchart represents an example of a decision tree:

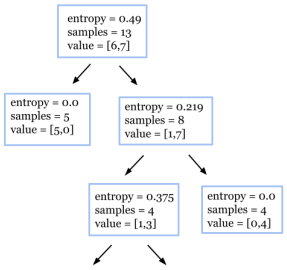

Figure 1: Unit of a decision tree. The nodes are represented by boxes and the branches by arrows. The value parameter stands for [number of classes, number of samples].

The model determines the splitting condition by comparing the entropy in the parent node against the entropy in the child node. The splitting condition that generates the largest difference in entropy, also called the information gain, is finally chosen. The information gain is defined as:

$$\text{Information Gain} = \Delta S = \sum_{i=1}^{c} p_i \log_2 p_i - \sum_{j=1}^{c} p_j \log_2 p_j \qquad (1)$$

In which p is the probability of randomly choosing a data point in class i and c is the number of classes. The splitting continues until the entropy of all leaf nodes is 0 or until the parameters of the model are satisfied. To determine the information gain, all possible splitting combinations must be tested, resulting in decision trees being relatively slow and inappropriate for large sets of data. Another weakness inherent to decision trees is that unless a maximum tree depth is defined, the tree will continue to split the data until all leaf nodes are in a state of zero entropy [10], resulting in overfitting.

Random forests are supervised machine-learning models generally less prone to overfitting than decision trees. In the initialization of the model, randomized features are extracted from a sub-section of the dataset in a process called bootstrap aggregation. These features are fed to a set of n decision trees that are inherent to the model. The class selected by most trees becomes the final output from the random forest. The adoption of randomness from bootstrapping improves the generalization capabilities of the model and thus improves its robustness [11].

1.3 Artificial Neural Networks

Artificial neural networks are data structures inspired by the biological nervous system used for function approximation purposes. The most primitive unit of a neural network is the perceptron, see figure 2.

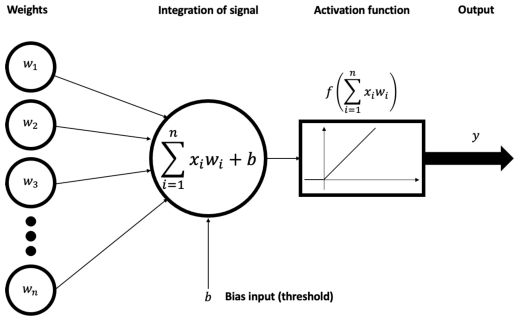

Figure 2: Schematic representation of a perceptron, including input data: x_n, weights: w_n, biases: b, weighted sum and activation function. (CC BY-SA 4.0)

In the initialisation of the model, each

node connection is assigned a randomized weight and bias. The input, in the form of an RGB image matrix, is then multiplied with the weights and summarized as a weighted sum, where b is the bias:

$$\sum_{i=1}^{n}(x_i w_i + b) \tag{2}$$

The model must be introduced to a dimension of non-linearity to be able to adapt itself to the input data [12]. This is achieved by the activation function which takes the weighted sum as input. Its output either serves as input to a new perceptron or as the final classification of the model. If the activation function had been linear, then the model would only be able to learn linear relationships between the input and outputs.

A neural network is composed of an input layer, at least one hidden layer with n neurons and an output layer. In a densely connected neural network, every node is interconnected by weights. The parts of a neural network which must be manually configured, e.g. the number of nodes per layer or the activation function, are called hyperparameters. The hyperparameters make up the model architecture and can influence the performance of the model significantly. Choosing the right hyperparameters is often the main challenge in constructing deep learning networks.

1.4 Loss Functions and Optimizers

Loss functions and optimizers are components of neural networks that are essential to their training process. In supervised neural networks, the loss function, for example, mean squared error or cross-entropy loss, is used to calculate the difference between the predicted class and the true class. An optimizing function numerically derives the loss function, with respect to its hyperparameters. This derivative helps the optimizer to tune the weights of the neural network in a process called backpropagation. Simultaneously, the input data is passed to the model n times, also called the number of epochs. This cycle is known as model training and ideally continues until the loss function converges at one of its local minimum values.

1.5 Convolutional Neural Networks

A convolutional neural network (CNN) automatically extracts features that are relevant for the classification of the image data. A typical convolutional neural network architecture consists of convolutional layers, see figure 3, that are alternated by pooling layers. This structure is finally interconnected with a densely connected neural network which gives the final classification.

The convolutional layer, a subunit of the convolutional neural network, initially convolves a tensor into feature maps that contain the extracted features from the initial tensor. In the domain of greyscaled medical images, this tensor is a two-dimensional matrix, see figure 3.

The convolving operation involves a filter matrix, also known as a kernel, that discretely moves over the input matrix. For each step, the sum of the dot product between the input matrix and the kernel is calculated and saved in the feature map. In the initialization of the model, the kernel values are randomized. Yet, as the model backpropagates, these values are tuned by the optimizing function in the training process to extract increasingly complex attributes. The convolving operation is summarized in equation 3.

$$\sum_{i=1}^{m}\sum_{j=1}^{n}(T_{ij} \cdot K_{ij}) \tag{3}$$

Where K_{ij} is the kernel and T_{ij} is the input tensor with the dimensions $m \times n$. The position of the kernel is represented by

$i \times j$. An approach used in this project is to apply a kernel of dimensionality 3×3. This example is visualized in figure 3, where the variables i, j, m, n equal 3.

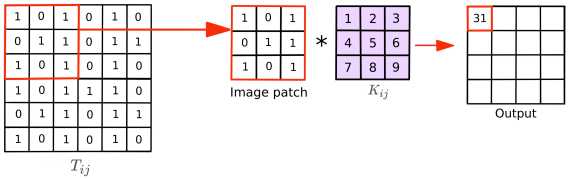

Figure 3: Schematic representation of a convolutional layer. The sum of the cross products between the image patch and the kernel is calculated and projected onto the feature map.

Multiple feature maps can be extracted from a single input matrix. The feature maps are either transferred to another convolutional layer or to a densely connected neural network, where they are ultimately classified. To minimize computing time, the feature maps can be passed through pooling layers, where their dimensionality is reduced.

Figure 4 visualizes the transformation process of an image after being applied to two convolutional layers, g_1 and g_2. This particular kernel extracts information about variance in light intensity.

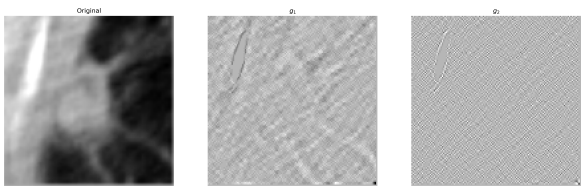

Figure 4: Application of two convolutional layers, g_1 and g_2, onto the original image.

1.6 Image Augmentation, Feature Learning, Transfer Learning

A common phenomenon for machine learning models is to develop more parameters than what is realistic for the data they are trained on. Just like a polynomial can be modified to fit n data points, a machine learning model can overadjust itself and thereby lose its ability to accurately classify new data. This phenomenon is called overfitting and can be partly counteracted by image augmentation, feature learning and transfer learning.

Image augmentation is a technique that can be implemented to increase the size of the training dataset. This is achieved by applying geometrical or intensity transformations to the data with techniques such as filter application, rotation, flipping and zooming. When dealing with limited datasets, a large number of features increases the risk of overfitting. By reducing features that are associated with one another, the authenticity of the dataset increases. [13]

As the architecture of a convolutional neural network gets increasingly intricate, the model transitions from being able to detect primitive features such as edges, corners and diagonals to more complex features. The better a model is at detecting these primitive structures, the better suited it will be for more advanced tasks. Pretrained models trained on extensive datasets such as the ImageNet [14] can be imported and integrated, i.e. fine-tuned, into a preexisting neural network to improve its generalization capabilities. See figure 5 for example datapoints of the ImageNet dataset. [15]

1.7 Dimensionality Reduction

Dimensionality reduction simplifies datasets by reducing the number of variables, addressing the increased computational load and risk of overfitting in high-dimensional spaces. This boosts model efficiency and accuracy through two strategies: feature selection, which identifies the most relevant features, and feature extraction, which transforms data into a lower-dimensional space while preserving vital information. This project

Figure 5: 400 sample images from the ImageNet dataset [14].
(Creative Commons Licence)

implements the Principal Component Analysis, L1 Regularization and Sequential Feature Selection.

Principal Component Analysis (PCA) is a feature extraction method that reduces dimensionality by transforming the original data into a set of linearly uncorrelated variables known as principal components. PCA captures the maximum variance in the data with the fewest number of principal components. [16]

L1 Regularization (Lasso) is a type of regularization technique used in regression models that performs feature selection by shrinking the coefficients of less important features to zero and thereby removing them from the model. [17]

Sequential Feature Selection (SFS) is a feature selection method that adds or removes features iteratively based on a criterion, such as the performance of a classifier, aiming to find the optimal set of features that improves model performance.[18]

1.8 Generative Adversarial Networks (GANs)

Generative Adversarial Networks (GANs) are unsupervised machine learning frameworks employed to generate data with the characteristics of a given training dataset. GANs operate through an interplay between two key components: a generator and a discriminator. The loss experienced by one component corresponds to the gain of the other, resulting in a "competitive" training process. [19]

The generator and discriminator components are typically composed of convolutional neural networks (CNNs). The discriminator's role is to assess and classify the images generated by the generator. Initially, the generator starts with random input, which is optimized to produce images that are increasingly similar to those found in the original training dataset. [19]

The training process ideally continues until the generator has achieved a such level of proficiency that the discriminator can no longer distinguish between the images generated by the generator and those originating from the training dataset. [19]

1.9 Model Evaluation: Metrics

When validating a machine learning model, various metrics are used to quantify its performance. In the manual optimization of the model, this information gives an indication of what hyperparameters need to be tuned. While a single metric can be misrepresentative, combining multiple can give a more accurate representation of the model performance [20]. One method for model optimization is hyperparameter tuning, where hyperparameters are adjusted while measuring the model's performance. Another method for model evaluation is K-fold cross-validation. This algorithm trains the model several times, using different validation data for each training process, and thus reduces the risk of biased datasets.

Creating a confusion matrix, containing the four possible outcomes of binary classification: true positive (T_P), false positive (F_P), true negative (T_N), and false negative (F_N), is a prerequisite for calculating certain metrics. To clarify, a false positive diagnosis occurs when a patient's tumour is predicted as malignant when it is benign.

1.9.1 Accuracy

Accuracy is a metric used to measure the proportion of correctly classified predictions. The definition of accuracy is found in expression 4.

$$Accuracy = \frac{T_P + T_N}{T_P + T_N + F_P + F_N} \quad (4)$$

Where T_P, T_N, F_P and F_N correspond to true positive, true negative, false positive and false negative.

1.9.2 Loss

The loss function is an essential part of the learning algorithm which quantifies the mismatch between the model's predicted outputs and the actual true values. Binary cross-entropy, see equation 5, is commonly used in the field of convolutional neural networks to calculate loss [20].

$$\text{Loss} = -\sum_{i=1}^{N} [y_i \log(p_i) + (1 - y_i) \log(1 - p_i)] \quad (5)$$

In equation 5, y represents the true label, 1 for a positive class and 0 for a negative, and p is the predicted probability of the instance being in the positive class. This is repeated for all data points in the dataset.

Furthermore, the loss can serve as a certification of the accuracy metric, particularly in scenarios involving imbalanced datasets.

2 Method

The project was divided into two pipelines: P1 and P2, where P1 involved the creation of a random forest model, data preprocessing, radiomic feature extraction, dimensionality reduction, hyperparameter tuning, synthetic minority oversampling techniques and k-fold cross-validation analysis. The second pipeline involved image synthetisation by applying a generative adversarial network and then fine-tuning two pre-trained models: VGG16 and InceptinetV3, which were rigorously compared. Finally, the models developed in P1 and P2 were compared against one another, culminating in the selection of an optimal candidate.

2.1 Data Preprocessing

The Kaggle Data Science Bowl 2017 dataset [2], containing 1297 low-dose computer tomography images of pulmonary nodules was used as training, testing and evaluation data. Figure 6 displays 32 samples from the dataset, where the labels '0' and '1' denote the benign and malignant classes, respectively.

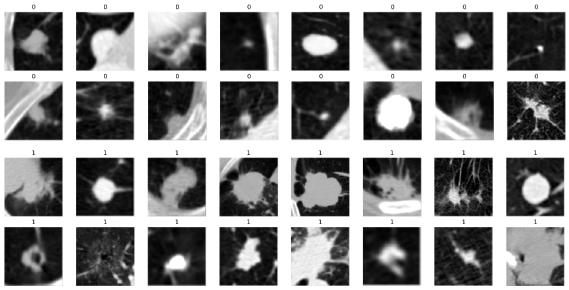

Figure 6: 32 samples from the Kaggle Data Science Bowl 2017 dataset [2]. An image is labelled as malignant if it has a title of 1, whereas a title of 0 indicates that the image is benign.

The images depicted in figure 6 are derived from low-dose computer tomography (CT) scans of the entire chest region of the patient. This is exemplified by the 2D axial images presented in figure 7.

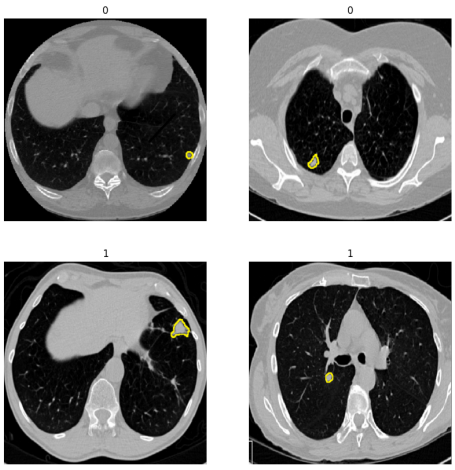

Figure 7: 2D axial images of the entire chest regions, with segmented pulmonary nodules. An image is labelled as malignant if it has a title of 1, whereas a title of 0 indicates that the image is benign.

2.2 P1: Radiomic Feature Extraction

A feature extracting method from the PyRadiomics python library was implemented to extract 1070 radiomic features from the dataset, including energy and entropy, which measure voxel intensity distribution, as well as statistical descriptors like mean intensity, maximum intensity, and geometric properties such as volume and surface area.

2.3 P1: Hyperparameter Tuning

The random forest machine learning model was initiated and maximum tree depth, $0-200$, and the number of estimators, $0-120$, were tuned combination-wise via the GridSearchCv module to optimize the performance of the model. See results in figure 10.

2.4 P1: Synthetic Minority Oversampling Techniques

A synthetic minority oversampling technique (SMOTE) was applied to counteract the imbalance between the two classes of the dataset; benign (67.5%) and malignant (32.5%). This imbalance had a degrading effect on the performance of the model on the minority class. The SMOTE algorithm creates synthetic data points by selecting a random example from the minority class, finding its k nearest neighbours, choosing one neighbour randomly, and creating a synthetic example at a randomly selected point between the two [21]. As a result, the relationship between the classes is balanced.

2.5 P1: Dimensionality Reduction and Feature Selection

To reduce the risk of overfitting, three different dimensionality reduction techniques were applied to the dataset sequentially: L1 regularization (lasso), principal component analysis (testing 2-31 components) and sequential feature selection (extracting 3 features during 3-fold cross-validation). After implementing each method individually, the AUC of the model was computed and the best candidate was chosen. See results in table 1.

2.6 P1: K-fold Cross Validation Analysis

Five-fold cross-validation was applied to the dataset to confirm the absence of statistical inequalities when splitting the data into training and validation sets. For each fold, the mean accuracy of the model was validated. See final model results in table 2.

2.7 P2: Generative Adversarial Network (GAN)

A generative adversarial network architecture was defined and trained with the hyperparameters of 200 epochs, a batch size of 6, and a learning rate of 1×10^{-5}. Binary cross-entropy was used to calculate loss and Adam was used as an optimizer for both the discriminator and the generator. The training set comprised 293 low-dose computer tomography images of malignant tumours from the Kaggle Data Science Bowl 2017 data set [2]. Figure 8 illustrates the evolution of the generator's performance, showcasing its incremental refinement as the number of epochs increases. To achieve a balanced training set with a 1:1 ratio between the two classes, 455 synthetic images of malignant tumours were generated, equalizing the number of benign and malignant cases. The validation set consisted of 240 images, evenly distributed with 120 images per class.

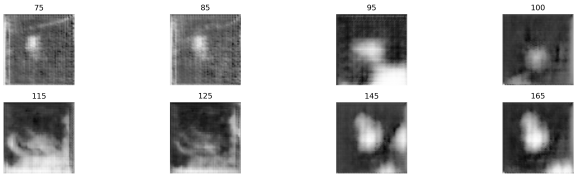

Figure 8: Feature maps produced by the generator after x epochs of training.

2.8 P2: Pretrained VGG16 InceptionV3 Fine-tuning

A convolutional neural network with the VGG16 architecture, see figure 9, pre-trained on the ImageNet dataset [14], containing $14,000,000$ images, was imported from Keras. The VGG16 was integrated with a three-layer densely connected neural network, incorporating dropout and batch-normalization layers to enhance model robustness and generalization. Leaky ReLU was selected as the activation function to address the vanishing gradient problem. To further mitigate overfitting, L2 regularization was employed across these layers with a lambda value of 0.06. Additionally, a learning rate schedule starting at 0.01 was implemented to fine-tune the training process, adjusting the learning rate dynamically to optimize convergence and model performance. The model was thereafter fine-tuned on the training set with the added synthesised images from the generative adversarial network. The same procedure was also applied to a pre-trained Inception V3 architecture, in place of the VGG16. Binary cross-entropy and accuracy for each model were extracted and summarized in table 2.

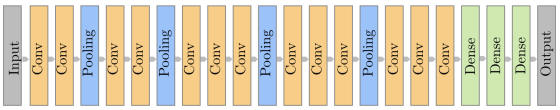

Figure 9: Schematic representation of the VGG16 architecture, which comprises 13 convolutional layers, 4 max-pooling layers and 3 densely connected layers.

3 Results

In this section, a comprehensive evaluation of the performance metrics from both the Random Forest and Deep Learning models is presented. By contrasting the models, we aim to highlight their respective strengths and weaknesses and to provide insights into their suitability for machine learning-guided lung tumour diagnosis.

3.1 Random Forest Metrics

The validation accuracy of the random forest model using extracted features was systematically assessed through the GridSearch-CV algorithm and compiled in figure 10. The validation accuracy initially increases rapidly and reaches a relative plateau of maxima in the intervals of 25-80 estimators and maximum depths of 110-50.

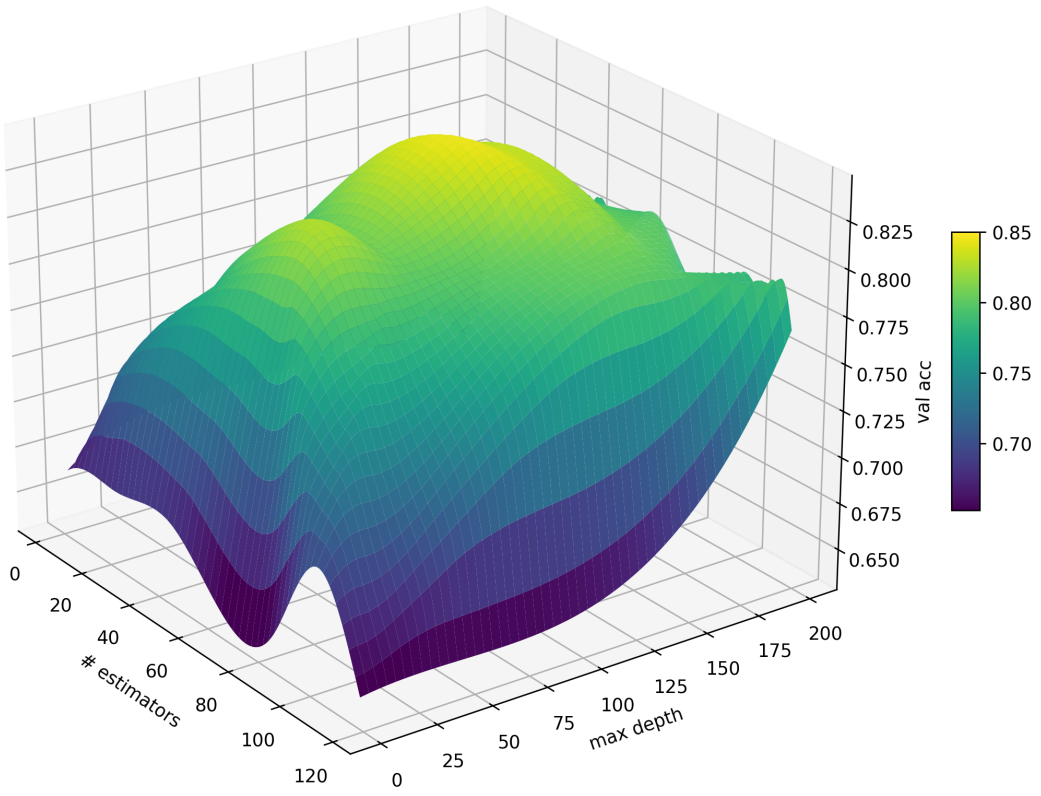

Figure 10: Visualization of mesh grid showing the relationship between estimators, maximum depth, and accuracy in model performance.

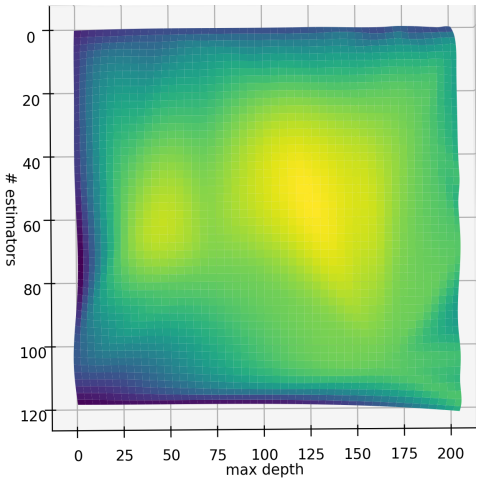

Figure 11: Complementary top-view mesh visualization of Figure 10, illustrating the variance of validation accuracies across the horizontal plane.

Table 1 summarizes the mean AUC results from a five-fold cross-validation, following the application of Lasso, PCA, and SFS filtering. During the validation process, a range of 2-30 features were evaluated for PCA, while exactly 10 features were considered for SFS.

Table 1: Mean AUC after filter application.

Filter Method	**Mean AUC**
Lasso	0.777 ± 0.062
PCA dimensionality reduction	0.810 ± 0.035
SFS algorithm	0.876 ± 0.021

3.2 Convolutional Neural Network Metrics

Figure 12 presents the compiled accuracy and loss metrics from n epochs of fine-tuning the pre-trained VGG16 convolutional neural network. The left graph indicates that the validation accuracy begins to converge near the 20th epoch, eventually stabilizing at an approximate mean accuracy of 0.65 with a standard deviation of 0.020. In contrast, the training accuracy persists in its ascent, suggesting that the model is overfitting on the training data. This phenomenon is confirmed by the divergence observed in the corresponding loss curves, where the training and validation loss trajectories deviate from one another. Data corresponding to the fine-tuning of the InceptionV3 model are summarised in table 2.

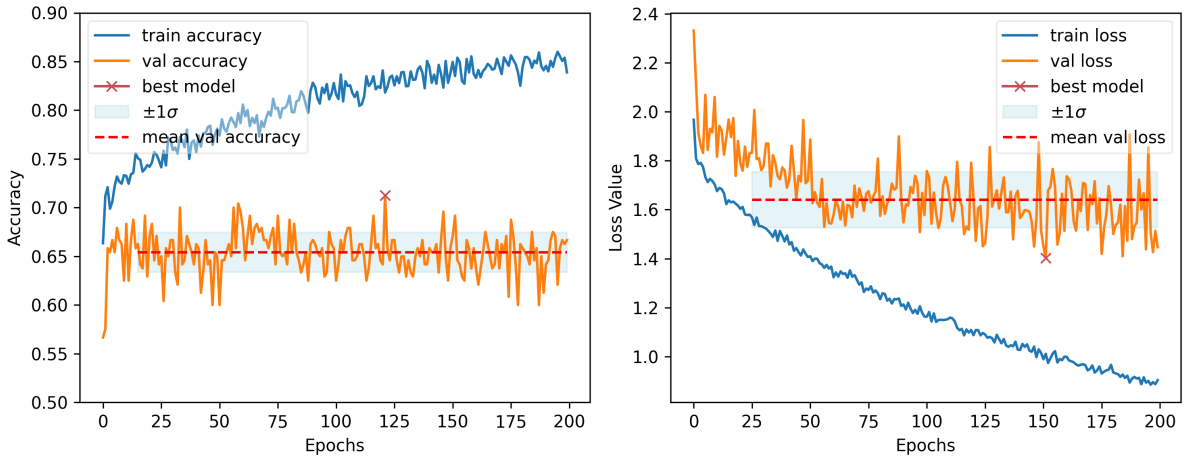

Figure 12: VGG16 fine-tuning learning curves. The orange graphs represent validation scores, while the blue graphs represent training scores. As the validation accuracy converges, the mean loss and accuracy values with standard deviation boundaries are plotted.

Table 2 summarises the mean validation accuracy and loss for the fine-tuned VGG16 and InceptionV3 models, in addition to detailing the performance metrics from the random forest model.

Table 2: Model summary.

Architecture	**Validation Accuracy**	**Validation Loss**
VGG16 w. pre-training	0.654 ± 0.022	1.64 ± 0.112
InceptionV3 w. pre-training	0.614 ± 0.024	3.40 ± 1.378
Random Forest w. feature-extraction	0.733 ± 0.045	-

The data presented in table 2 are visually represented as violin plots in figure 13. The observed larger variance in the random forest model's metrics may be attributable to the inclusion of all metrics in its analysis, as opposed to the selective consideration of post-convergence metrics in the deep learning models.

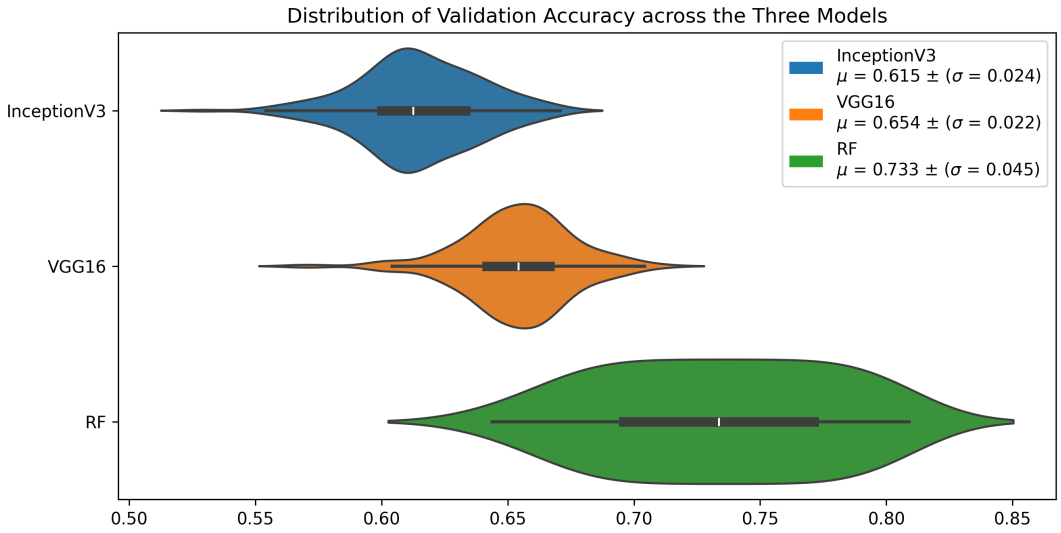

Figure 13: Visualization of table 2 using violin plots, where the violin body represents the distribution of model accuracy, the central error bar indicates the range within one standard deviation from the mean, while the extended error bar denotes the maximum and minimum values, excluding statistical outliers.

Table 3 summarises the results of pairwise T-tests comparing the validation accuracy of each respective model. This data is visualised in figure 14 as Gaussian curves for each model, offering a visual representation of the t-test analysis. The statistical outcomes of the t-tests, with a significance level α of 0.05, demonstrate that the random forest model's performance is **significantly** superior compared to the two evaluated deep learning models.

Table 3: Pairwise T-tests

Architecture-Pair	**T-statistic**	**p-value**	**Significant difference**
InceptionV3 vs. VGG16	-16.93	$p < .05$	✓
VGG16 vs. RF	-18.02	$p < .05$	✓
InceptionV3 vs RF	-25.78	$p < .05$	✓

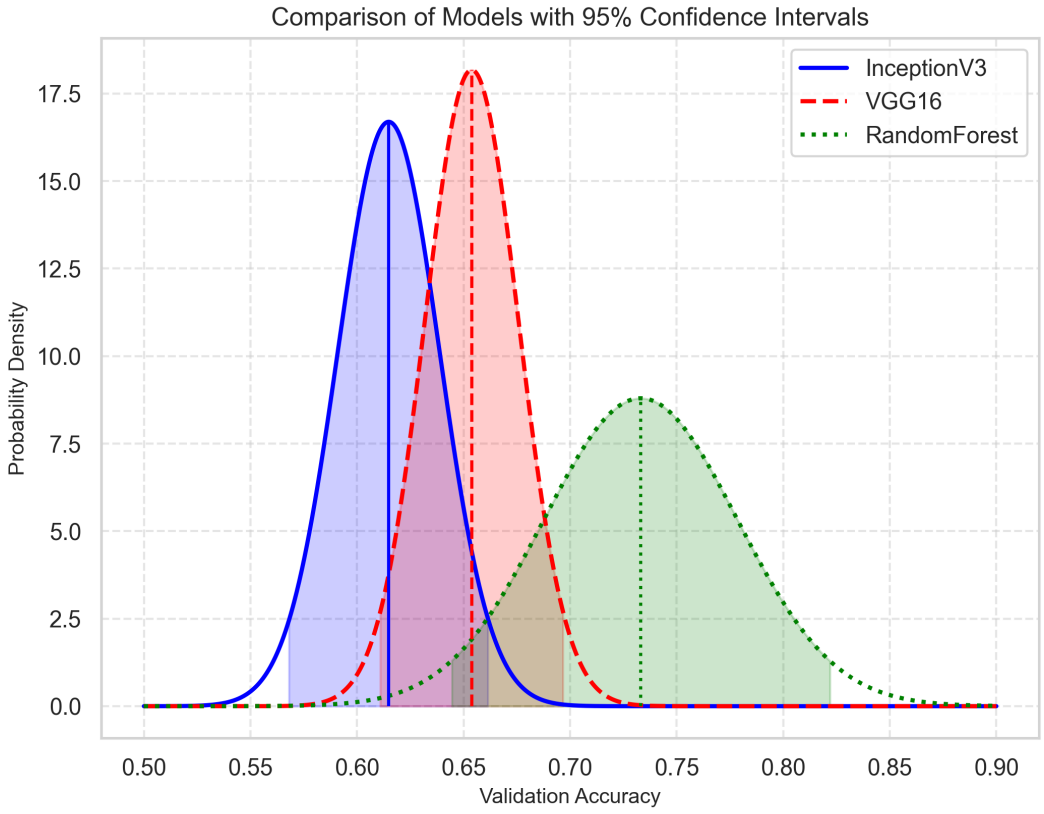

Figure 14: The distribution of validation accuracy for each model is depicted as gaussian curves, with the area under the curves around the mean representing the 95% confidence intervals.

4 Discussion

4.1 Model Training

All three models achieved convergence, indicating that further training would not yield substantial performance improvements. However, the divergence between validation and training accuracies observed in the deep learning models raises concerns regarding overfitting to the training dataset. This suggests that the relationship between the amount of data in the dataset and the amount of parameters in the model architecture is insufficient. To address this, optimizing hyperparameters such as increasing model regularization through additional dropout or L2 regularization could be considered potential measures. As the utilization of a generative adversarial network balanced the distribution of samples across the two classes for the deep-learning model, any performance exceeding 50% should be interpreted as an enhancement in model performance, effectively excluding the effect of random chance. This underscores the efficacy of generative adversarial networks in enhancing data distribution for balanced learning outcomes. Similarly, the implementation of the SMOTE algorithm for the random forest eliminates random chance as a potential explanation for the model's improvement. In summary, we conclude that the models have converged, and the enhancements in model performances can be attributed to effective training rather than random chance.

4.2 Model Comparison

The overlapping confidence intervals depicted in figure 14 might initially imply indistinguishability among model performances. However, the examination through pairwise t-tests reveals significant distinctions: the fine-tuned VGG16 model outperforms the fine-tuned InceptionV3 with statistical significance, and the random forest model surpasses VGG16 with statistical significance, establishing itself as the superior alternative. The pairwise t-test analysis clarifies that the observed performance differences cannot be attributed to chance, strengthening the validity of the model comparisons.

The superior performance of the random forest model could potentially be derived from the interplay between the quantity of training data and the complexity of the model. While the random forest model underwent testing across a spectrum of complexity levels through the GridSearch algorithm, starting from 0 estimators and depths, the VGG16 models initially possessed a fixed number of parameters. These parameters were subsequently reduced through the application of a fixed dropout and L2 regularization. However, this approach primarily focused on reducing complexity, and it did not systematically explore lower levels of complexity. This suggests a potential misalignment between the available dataset size and the complexity of the VGG16 model, which could have contributed to the observed differences in model performance.

In summary, the feature extraction method employed from the PyRadiomics Python library, which extracted a total of 1070 features, in combination with the synthetic minority over-sampling technique (SMOTE), demonstrated greater efficiency when applied to limited datasets compared to the fine-tuned VGG16 model trained on both original and synthetically generated data produced by the generative adversarial network.

4.3 Applicability in Healthcare

A study focused on the Bahcesehir Mammographic Screening program in Türkiye evaluated the breast cancer detection rate among radiologists [22]. The study reported an initial average accuracy of 67.3% for cancer detection when evaluated solely by humans. However, when supervised by the Lunit deep learning model [23], this accuracy improved to 83.6%. While the random forest model developed in this study achieved an average accuracy of 73.3%, surpassing the average accuracy of the radiologists, it remains insufficient for practical clinical implementation, in particular for more mortal types of cancer than breast cancer. The much lower five-year survival rate for lung cancer compared to breast cancer, as supported by lung cancer statistics [24] and breast cancer statistics [25], underscores the urgency of implementing more extensive diagnostic procedures with higher accuracies. Furthermore, there are legal considerations, including the determination of responsibility, that would need to be resolved before the full integration of machine learning methods into the realm of cancer diagnosis [26].

4.4 Further Studies

In the preprocessing phase, the original three-dimensional NIFTI files were transformed into the 2D JPEG format, which is less data-intensive. NIFTI images possess unique features, such as dimensionality, bit depth, and tissue density. For instance, lung tissue density is denoted by pixel intensity, typically ranging from -400 to 1000. However, when converted to the JPEG format, the intensity range must be adjusted to the 0 to 255 range due to JPEG's 8-bit compression. This results in information loss that could potentially degrade model performance.

Furthermore, only the axial view (the head-to-foot axis) was converted from the three-dimensional NIFTI format to JPEG. Including the sagittal (parallel to the sides of the patient) and coronal (perpendicular to the front of the patient) views would have provided a more comprehensive representation of the human respiratory region.

To circumvent these issues, the pydicom library could be employed to directly convert DICOM files into readable numpy arrays. While this approach would likely improve model performance, it would also significantly increase training time.

Moreover, the original resolution of the images was reduced to 244 x 244 x 1 to fit the VGG16 model's requirements. Given that pulmonary nodules occupy a small number of pixels, this reduction in resolution may result in the loss of critical information about their internal heterogeneity and shape, further impacting the model's effectiveness.

Additionally, all convolutional layers of the pre-trained models were utilized, which are highly optimized for the ImageNet dataset. This dataset does not necessarily reflect the characteristics of the pulmonary nodules. Reducing the number of imported layers and fine-tuning more layers could potentially enhance the model's generalizability and performance.

5 Conclusion

This study reveals that based on the analysis of the Kaggle Data Science Bowl 2017 data set [2], the random forest model with feature extraction surpasses the fine-tuned VGG16 and InceptionV3 models in validation accuracy with statistical significance, underscoring its superior performance.

In addition, we can conclude that the random forest model with feature extraction from the PyRadiomics library, combined with the synthetic minority over-sampling technique (SMOTE), proved more efficient on limited datasets than the fine-tuned VGG16 model trained on data augmented by a generative adversarial network (GAN). Nonetheless, if the dataset had been larger and denser in terms of information, then the deep learning approach might have yielded improved performance.

6 Code Access

The code, models, figures and relevant material used in this project are openly accessible on the following GitHub repository:

`https://github.com/CarlViggo/GY-arbete`

References

[1] Kleber, H., D., and Gold, M., S., , "Use of psychotropic drugs in treatment of methadone maintained narcotic addicts," *Ann N Y Acad Sci*, vol. 311, pp. 81–98, 1978.

[2] "Data science bowl 2017." https://www.kaggle.com/c/data-science-bowl-2017. Accessed: 07 08, 2023.

[3] Bray, F., Ferlay, J., Soerjomataram, I., Siegel, R. L., Torre, L. A., Jemal, A, "Global cancer statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries," *A Cancer Journal for Clinicians*, vol. 68, pp. 394–424, 2018.

[4] Xu, J., Liao, K., Yang, X. et al, "Using single-cell sequencing technology to detect circulating tumor cells in solid tumors," *Molecular Cancer*, 2021. https://doi.org/10.1186/s12943-021-01392-.

[5] Otero, Hansel J. and Rybicki, Frank J. and Greenberg, Dan and Neumann, Peter J., "Twenty Years of Cost-effectiveness Analysis in Medical Imaging: Are We Improving? ," *Radiology*, vol. 249, no. 3, pp. 917–925, 2008. https://doi.org/10.1148/radiol.2493080237.

[6] Park, J. E. and Kim, H. S. , "Radiomics as a Quantitative Imaging Biomarker: Practical Considerations and the Current Standpoint in Neuro-oncologic Studies," *Nucl Med Mol Imaging*, vol. 52, pp. 99–108, Apr 2018.

[7] E. Harefa and W. Zhou, "Performing sequential forward selection and variational autoencoder techniques in soil classification based on laser-induced breakdown spectroscopy," *Analytical Methods*, vol. 13, pp. 4926–4933, 2021.

[8] I. T. Jolliffe and J. Cadima, "Principal component analysis: a review and recent developments," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374, p. 20150202, 2016.

[9] M. Yogeshwari and G. Thailambal, "Automatic feature extraction and detection of plant leaf disease using GLCM features and convolutional neural networks ," *Materials Today: Proceedings*, vol. 81, pp. 530–536, 2023. https://www.sciencedirect.com/science/article/pii/S2214785321028029.

[10] Wikipedia contributors, "Decision tree — Wikipedia, the free encyclopedia." https://en.wikipedia.org/w/index.php?title=Decision_tree&oldid=1164832164, 2023. [Online; accessed 11-July-2023].

[11] David O'Connor and Evelyn M.R. Lake and Dustin Scheinost and R. Todd Constable, "Resample aggregating improves the generalizability of connectome predictive modeling," *NeuroImage*, vol. 236, no. 11, p. 118044, 2021. https://www.sciencedirect.com/science/article/pii/S1053811921003219.

[12] N. Kulathunga *et al.*, "Effects of the nonlinearity in activation functions on the performance of deep learning models," 2020. Funded by the National Science Foundation (NSF), grant no: CNS-1831980.

[13] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, p. 60, 2019.

[14] "Imagenet." `https://www.image-net.org/index.php`. Accessed: 07 08, 2023.

[15] A. Jeddi, M. J. Shafiee, and A. Wong, "A simple fine-tuning is all you need: Towards robust deep learning via adversarial fine-tuning," *CoRR*, vol. abs/2012.13628, 2020.

[16] I. Jolliffe, *Principal Component Analysis*. Springer Series in Statistics, Springer, New York, NY, 2002.

[17] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[18] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.

[19] Y. Hong, U. Hwang, J. Yoo, and S. Yoon, "How generative adversarial networks and their variants work: An overview," *arXiv preprint arXiv:1711.05914*, 2017.

[20] "Evaluating machine learning models and their diagnostic value," *PubMed*, 2023. Accessed: 2024-01-03.

[21] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[22] Kizildag Yirgin, I. and Koyluoglu, Y. O. and Seker, M. E. and Ozkan Gurdal, S. and Ozaydin, A. N. and Ozcinar, B. and lu, N. and Ozmen, V. and Aribal, E. , "Diagnostic Performance of AI for Cancers Registered in A Mammography Screening Program: A Retrospective Analysis," *Technol Cancer Res Treat*, vol. 21, p. 15330338221075172, 2022.

[23] "lunit." `https://www.lunit.io/en`. Accessed: 07 09, 2023.

[24] Di Girolamo, C. and Walters, S. and Benitez Majano, S. and Rachet, B. and Coleman, M. P. and Njagi, E. N. and Morris, M., "Characteristics of patients with missing information on stage: a population-based study of patients diagnosed with colon, lung or breast cancer in England in 2013," *BMC Cancer*, vol. 18, p. 492, May 2018.

[25] DeSantis, C. E. and Bray, F. and Ferlay, J. and Lortet-Tieulent, J. and Anderson, B. O. and Jemal, A. , "International Variation in Female Breast Cancer Incidence and Mortality Rates," *Cancer Epidemiol Biomarkers Prev*, vol. 24, pp. 1495–1506, Oct 2015.

[26] N. Naik, B. M. Z. Hameed, D. K. Shetty, D. Swain, M. Shah, R. Paul, K. Aggarwal, S. Ibrahim, V. Patil, K. Smriti, S. Shetty, B. P. Rai, P. Chlosta, and B. K. Somani, "Legal and ethical consideration in artificial intelligence in healthcare: Who takes responsibility?," *Frontiers in Surgery*, vol. 9, 2022.

A Complementary Data

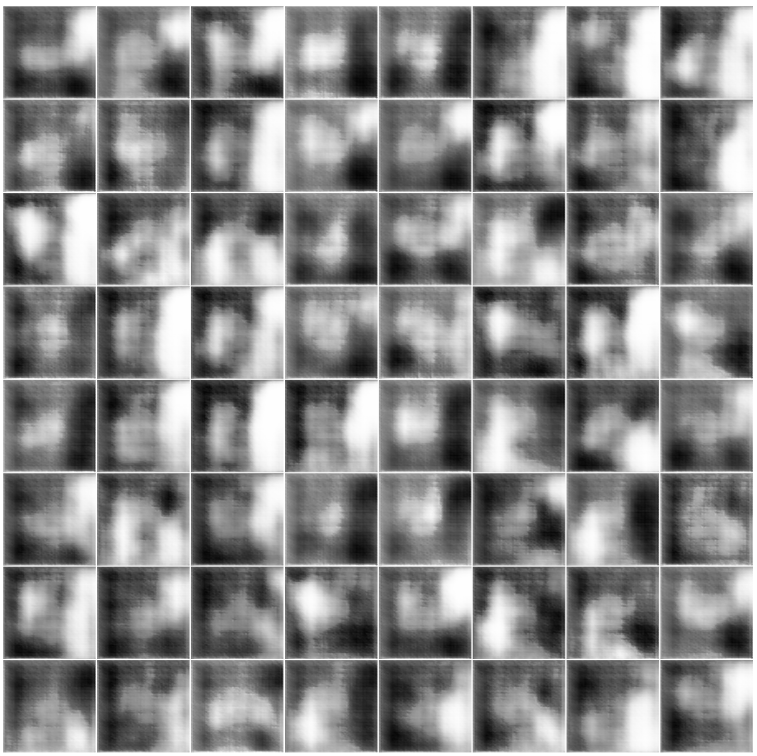

Figure 15: 64 sample images generated by the generative adversarial network used in this study.