HHS Public Access

Author manuscript

ACM BCB. Author manuscript; available in PMC 2020 July 07.

Published in final edited form as:

ACM BCB. 2019 September ; 2019: 485–493. doi:10.1145/3307339.3342166.

Fusion in Breast Cancer Histology Classification

Juan Vizcarra[1], **Ryan Place**[2], **Li Tong**[1], **David Gutman**[3], **May D. Wang**[1,*]

[1]Dept. of Biomedical Engineering, Georgia Institute of Technology and Emory University, Atlanta, GA 30332

[2]School of Biological Sciences, Georgia Institute of Technology, Atlanta, GA 30332

[3]Department of Neurology, Emory University, Atlanta, Georgia, United States

Abstract

Breast cancer is a deadly disease that affects millions of women worldwide. The International Conference on Image Analysis and Recognition in 2018 presents the BreAst Cancer Histology (ICIAR2018 BACH) image data challenge that calls for computer tools to assist pathologists and doctors in the clinical diagnosis of breast cancer subtypes. Using the BACH dataset, we have developed an image classification pipeline that combines both a shallow learner (support vector machine) and a deep learner (convolutional neural network). The shallow learner and deep learners achieved moderate accuracies of 79% and 81% individually. When being integrated by fusion algorithms, the system outperformed any individual learner with the highest accuracy as 92%. The fusion presents big potential for improving clinical design support.

Keywords

Histology; Fusion; SURF; breast cancer; deep learning; support vector machines

1. Introduction

In the United States, cancer ranks as the second highest cause of death. It is estimated that millions of new cases will appear in the following decades [1]. In 2019, The American Cancer Society estimated that there would be approximately 62,930 new cases of in situ breast cancer alone, which accounts for around 30% of all new cancer diagnosis in women [2]. Even with this rapid rise in occurrence, the gold standard of breast cancer diagnosis is for a pathologist to examine tissue samples manually. Breast tissues are usually stained with several different kinds of chemicals (e.g., hematoxylin and eosin (H&E)) to accentuate certain features, such as nuclei and tissue structure. These staining procedures help pathologists identify critical features representing both the type and stage of the neoplasm. Although this gold-standard method is well trusted, it is well-known to be time consuming and subjective, where even experienced pathologists specializing in this field will often disagree on the severity of cancer present in any one tissue [3].

*Corresponding author contact: maywang@bme.gatech.edu.

Driven by the fast advancement in high throughput and high-resolution digital slide scanners, computational approaches are becoming viable options in the analysis of these stained tissue images [4]. Machine Learning (ML) techniques such as Deep Learning (DL) methods have also been increasingly applied to challenges dealing with histological images. For example, a semiautomated computational approach may help reduce inter-pathologist variability in clinical practice. Clinicians may also benefit from the reduction in workload currently required for certain tasks, like cell counting [5]. Already such methods have proven to be successful in areas such as dermoscopy and dermatology [6]. Recently, the 2018 ICIAR presents the BACH challenge aiming to accurately classify breast tissue into cancer subtypes (i.e. benign vs in situ) using ML: Part A consists of selected tiles of histopathological images of H&E stained breast cancer biopsy, and Part B consists of the whole-slide image (WSI) of the same cohort [7].

In our work, we focus on Part A that deals with images tiled from whole slide images and with the same size (2048 by 1536 pixels) from four different classes. We aim to develop a novel classification scheme to improve the prediction of these images for accurate diagnosis of breast cancer. The dataset consisted of four hundred images in total, consisting of one hundred normal, invasive, in situ, and benign breast tissue images respectively. We propose a pipeline with modularity while enabling interpretability of the important features by DL approaches. Interpretability is important because it not only produces good results but also allows clinicians to gain new information about the data [8]. The modularity makes it easier to improve performance and interpretability. In the following sessions, we will first provide literature critique that inspired the development of the pipeline, followed by a study of the BACH data for clinical diagnosis decision support using various fusion techniques.

2. Literature Survey

2.1 Color Normalization

In histology images, the most common first step in preprocessing is color normalization. There have been many techniques utilized over the years to make sure that image classification is determined by substantial feature differences rather than color variation. The Reinhard method is a linear transformation for color normalization that uses the color mean and standard deviation of a source and target image to make images have similar stain intensities [9]. First, the images are converted into the LAB color space (L stands for lightness, and the A & B channels are transformations of yellow-blue and red-green, respectively). The mean of each channel is zeroed, and then the standard deviation of each channel is scaled by the standard deviation of the reference image. Then the image is converted back from LAB to red-green-blue (RGB) color space to produce a color-normalized image. This approach treats color variation within an image as constant. Issues arise when there is too much variation between the images, such as an image with a lot of non-tissue space, or artifacts in the images such as sharpie marks or folded tissue. Also, it normalizes the entire dataset to one reference image. Thus, the selection of the reference image can influence classification accuracy as no one image is representative of the entire dataset.

Another approach involves the implementation of color deconvolution, a process that aims to separate an RGB image into channels that represent specific color stains in the image. This method is shown by Macenko et al. 2009 and Ruifrok and Johnston 2001 [10,11]. To properly employ this method, a stain matrix needs to be estimated. This matrix is case specific and aims at separating each RGB pixel into its stain channels. In the case of an H&E image, this would separate any RGB pixel into an H channel, an E channel, and a channel representing everything else. Because it does not assume the color is the same across the image, this method performs better than Reinhard normalization. However, it suffers from inconsistencies in estimating the stain matrix. Thus, several techniques have been developed over the years for estimating this stain matrix in a reliable and scalable manner [11,12]. For example, the issue of the stain matrix estimation was further investigated by Khan et al. 2014 which introduced a machine learning classification to estimate a stain matrix for individual images using a nonlinear approach to map source to target stain channels [13].

With the Oct-tree quantization approach, a set of 255 color histogram prototypes were generated from stain color descriptors for each image [14]. A stain color descriptor (SCD) is calculated by finding the mean and covariance of all the histograms of the dataset, then each of the histograms is projected into a lowdimensional space. Annotations are required at the pixel level, so a relevant vector machine (RVM) can be trained using the RGB features individually as well as the SCD. The classifier outputs probabilistic information used to calculate the contribution of each pixel to a stain or background. This method learns from the entire dataset when estimating the stain matrix, so it is even more robust and versatile than the Macenko method, which uses a manual selection or a pre-defined algorithm. However, it requires pixel-level labeling to train the classifier, making it challenging to scale and resulting in variation based on the labeling fidelity.

2.2 Deep Learners

Inception V3 is a convolutional neural network developed by Google and was the runner up of The ImageNet Large Scale Visual Recognition Challenge (ILSVRC). One of the hallmarks of this model is its inception modules. Within these modules, there are many parallel paths where the image is passed through with many smaller filters. The output of all these paths is assembled at the end. These inception modules are passed through multiple times during training. Many papers, such as Yi et al. 2017 and Coudray et al. 2018, have demonstrated an 85% or higher accuracy of breast cancer tissue classification using this architecture [15,6]. Another architecture developed by Google is that of Inception ResNet V2. Szegedy et al. 2017 used a model to combine the Inception model with the ResNet model. The main difference is the introduction of the ResNet model, which allows the image to bypass blocks in the architecture. The residual block allows the image to travel deeper into the network with less effort [16].

One pitfall of these networks is that they take a long time to train and require a lot of data to train properly. Thus, transfer learning is proposed by removing its classification layer and the fully connected layer right after the last average pooling layer. It allows new layers to be added on top and the weights of the fully trained model to be “transferred” into use for classification of new data. Layers of the trained network can also be unfrozen, meaning that

their weights can be updated during the new training [17, 18]. Transfer learning has been previously utilized in the classification of breast cancer images. Sharma et al. used breast cancer images from the Breast Cancer Histopathological Database (BreakHis) with several DL architectures that were pre-trained [19]. Depending on the architecture used, transfer learning showed better performance than simply training a model from scratch (random initialization of model weights) [20]. The BACH 2018 grand challenge dataset has been used in conjunction with transfer learning as well. Vesal et al. 2018 used this dataset along with Inception V3 and ResNet-50 pre-trained architectures to perform the same classifications as our model, obtaining above 90% test accuracy [21]. Other works developed by the authors have shown various other neural network approaches that show promise in histopathology imaging challenges [22]. Based on these findings, we decide to use transfer learning because of its decreased training time as well as its demonstrated accuracy when classifying breast cancer histology images.

Another group who tackled our question in their own way was Nazeri et al. 2018. They developed their own classification pipeline to use on the BACH dataset based on CNNs and an ensemble approach. This pipeline was made up of two components. One was a patch-wise CNN, and the other was an image-wise CNN. The patch-wise CNN breaks each image into smaller patches to pass to the network. The patch-wise CNN works by utilizing a series of 3x3 convolutional layers (CL) followed by a pooling layer which would cause the number of channels doubling after each down-sampling, a 2x2 CL with a stride of two, and a 1x1 CL which obtained the spatial average of feature maps. The vector is finally run through a softmax layer. Twelve feature maps with dimensions 64x64xC, with C being a hyper-parameter that controls the depth, are fed into the imagewise CNN. The image-wise CNN follows a similar pattern series of 3x3, followed by 2x2 with a stride of 2, and 1x1 convolutional layers. One interesting note about this model was that no color normalization was performed, but image augmentation was added to remove color variations. This model was able to obtain accuracies of 93.75% with their best CNN and 95% with an ensemble method [23]. This to our knowledge is the highest accuracy obtained when running classification on the ICIAR BACH dataset, and will be used as a comparison.

2.3 Fusion

If one classifier is not enough to get the accuracy needed, then fusing two may be able to achieve what is needed. The process of running the same data set through distinct classifiers and subsequently fusing the results to produce a single output has been a common technique used to improve classification accuracy for several decades. The general idea is that multiple independent classifiers can be developed for one problem, with the goal being to use a set of given features and an architecture (the model) to classify the data appropriately. However, classifiers may perform similarly, even when using very different feature information. Thus, finding optimal ways to combine classifiers and features may result in performance better than any single classifier alone. This “optimal” combining scheme can be difficult to pin down and may vary depending on the task at hand. Previous work has shown theoretical approaches to finding optimal schemes. However, these early works have not used histological data and primarily focused on theoretical approaches [24,25]. Additionally,

these early works did not focus on deep learning, mostly because research in that area was not as prevalent during that time.

Recent work by Masoud et al. 2017 hypothesized that even well-performing neural networks would miss key information due to their specific architecture [26–28]. Specifically, they used a dataset that consisted of 700 images of African art artifacts from 7 different African tribes. A support vector machine (SVM) and a CNN were trained separately and once trained the models were combined using a fusion algorithm to provide a fusion prediction. The fusion method was based on an optimal score. It is defined by the authors as the minimum probability score that achieves zero error classification for all the predicted labels that have prediction accuracy greater than or equal to the optimal one.

If both models gave the same prediction, then that was the fusion prediction given. If two models disagreed, then use the predicted label of the classifier that has a prediction accuracy greater than or equal to its optimal score. If both models meet that criteria, then take the predicted label of the model that has a higher validation score. One downside of this method is that it needs two models trained in parallel in which neither of the models can contribute to the other during training.

Kan et al. 2019 proposed another fusion method. This one fixes the problem of the Masoud paper where the fusion happens during the training and not just at the end [29]. This is done by a unit the authors call Fusion-net, which is comprised of a converter and a merger. The main novelty of this method is that the features used in SVM (or shallow learner) can be back-propagated into the CNN, which means they will help update the weights. Three different converters were tested: an extreme learning machine, an autoencoder, and a fully connected network. Out of these three converters, the fully connected network was found to suppress useless information and extracted the most useful information. The authors also proposed a unique loss function called class-metric loss. This new loss function, combined with the standard softmax loss, was shown to have the highest performance improvement. The class-metric loss contains the structural information, while the softmax loss contains the label information. Similarly, to the Masoud paper, the dataset was not over histology images, but the Stanford Online Products' and In-shop Clothes Retrieval datasets. Because these new techniques were implemented on non-medical datasets containing thousands of images, we decided to move forward with using a more straightforward fusion technique, but this method would be a potential next step.

3. Methodology and System Design

3.1 Data

We used the ICIAR 2018 Grand Challenge on Breast Cancer Histology (BACH) images. We focused on part A of the challenge: the classification of H&E images into four classes: normal, benign, *in situ* carcinoma, and invasive carcinoma. One hundred images for each class were provided, totaling 400 histology images for the whole dataset [7]. The data was divided into 80:20 training and testing split while keeping the contribution of each class the same for training and testing. The testing set was not used until after training when

implementing fusion. During training, we used 5-fold cross-validation to tune hyperparameters, using an 80:20 split for training and validation.

3.2 System Design

Our design is a parallel training approach in which we use a shallow learner and a deep learner to train separately (Figure 1). The trained models are then fused to provide a final prediction on the data. The "top" path keeps the original size of the images, obtains hand-crafted features, and runs an SVM over those features (see section 3.5). The "bottom" path down-samples the images and runs that through a convolutional neural network. Fusion algorithms are implemented at the end, which gives the final prediction of which class each image is a member of. All these steps are further explained in the following sections.

3.3 Color Normalization

As stated earlier, color normalization is a critical step for image classification. The difference in hues between images of the same class should not be a feature that is different between them. For color normalization, the Reinhard method was used [9]; the Reinhard method works for the ICIAR BACH dataset and is easily implemented. Manual review of the images after color normalization only identified one instance that required manual modification. Our previous experience with Reinhard normalization suggests this is not the usual case, and more robust techniques may be considered for other datasets (Figure 2). We selected a reference image from the training dataset at random, shown in Figure 2. We re-ran our analysis with different reference images but saw no result variations during the training of the neural network or shallow learner (not shown).

3.4 Image Down-sampling

The original resolution given for the images was 2048 x 1536 pixels, far larger than most pre-trained image models (and GPUs) can accept without modification. Each image was down-sampled to 244 x 244 before being passed into the neural network. As seen in Figure 3, this makes sure that the entire image is still being passed through to the network, at the cost of losing potentially useful information content during the compression process.

3.5 Shallow Learners

As seen in Kan et al. 2017 and Masoud et al. 2017, image descriptors can be obtained using a variety of methods. An image descriptor is an n-sized vector describing essential aspects of an image, such as blob, corners, or color moments [30]. We follow the approach detailed in Masoud et al. 2017 to obtain SURF (Speeded Up Robust Features) descriptors for our images. That work also obtained HOG descriptors (Histogram of Oriented Gradients) to pair up with their SURF descriptors. We chose to skip the HOG features because orientation in histology images should not be an important feature, as tissue can be flipped in any directory during slide preparation. SURF is a variation on scale invariant feature transform (SIFT) descriptors, which captures local features in an image. SIFT and SURF are mostly color invariant, with key points being recognized by high contrast regions. In histology images SURF tends to recognize cellular-like objects, capturing information regarding cell density and nuclei morphology (Figure 4). We hypothesize that this captured information would lead

to reliable classification between the image classes, since it would capture the key difference between normal and abnormal cells [26].

For each image, we have m number of n-sized feature vectors, where m denotes the number of SURF objects captured for that image. The value of m will vary from image to image based on its content, but the n-sized feature vector is inherent to the SURF algorithm (normally 32 or 64, 64 in this work). To be able to use SURF descriptors in a traditional shallow learner, we first need to transform them into a single feature space, such that each image has the same number of features. To accomplish this transformation, an encoding method, based on the popular bag of words method used for natural language processing, was used: the bag of visual words (BOVW) [31]. Briefly, the image descriptors are extracted from the entire training set. Then a minibatch KMeans model for n clusters is trained. All vectors from all training images must be used. Mini-batch KMeans is used to avoid computational memory limits during training of the KMeans model. After training, we create a histogram of "words" for each image. We extract the SURF features for each image again and assign each SURF descriptor to a cluster center from the trained KMeans model. This creates a histogram of words for each image, with the size being determined by the number of clusters chosen for the KMeans model. The counts of the histogram are used as the feature space for classification using a shallow learner.

For our analysis, we used a support vector machine (SVM) classifier, due to it being able to train effectively with smaller datasets. Best model selection was guided by the accuracy from 5-fold cross-validation during training. Hyperparameter tuning was performed in an exhaustive manner by training on multiple combinations of SVM hyperparameters: penalty parameter C, kernel (radial basis function used), and kernel coefficient gamma. Apart from these parameters, we also tested the accuracy from using varying cluster numbers (referred to as words) in the KMeans model (100 to 10,000) and the Hessian Threshold (100 to 3,000). The Hessian Threshold affects how many descriptors are extracted from an image using the SURF algorithm. Low values such as 100 lead to tens of thousands of possible descriptors while high values such as 10,000 can lead to only a few hundred. Color normalization and orientation of descriptors were also tested.

3.6 Deep Learners

For deep learning packages, we used Tensorflow and Keras [32, 33]. The Keras package contains Inception V3 and Inception ResNet V2 pre-trained, both models can be loaded with the weights from ImageNet. The last five layers of the pre-trained models were removed, and a two-layer fully connected unit was added at the end. The first layer of the unit contained 1024 nodes with ReLU activation and L2 regularization, and the second contained 4 nodes, one for each classification, and used SoftMax as its decision layer. ReLU activation function is defined as $f(x) = \max(0,x)$; this means that when $x < 0$ it outputs 0, but when $x >= 0$ then it follows a linear function [17]. L2 regularization is used to decrease the loss of the learner by penalizing complex models [18]. In this model, we set the penalty equal to the learning rate. As specified earlier, the images were down-sampled to 244 x 244.

To perform hyperparameter tuning, we used 5-fold cross validation by randomly splitting the training data into 80% training and 20% validation in a stratified manner (class balance was

kept). We repeated the entire training for every fold and averaged the prediction accuracy on the validation sets. During the prediction of the validation data, we also recorded the classifier accuracy and the per-class accuracy for each trained fold for fusion schemes. The hyperparameters that were tuned during training were learning rate, momentum with stochastic gradient descent (SGD), L2 regularization, and batch size. The structure of deep learners is shown in Figure 5.

3.7 Fusion

$$CCA = \frac{\sum \text{prob of correct prediction } i}{\text{total \# correct predictions}}$$

$$CCA_x = \frac{\sum \text{prob of correct prediction i for class x}}{total\ \#\ correct\ predictions\ for\ class\ x}$$

Fusion 1: equal classifier algorithm

1. Choose model with highest predicted class probability

Fusion 2: best class classifier algorithm

1. Choose model with highest predicted class model accuracy

Fusion 3: scaled to distance to CCA algorithm

1. For each model: calculate the difference between predicted class probability and model CCA

2. Choose model with highest difference calculated in 1

Fusion 4: scaled to distance to CCAx algorithm

1. For each model: calculate the difference between predicted class probability and CCA for predicted class

2. Choose model with highest difference calculated in 1

Like the Masoud et al. 2017 paper, we implemented the fusion algorithm, but we developed our own algorithms used in fusion. Four fusion algorithms were proposed. All algorithms start the same: if the classifiers predict the same class, then that prediction is taken. When they disagree, then the different algorithms are applied to decide on the final prediction. To accomplish this, some key statistics were obtained during training and validation evaluation. Validation accuracy was calculated as the average of prediction accuracy on the 5 folds during training. This was done at the classifier level and the class level. A key term is introduced here, the correct classification accuracy (CCA). This is calculated by summing the model prediction probability (a decimal from 0 to 1) for every correct prediction on the

validation set. This value is divided by the number of correct predictions (averaging). The CCA is calculated for the model as a whole and each class individually.

Of note, SVMs do not inherently provide probability prediction outputs, which are essential for our fusion method. The SVM module (sklearn.svc.SVC) implemented in Python's scikit-learn package provides probability outputs for each prediction, implementing the method described in Plat et al. 1999 [34]. More sophisticated fusion schemes, such as those investigated in Kuncheva et al. [25], rely on theoretical approaches to narrow down a general "best" fusion scheme. However, the goal of our approach was to keep it simple to provide easy interpretability at the start. Future work would be dedicated to applying more complex fusion approaches.

4. Results

Figure 6 shows the results of validation accuracy during training the SVM. The highest accuracy achieved happened using a Hessian Threshold of 100, 500 words, and no color normalization used. SURF allows the calculation of orientation of the descriptors, which significantly increases the computational time but is critical in providing meaningful descriptors. The highest validation accuracy achieved by the SVM was 78%.

Two best models were achieved using the deep learners, one using the Inception V3 architecture and the other using the Inception ResNet V2 architecture. For each model, we tested several hyperparameters: learning rate (LR), momentum used during stochastic gradient descent (M), and batch size (BS). The results of these various combinations are shown in Figures 7 and Figure 8 for Inception V3 and Inception ResNet V2, respectively. Best validation accuracy for Inception V3 was 83% and 84% for ResNet V2. Validation accuracy was calculated by averaging the accuracy in the last ten epochs of training.

Fusion algorithm results can be seen in Figure 9, 10, 11, and 12. The results are shown for accuracy on the testing data set for both the four fusion methods and the learners. We fused the shallow learner with each deep learning model and fused the two deep learners. Finally, we implemented fusion with all three learners. In each fusion, we achieved improvements in testing accuracy. Fusion 1 and fusion 3 gave improvements in every combination of learners, achieving the highest when fusing the shallow learner and ResNet V2 (92%). The testing accuracies on the shallow learner was 79%, Inception V3 was 81%, and Inception ResNet V2 was 79%.

5. Discussion

The ICIAR BACH challenge has occurred for a few years, and 2018 challenge winners have been announced. There was a tie for first place, and their accuracy was 87%, third place was 86%, fourth place was 84%, and fifth place was another tie with 83% accuracy. Based on those results, our highest testing accuracy of 92% would have been competitive. It would be interesting to see how our model would have performed on the dataset the challenge ultimately tested on. Comparing our method with the winners, Chennamsetty et al. group 216 was running a pre-trained Resnet-101 and two Densenet-161 in ensemble and Kwok group 248 was running a pre-trained Inception Resnet v2. Group 248 was also attempting

the part B of the ICIAR BACH challenge which uses whole slide images (WSI), and they used these WSI in their training as well. One interesting thing about each of these models is that neither of them performed color normalization while we did [7]. Comparing to Nazeri et al. 2018, who achieved 95% accuracy with an ensemble method, our fusion performed comparably [23].

One downside of the fusion algorithms is that they are difficult to develop and maybe even harder to validate. This can be seen by the Fusion 2 algorithm, which performed well in some fusions but worse in others. It even performed worse than the individual learners when fusing the two Inception models. However, it is important to note that some fusion algorithm showed great promise, such as algorithm 3. Fusion 3 performed 1%, 13%, 4%, and 5% better than the best single learner in each fusion. This is similar to results obtained by Masoud et al., which showed an approximate 5% increase with fusion. We hypothesize that Fusion 3 algorithm showed the greatest reliability because it considers the class accuracies for each model. This can be thought as combining models specifically tailored to recognize one or more classes better than others; akin to using complementary models to fill in the weaknesses of each other.

Interesting to note is that the closer the individual learners were to each other, the best the fusion worked. This can be seen by the high increase in accuracy when fusing the ResNet and shallow learner (both had 79% accuracy). Surprisingly, adding Inception V3 in the three-model fusion resulted in decreased accuracy, though still higher than individual learners.

6. Conclusions

As shown in the results, the difference between the testing accuracy of the fusion algorithms and Inception v3 was significant. Even though we took a different approach compared to Masoud et al.'s original method, we were still able to reach favorable results. This supports the hypothesis that we can adopt a method shown to work on images unrelated to medicine, to work with these images with appropriate modification to the approach.

To further validate our results, we would need to test these algorithms on larger datasets. When dealing with small datasets, the results can be skewed depending on what subset of the data ends up in the testing set. This can be remedied by increasing the data samples or alternatively averaging the results from training and testing on different splits.

One of the benefits of the pipeline created is its modular form. We can replace nearly every aspect of the pipeline and see what results we can get. The Speeded Up Robust Features (SURF) image descriptors can be switched with another type of image descriptor, such as color moments or nuclear segmentation values. Bag of Visual Words (BOVW) can be changed with another encoding method, such as the improved fisher kernel [35]. The shallow learners can be changed as well as the deep learners, and other color normalization schemes attempted. This allows for the flexibility of pursuing further studies using this workflow. Ultimately, we believe that this method would allow researchers to be able to use the power of deep learning architectures while also adding interpretable features.

ACKNOWLEDGMENTS

This research was possible due to the support provided by the David Gutman research group from Emory University's Neurology department. With this support, we were able to access several computers with high-end GPUs, such as Titan V, which allowed running deep learning models efficiently. The authors would also like to acknowledge the Giglio Cancer Research fund, Petit Institute Faculty Fellow Fund, and Carol Ann and David Flanagan Faculty Fellow Fund to Professor May D. Wang.

REFERENCES

[1]. Ferlay Jacques, Soerjomataram Isabelle, Dikshit Rajesh, Eser Sultan, Mathers Colin, Rebelo Marise, Parkin Donald Maxwell, Forman David, and Bray Freddie. 2015 Cancer incidence and mortality worldwide: sources, methods and major patterns in GLOBOCAN 2012. International journal of cancer. Journal international du cancer 136, 5: E359–86. https://doi.org/ [PubMed: 25220842]

[2]. Siegel Rebecca L., Miller Kimberly D., and Jemal Ahmedin. 2019 Cancer statistics, 2019. CA: a cancer journal for clinicians 69, 1: 7–34. 10.3322/caac.21551 [PubMed: 30620402]

[3]. Orlando Laura, Viale Giuseppe, Bria Emilio, Eufemia Stefania Lutrino Isabella Sperduti, Carbognin Luisa, Schiavone Paola, Quaranta Annamaria, Fedele Palma, Caliolo Chiara, Calvani Nicola, Criscuolo Mario, and Cinieri Saverio. 2016 Discordance in pathology report after central pathology review: Implications for breast cancer adjuvant treatment. Breast 30: 151–155. 10.1016/j.breast.2016.09.015 [PubMed: 27750105]

[4]. Zarella Mark D., Bowman Douglas, Aeffner Famke, Farahani Navid, Xthona Albert, Absar Syeda Fatima, Parwani Anil, Bui Marilyn, and Hartman Douglas J.. 2018 A Practical Guide to Whole Slide Imaging: A White Paper From the Digital Pathology Association. Archives of pathology & laboratory medicine. 10.5858/arpa.2018-0343-RA

[5]. Kothari Sonal, Chaudry Qaiser, Wang May D, "Automated cell counting and cluster segmentation using concavity detection and ellipse fitting techniques", 2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro, pp. 795–798

[6]. Coudray Nicolas, Ocampo Paolo Santiago, Sakellaropoulos Theodore, Narula Navneet, Snuderl Matija, Fenyö David, Moreira Andre L., Razavian Narges, and Tsirigos Aristotelis. 2018 Classification and mutation prediction from non–small cell lung cancer histopathology images using deep learning. Nature medicine 24, 10: 1559–1567. 10.1038/s41591-018-0177-5

[7]. Aresta Guilherme, Araújo Teresa, Kwok Scotty, Chennamsetty Sai Saketh, Safwan Mohammed, Alex Varghese, Marami Bahram. 2019 BACH: Grand Challenge on Breast Cancer Histology Images. Medical image analysis 56 (August 2019), 122–139. DOI: 10.1016/j.media.2019.05.010 [PubMed: 31226662]

[8]. Kothari Sonal, Phan John H, Osunkoya Adeboye O, Wang May D, "Biological interpretation of morphological patterns in histopathological whole-slide images", 2012 International Conference Proceedings of the ACM Conference on Bioinformatics, Computational Biology and Biomedicine (ACMBCB2012), pp. 218–225

[9]. Reinhard E, Adhikhmin M, Gooch B, and Shirley P. 2001 Color transfer between images. IEEE Computer Graphics and Applications 21, 34–41. 10.1109/38.946629

[10]. Macenko M, Niethammer M, Marron JS, Borland D, Woosley JT, Guan Xiaojun, Schmitt C, and Thomas NE. 2009 A method for normalizing histology slides for quantitative analysis. In 2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro, 1107–1110. 10.1109/ISBI.2009.5193250

[11]. Ruifrok AC and Johnston DA. 2001 Quantification of histochemical staining by color deconvolution. Analytical and quantitative cytology and histology / the International Academy of Cytology [and] American Society of Cytology 23, 4: 291–299. Retrieved from https://www.ncbi.nlm.nih.gov/pubmed/11531144

[12]. Alsubaie Najah, Trahearn Nicholas, Ahmed Raza Shan E., Snead David, and Rajpoot Nasir M.. 2017 Stain Deconvolution Using Statistical Analysis of Multi-Resolution Stain Colour Representation. PloS one 12, 1: e0169875 10.1371/journal.pone.0169875 [PubMed: 28076381]

[13]. Khan Adnan Mujahid, Rajpoot Nasir, Treanor Darren, and Magee Derek. 2014 A nonlinear mapping approach to stain normalization in digital histopathology images using image-specific color deconvolution. IEEE transactions on bio-medical engineering 61, 6: 1729–1738. [PubMed: 24845283]

[14]. Gervautz M and Purgathofer W. 1988 A Simple Method for Color Quantization: Octree Quantization In New Trends in Computer Graphics, 219–231. 10.1007/978-3-642-83492-9_20

[15]. Yi Darvin, Sawyer Rebecca Lynn, David Cohn Iii, Dunnmon Jared, Lam Carson, Xiao Xuerong, and Rubin Daniel. 2017 Optimizing and Visualizing Deep Learning for Benign/Malignant Classification in Breast Tumors. arXiv [cs.CV]. Retrieved from http://arxiv.org/abs/1705.06362

[16]. Szegedy Christian, Ioffe Sergey, Vanhoucke Vincent, and Alemi Alexander A.. 2017 Inception-v4, inception-resnet and the impact of residual connections on learning. In Thirty-First AAAI Conference on Artificial Intelligence Retrieved from https://www.aaai.org/ocs/index.php/AAAI/AAAI17/paper/viewPaper/14806

[17]. Agarap Abien Fred. 2018 Deep Learning using Rectified Linear Units (ReLU). arXiv [cs.NE]. Retrieved from http://arxiv.org/abs/1803.08375

[18]. Cortes Corinna, Mohri Mehryar, and Rostamizadeh Afshin. 2012 L2 Regularization for Learning Kernels. arXiv [cs.LG]. Retrieved from http://arxiv.org/abs/1205.2653

[19]. Spanhol F, Oliveira LS, Petitjean C, Heutte L, A Dataset for Breast Cancer Histopathological Image Classification, IEEE Transactions on Biomedical Engineering (TBME), 63(7):1455–1462, 2016. [pdf]

[20]. Sharma Shallu and Mehra Rajesh. 2018 Breast cancer histology images classification: Training from scratch or transfer learning? ICT Express 4, 4: 247–254. 10.1016/j.icte.2018.10.007

[21]. Vesal Sulaiman, Ravikumar Nishant, Davari Amirabbas, Ellmann Stephan, and Maier Andreas. 2018 Classification of breast cancer histology images using transfer learning. arXiv [cs.CV]. Retrieved from http://arxiv.org/abs/1802.09424

[22]. Tong L, Sha Y, and Wang MD 2019 “Improving Classification of Breast Cancer by Utilizing the Image Pyramids of Whole-Slide Imaging and Multi-Scale Convolutional Neural Networks”. IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC) 1 (July 2019) 696–703. DOI: 10.1109/COMPSAC.2019.00105

[23]. Nazeri Kamyar, Aminpour Azad, and Ebrahimi Mehran. 2018 Two-Stage Convolutional Neural Network for Breast Cancer Histology Image Classification In Image Analysis and Recognition, 717–726.

[24]. Kittler J, Hatef M, Duin RPW, and Matas J. 1998 On combining classifiers. IEEE transactions on pattern analysis and machine intelligence 20, 3: 226–239. 10.1109/34.667881

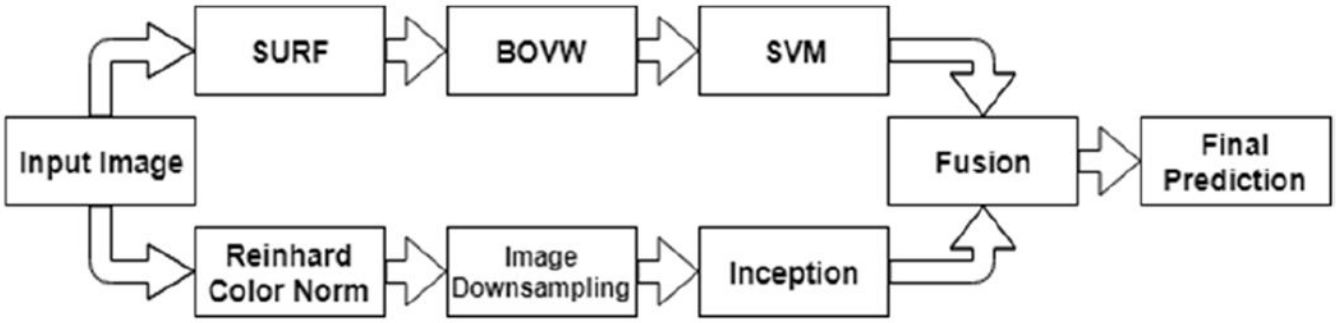

Figure 1:
Architecture of our pipeline. Images are passed through color normalization; the top path does not use color-normalized images because SURF is color invariant.

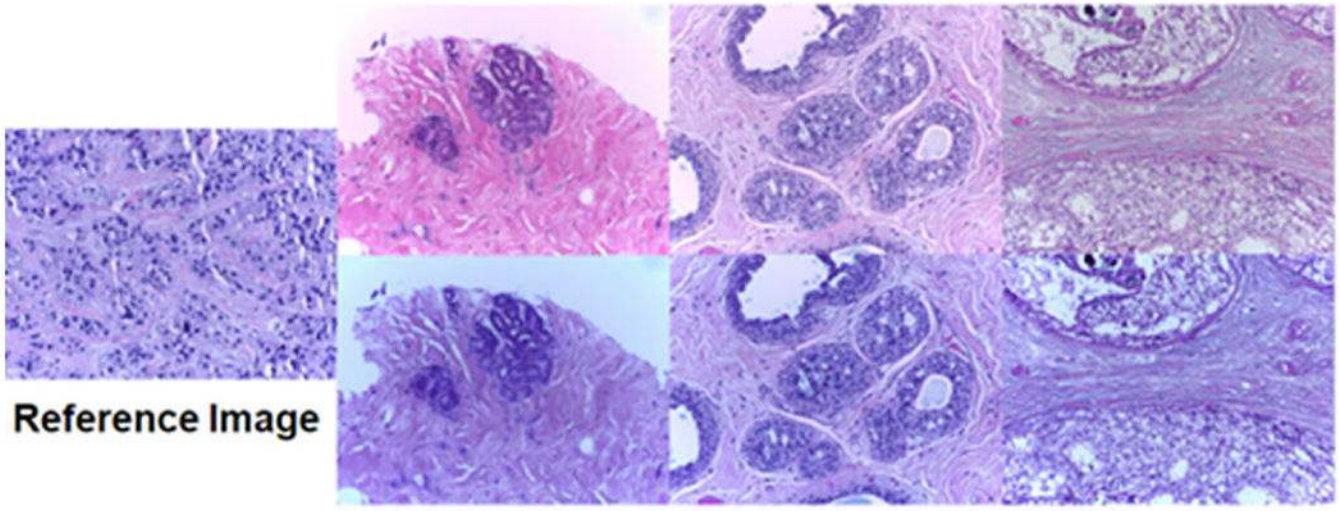

Figure 2:
Color normalization example. The left image is the reference image used in this work. Right: top row are original "raw" images, bottom are the same images, but color normalized to reference image.

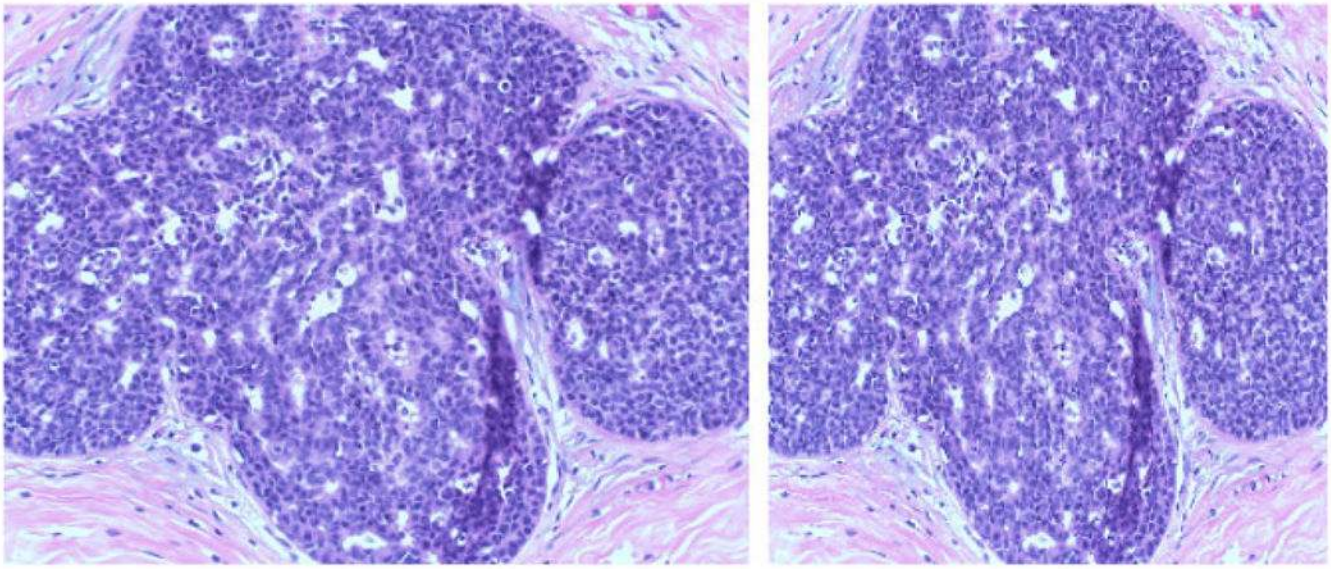

Figure 3:
Down-sampling helps run the neural networks efficiently, but do cause changes in the images as seen above (left image is the original and right is down-sampled).

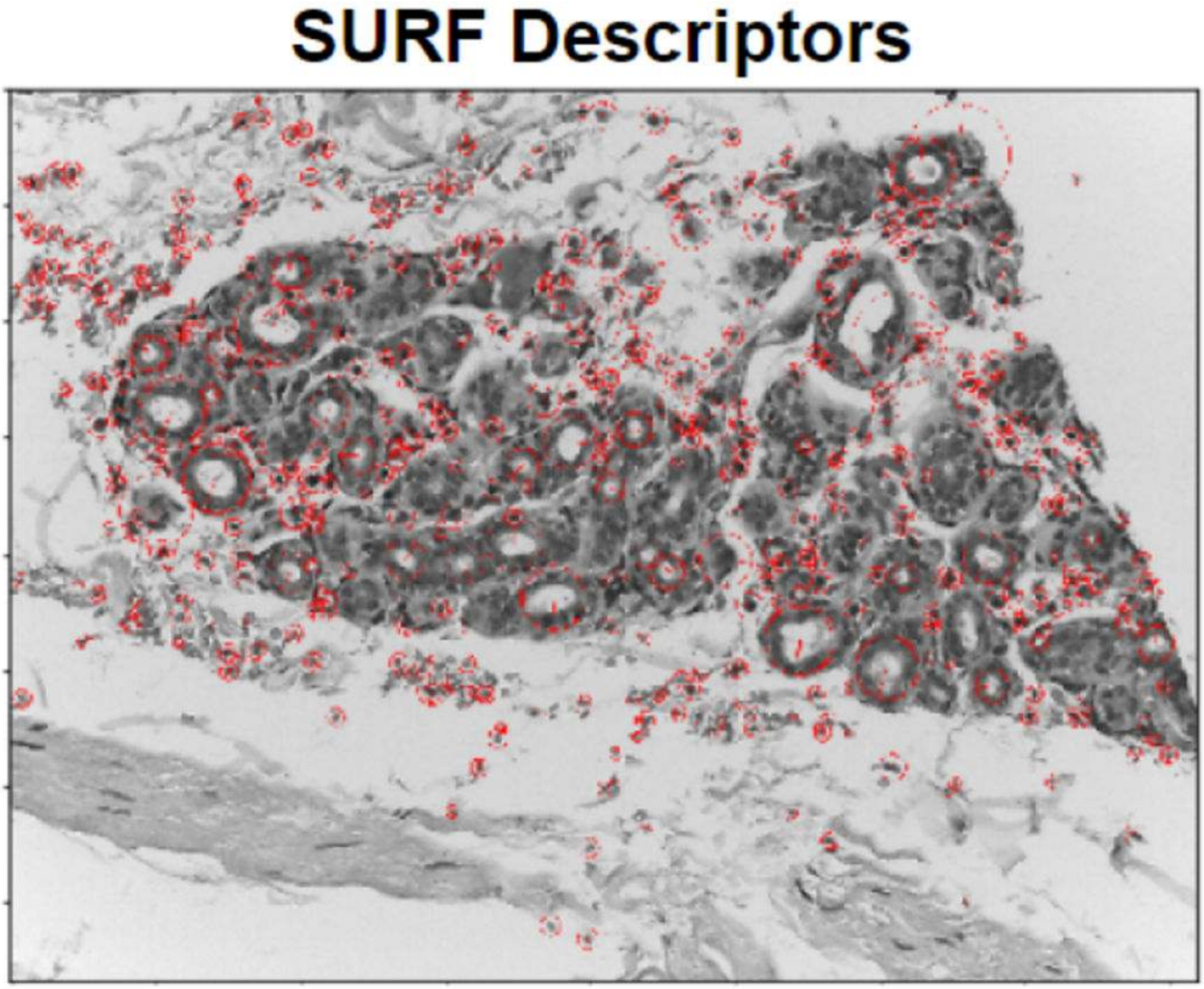

Figure 4:
SURF descriptors (red) drawn on a histology image. Note how the descriptors capture cell-like structures in the images. SURF is color invariant and thus the image is shown in grayscale.

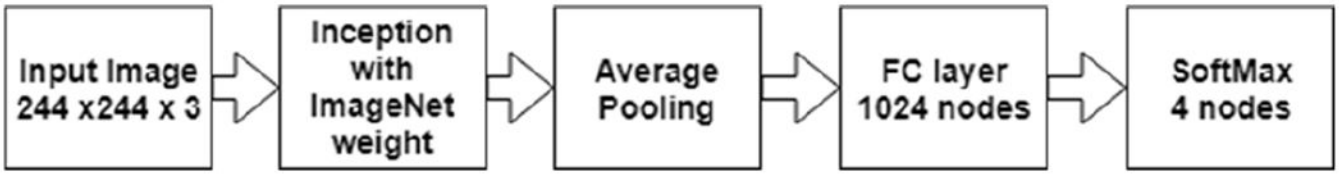

Figure 5:

The structure of deep learners. Both Inception V3 and Inception ResNet V2 were loaded with ImageNet weights. After an average pooling, a two-layer fully connected unit was used for weight updates based on the histopathology images and classification.

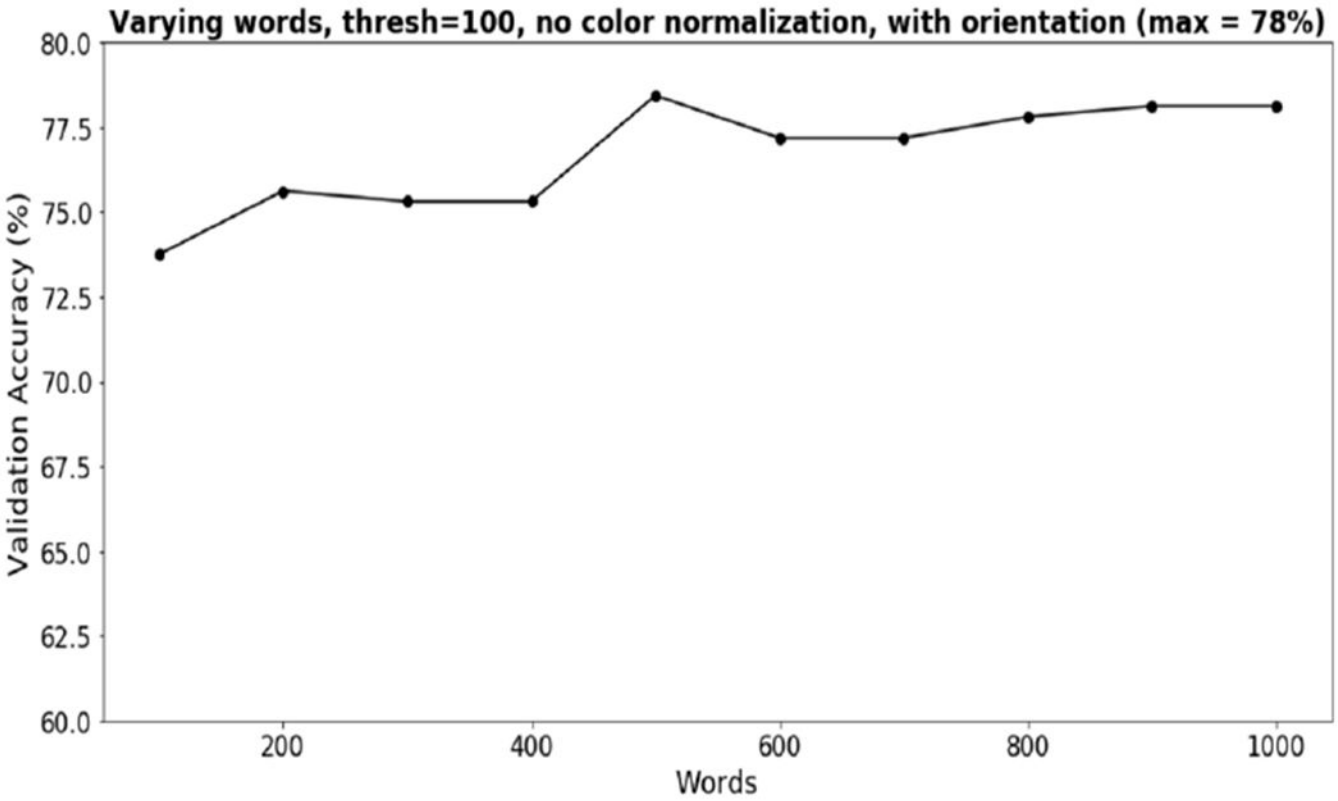

Figure 6:
Cross-validation accuracy achieved during SVM training with different number of words used.

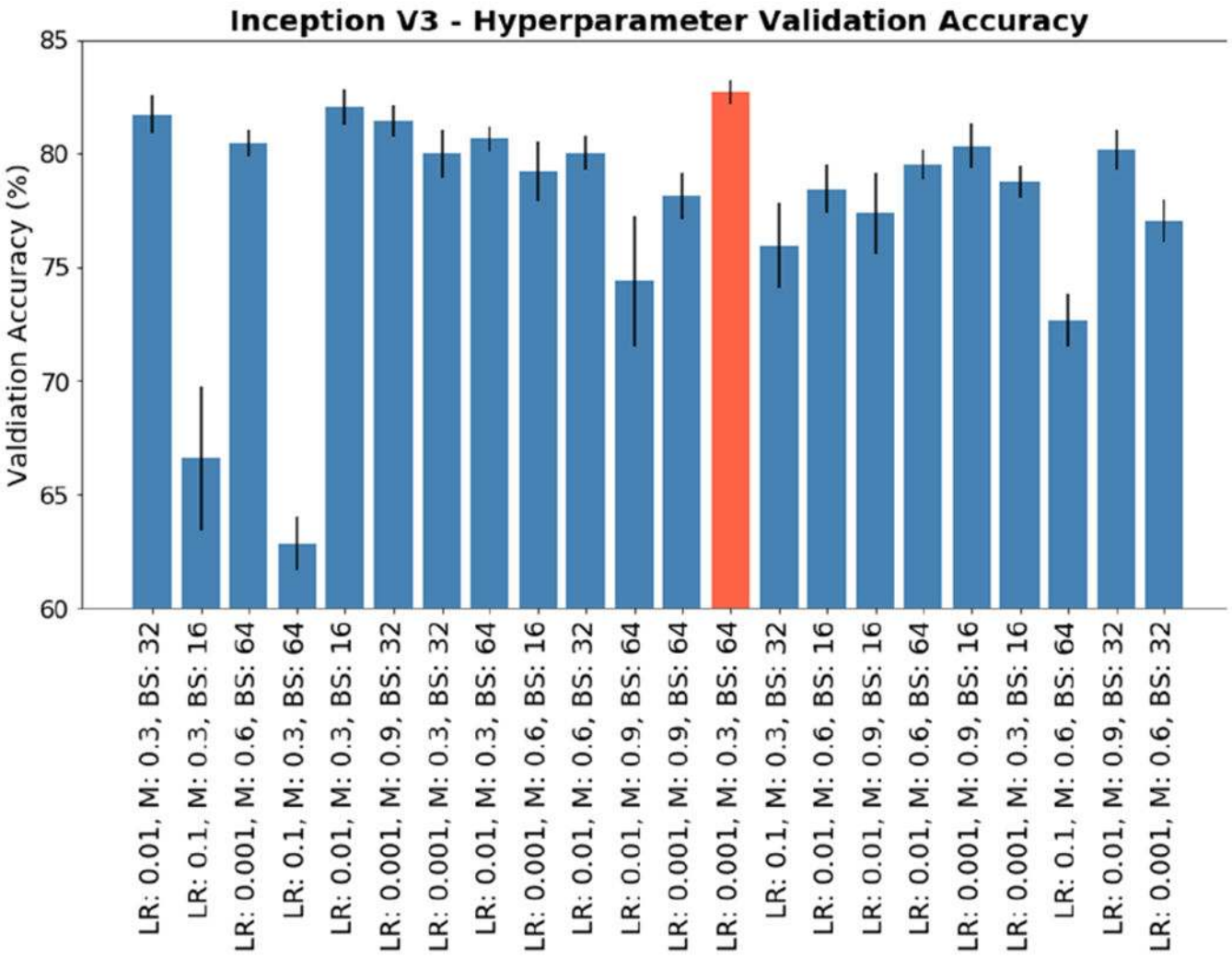

Figure 7:
Cross-validation accuracy achieved for Inception V3. Standard error bars are shown from taking the accuracy during the last ten epochs of training.

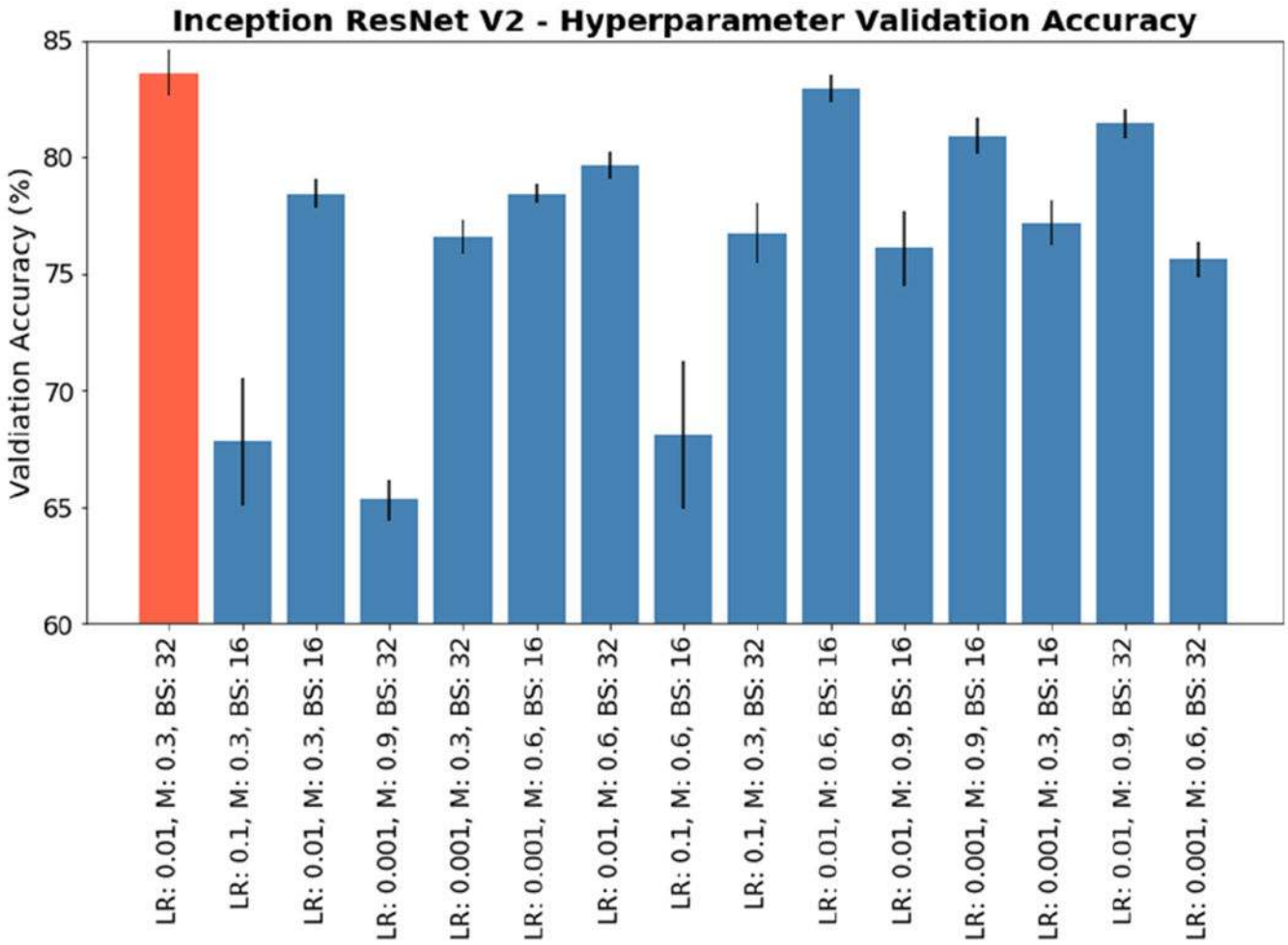

Figure 8:
Cross-validation accuracy achieved for Inception ResNet V2. Standard error bars are shown from taking the accuracy during the last ten epochs of training.

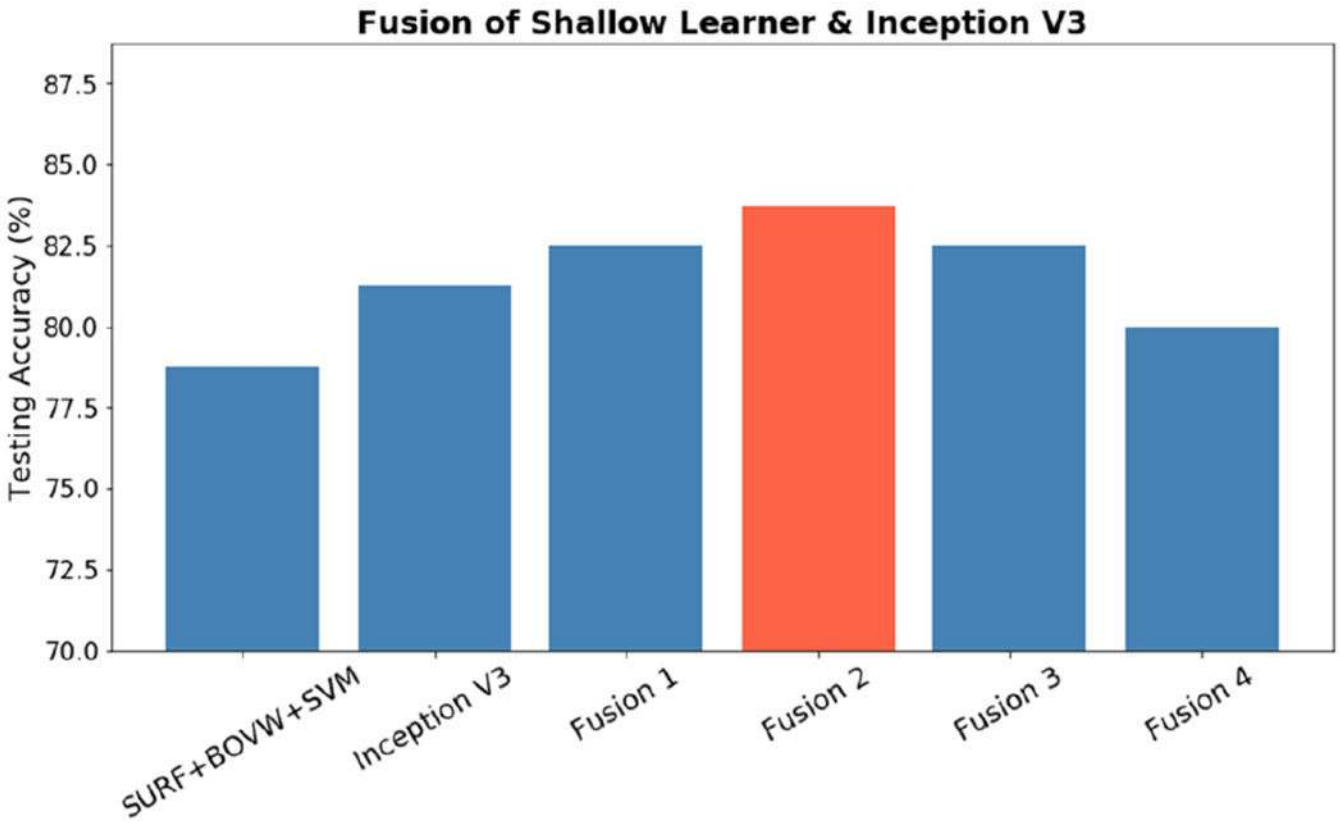

Figure 9:
Test accuracy for fusing shallow learner and Inception V3. The orange bar is the best result.

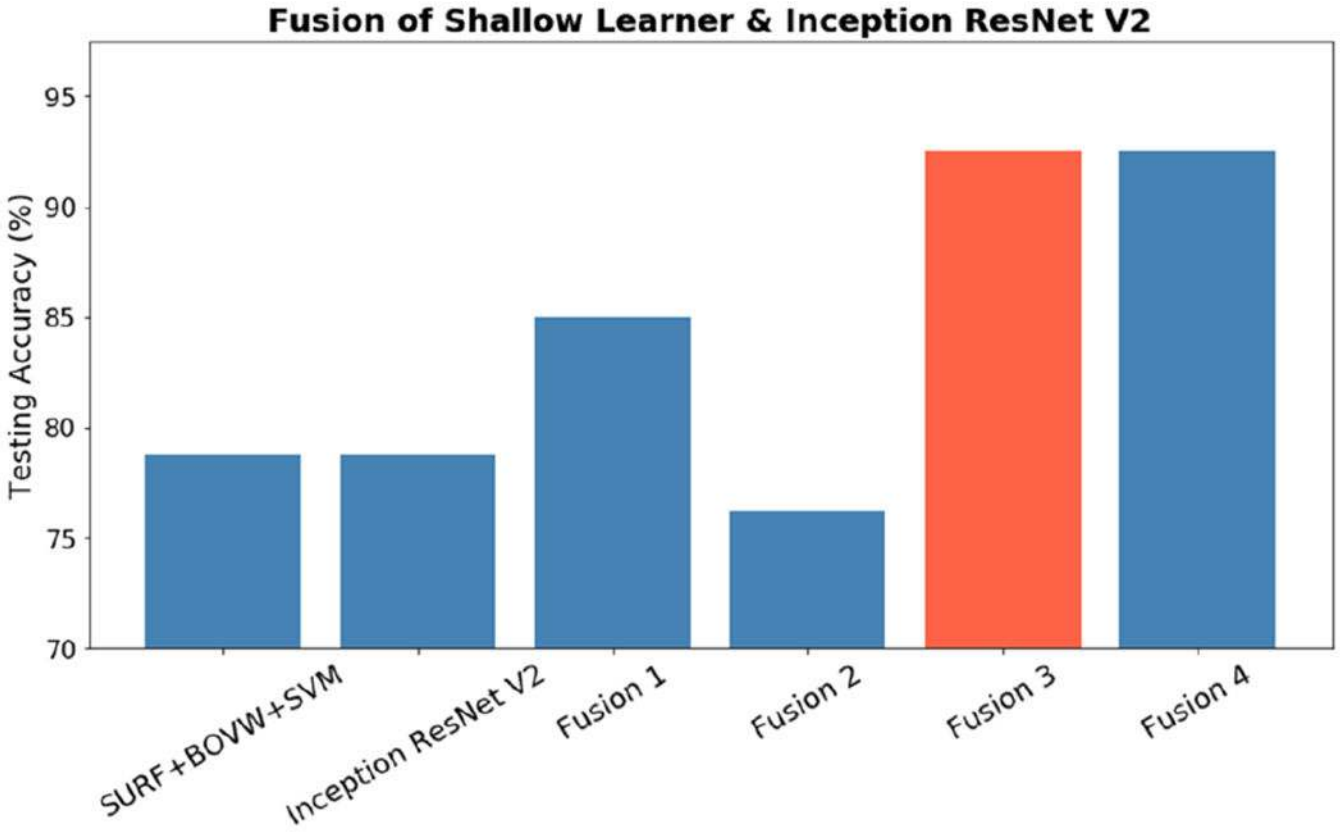

Figure 10:
Testing accuracy results for fusing shallow learner and Inception ResNet V2. The orange bar is the best result.

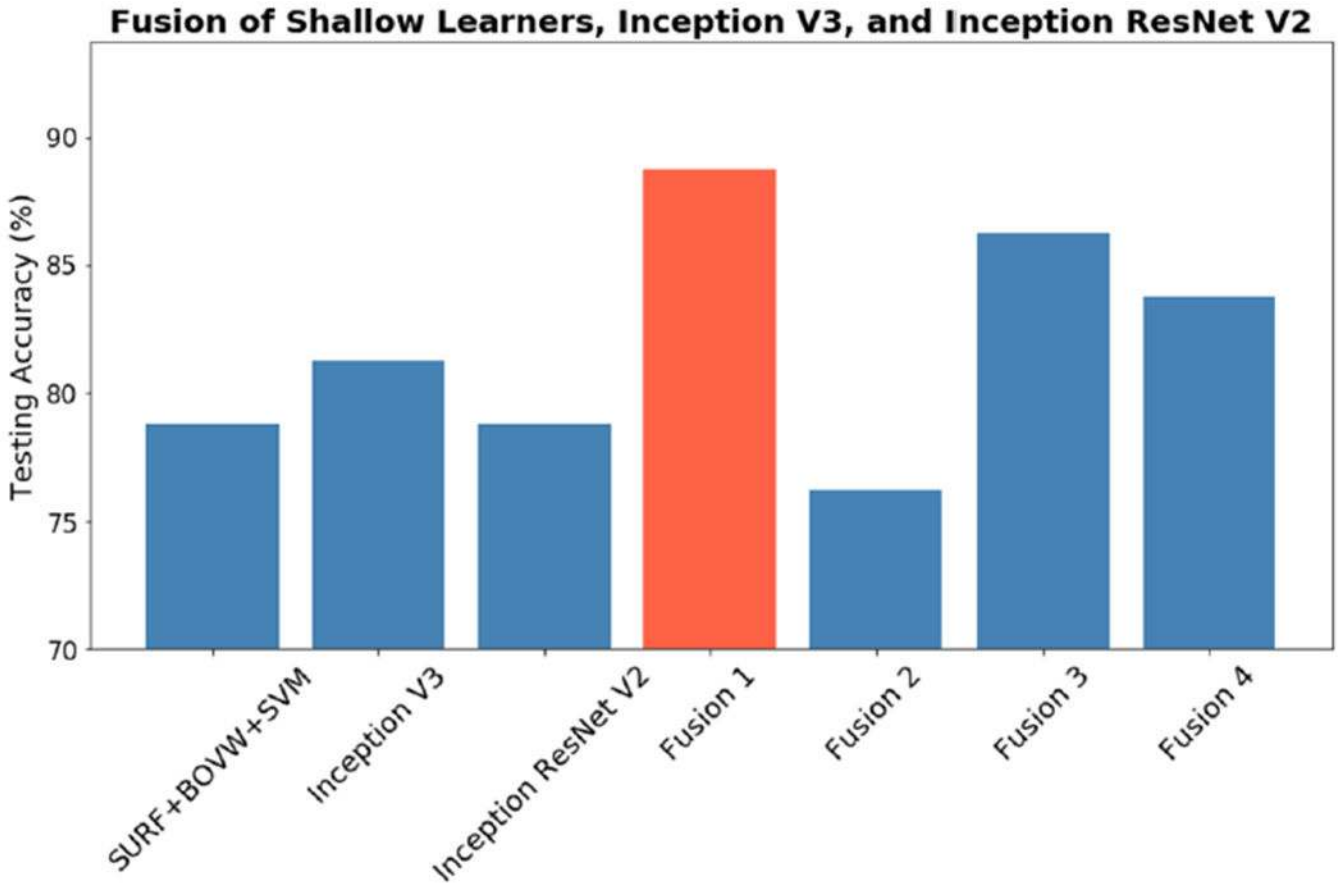

Figure 11:
Testing accuracy results for fusing the two inception models. The orange bar is the best result.

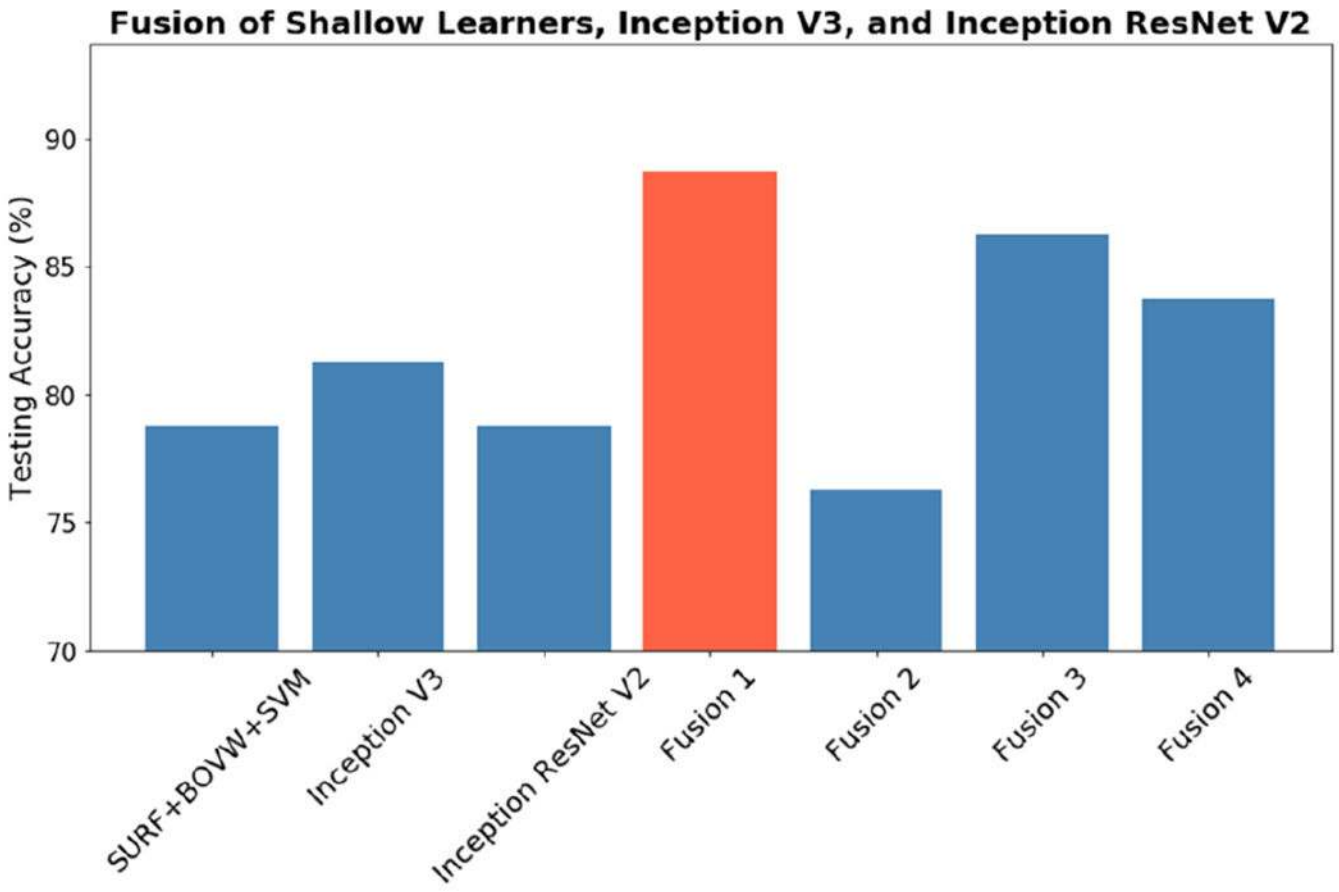

Figure 12:
Testing accuracy results for all three models. The orange bar is the best results.