Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

New Fully Automatic Approach for Tissue Identification in Histopathological Examinations using Transfer Learning

ARTICLE INFO

Article history:
Received May 12, 2021

Keywords: Classification of Breast Cancer, Convolutional Neural Networks, Transfer Learning, Computer-Assisted Diagnosis, Deep Learning

ABSTRACT

The use of computational techniques in the processing of histopathological images allows the study of the structural organization of tissues and their changes through diseases. The present work proposes analyzing these images with the help of CNN's associated with Machine Learning extractors through the Transfer Learning process. The BreakHis dataset was used in the experiments, consisting of histopathological images of breast cancer with different tumor enlargement scales classified as Malignant or Benign. In this study were performed various combinations of Extractor-Classifiers, thus seeking to compare the best model. Among the results achieved, the best Extractor-Classifier set formed was CNN DenseNet201, acting as an extractor, with the SVM RBF classifier, obtaining accuracy of 95.39% and precision of 95.43% for the 200X magnification factor. Different models were generated, compared to each other, and validated based on methods in the literature to validate the experiments, thus showing the effectiveness of the proposed model.

1. Introduction

Breast cancer is the most common cause of cancer death among women aged 40 to 45 years and being the main factor of mortality in females [1]. According to the World Health Organization (WHO), in 2020, about 2.3 million women were diagnosed with breast cancer. By the end of the same year, about 7.8 million women lived after receiving the diagnosis. Finally, in 2020, there were about 685,000 deaths worldwide; these data affirm breast cancer as the most prevalent in the world [2].

In this context, a patient with this type of pathology spends about the US $ 2,300 a year directly, and about the US $ 3325 to $ 5,545 with indirect costs due to the medical expenses of the disease [3].

An open surgical biopsy (OSB) is the most recommended exam when there is a need for an accurate diagnosis for surgical preparation, especially when it is not possible to determine the cause of the tumor with a needle biopsy [4] or other techniques. The OSB obtains practically all the suspicious injuries in its procedure with a high rate of reliability. Then, this exam is a technique considered adequate for defining the entire pathology in cases where the mass is small and complex to locate by touch or if an area looks suspicious on a mammogram but cannot be felt [5].

The microscope images provided by the OSB exam are called histopathological images [6]; after obtaining this type of image, it is necessary to classify the tumor as malignant or benign. Usually, this process is manual, performed by the specialist physician [7], which can cause a delay in classification and numerous factors that cause errors such as high workload or external factors such as lighting [8]. The following are studies that used mainly Learning Transfer and the BreakHis dataset to classify breast cancer.

Many Computers Assisted Diagnosis methods (CAD) have been proposed over the years to make this type of classification and several other medical applications faster and more accurate. These methods are mainly based on Digital Image Processing [9], Machine Learning [10], and deep learning [11], which are characterized by using Artificial Intelligence, or assisting, in human activities in an intuitive way and need [12]. Works such as Sharma et al. [13] proposed two classification methods based on Machine and Deep Learning Learning algorithms for the set of the multiclass present in the BreakHis [14] dataset. The first proposal is based on handcrafted resources extracted using Momentos de Hu, color histogram, and Haralick textures for later classification of these attributes to be classified by conventional classifiers. The second proposal is based on VGG16, VGG19, and ResNet50 networks to extract and classify features. The results were satisfactory, especially for convolutional neural networks, reaching an accuracy of 93.97%.

Taking into account the importance of CAD systems for quickly and efficiently classification of pathologies, and based on different techniques in the literature with the same purpose for the same dataset, the proposed method seeks to classify the classes of histopathologies of breast cancer, especially the classification between benign or malignant classes through the use of Machine and Deep Learning algorithms. As contributions, this study presents significantly better results than the methods found in the current literature, besides having a quick classification time. We show excellent results for classification according to the BreakHis dataset magnification factor and for generalized classification of the dataset, i. e., with all scales at once.

2. Related works

The use of *deep learning* has gained strength in recent years, when applied as an approach to problems involving *machine learning*, such as object recognition and classification [15]. A convolutional neural network (CNN) is composed of convolutional layers (used to generate attributes), layers of *pooling* (used to join information from a region) and layers called *fully-connected*, used in classification.

Have been used Histopathological images (H& E) to detect different pathologies, such as breast cancer, colorectal cancer, and lung cancer. The various uses of this type of medical image demonstrate the vast scope for improvements in histological exams and the importance of contrasting these approaches [16]. The (*transfer learning*) technique uses CNN's convolutional and *pooling* layers, while replacing the *fully-connected* layers with traditional classifiers (such as KNN, SVM, etc.). In this way, the convolutional neural network functions as a feature extractor.

In this sense, in the studies by Celik *et al.* [17] there was an investigation regarding the automatic detection of invasive ductal carcinoma (IDC), which is the most common subtype of breast cancer, using the technique deep transfer, especially with the aid of the ResNet-50 and DenseNet-161 extractors for IDC detection. Was applied the method developed by the authors to the BreaKHis image data set for classification between benign and malignant tumors of the breast, obtaining mean accuracy of 88% for the classification of the whole set, without separation by scale. However, it is worth mentioning that has been balanced the dataset for better performance, which may compromise the method's generalizability.

The work of Zhi *et al.* [18] investigates the use of transfer learning with Convolutional Neural Networks to automatically diagnose breast cancer in spots of histopathological images provided by BreakHis. The authors Combined transfer learning with CNN VGGNet in a more superficial custom architecture. In addition, as far as classification is concerned, it was separated into scales the dataset, and the two main classes were considered malignant and benign. The method obtained satisfactory results, with metrics superior to the same model of CNN trained without transfer Learning, with an accuracy of 94% for the classification on the scale of 200%. However, the comparison between the proposed model and other methods found in the literature for the BreakHis dataset was based only on accuracy, which may compromise the reliability of such a comparison, such as the absence of observation of other points of the method, such as its false detection positives and false negatives.

Also, taking into account the advantages of this approach, and the same dataset, Mehra *et al.* [19] had Transfer Learning compared to networks fully trained in the histopathological imaging modality. It analyzed three pre-trained networks: VGG16, VGG19, and ResNet50. Was observed the behavior of the networks to enlarge the image scale, the classification took into account the two main types of cancer. The method obtained 92% accuracy and 95% ROC, proving the effectiveness of using Transfer Learning over networks. However, the authors balanced the dataset for better performance without specifying this procedure in more detail, which makes the method's performance less reliable. In addition, it wasn't performed classification by scale.

Song *et al.* [20] presented an approach based on transfer of learning for the classification of histopathological images. In addition, we used the Fisher Image Coding Vector (FV) resource of local characteristics, extracted using the model of Convolutional Neural Network (CNN) pre-trained on ImageNet. Again, it uses BreaKHis the imaging data set to classify benign and malignant breast tumors. The authors used the only accuracy as an evaluation metric, which precludes a broader comparison with state-of-the-art works, and only performed the classification by scale.

Deniz *et al.* [21] guided the theme focused on breast cancer in which the transfer of learning and methods of extracting characteristics are used to adapt a pre-trained CNN model to the problem in question. The AlexNet and Vgg16 models are used for resource extraction. The attributes obtained are classified by support vector machines (SVM). The data set used the BreakHis, being divided between the image scales for the malignant and benign classes. The method reached 95% accuracy with the SVM RBF classifier in the 200X scale. However, the classification was not carried out without distinction by the scale factor, which could prove the ability to generalize the method.

So, considering the importance of a CAD system for the classification of histopathologies quickly and efficiently, based on different techniques in the literature with the same purpose. In addition to the BreakHis dataset, the proposed method seeks to classify the other classes of histopathologies of breast cancer, especially the classification between benign or malignant, through Machine and Deep Learning algorithms. As contributions, this study presents significantly better results than the methods found in the current literature, besides having a quick classification time and showing the classification according to

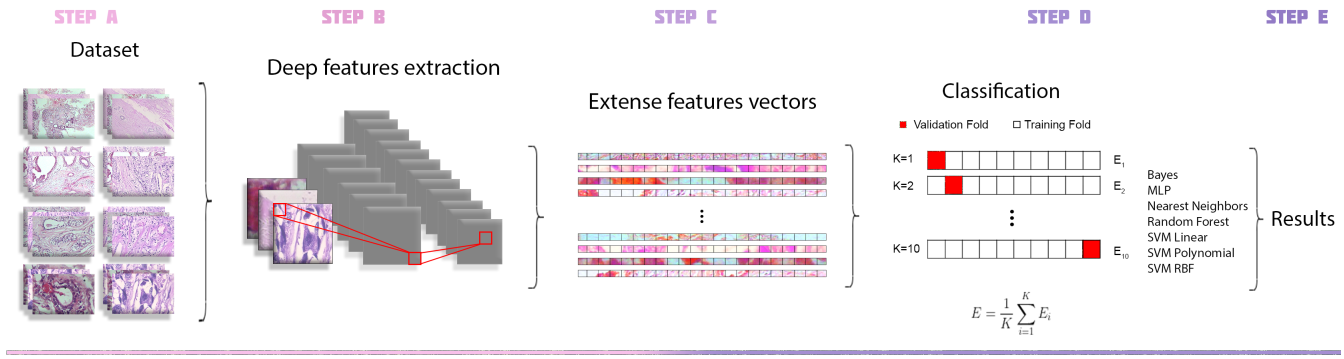


Fig. 1. Flowchart of transfer learning process, the methodology used in this work. The Step A represents the split of dataset, depending on magnification factor. The Step B presents the feature extraction of images by each CNN model. The Step C represents the extense feature vectors resulting from extraction stage. The Step D shows the classification stage of each feature vector into benign or malignat classes. And the Step E represents the results reached by each model in terms of the evaluation metrics.

the BreakHis dataset scale factor and excellent results in the generalized classification of the dataset, that is, with all scales at once.

3. Materials and methods

In this section, the methods used in the experiments are presented in different combinations of ways through the transfer learning technique for the generation of other computational models to classify tissues in histopathological exams. In this section, the methods used in the experiments are presented, in the different combinations of algorithms, using the transfer learning technique to generate diverse computational models to classify tissues in histopathological exams and the evaluation metrics used in the experiments.

3.1. Attribute extraction through convolutional neural networks

The Transfer Learning method consists of using a convolutional neural network (CNN), pre-trained with an extensive database, of extracting shape and texture characteristics from another dataset, to remove the attributes of the images used in the classification step, using traditional classifiers [22]. In other words, to be used exclusively as a feature extractor, CNN lacks its classification layers, called fully-connected layers [23, 24]. Thus, the network does not need to be trained, unlike what happens with other methods such as fine-tuning or learning from scratch, which seeks to train the network with the dataset in question [25].

In this work, we used CNN's topologies with weights initially trained with the large ImageNet [26] image bank, which consists of millions of images of everyday objects in 1000 different classes. The fully connected layers were removed for all of them, with the output of each network being a vector resulting from the last convolutional or pooling layer.

The CNN's architectures used in this paper were two: the VGG [27] architecture and the DenseNet [28] architecture. The first was implemented in two different configurations (VGG16 and VGG19), while the second was implemented in three (DenseNet121, DenseNet169, and DenseNet201).

The VGG16 and VGG19 configurations differ by their number of weights, with 16 layers and the second with 19 layers. This architecture is differentiated by using small convolutional filters, which increases its depth power [27].

In parallel, the three different DenseNet configurations are differentiated by the different number of layers that compose them, being formed by convolutional, transition layers and by the so-called "DenseBlocks." Its main characteristic is its dense connections between the layers, feeding a system of solid propagation of attributes to the subsequent layers and its reuse, requiring few parameters [28].

3.2. Classification using machine learning techniques

After the feature extraction stage, the attribute vectors extracted by the topologies presented in the previous section have different sizes, depending on the network, and are provided to the seven classifiers used in this work. They are Naive Bayes, MLP, Nearest Neighbors, Random Forest, and three different versions of SVM (Linear, Polynomial, and RBF).

The first classifier used, Naive Bayes, is an algorithm that makes a statistical analysis of the vector of attributes, based on the Bayes Decision Rule, on conditional analysis, and the probability density function. The method calculates a probability value for a sample to belong to each of the classes in question at the end labels, it with the most likely class [29].

The MLP classifier (Multi-layer Perceptron) is an algorithm composed of several layers of the artificial neuron, called perceptron [30]. Between the input, which receives the attributes, and the output of the MLP, there are several layers of perceptrons with different weights that propagate the initial information throughout its length, learning from the values provided to it to predict the sample class in its output [31].

Nearest Neighbors, or KNN, is a supervised machine learning method that classifies a sample through its spatial distribution with the others already labeled [32]. It is based on its k parameter, which must be odd and is the number of neighbors closest to the current sample. Therefore, the new sample receives the most frequent label among the k neighbors.

Based on the decision trees and the form of classification of the human brain, we have the Random Forest [33] algorithm. This method is considered unsupervised and has a random startup, using estimators to handle the input information.

The Support Vector Machine (SVM), in turn, are classification methods that use statistical analysis and optimal separation hyperplanes that depend directly on the kernel used to analyze the spatial layout of the samples [34]. The kernels used for SVM's were Linear, Polynomial, and RBF (Radial Basis Function).

3.3. Evaluation Metrics

The metrics used to assess the classification of each combination were: accuracy (Acc), precision, recall or sensitivity, F1-Score (F1), and Matthews Correlation Coefficient (MCC).

All equations use true positives (VP), false negatives (FN), true negatives (VN), and false positives (FP), all present in the confusion matrix to calculate the evaluation metrics that can be seen in the Equations of 1 à 5. A confusion matrix is a tool that compares the actual class of each classified sample and the class predicted by the method.

$$Acc = \frac{VP + VN}{VP + VN + FP + FN} \tag{1}$$

$$Precision = \frac{VP}{VP + FP} \tag{2}$$

$$Recall = \frac{VP}{VP + FN} \tag{3}$$

$$F1 = \frac{2VP}{2VP+FP+FN} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{4}$$

$$MCC = \frac{VP \times VN - FP \times FN}{\sqrt{(VP+FP)(VP+FN)(VN+FP)(VN+FN)}} \tag{5}$$

Accuracy (ACC) is a metric that measures the number of samples correctly classified by the total number of samples, and since it deals directly with the method's success rate, it is often placed as the main one. F1-Score, on the other hand, uses intermediate metrics, Precision and Recall, to measure a balance between them through their harmonic mean. Thus, a high F1-Score index is considered a uniformity factor in the classification of tissues, as it shows a low number of false positives and false negatives [16].

Finally, the Matthews Coefficient (MCC) is a correlation measure widely used in binary classification problems. Based on Pearson's correlation indices, this coefficient can be treated as a balanced measure even when there is an unbalanced dataset.

There is a metric result for each class in question., simple arithmetic averages of the class results for representation in tables were made to facilitate the analysis of the results. Values will be represented as a percentage.

3.4. Statistical Test

The Kolmogorov-Smirnov test is non-parametric, commonly used to assess the statistical similarity between two attribute vectors. Through the statistical test, it is possible to affirm with a more significant property if an average with its respective standard deviation is different from the other, since not only its point values are compared, but the vector of values that generated it [35].

Table 1. BreaKHis dataset images distribution according to class, subclass and magnification factor [37].

Class	Subclass	Magnification Factors				Total
		40X	100X	200X	400X	
Benign	Adenosis Fibroadenoma Phyllodes Tumor Tubular Adenoma	625	644	623	588	2,480
Malignant	Ductal Carcinoma Lobular Carcinoma Mucinous Carcinoma Papillary Carcinoma	1,370	1,437	1,390	1,232	5,429
		Total				7,909

It is necessary to establish a reliability coefficient, that is, a α to carry out such a comparison. The comparison algorithm between the attribute vectors provides a value of P at the end of its calculation. This P value is then compared to the alpha, accepting the hypothesis of statistical similarity if it is greater and rejecting the hypothesis if it is less than the value α. Usually the value of α chosen is 5 % (0.05) [36].

4. Methodology

This section presents the methodology proposed in that study for the transfer learning approach. The section is divided into subsections addressing the dataset used in the experiments and the methodology and methodology applied to the study, and the parameters of each model used.

4.1. BreaKHis database

BreaKHis is a database composed of thousands of biopsy images, acquired through microscopes, of tissues present in benign and malignant tumors in breasts [37]. The dataset was formed between January and December 2014, with patients invited by the R&D Laboratory in Brazil.

The samples were collected using open surgical biopsies (OSB) and prepared for study through a microscope attached to a digital camera. The resulting dataset consists of 7,909 images with 3 RGB channels, 8 bits each, PNG format, and dimensions of 700 × 460 pixels.

The tissue images, divided into the main benign-malignant classes, are further subdivided into 8 other subclasses, according to the type of lesion, as shown in Table 1. Finally, there is also a division according to the magnification factor, 40X, 100X, 200X, and 400X.

Following other studies that used the same dataset and in a context of aid to medical diagnosis, we opted for the binary classification between the benign and malignant classes for each of the magnification factors. Thus, subclasses were not considered for this work.

In addition, BreaKHis samples can also be grouped among the 82 patients who volunteered to form the dataset. In this way, several images would be assigned to the same patient, which

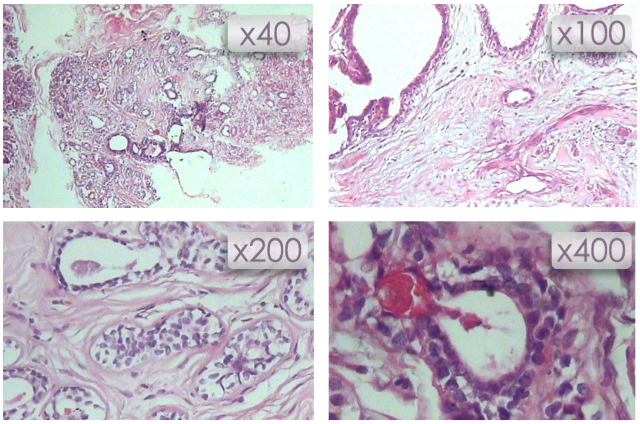


Fig. 2. Sample images of benign class for each magnification factor from BreaKHis dataset.

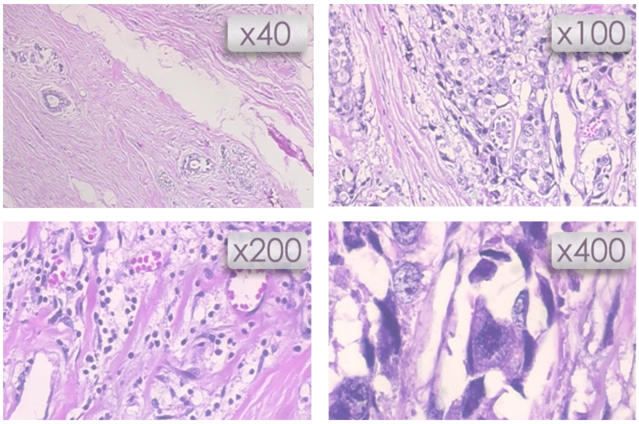


Fig. 3. Sample images of malignant class for each magnification factor from BreaKHis dataset.

would promote a different classification work. It is worth mentioning that the classification was made only at the image level for this work, for the different magnification factors, and the classification at the patient level was disregarded.

We also emphasize that there was no work to increase or balance the data set. Therefore, only the original images were provided to CNN's, only with the benign/malignant label.

4.2. Methodology of the Proposed Study

To differentiate the histopathological exam images obtained through biopsy, in their benign / malignant classes (for each subdivision of the magnification factors), the attributes of each of the images are extracted, through five different CNN's (DenseNet121, DenseNet169, DenseNet201, VGG16, and VGG19), used as extractors, for fully automatic classification of the end of the extraction process, subsequently using seven different classifiers (Naive Bayes, MLP, Nearest Neighbors, Random Forest, Linear SVM, SVM Polynomial and SVM RBF), both models process in parallel. Then, the performance of each extractor-classifier combination is evaluated. The CNN architectures mentioned in this study and the classifiers are presented in Section 3.

Figure 1 details the approach of the proposed methodology using transfer learning techniques. The methodology was subdivided into Stages A to E, addressing the different points of each Stage.

In Step A, the images from the BreaKHis dataset, presented above, were divided according to their magnification scales (40X, 100X, 200X, and 400X), forming four different datasets with different amounts of images, labeled as benign or malignant.

The resize technique is performed for each image, as preprocessing, to adapt them to the entrance to the network, depending on the topology used. All other CNN's parameters and configurations used in this work were kept as standard by the network itself, as used in the works compared in Section 3.

In Step B, the CNN topologies mentioned above receive each of these datasets formed to act as feature extractors, forming a sub-dataset composed of the attribute vectors extracted from each of the images and their respective class (benign - 0, malignant - 1), resulting in Step C. The attribute vectors have different sizes, depending on the CNN used in the extraction: 512 attributes in VGG16 or VGG19 and 1024, 1664, and 1920 attributes in the cases of DenseNet121, DenseNet169, and DenseNet201, respectively.

In Step D, the data set composed of the attribute vectors generated in Step B and C leads the classifiers to carry out a supervised classification of each of these extracted images. The was done by cross-validation, using the K-fold method of training with 10 folders. For each folder, the images were separated in the proportion of 90% / 10% for training/validation.

A random search algorithm was used to find the best settings for each of the classifiers for the current problem. The random search method was used to search for the best hyperparameters for each of them, according to the search intervals found in Table 2. 20 iterations were made with 5-fold cross-validation to search for the parameters of each classifier.

Table 2. Parameters intervals provided to Random Search algorithm for each classifier used in the transfer learning approach.

Classifier	Parameters of Search	Intervals of Search
Naive Bayes	-	-
MLP	Hidden layers	[2, 1001]
Nearest Neighbors	Number of Neighbors	1, 3, 5, 7, 9, 11, 13, 15
Random Forest	Maximum depth	6, Unlimited
	Bootstrap	True, False
	Criterion	Gini, Entropy
SVM Linear	Regularization parameter (C)	$2^x \mid x \in [-5, 15]$
SVM Polynomial	Degree	3, 5, 7, 9
	Regularization parameter (C)	$2^x \mid x \in [-5, 15]$
SVM RBF	Gamma	$2^x \mid x \in [-15, 3]$
	Regularization parameter (C)	$2^x \mid x \in [-5, 15]$

In addition to the parameters found through random search, other fixed parameters for the classifiers stand out: For MLP, a limit of 1000 iterations and an initial learning rate of 5×10^{-4}; for Random Forest, in addition to random initialization, 3000 estimators were used; and, finally, for the three types of SVM, fixed tolerances of 10^{-3} were used. The other classifying parameters, which were not mentioned here, were specified sets and equal to the default values of the scikit-learn library 0.20.2, used in the implementation of these methods.

Finally, in Step E, the evaluation metrics seen in Section 3 were used to evaluate each model formed by the CNN-classifier combination. The results of these metrics are presented in Section 5. In addition to the comparison between the models, to find the best combination for the problem in question, the best model is compared with other works in the literature that used the same database in binary classification problems, both for the magnification factors individually, as for the entire dataset, without distinction between scales.

5. Experimental Results

This section deals with the results obtained by each of the 35 CNN X classifier combinations (5 X 7), for each of the magnification factors, in the light of the evaluation metrics presented in the previous section. In addition, in the following subsections, the best combination made by the study (Best model) was compared with other works in the literature. Finally, a classificatory experiment is performed, using the entire dataset, simulating an augmentation by scale, To validate the investigation also compared with literature work.

All processing was performed using the Linux operating system (Ubuntu 16.04 distribution) with 16GB of RAM and AMD Ryzen 5 3400G processor. The extraction processes with CNN's were accelerated by means of an Nvidia GeForce GTX 1660 Super GPU with 6GB dedicated memory. The *deep learning* models (VGG's and DenseNet's) were implemented in the Python programming language (version 3.7) through the libraries TensorFlow-GPU 1.14, Keras 2.2.4, and using OpenCV 4.1.0, while the classifiers were implemented with the sci-kit-learn library 0.20.2.

Table 3 shows the results obtained with the magnification factor of 40X for a benign-malignant binary classification of histopathological images. For all the CNN architectures used, it can be seen that the SVM RBF classifier achieved accuracy above 91.9%. The accuracy and recall results for that same classifier were also above 90%. For the remaining metrics, F1-Score and Matthews Coefficient, the results were in the range of 80% or 90%, which indicates a uniformity and balance in the classification of both classes.

Table 3 also shows that the DenseNet architecture is superior as a feature extractor when compared to the VGG architecture. However, this difference is not very significant for most classifiers. Unlike SVM RBF, SVM Polynomial was the lesser of all classifiers, not reaching any metric value above 50%, which shows that this classifier tends to make more mistakes than to correct for regions belonging to breast tissue. Finally, the MLP and Nearest Neighbors classifiers also achieved results close to that of SVM RBF for most metrics, but they lose to it when it comes to training time (higher processing) and test time, respectively, since the time of MLP training is around 120 to 190 ms, while the Nearest Neighbors prediction time is in the 9 to 30 ms range, with a high standard deviation (while the SVM RBF trains and predicts at a faster average than the others two and with a smaller standard deviation).

Table 4 shows the classification results for the BreaKHis dataset, restricted to the 100X magnification factor. Much like the results in the previous table, the combination DenseNet201 with SVM RBF also achieved the highest values for all metrics. Again, the Naive Bayes and SVM Polynomial classifiers proved to be the lowest for this breast tissue classification problem, despite an improvement in SVM Polynomial for these 100X images.

The other metric values reached by the different combinations had results very close to the 3, which shows that the magnification factor of 100X does not have much difference compared to 40X, probably because they are very close to each other. Despite the accuracy values being near, for the 100X factor, there was a decrease in the standard deviation in practically all combinations, showing that the models obtained minimally better results to classify the tissue in Histopathological images. The F1-Score values in the table, in turn, linked to the MLP, Nearest Neighbors, Random Forest, SVM Linear, and SVM RBF classifiers are in the same range between 80% and 92%. The Matthews coefficient, in turn, has its values highlighted for MLP, Nearest Neighbors, and SVM RBF, mainly for the DenseNet169 and DenseNet201 networks, with values from 85% to 88%. On the other hand, once again, SVM Polynomial could not score on the Matthews Coefficient. That is, it classifies the samples entirely randomly, without any correlation.

Finally, in Tables 5 and 6 we can see the results of each classification made for the magnification factors of 200X and 400X, respectively. The combination DenseNet201 and SVM RBF reached their best values among all image scales for the 200X variation, when it had 95.38% accuracy, with a standard deviation of only 0.40%. Accuracy and recall also achieved approximately 95.5% and 94.5%, respectively, both with a low standard deviation. About the F1-Score and Matthews Coef-

Feature Extractor	Classifier	Accuracy	Precision	Recall	F1-Score	Matthews	Train Time	Predict Time
DenseNet121	Bayes	73.83±0.68	79.63±4.12	59.15±0.56	58.00±0.70	32.90±3.14	0.049±0.022	0.106±0.010
	MLP	90.83±1.34	89.66±2.14	89.10±1.31	89.30±1.46	78.73±3.03	187.838±8.430	0.677±0.318
	Nearest Neighbors	91.73±0.69	91.21±0.73	89.37±1.26	90.17±0.91	80.54±1.69	0.104±0.015	16.512±2.471
	Random Forest	87.67±2.76	85.90±3.37	85.37±3.11	85.59±3.17	71.25±6.38	15.718±1.708	5.709±0.454
	SVM Linear	90.38±1.54	88.37±1.86	90.08±1.39	89.09±1.66	78.43±3.19	6.940±0.638	3.758±0.195
	SVM Polynomial	31.33±0.00	15.66±0.00	50.00±0.00	23.85±0.00	0.00±0.00	17.794±1.158	11.071±0.536
	SVM RBF	93.13±1.28	92.17±1.83	91.91±1.23	92.01±1.45	84.07±2.92	8.139±0.764	4.701±0.499
DenseNet169	Bayes	78.20±0.88	83.31±0.75	66.24±1.56	67.79±1.98	46.47±2.46	0.083±0.030	0.087±0.029
	MLP	92.98±1.95	91.87±2.42	91.85±2.06	91.85±2.23	83.72±4.47	117.196±8.137	0.590±0.323
	Nearest Neighbors	93.78±1.11	93.29±0.95	92.17±1.82	92.67±1.40	85.44±2.66	0.123±0.018	24.993±3.575
	Random Forest	89.32±1.14	90.04±1.47	84.83±1.51	86.79±1.45	74.68±2.79	14.133±1.672	10.158±3.413
	SVM Linear	91.63±1.71	89.89±1.95	91.25±1.99	90.45±1.90	81.12±3.74	9.602±0.873	5.606±0.351
	SVM Polynomial	31.33±0.00	15.66±0.00	50.00±0.00	23.85±0.00	0.00±0.00	26.836±0.972	17.894±0.309
	SVM RBF	94.09±1.32	93.63±1.83	92.56±1.32	93.05±1.52	86.18±3.07	11.621±0.960	7.104±0.440
DenseNet201	Bayes	78.35±1.05	84.11±1.62	66.31±1.74	67.87±2.23	47.11±3.04	0.071±0.030	0.100±0.034
	MLP	93.18±0.81	92.64±1.02	91.38±1.02	91.96±0.96	84.01±1.92	121.425±2.298	0.836±0.076
	Nearest Neighbors	93.18±1.36	93.60±1.58	90.43±1.70	91.79±1.65	83.96±3.27	0.122±0.031	29.353±4.152
	Random Forest	88.12±1.45	86.55±1.85	85.65±1.54	86.04±1.61	72.19±3.21	23.778±2.721	6.171±1.180
	SVM Linear	91.63±0.80	90.07±0.88	90.69±1.23	90.35±0.96	80.75±1.98	10.820±0.638	6.393±0.282
	SVM Polynomial	31.33±0.00	15.66±0.00	50.00±0.00	23.85±0.00	0.00±0.00	30.694±1.194	20.596±0.292
	SVM RBF	**94.94±0.87**	**94.63±1.05**	**93.53±1.11**	**94.04±1.04**	**88.15±2.06**	13.418±0.768	8.655±0.804
VGG16	Bayes	73.83±0.88	72.83±1.87	61.24±1.40	61.46±1.85	32.00±2.95	0.040±0.030	0.074±0.018
	MLP	87.87±1.31	86.60±1.38	84.99±2.51	85.58±1.79	71.53±3.32	129.823±9.492	0.394±0.174
	Nearest Neighbors	89.42±0.99	89.70±1.03	85.30±1.42	87.03±1.29	74.86±2.43	0.045±0.006	9.743±1.467
	Random Forest	84.96±1.65	86.91±2.31	77.74±2.22	80.34±2.32	63.98±4.37	34.466±3.962	8.064±1.655
	SVM Linear	87.02±1.03	84.69±1.33	85.94±1.19	85.20±1.12	70.61±2.27	5.278±0.586	2.730±0.110
	SVM Polynomial	31.33±0.00	15.66±0.00	50.00±0.00	23.85±0.00	0.00±0.00	10.200±0.302	5.609±0.037
	SVM RBF	91.98±0.79	91.62±0.55	89.55±1.62	90.44±1.07	81.13±1.99	8.142±0.604	3.119±0.177
VGG19	Bayes	74.59±1.53	73.68±3.49	62.62±1.74	63.31±2.17	34.57±4.90	0.047±0.024	0.074±0.015
	MLP	88.77±0.58	87.45±1.20	86.39±1.15	86.79±0.63	73.81±1.35	178.718±8.357	0.556±0.248
	Nearest Neighbors	89.47±0.69	90.43±0.92	84.90±1.28	86.94±1.01	75.10±1.61	0.046±0.006	9.576±1.640
	Random Forest	86.17±2.93	85.12±3.33	81.84±3.90	83.13±3.70	66.86±7.21	31.504±4.190	6.997±2.559
	SVM Linear	86.42±1.70	83.98±1.94	85.28±2.31	84.52±2.00	69.25±4.11	4.965±0.543	2.400±0.087
	SVM Polynomial	31.33±0.00	15.66±0.00	50.00±0.00	23.85±0.00	0.00±0.00	10.386±1.002	5.501±0.056
	SVM RBF	91.98±1.30	91.92±1.61	89.20±1.58	90.38±1.56	81.07±3.12	9.056±1.115	3.729±0.295

Table 3. Metrics (%), train time (ms) and predict time (ms) obtained by each combination of feature extraction and classifier for the 40X magnification factor. In bold, the best results per metric and the best combination method.

ficient (MCC), they have also achieved the best results so far, with 93.7% and 89%.

However, the magnification of 400X showed a decrease in the values achieved by the metrics for all networks under study. This approach already makes it more challenging to differentiate by CNN topologies, not showing much difference between classes. Despite this, the difference is low in accuracy, with the best combination reaching 92.64% on average. The standard deviation has risen above 1.2% in virtually all cases. For the MCC, there is a more significant difference compared to the other magnifications seen, having its best result again for the SVM RBF combination with DenseNet201, reaching almost 83% on average, with a standard deviation of 2.7%.

Finally, analyzing the training times and prediction of the tables, it is clear that they depend only on the classifiers and the size of the vector of attributes coming from CNN. Therefore, the time bands do not vary much depending on the table. It is noticeable that the MLP training time varies a lot on average, with a very high standard deviation, and is mostly quite high and requires a high level of processing due to the complexity of its algorithm. On the other hand, the Nearest Neighbors classifier, despite being trained quickly, predicts more slowly among the other classifier options. Furthermore, the times of the VGG architecture (either 16 or 19) are shorter than the times of the DenseNet architecture since the number of attributes in the vector of each sample is at least half the number of attributes resulting from the competing architecture. It should be noted that the values are in milliseconds, and therefore, there is no stark difference between the values.

5.1. Comparison of the best combination with the methods in the literature for each magnification factor

Since, for all magnification factors, the best combination of CNN-classifier was analyzed using the DenseNet201 architecture and the SVM RBF classifier. In this way, the results of this model were used for comparison with other works in the literature as shown in the table 7. In the same way, these authors also classified the images in a binary way, distinguishing the magnification factors 40X, 100X, 200X, and 400X.

Table 7 shows the accuracy compared to the best model of this study with other methods in the literature that also made the benign-malignant distinction, separating the dataset by the magnification factor. As specified in Section 2, Song *et al.* [20] used CNN-based FV descriptor with adaptation layer to classify the dataset between the two classes. On the other hand, Zhi *et al.* [18] also used the transfer learning technique with a VGGNet-based architecture custom model with a patch-based augmentation. Finally, Deniz *et al.* [21] fine-tuned the AlexNet deep model for the demand, achieving better results than in their other attempts with the use of VGG16 and the concatenated vectors of VGG16 and AlexNet.

Feature Extractor	Classifier	Accuracy	Precision	Recall	F1-Score	Matthews	Train Time	Predict Time
DenseNet121	Bayes	71.94±0.83	81.81±4.83	54.92±1.15	50.71±2.04	24.85±4.12	0.043±0.025	0.106±0.029
	MLP	91.30±1.19	90.23±1.64	89.29±1.12	89.72±1.33	79.50±2.67	35.253±1.438	0.185±0.071
	Nearest Neighbors	88.80±0.83	89.40±1.59	84.01±0.69	86.03±0.92	73.20±2.10	0.105±0.012	18.264±2.092
	Random Forest	87.89±1.46	87.46±2.02	83.61±1.90	85.12±1.80	70.95±3.60	11.120±1.281	6.901±2.060
	SVM Linear	89.96±1.38	87.91±1.72	89.17±1.19	88.47±1.49	77.06±2.89	7.637±0.108	4.216±0.229
	SVM Polynomial	61.39±15.27	30.70±7.64	50.00±0.00	37.38±6.91	0.00±0.00	19.537±1.345	11.979±0.301
	SVM RBF	93.22±1.61	92.41±2.13	91.66±1.58	92.02±1.84	84.07±3.70	8.904±0.728	5.265±0.702
DenseNet169	Bayes	74.53±0.44	84.08±1.57	59.15±0.83	57.83±1.39	35.25±1.26	0.081±0.022	0.098±0.016
	MLP	92.02±1.31	90.82±1.43	90.54±1.92	90.63±1.59	81.35±3.12	161.009±12.553	0.693±0.269
	Nearest Neighbors	91.74±1.34	91.55±1.63	88.83±1.67	90.02±1.64	80.34±3.25	0.120±0.018	27.918±4.142
	Random Forest	87.60±2.88	86.67±3.88	83.78±3.14	84.97±3.39	70.38±6.89	10.790±1.243	7.001±1.888
	SVM Linear	91.78±0.77	90.16±0.98	90.88±1.00	90.48±0.87	81.03±1.74	11.390±0.739	6.684±0.187
	SVM Polynomial	61.39±15.27	30.70±7.64	50.00±0.00	37.38±6.91	0.00±0.00	29.500±1.170	19.383±0.132
	SVM RBF	93.37±0.97	92.20±1.15	**92.33±1.19**	92.26±1.12	84.53±2.25	11.090±0.714	6.682±0.528
DenseNet201	Bayes	79.24±0.51	85.59±0.75	67.10±0.90	69.01±1.12	49.32±1.44	0.102±0.046	0.149±0.021
	MLP	93.22±0.86	92.52±0.80	91.54±1.52	91.97±1.09	84.04±2.11	48.209±3.841	0.220±0.113
	Nearest Neighbors	91.93±1.13	92.09±1.22	88.80±1.74	90.18±1.45	80.81±2.77	0.141±0.020	31.707±3.707
	Random Forest	88.56±0.91	87.60±1.20	85.12±1.17	86.18±1.11	72.67±2.22	20.628±2.545	6.859±2.040
	SVM Linear	92.89±1.18	91.53±1.04	91.94±1.94	91.70±1.46	83.46±2.96	14.042±0.807	8.433±0.321
	SVM Polynomial	61.39±15.27	30.70±7.64	50.00±0.00	37.38±6.91	0.00±0.00	33.403±0.371	22.413±0.353
	SVM RBF	**94.18±1.34**	**94.23±1.16**	92.06±2.11	**93.01±1.67**	**86.25±3.19**	22.408±2.560	11.321±1.032
VGG16	Bayes	74.10±1.13	85.69±0.45	58.23±1.97	56.19±3.24	34.01±3.98	0.056±0.032	0.079±0.021
	MLP	88.13±1.38	87.11±1.97	84.59±1.47	85.66±1.61	71.65±3.30	12.570±0.536	0.064±0.042
	Nearest Neighbors	87.22±0.98	87.41±0.90	82.00±1.74	83.96±1.47	69.18±2.49	0.051±0.004	10.680±1.631
	Random Forest	86.64±0.99	84.96±1.31	83.30±1.44	84.00±1.23	68.23±2.46	15.960±2.162	5.360±1.444
	SVM Linear	86.78±1.72	84.35±2.07	85.50±1.90	84.82±1.92	69.83±3.84	4.979±0.444	2.451±0.122
	SVM Polynomial	61.39±15.27	30.70±7.64	50.00±0.00	37.38±6.91	0.00±0.00	11.150±0.800	6.053±0.088
	SVM RBF	90.05±1.05	90.71±1.73	85.77±1.33	87.67±1.31	76.30±2.67	10.122±0.926	4.319±0.213
VGG19	Bayes	77.27±0.73	84.17±2.36	63.92±0.90	64.87±1.20	43.60±2.62	0.056±0.034	0.080±0.023
	MLP	87.60±1.21	86.08±1.38	84.43±1.62	85.16±1.50	70.48±2.95	31.048±2.769	0.117±0.060
	Nearest Neighbors	86.30±1.57	86.69±2.09	80.49±2.14	82.62±2.11	66.88±4.04	0.046±0.004	10.614±0.866
	Random Forest	86.64±1.14	87.07±1.26	80.95±1.74	83.08±1.60	67.73±2.88	17.845±2.145	6.817±0.895
	SVM Linear	87.22±0.38	84.79±0.55	86.16±0.77	85.36±0.38	70.93±0.85	5.027±0.420	2.454±0.167
	SVM Polynomial	61.39±15.27	30.70±7.64	50.00±0.00	37.38±6.91	0.00±0.00	11.243±0.988	5.991±0.196
	SVM RBF	90.82±1.07	90.72±1.34	87.62±2.06	88.85±1.45	78.25±2.70	8.794±0.905	3.290±0.234

Table 4. Metrics (%), train time (ms) and predict time (ms) obtained by each combination of feature extraction and classifier for the 100X magnification factor. In bold, the best results per metric and the best combination method.

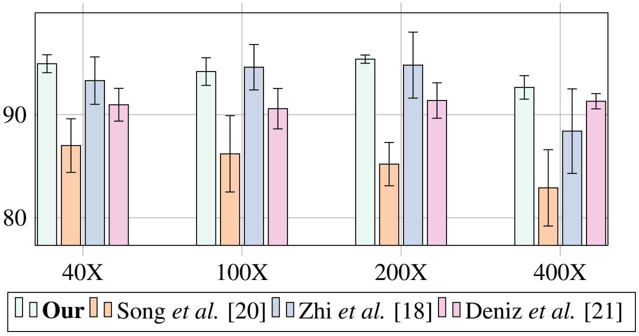
As can be seen in the table 8, the Kolmogorov-Smirnov statistical test rejected the hypothesis, with a α of 1%, that the attribute vectors have statistical similarity, this refusal even for such a small α value may have been due to the dense amount of samples in each of the attribute vectors used to generate the averages and standard deviations observed in the table 7.

Still, in Table 7, it is possible to see that our method, when it comes to accuracy, matches the other techniques in the literature for the magnification factors of 40X, 200X, and 400X. In addition, we also come very close to the average accuracy of Zhi et al. [18], the best result for the magnification factor of 100X, even without using the augmentation technique used by the author. We present a deviation - much smaller standard compared to his (double that obtained by this study).

The same results can be seen in the graph of Figure 4, for the purpose of a better comparison between the accuracy obtained by the methods. From the figure, it is clear that the accuracy achieved by our method has a smaller standard deviation for practically all magnification factors in comparison with other methods.

As can be seen in the table 8, the Kolmogorov-Smirnov statistical test rejected the hypothesis, with a α of 1%, that the attribute vectors have statistical similarity, this refusal even for such a small α value may have been due to the dense amount of samples in each of the attribute vectors used to generate the averages and standard deviations observed in the table 7.

Fig. 4. The comparison between accuracies (%) reached by our method and by the other methods from related works per magnification factor.



The comparisons made in the table and the respective figure are only at the average accuracy level. The authors did not inform the values of other metrics of the works in comparison.

5.2. *Comparison of the best combination with the literature methods for the complete dataset*

One last experiment was carried out, with all the images from the dataset, with supervised classification between the benign and malignant classes, this time without distinction of magnification between the images. The model generated and chosen by this study, based on Tables 3 to 6, was also the one that uses the DenseNet201 network with the SVM RBF classifier.

The results obtained were 94.88±0.57 accuracy, 94.64±0.49 precision, 93.97±0.70 recall, 93.38±0.89 F1-Score and

Feature Extractor	Classifier	Accuracy	Precision	Recall	F1-Score	Matthews	Train Time	Predict Time
DenseNet121	Bayes	79.58±1.47	83.35±1.20	68.43±2.56	70.46±3.00	49.49±4.02	0.053±0.036	0.094±0.041
	MLP	90.11±1.45	88.49±1.86	88.41±1.58	88.43±1.66	76.90±3.33	246.854±12.726	0.685±0.199
	Nearest Neighbors	90.76±1.98	90.34±2.05	87.68±2.87	88.80±2.53	77.96±4.86	0.114±0.011	17.307±2.990
	Random Forest	89.32±2.40	88.89±2.51	85.66±3.55	86.94±3.12	74.44±5.89	16.204±1.637	6.762±1.985
	SVM Linear	89.62±1.36	87.37±1.56	89.38±1.47	88.20±1.51	76.72±2.92	6.618±0.556	3.454±0.197
	SVM Polynomial	53.73±18.68	26.86±9.34	50.00±0.00	33.92±8.45	0.00±0.00	18.467±0.494	11.247±0.233
	SVM RBF	93.49±0.53	93.33±0.90	91.30±0.60	92.21±0.62	84.60±1.27	14.919±0.821	6.405±0.231
DenseNet169	Bayes	77.55±0.85	82.71±2.06	64.78±1.33	66.03±1.71	43.94±2.69	0.097±0.032	0.064±0.010
	MLP	92.10±0.95	91.40±1.51	90.12±1.71	90.62±1.16	81.49±2.39	374.174±24.015	1.402±0.838
	Nearest Neighbors	92.94±1.10	92.61±1.33	90.68±1.37	91.56±1.34	83.27±2.67	0.109±0.024	26.162±2.925
	Random Forest	87.88±0.87	86.21±1.12	85.11±1.07	85.61±1.03	71.31±2.06	10.352±1.196	6.788±1.869
	SVM Linear	91.36±0.79	89.51±0.72	90.82±1.73	90.05±1.03	80.31±2.25	11.441±0.548	6.718±0.269
	SVM Polynomial	53.73±18.68	26.86±9.34	50.00±0.00	33.92±8.45	0.00±0.00	27.726±1.383	18.152±0.530
	SVM RBF	93.34±0.90	92.74±1.30	91.68±1.51	92.12±1.07	84.39±2.19	10.987±0.577	6.693±0.475
DenseNet201	Bayes	82.96±1.29	87.35±2.17	73.40±2.05	76.24±2.19	59.08±3.54	0.089±0.035	0.108±0.027
	MLP	92.40±0.29	91.67±0.38	90.38±0.60	90.97±0.38	82.03±0.73	396.417±36.733	2.052±0.802
	Nearest Neighbors	93.54±0.77	93.64±0.84	91.17±1.40	92.23±0.99	84.75±1.81	0.136±0.019	30.310±4.724
	Random Forest	89.87±0.67	89.77±1.06	86.15±1.19	87.61±0.90	75.82±1.66	13.735±1.806	6.778±1.913
	SVM Linear	90.61±0.68	88.93±1.06	89.49±1.11	89.11±0.72	78.40±1.36	9.727±0.710	5.600±0.129
	SVM Polynomial	53.73±18.68	26.86±9.34	50.00±0.00	33.92±8.45	0.00±0.00	31.196±0.650	21.026±0.403
	SVM RBF	**95.38±0.40**	**95.43±0.59**	**93.69±0.51**	**94.49±0.48**	**89.10±0.97**	15.662±0.859	10.064±0.493
VGG16	Bayes	74.66±1.27	84.11±2.80	59.33±1.90	58.07±2.84	35.59±4.74	0.048±0.038	0.077±0.024
	MLP	89.47±1.45	88.21±1.82	86.88±1.83	87.47±1.74	75.07±3.46	31.354±4.076	0.116±0.057
	Nearest Neighbors	87.83±0.52	87.40±0.64	83.43±0.80	85.01±0.70	70.72±1.33	0.052±0.008	10.039±0.536
	Random Forest	86.14±0.44	84.79±0.70	81.99±0.65	83.15±0.52	66.72±0.99	13.143±1.720	5.684±1.489
	SVM Linear	88.82±0.97	86.69±1.34	87.70±0.85	87.12±1.00	74.38±1.97	4.623±0.359	2.191±0.121
	SVM Polynomial	53.73±18.68	26.86±9.34	50.00±0.00	33.92±8.45	0.00±0.00	10.814±0.655	5.624±0.192
	SVM RBF	92.20±0.79	91.77±0.60	89.75±1.36	90.65±1.03	81.49±1.95	8.148±0.585	3.245±0.268
VGG19	Bayes	80.38±1.24	85.14±2.37	69.32±1.72	71.64±2.06	52.08±3.79	0.047±0.030	0.068±0.019
	MLP	90.66±1.61	89.94±1.72	87.88±2.31	88.77±2.03	77.78±3.94	21.890±1.504	0.116±0.028
	Nearest Neighbors	86.74±0.50	87.56±0.80	80.79±0.54	83.10±0.60	68.01±1.25	0.044±0.003	9.848±1.696
	Random Forest	87.43±1.21	87.84±1.59	82.13±1.62	84.19±1.59	69.73±3.05	12.120±1.257	8.292±2.995
	SVM Linear	89.37±1.37	87.30±1.50	88.23±1.86	87.71±1.63	75.51±3.27	4.235±0.394	2.032±0.076
	SVM Polynomial	53.73±18.68	26.86±9.34	50.00±0.00	33.92±8.45	0.00±0.00	10.381±0.843	5.611±0.143
	SVM RBF	92.35±1.53	92.33±1.78	89.55±2.02	90.75±1.90	81.82±3.73	7.830±0.676	2.938±0.194

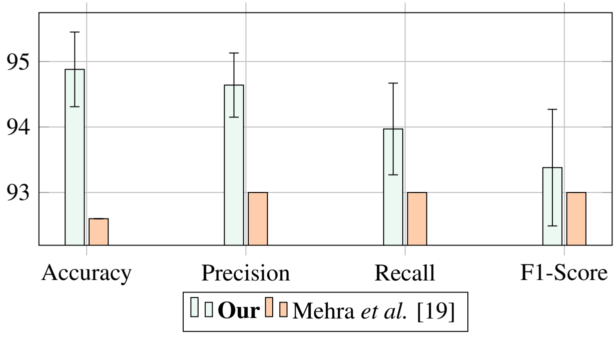
Table 5. Metrics (%), train time (ms) and predict time (ms) obtained by each combination of feature extraction and classifier for the 200X magnification factor. In bold, the best results per metric and the best combination method.

88.00±1.35 for the Matthews Correlation Coefficient (MCC). In general, the results are comparable to the results obtained individually for each magnification factor. If compared, for example, with Table 5, which shows the results of the models for the 200X images, there was a drop in the evaluation metrics. In contrast, for the results of 400X images, shown in Table 6, there was an improvement of 2% in the average accuracy, with a decrease of standard deviation and an even more notable improvement for the others metrics, culminating in an increase of almost 6% MCC with a lower standard deviation of around 50%.

These results are most likely due to the model better discerning images with magnification in the 200X range and starting to confuse images with higher magnification, like 400X. Thus, many authors (mentioned above) still defend the separation of the dataset in question in different magnification factors to find the best way to classify it.

Among cited authors, only Mehra *et al.* [19] and Celik *et al.* [17] made a study with all the images, generalizing the scales of 40X, 100X, 200X and 400X in a single dataset. However, the latter author has restricted himself to the detection of invasive ductal carcinoma (IDC), as mentioned in Section 2, and therefore does not fit in comparison to our results. Mehra *et al.* tested a full or partial training of convolutional neural networks (models VGG16, VGG19 and ResNet50) also with the BreaKHis dataset, but in a balanced and augmented way. Table 9 compares our best model (transfer learning with DenseNet201 + SVM RBF) with the best method among those used by the mentioned author (fine-tuning of VGG16 model, with logistic regression).

Fig. 5. Comparison between our method and the other method from related works per metric. The results below is from using of all dataset, without magnification factor distinction. The other author did not specify the stardard deviation.



Likewise, Figure 5 illustrates the comparison between the methods, shown in numbers in Table 9. By analyzing the figure, it can be seen that, even without fine-tuning the network and without balancing or increasing the dataset, the results obtained by our transfer learning method are superior to the author's results for the metrics in question. The standard deviations of the comparison method were not specified by the author, leaving a gap regarding the validation of the results and the possible

Feature Extractor	Classifier	Accuracy	Precision	Recall	F1-Score	Matthews	Train Time	Predict Time
DenseNet121	Bayes	73.29±1.41	79.34±3.71	59.47±2.01	58.12±2.98	33.27±5.57	0.064±0.053	0.071±0.038
	MLP	89.23±1.91	87.87±2.43	87.46±2.02	87.64±2.16	75.33±4.36	140.998±5.360	0.373±0.183
	Nearest Neighbors	88.74±1.27	88.32±1.42	85.46±1.63	86.65±1.54	73.71±3.00	0.087±0.012	15.624±0.697
	Random Forest	86.92±0.88	86.04±1.01	83.49±1.49	84.52±1.18	69.47±2.19	11.939±1.040	6.299±0.245
	SVM Linear	89.18±0.81	87.55±1.08	87.87±0.60	87.69±0.83	75.42±1.63	7.269±0.156	3.961±0.035
	SVM Polynomial	53.48±17.35	26.74±8.67	50.00±0.00	33.96±7.83	0.00±0.00	15.096±0.184	9.344±0.315
	SVM RBF	92.42±0.64	92.28±0.84	90.22±0.89	91.12±0.77	82.47±1.50	11.703±0.353	4.953±0.378
DenseNet169	Bayes	75.60±1.32	80.17±2.06	63.49±2.00	63.97±2.66	40.28±3.99	0.083±0.028	0.064±0.018
	MLP	90.06±1.01	89.28±1.36	87.81±1.34	88.43±1.18	77.06±2.30	146.267±5.852	0.571±0.242
	Nearest Neighbors	90.33±1.28	89.52±1.09	88.10±2.02	88.73±1.64	77.60±3.11	0.111±0.013	24.793±0.462
	Random Forest	87.37±2.04	85.93±2.30	84.90±2.58	85.34±2.41	70.81±4.77	7.231±0.218	6.666±0.267
	SVM Linear	90.11±1.60	88.43±1.70	89.32±2.14	88.82±1.85	77.74±3.76	11.047±0.114	6.557±0.127
	SVM Polynomial	53.48±17.35	26.74±8.67	50.00±0.00	33.96±7.83	0.00±0.00	24.843±0.219	16.189±0.172
	SVM RBF	92.42±1.72	91.88±1.76	**90.63±2.30**	91.19±2.05	82.49±4.02	13.303±1.016	7.689±0.199
DenseNet201	Bayes	82.53±1.98	86.02±2.03	74.21±2.95	76.69±3.15	59.00±5.00	0.099±0.021	0.113±0.026
	MLP	90.71±1.27	89.90±1.34	88.65±1.77	89.21±1.56	78.53±3.03	84.052±8.840	0.352±0.129
	Nearest Neighbors	90.82±1.20	90.63±0.87	88.15±2.14	89.17±1.60	78.72±2.93	0.146±0.020	27.270±0.638
	Random Forest	89.07±1.93	89.52±2.03	85.12±2.69	86.79±2.47	74.50±4.65	19.350±1.236	9.986±0.552
	SVM Linear	90.71±0.90	89.36±0.98	89.45±1.21	89.40±1.07	78.81±2.13	10.822±0.217	6.512±0.101
	SVM Polynomial	53.48±17.35	26.74±8.67	50.00±0.00	33.96±7.83	0.00±0.00	26.524±0.138	17.781±0.120
	SVM RBF	**92.64±1.14**	**92.41±1.48**	90.60±1.33	**91.41±1.34**	**82.98±2.70**	21.835±0.409	10.720±0.385
VGG16	Bayes	72.25±1.22	79.07±2.96	57.72±1.90	55.29±3.10	29.78±4.59	0.065±0.039	0.088±0.024
	MLP	86.65±1.23	85.16±1.47	83.96±1.68	84.48±1.52	69.11±2.97	17.253±1.703	0.108±0.057
	Nearest Neighbors	84.28±1.16	82.54±1.28	80.88±1.68	81.57±1.46	63.39±2.82	0.042±0.002	9.316±0.315
	Random Forest	85.82±1.33	84.65±1.03	82.24±2.34	83.20±1.89	66.83±3.42	9.322±0.250	6.547±0.507
	SVM Linear	86.54±1.02	84.38±1.21	85.66±1.10	84.92±1.10	70.03±2.18	4.984±0.175	2.443±0.038
	SVM Polynomial	53.48±17.35	26.74±8.67	50.00±0.00	33.96±7.83	0.00±0.00	9.471±0.179	5.108±0.041
	SVM RBF	88.52±1.27	87.09±1.60	86.54±1.34	86.79±1.42	73.62±2.86	5.469±0.259	2.419±0.158
VGG19	Bayes	77.31±1.11	81.62±1.19	66.22±1.81	67.51±2.37	45.22±2.96	0.046±0.036	0.072±0.031
	MLP	87.64±2.03	86.12±2.22	85.40±2.62	85.72±2.41	71.51±4.77	40.986±1.050	0.156±0.061
	Nearest Neighbors	85.71±1.55	85.42±1.74	81.13±2.35	82.68±2.09	66.39±3.86	0.042±0.003	9.172±0.274
	Random Forest	86.98±1.33	87.07±1.49	82.65±2.32	84.22±1.85	69.55±3.30	14.121±0.705	7.529±0.572
	SVM Linear	87.69±1.22	85.88±1.61	86.65±1.79	86.11±1.35	72.50±2.84	4.406±0.190	2.112±0.106
	SVM Polynomial	53.48±17.35	26.74±8.67	50.00±0.00	33.96±7.83	0.00±0.00	9.090±0.178	4.838±0.045
	SVM RBF	89.67±1.00	88.18±1.57	88.60±1.19	88.28±1.05	76.75±2.17	4.721±0.096	2.435±0.053

Table 6. Metrics (%), train time (ms) and predict time (ms) obtained by each combination of feature extraction and classifier for the 400X magnification factor. In bold, the best results per metric and the best combination method.

Table 7. Mean accuracies (%) and their respective standard deviations of our best method (DenseNet201 + SVM RBF) compared to other methods from related works for each magnification factor. The other authors only presented accuracy as evaluation metric.

Method	Magnification Factors			
	40X	100X	200X	400X
Our best method	**94.94±0.87**	94.18±1.34	**95.38±0.40**	**92.64±1.14**
Song *et al.* [20].	87.00±2.60	86.20±3.70	85.20±2.10	82.90±3.70
Zhi *et al.* [18].	93.30±2.30	**94.60±2.20**	94.80±3.20	88.40±4.10
Deniz *et al.* [21].	90.96±1.59	90.58±1.96	91.37±1.72	91.30±0.74

Table 8. Result of the Kolmogorov-Smirnov test for the best method proposed by this study and methods found in the literature with α of 1%.

Method	Magnification Factors			
	40X	100X	200X	400X
Our X Song *et al.* [20].	≠	≠	≠	≠
Our X Zhi *et al.* [18].	≠	≠	≠	≠
Our X Deniz *et al.* [21].	≠	≠	≠	≠

Table 9. Performance of our best method (DenseNet201 + SVM RBF) in comparison to the other, from related works, for a classification experiment using all BreaKHis dataset, without magnification factor distinction. The other author did not present MCC or time values.

Method	Métricas			
	Accuracy	Precision	Recall	F1-Score
Our best method	**94.88±0.57**	**94.64±0.49**	**93.97±0.70**	**93.38±0.89**
Mehra *et al.* [19].	92.60	0.93	0.93	0.93

high levels of standard deviations compromising the real values obtained by the presented study.

6. Conclusions and Future Work

This study presents a method to classify tissue images composed of histopathological exams on four binary scales between benign and malignant. The approach is divided into two stages: I) feature extraction using CNNs using the learning transfer technique, and II) automatic image classification using machine learning methods.

The results show that CNNs, combined with the learning transfer technique and machine learning methods, can be used as resource extractors for this problem. Thus, we can say that the experiments in this study achieved low computational costs, with an accuracy of 95.38% with the DenseNet201 extractor combined with the SVM RBF classifier in the 200x class between malignant and benign, using the BreaKHis database, in addition to F1-Score 93.69% and 10 ms in the mean test time, as shown in Table 5. In this sense, the proposed method using the best model (Model: VVG16 + SVM RBF) brings significant gains for medical applications to aid pre-diagnosis in order to identify tissues in histopathological exams with excellent results.

Therefore, the study focuses on intensifying the analysis, looking for parameters that enable better and better results. For future work, the proposal of this study's model for different tissue images, such as melanoma and different types of skin diseases, is proposed, in order to assess the generalization of the model. for different types of problems and dataset.

Acknowledgments

References

[1] Syed, L, Jabeen, S, Manimala, S. Telemammography: a novel approach for early detection of breast cancer through wavelets based image processing and machine learning techniques. In: Advances in Soft Computing and Machine Learning in Image Processing. Springer; 2018, p. 149–183.

[2] Organization, WH. Breast cancer. ???? URL: https://www.who.int/news-room/fact-sheets/detail/breast-cancer#:~:text=In%202020%2C%20there%20were%202.3,the%20world%27s%20most%20prevalent%20cancer.

[3] De Vrieze, T, Nevelsteen, I, Thomis, S, De Groef, A, Tjalma, WA, Gebruers, N, et al. What are the economic burden and costs associated with the treatment of breast cancer-related lymphoedema? a systematic review. Supportive Care in Cancer 2020;28(2):439–449.

[4] Nuciforo, S, Fofana, I, Matter, MS, Blumer, T, Calabrese, D, Boldanova, T, et al. Organoid models of human liver cancers derived from tumor needle biopsies. Cell reports 2018;24(5):1363–1376.

[5] Pugliese, N, Di Perna, M, Cozzolino, I, Ciancia, G, Pettinato, G, Zeppa, P, et al. Randomized comparison of power doppler ultrasonography-guided core-needle biopsy with open surgical biopsy for the characterization of lymphadenopathies in patients with suspected lymphoma. Annals of hematology 2017;96(4):627–637.

[6] Komura, D, Ishikawa, S. Machine learning methods for histopathological image analysis. Computational and structural biotechnology journal 2018;16:34–42.

[7] Qi, Q, Li, Y, Wang, J, Zheng, H, Huang, Y, Ding, X, et al. Label-efficient breast cancer histopathological image classification. IEEE journal of biomedical and health informatics 2018;23(5):2108–2116.

[8] Bhise, V, Rajan, SS, Sittig, DF, Morgan, RO, Chaudhary, P, Singh, H. Defining and measuring diagnostic uncertainty in medicine: a systematic review. Journal of general internal medicine 2018;33(1):103–115.

[9] Robertson, S, Azizpour, H, Smith, K, Hartman, J. Digital image analysis in breast pathology—from image processing techniques to artificial intelligence. Translational Research 2018;194:19–35.

[10] Liakos, KG, Busato, P, Moshou, D, Pearson, S, Bochtis, D. Machine learning in agriculture: A review. Sensors 2018;18(8):2674.

[11] Chen, J, Ran, X. Deep learning with edge computing: A review. Proceedings of the IEEE 2019;107(8):1655–1674.

[12] Kooi, T, Litjens, G, Van Ginneken, B, Gubern-Mérida, A, Sánchez, CI, Mann, R, et al. Large scale deep learning for computer aided detection of mammographic lesions. Medical image analysis 2017;35:303–312.

[13] Sharma, S, Mehra, R. Conventional machine learning and deep learning approach for multi-classification of breast cancer histopathology images—a comparative insight. Journal of digital imaging 2020;33(3):632–654.

[14] Benhammou, Y, Achchab, B, Herrera, F, Tabik, S. Breakhis based breast cancer automatic diagnosis using deep learning: Taxonomy, survey and insights. Neurocomputing 2020;375:9–24.

[15] Tsochatzidis, L, Costaridou, L, Pratikakis, I. Deep learning for breast cancer diagnosis from mammograms—a comparative study. Journal of Imaging 2019;5(3):37.

[16] Ohata, EF, das Chagas, JVS, Bezerra, GM, Hassan, MM, de Albuquerque, VHC, Reboucas Filho, PP. A novel transfer learning approach for the classification of histological images of colorectal cancer. The Journal of Supercomputing 2021;:1–26.

[17] Celik, Y, Talo, M, Yildirim, O, Karabatak, M, Acharya, UR. Automated invasive ductal carcinoma detection based using deep transfer learning with whole-slide images. Pattern Recognition Letters 2020;133:232–239.

[18] Zhi, W, Yueng, HWF, Chen, Z, Zandavi, SM, Lu, Z, Chung, YY. Using transfer learning with convolutional neural networks to diagnose breast cancer from histopathological images. In: International Conference on Neural Information Processing. Springer; 2017, p. 669–676.

[19] Mehra, R, et al. Breast cancer histology images classification: Training from scratch or transfer learning? ICT Express 2018;4(4):247–254.

[20] Song, Y, Zou, JJ, Chang, H, Cai, W. Adapting fisher vectors for histopathology image classification. In: 2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017). IEEE; 2017, p. 600–603.

[21] Deniz, E, Şengür, A, Kadiroğlu, Z, Guo, Y, Bajaj, V, Budak, Ü. Transfer learning based histopathologic image classification for breast cancer detection. Health information science and systems 2018;6(1):1–7.

[22] Zhu, Z, Albadawy, E, Saha, A, Zhang, J, Harowicz, MR, Mazurowski, MA. Deep learning for identifying radiogenomic associations in breast cancer. Computers in biology and medicine 2019;109:85–90.

[23] Paul, R, Hawkins, SH, Balagurunathan, Y, Schabath, MB, Gillies, RJ, Hall, LO, et al. Deep feature transfer learning in combination with traditional features predicts survival among patients with lung adenocarcinoma. Tomography 2016;2(4):388.

[24] Orenstein, EC, Beijbom, O. Transfer learning and deep feature extraction for planktonic image data sets. In: 2017 IEEE Winter Conference on Applications of Computer Vision (WACV). IEEE; 2017, p. 1082–1088.

[25] Tajbakhsh, N, Shin, JY, Gurudu, SR, Hurst, RT, Kendall, CB, Gotway, MB, et al. Convolutional neural networks for medical image analysis: Full training or fine tuning? IEEE transactions on medical imaging 2016;35(5):1299–1312.

[26] Deng, J, Dong, W, Socher, R, Li, LJ, Li, K, Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. Ieee; 2009, p. 248–255.

[27] Simonyan, K, Zisserman, A. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:14091556 2014;.

[28] Huang, G, Liu, Z, Van Der Maaten, L, Weinberger, KQ. Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2017, p. 4700–4708.

[29] Theodoridis, S, Koutroumbas, K. Pattern recognition, academic press. Burlington, MA[Google Scholar] 2008;.

[30] Rosenblatt, F. The perceptron: a probabilistic model for information storage and organization in the brain. Psychological review 1958;65(6):386.

[31] Haykin, SS, et al. Neural networks and learning machines/simon haykin. 2009.

[32] Aha, DW, Kibler, D, Albert, MK. Instance-based learning algorithms. Machine learning 1991;6(1):37–66.

[33] Ho, TK. The random subspace method for constructing decision forests. IEEE transactions on pattern analysis and machine intelligence 1998;20(8):832–844.

[34] Vapnik, V. N.(1998). statistical learning theory. ????

[35] Xiao, Y. A fast algorithm for two-dimensional kolmogorov–smirnov two sample tests. Computational Statistics & Data Analysis 2017;105:53–58.

[36] Ho, J, Tumkaya, T, Aryal, S, Choi, H, Claridge-Chang, A. Moving beyond p values: data analysis with estimation graphics. Nature methods 2019;16(7):565–566.

[37] Spanhol, FA, Oliveira, LS, Petitjean, C, Heutte, L. A dataset for breast cancer histopathological image classification. Ieee transactions on biomedical engineering 2015;63(7):1455–1462.