International Journal of Electrical and Computer Engineering (IJECE)
Vol. 99, No. 1, Month 2099, pp. 1~1x
ISSN: 2088-8708, DOI: 10.11591/ijece.v99i1.pp1-1x

A Robust Transfer Learning Approach For Colorectal Cancer Identification Based on Histopathology Images

Toto Haryanto[1], Helmi Al Farel[2], Heru Suhartanto[2], Kusmardi[3]
[1]Department of Computer Science, IPB University, Bogor, Indonesia 16680
[2] Faculty of Computer Science, Universitas Indonesia, Depok, Indonesia, 16424
[3]Faculty of Medicine, Universitas Indonesia, Jakarta, Indonesia 10430

Article Info

Article history:

Received month dd, yyyy
Revised month dd, yyyy
Accepted month dd, yyyy

Keywords:

convolutional neural network
colorectal cancer
hyperparameters
transfer learning

ABSTRACT

Diagnosis and treatment of cancer at the benign stage is very important. Recently, pathologists have been using computer-aided diagnostics with machine-learning techniques to diagnose patients from medical images. However, the amount of data required for machine learning is large, but the number of medical images available is usually limited. Transfer learning is a technique that can handle limited amounts of data. Transfer learning is a technique that transfers knowledge gained when learning to solve a problem to use it to solve a different issue. In machine learning, choosing an optimum architecture and hyperparameters is very important because it affects performance. In our experiment, we did a hyperparameter optimization of a CNN that classifies images containing healthy and cancer tissue. The research concludes that CNN with architecture DenseNet121, freeze rate 75%, zero hidden layers on the classifier, learning rate 0.001, and optimizer RMSProp have the best performance with 98% accuracy and 19.5 seconds training time.

Corresponding Author:

Toto Haryaanto
Department of Computer Science, Faculty of Mathematics and Natural Sciences, IPB University
Jl. Meranti Wing 20 Level 5, Dramaga, Bogor 16680, Indonesia
Email: totoharyanto@apps.ipb.ac.id.

1. INTRODUCTION

According to research by the World Health Organization, cancer is one of the main causes of death in the world [1]. Based on the data collected, it is estimated that there will be 19.3 million new cancer cases and 10 million deaths due to cancer in 2020 [2]. Cancer is a condition in which cells grow uncontrollably and spread to other organs. The uncontrolled multiplication of cells can form tumors. Tumors can be benign or malignant. Malignant tumors can spread to other tissues through the lymph node system to form new tumors. The most crucial thing in cancer treatment is detecting it and treating it at a benign stage

Commonly, there are test methods used to detect cancer, especially colorectal cancer, such as colonoscopy and flexible sigmoidoscopy [3]. A pathologist performs this test method by analyzing colorectal tissue in situ using a tool. With the development of computer technology, pathologists are using tools in techniques such as computed tomography to obtain detailed tissue in the form of medical images that can be further analyzed. Currently, there are many studies to analyze medical images using a computer called Computer-aided diagnosis (CAD)[4]. Computer-Aided Diagnosis (CAD) with medical images is increasingly being used to assist pathologists in the process of detecting, diagnosing and prognosing disease [4]. Most of the use of CAD is to analyze cytology images because cytology images can be obtained through different processes minimally-invasive [5]. However, histopathological images provide a more comprehensive picture of the disease and its effects on tissues, because the image collection process preserves tissue structure. Thus, histopathological image analysis is the gold standard in diagnosing diseases such as cancer [6].

Machine learning (ML) is a technology that has been used to help pathologists detect various types of cancer. For example, Zotin et al. [7] extracted various types of structures from X-Ray images and used a neural network to detect lung cancer. [8] used a convolutional neural network to detect colon cancer on histopathological images. Machine learning is a branch of artificial intelligence that allows computer systems to learn from examples, data, and experiences. The challenges and opportunities of machine learning for digital pathology image analysis are comprehensively explained [9]. Machine learning to solve problems that are quite complex, such as creating a model that can accurately detect cancer in medical images. Machine learning requires a lot of data to solve complex problems, whereas medical images are generally available in limited quantities.

To overcome this issue, Haryanto et al. (2021) propose the conditional sliding window (CSW) technique to reproduce data by taking sub-samples from the images in the dataset [10]. The data augmentation method also produces data by giving a transformation to an image, for example, a geometric transformation, such as rotation, or a color transformation, such as adding brightness. In the article by Haryanto et al. (2021), they built a colorectal cancer detection system by training a Convolutional neural network using datasets processed using CSW and augmentation. The technique proposed the resulting model with an accuracy of 81%.

Transfer learning is one technique that is currently being researched to address the issue of limited amounts of data. Transfer learning is a method in which the knowledge gained during the learning process to solve a problem is used in the process of solving different problems. Transfer learning techniques have been successfully applied to create models that can detect breast cancer, cervical cancer [11], skin cancer [12], and lung cancer [13], [14] that exceed the performance of models trained from scratch. The application of machine learning or deep learning can be improved by optimizing the hyperparameter values. There are some research related to the overview of transfer learning, such as [15], [16], and comprehensive literature review [17]. Hyperparameters in transfer learning are variables that control the learning and training process of the model. There are several hyperparameters that, if they have the right values, can improve performance both in terms of accuracy and the required training time. Each of these hyperparameters has a large selection of values, so it is necessary to search for a set of hyperparameter values that produce the model with the best performance. This research will look for combinations of hyperparameter values that produce the best performance in terms of accuracy and training time with transfer learning techniques and analyze the effect of these hyperparameters.

2. METHOD (10 PT)

2.1 Dataset

The dataset used in this study was provided by Warwick University in the 2015 Gland Segmentation Challenge Contest (GlaS) [18], [19]. The Warwick dataset consists of 165 histopathological images taken from 16 colorectal tissues stained with H&E. There are 74 images annotated with benign cancer (Benign) and 91 images annotated with malignant cancer (Malignant). In the GlaS'2015 event, the dataset was divided into three parts, namely Training, Test Part A, and Test Part B. This division will be explained further later. Most of the images have a size of 755x522 pixels, but there are some that have different sizes. Each colorectal tissue used comes from a different patient and the tissue is processed at different occasions. As a result, this dataset has a high level of variation both from the aspect of staining distribution and network architecture. The detail of dataset benign and malignant for colorectal histopathology images shown in Table 1.

Table 1. Warwick Dataset for Colorectal Histopathology

Class Label	**Number data**		
	Training	**Validation**	**Testing**
Beningn	37	33	4
Malignant	48	27	16

Usually, in the process of taking histopathological images, lymph nodes (lymph glands) are taken from the patient's body and placed on a glass slide for analysis by a pathologist [20] .The tissue on the glass slide is then scanned using a high-speed whole slide scanner. This process is called the Whole Slide imaging and results obtained are called whole slide images (WSI). For research and education needs, digitized WSI provides many conveniences [21]. Prior to scanning, the glass slide will be stained with hematoxylin and eosin (H&E). Hematoxylin and eosin give color to tissues so that cell structures such as cell nuclei are easier to see. As can be seen in Figure 3 the results of the staining on a glass slide gives the tissue a pink-purple color. There are variations in color and size for each WSI collection due to differences in lab conditions, scan tools, staining protocols, or tissue types [22].

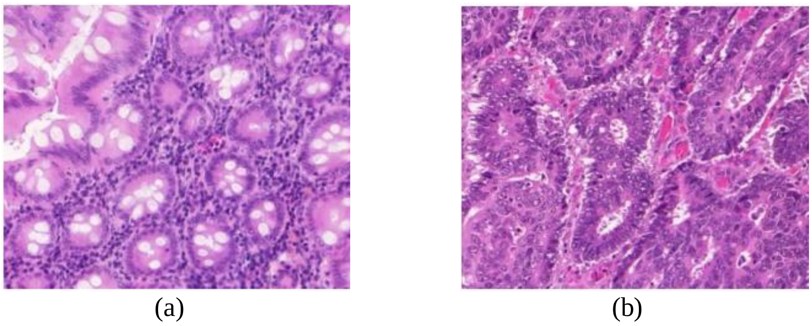

Fig 1. (a) benign tissues (b) malignant tissue

Large datasets are needed to represent population, so the system can generalize. The Warwick dataset has a very limited number, so it is necessary to carry out the Data Generation process. The data generation process is achieved using the Conditional Sliding Window (CSW) technique.This technique produces data by subsampling from the original data, so that the amount of data increases. Subsampling is done by cropping an area of the image and then cropping the next area. The size of the cropped area (window size) is set first. Haryanto et al. (2021) experimented with several different window sizes and different overlap schemes. Previous research found that a window size of 300x300 pixels with an overlap of 50 pixels produces best performance. Overlap is done to reduce loss of information [10]. This study will use the same scheme, namely window 26 size 300x300 with an overlap of 50 pixels. Table 2 shows the amount of data before and after the data generation process using CSW.

Table 2 Result of data generated by Conditional Sliding Windows (CSW)

Class Label	Number data	
	Before CSW	After CSW
Benign	74	430
Malignant	91	532

2.2 Data Augmentation

The data augmentation process is carried out to overcome the limited amount of data further. Data augmentation is a technique for increasing the number and quality of datasets [23]. Some examples of augmentation of image data are geometric transformations, color transformations, random erasing, adversarial training, and neural style transfers (Shorten & Khoshgoftaar, 2019). In this study, geometric transformation and color transformation methods are used to augment the data. The geometric transformation used follows previous research [10]. At the same time, the color transformation used follows [24], who won the CAMELYON'16 Challenge, which is a competition for making a breast cancer detection system. Details regarding the augmentation techniques used and their value ranges are presented in Table 3. We Visualize the result of augmentation techniques in Figure 2

Table 3 Detail applied augentation for colorectal histhopathology dataset

Augmentation Techniques	Values Range
Random Horizontal Flip	{0,1}
Random Vertical Flip	{0,1}
Random Rotation	-30 - 30
Random Adjust Brigness	-0,25 – 0.25
Random Adjust Hue	-0,4 – 0.4
Random Adjust Saturation	-0,25 – 0,25
Random Adjust Contrast	-0,75 – 0,75

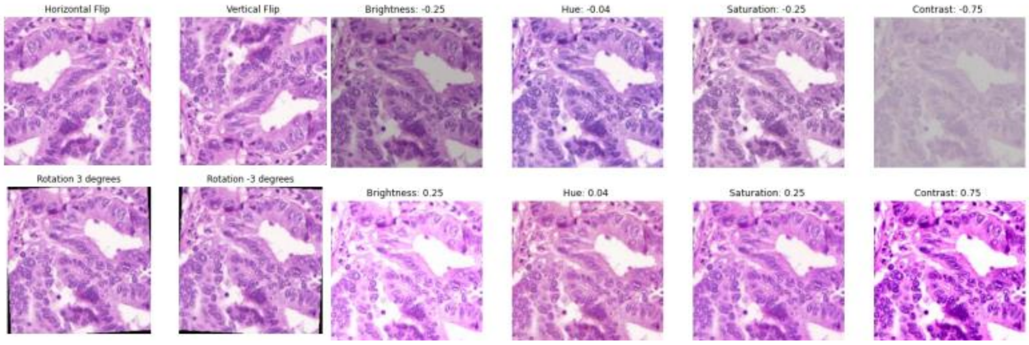

Fig 2. Visualization of augmentation techniques

2.3 Transfer Learning & Hyperparameter Optimization

Transfer learning is the process of using learning that has already been done to solve a problem to solve a different problem. The learning outcomes (optimized parameters) can be used as the initial parameters of another model that solves different problems, so that the model does not need to start with randomly filled parameters. Then this model will be retrained through a re-learning process (fine-tuning) so that it can solve different problems. There are several variables that affect the ability of the model to learn. This variable is called a hyperparameter. Some examples of hyperparameters are architecture, learning rate, optimizer, regularization, batch size, and initial weight. Models that use their parameters as initial parameters for other models need to be trained on a large number of datasets, in another sense, the model must be a robust model. Transfer learning is effective in solving problems that use images as a modality because images generally have similar low-level characteristics. Because of its shape, there are additional properties when CNN uses transfer learning. CNN consists of many layers and in general it can be divided into two parts, namely feature extractor and classifier. The feature extractor section is the part that processes input in the form of images into feature maps. While the classifier is the part that uses the resulting feature maps to produce label output. In transfer learning, a new model is created, and its parameters are initialized with parameters from the learned model to solve another problem. There is an option not to retrain some of the initial/low-level layers due to the nature of low-level similarity, meaning that the parameters in the low-level layers produce a good feature map for new problems and do not need to be retrained. This is called the Freeze rate.

Hyperparameters are very important in machine learning because they directly control the training process and have a significant effect on a model performance machine learning [25]. Hyperparameters are variables that must be determined before the training process begins. The process of finding the most optimal set of hyperparameter values is called hyperparameter optimization [25]. In this research, hyperparameters are included in the process search: CNN architecture, freeze rate, number of layers in the classifier, optimizer, learning rate and initial weight. There are two methods for hyperparameter optimization, grid search and random search. Grid search usually used for small search spaces of hyperparameters, but if we have so many hyperparameters, random search is a more suitable techniques. The specification of architectures for transfer learning adopted in this research shown in Table 4.

Architectures	**Freeze rate**	**Number of fully connected layer**	**Optimizer**	**Learning rate**
DenseNet121 DenseNet161 Inception_v3 MobileNet_v2 ResNet18 ResNet50 ResNext50	0%, 25%, 50%, 75%, 100%	0,1,2	Adam RMSProp SGD	0,01 0,001 0,0001

The hyperparameter optimization process is carried out using the grid search method. The program will build and train the model with every combination of the hyperparameter set to reach the conclusion of the hyperparameter combination with the best performance. Because there is an element of randomness in the learning process, this process is repeated five times so that the conclusions reached are correct. Table 3.4 displays the hyperparameters and their values to be used in the search process. From the table, it can be calculated that the total set of combinations investigated is 945 combinations model candidates.

2.4 Experiment Setup

For each combination, the model will be trained for a maximum of 100 epochs. In each epoch, there are 2 phases, namely the training phase and the validation phase. In the training phase, the model will predict the labels from the input set train, which will then be evaluated using the Cross Entropy function. The loss value will be used to adjust the model parameters with the optimizer algorithm and learning rate. The duration of this phase will be recorded for each epoch, the loss value, accuracy, sensitivity and specificity are also recorded. In the validation phase, the model will again be evaluated with the validation set as input. Loss values, accuracy, sensitivity and specificity are also recorded in this phase. This training process also uses the Early Stopping mechanism.

The training process will stop 5 epochs after the model reaches both sensitivity and specificity evaluations above 95%. This was done because the pre-experiments showed quite a number of combinations that converged in a few epochs. so the authors decided to set the threshold early stopping. After running the last epoch, the model will be filled again with the parameters in the epoch with the smallest loss. The training process uses a batch with size 32. A batch size of 32 will accelerate the step towards convergence but is quite computationally heavy. Each model in this study was trained using the Nvidia DGX A100 system hardware, which is explained in Table 5. The library used to train the model in this study is Pytorch. Pytorch provides many pre-trained models to solve classification problems on ImageNet datasets. Whereas the pretrained model on CAMELYON has a resnet architecture so that hyperparameter optimization is only done with this architecture.

Table 5 Hardware specification for trainng

Hardware	Specification
CPU	Dual AMD Rome 7742
RAM	32GB
GPU	NVIDIA Tensor Core GPU A100 FP64 Tensor Core : 19,5 TFLOPS VRAM 63 GB GPU Memory bandwith : 1,935GB/s

2.5 Model Evaluation

The evaluation phase begins by using a test set as the input model. The last layer of the model consists of 2 neurons which represent labels (benign, malignant). The Softmax function is applied to convert the model output into probabilities of each class. Thus, the label predicted by the model can be concluded as the class that has the greatest probability. From Table 2, it can be seen that the class distribution of each section has a different ratio. In the test section, the amount of data with a malignant class is far more than data with a benign class. In contrast to the training section, which has almost the same ratio, and the validation section, which has more benign data than malignant data. This adds complexity to the problem because the model must be able to ignore distributions (probabilities) in making inferences.

Because the type of problem in this study is binary classification, the determination of evaluation metrics (true positive, false positive, true negative, false negative) can be calculated easily. From these four assessment metrics, measurement of model accuracy performance can be done with accuracy, recall, precision, f1-score, sensitivity, specificity metrics. ROC and AUC metrics can also be calculated using the results of calculating the probability of malignant class test data. The metric used as a reference for model accuracy performance is the harmonic mean which is defined in equation (4). The value of the accuracy metric can be affected by the distribution of the data. The harmonic mean metric is used because the class distribution in the test set is unbalanced, the harmonic mean measures the accuracy of the model as if we were 'blind' to the class distribution.

$$\text{sensivity} = \frac{TP}{TP+FN} \tag{1}$$

$$\text{specificity} = \frac{TN}{TN+FP} \tag{2}$$

$$\text{accuracy} = \frac{TP+TN}{TP+FP+TN+FN} \tag{3}$$

$$\text{harmonic mean} = \sqrt{sensitivity * specificity} \tag{4}$$

2.6 Testing for Real Dataset

To test the performance of the model, we used the model with the best combination of hyperparameters to be tested using original data obtained from the Department of Pathology and Anatomy, Faculty of Medicine, University of Indonesia. The image data has previously been annotated by a pathologist.

3. RESULTS AND DISCUSSION

3.1 Ekspriment Result based on performance accuracy and training time

Table 6 Initial performance Performance

No	Arsitektur - FreezeRate - LayerFCN - Learning rate - Optimizer	Harmonic Mean	Sensitivitas	Spesifisitas
1	inception_v3-50-0-0.01-Adam	0.989	0.979	1.000
2	mobilenet_v2-50-1-0.0001-RMSprop	0.984	0.968	1.000
3	resnext50_32x4d-75-2-0.01-SGD	0.984	0.968	1.000
4	resnext50_32x4d-75-1-0.0001-Adam	0.984	0.968	1.000
5	densenet121-75-2-0.01-RMSprop	0.984	0.968	1.000
6	resnext50_32x4d-50-1-0.0001-SGD	0.984	0.968	1.000
7	densenet161-75-2-0.01-RMSprop	0.984	0.968	1.000
8	mobilenet_v2-75-2-0.001-SGD	0.984	0.968	1.000
9	mobilenet_v2-75-1-0.01-Adam	0.984	0.968	1.000
10	densenet161-75-1-0.01-Adam	0.984	0.968	1.000

Determination of the best model can be done by grouping experimental results based on their accuracy, and looking for the fastest hyperparameter combination at that level of accuracy. With this method, we can determine the point where the trade-off between accuracy and training time is the most optimal. It can be seen in Table 7, the hyperparameter combination densenet121-75-0-0.001-RMSprop has an accuracy difference of only 0.5% with the hyperparameter combination with the best accuracy, namely inception_v3-50-0-0.01-Adam. The train time required by densenet121-75- 0- 0.001-RMSprop is only 19.5 seconds which is 30 seconds faster than the time required by inception_v3-50-0-0.01-Adam. So it can be said that densenet121 -75-0-0.001- RMSprop is one of the best hyperparameter combination candidates.

Table 7 Performance of transfer learning with full training time

No	Arsitektur - FreezeRate - LayerFCN - *Learning rate – Optimizer*	Total training time	*Harmonic Mean*
1	resnet18-75-0-0.0001-Adam	15.9 seconds	0.95
2	resnext50_32x4d-75-2-0.0001-RMSprop	17.2 seconds	0.96
3	densenet121-25-0-0.0001-RMSprop	18.5 seconds	0.97
4	inception_v3-50-0-0.01-SGD	18.6 seconds	0.95
5	resnext50_32x4d-75-2-0.0001-Adam	18.9 seconds	0.96
6	resnext50_32x4d-75-0-0.01-SGD	19.0 seconds	0.96
7	resnext50_32x4d-75-0-0.0001-RMSprop	19.1 seconds	0.97
8	**densenet121-75-0-0.001-RMSprop**	**19.5 seconds**	**0.98**
9	resnet18-50-1-0.0001-RMSprop	19.6 seconds	0.96
10	resnet18-0-1-0.01-SGD	19.7 seconds	0.95

3.2 Combination of the best hyperparamaters

The combination of densenet121-75-0-0.001-RMSprop has a training time of 3 seconds slower than the fastest combination, but with an accuracy difference of 3%. The increase in training time for 3 seconds is not significant compared to the 3% increase in accuracy. So it can be concluded that the combination of densenet121-75-0-0.001-RMSprop is the best hyperparameter combination in terms of accuracy and training time required (Table 8).

Table 8 Combination hyperparameters

Hyperparameter	**Values**
Architecture	DenseNet121
Freeze rate	75%
#layer FCN	0
Learning rate	0.001
Optimizer	RMSProp
Initial weight	ImageNet

3.3 The best model analysis

In this section, the model selected in the previous section will be analyzed more deeply. The analysis was carried out by evaluating the model with metrics such as recall, precision, f1-score.

Table 8 Confusion Matrix

Actual class	Predicted class	
	benign	Malignant
benign	24	0
malignant	3	93

In our Previous research [10] can be used as a baseline. In that study, the model created was named CNN 7-5-7. Table 9 contains a comparison of the performance of the CNN 7-5-7 and our new transfer learning model.

Table 9 Comparison between previous research

Metrics	**CNN 7-5-7**	**Our best** **Transfer learning model**
Accuracy	80.8%	97.5%
Sensitivity/ *recall*	76%	96.8%
Spesificity	100%	100%
Harmonic Mean	88%	98.4%
Presisi	51%	88.8%
F1-Score	61%	92.6%
AUC	90%	97.8%
time per *epoch*	6.68 seconds	2.2 seconds

3.4 Freeze rate analysis to training time

Based on the literature study conducted, not many studies have analyzed the effect of the freeze rate on model performance. CNN consists of many layers and in general it can be divided into two parts, namely feature extractor and classifier. The feature extractor section is the part that processes input in the form of images into feature maps. While the classifier is the part that uses the resulting feature maps to produce label output. A freeze rate of 0% means that all layers in the feature extractor are retrained and a freeze rate of 100% means that no layers in the feature extractor are retrained, in other words the feature extractor section uses the weights from the model trained with different datasets and only the classifier section retrains with the dataset. Warwick. A freeze rate of 25% means that 25% of the feature extractor layers from the very front are frozen (not retrained), and the remaining 75% are retrained. A 0% freeze rate will be used as the basis for performance changes to other freeze rate values. Table 10 shows the impact of freeze rate to the training time for each ransfer learning models.

Table 10 Freeze rate analysis to training time

Model **#trainable parameters**	**Metrices**	**Freeze rate**				
		(0%) **Basis**	**25%**	**50%**	**75%**	**100%**
MobileNetV2 2.226.434	Average total training time	2m0s	1m39s	1m18s	1m11s	2m38s
	difference with base	.	-0m21s	-0m42s	-0m49s	0m37s
DenseNet121 6.955.906	Average total training time	2m13s	1m11s	1m7s	1m16s	3m6s
	difference with base	.	-1m2s	-1m6s	-0m56s	0m53s
ResNet18 11.177.538	Average total training time	1m49s	1m34s	1m5s	0m57s	3m21s
	difference with base	.	-0m15s	-0m43s	-0m51s	1m31s
ResNext50 22.984.002	Average total training time	2m20s	1m13s	1m3s	0m59s	2m40s
	difference with base	.	-1m6s	-1m16s	-1m21s	0m20s
ResNet50 23.512.130	Average total training time	4m55s	2m45s	2m15s	1m53s	3m11s
	difference with base	.	-2m10s	-2m39s	-3m2s	-1m44s
Inception_V3 25.116.362	Average total training time	2m28s	1m41s	1m11s	1m14s	2m37s
	difference with base	.	-0m47s	-1m17s	-1m13s	0m9s
DenseNet161 26.476.418	Average total training time	5m19s	2m17s	2m11s	1m47s	5m7s
	difference with base	.	-3m2s	-3m8s	**-3m32s**	-0m12s

3.5 Freeze rate analysis to performance

This study also examines and analyzes the effect of the freeze rate on the performance of the model being trained. Table 11 shows the effect of the freeze rate on the proposed transfer learning model.

Table 11. Freeze rate analysis to performance of transfer learning

Model **#trainable parameters**	**Metrices**	**Freeze rate**				
		(0%) **Basis**	**25%**	**50%**	**75%**	**100%**
MobileNetV2 2.226.434	Average total training time	91.3%	94.5%	95.8%	96.0%	90.7%
	difference with base	.	3.2%	4.5%	4.8%	-0.5%
DenseNet121 6.955.906	Average total training time	88.2%	95.6%	94.6%	95.6%	89.8%
	difference with base	.	7.4%	6.5%	7.4%	1.6%
ResNet18 11.177.538	Average total training time	87.5%	94.8%	94.9%	95.1%	91.5%
	difference with base	.	7.3%	7.4%	7.6%	4.0%
ResNext50 22.984.002	Average total training time	89.2%	95.2%	94.2%	94.5%	93.6%
	difference with base	.	6.0%	5.0%	5.3%	4.4%
ResNet50 23.512.130	Average total training time	88.1%	93.2%	95.1%	95.2%	93.5%
	difference with base	.	5.0%	7.0%	7.1%	5.4%
Inception_V3 25.116.362	Average total training time	80.8%	88.5%	88.8%	92.4%	86.1%
	difference with base	.	7.7%	8.0%	11.6%	5.3%
DenseNet161 26.476.418	Average total training time	87.0%	95.5%	96.0%	95.7%	94.8%
	difference with base	.	8.5%	9.0%	8.7%	7.8%

3.6 Analysis impact of initial weight

In the transfer learning technique, the weights of the newly created model are filled with the weights of other models (but with the same architecture) that have been optimized to solve specific problems. The optimized weight is specific to solving that specific problem, in other words, other models trained with different datasets will have different optimized weights. So that we can determine that our model is filled with the weight (initial weight) of the model that was trained to solve different problems and datasets. In this study, the weights of the models trained with the ImageNet and Camelyon datasets will be used as initial weights. ImageNet was chosen because this dataset has a large number of classes and a very large amount of data. Camelyon was chosen because this dataset contains histopathological images of breast cancer, which are similar in characteristics to the dataset used in this study.

The discussion of initial weights is separated from other hyperparameters, because only the resnet18 architecture is available with the initial weight Camelyon16. The effect of the initial weight is analyzed using the same method as the freeze rate analysis process, which is to compare the performance between models that have the same configuration but have different initial weights. Of the models that have the Camelyon initial weight, the hyperparameter combinations that produce the best models are shown in Table 12.

Table 12 The best parameters combination for Camelyon initial weight

Hyperparameter	**Nilai**
Arsitektur	ResNet18
Freeze rate	0%
Number of layer FCN	1
Learning rate	0.0001
Optimizer	Adam
Initial weight	CAMELYON

3.7 Comparative analysis of the initial training time and performance of ImageNet and Camelyon

Table 13 shows the average change in model performance that has the same set of hyperparameter values but with different initial weight values at each freeze rate level. It was found that the model with the initial weight ImageNet achieves faster convergence than the model with the initial weight Camelyon at each freeze rate.

Table 13 Comparative analysis of ImageNet and Camelyon

Metrices	**Initial weight**	**Freeze rate**				
		0%	**25%**	**50%**	**75%**	**100%**
Total Training Time	Camelyon	2m0s	3m36s	2m56s	2m23s	3m42s
	ImageNet	**1m49s**	**1m34s**	**1m5s**	**0m57s**	**3m21s**
Accuracy	Camelyon	**88.0%**	92.2%	94.8%	**95.6%**	85.1%
	ImageNet	87.5%	**94.8%**	**94.9%**	95.1%	**91.5%**

Research related to histopathological image-based cancer is still an interesting study to do. Issues regarding limited medical data, performance and training time are some of the points of the researchers. The current transfer learning approach at least answers some of these problems. Various transfer learning architectures that exist today are a challenge for researchers to get models that are robust and ready to be implemented. Histopathological images of colon cancer are relatively difficult to obtain because of the position of the tissue cells in the patient's body. This greatly affects the availability of datasets. Therefore, with transfer learning, this limited data availability is helped by the template model of the previous architecture by implementing transfer learning. However, the optimization process must be carried out to answer the open problems contained in creating specific deep learning models to solve our problems.

The performance of a model is sometimes faced with a trade-off between performance and training time. For this reason, a hyperparameter tuning or optimization process is needed. Of the ten transfer learning architectures trained, the best combination was obtained as reported in Table 7. In terms of architecture, DenseNet121 was the best architectural option chosen with a harmonic mean value of 0.98 and a training time of 19.5 seconds per epoch. While page matters the selected layer to maintain or retrain, using 75% is best.

Transfer learning with fellow cancer data is currently still limited compared to the ImageNet dataset which has a large variety of datasets. In fact, from our research, using the initial weight with ImageNet is still better even when compared to the Camelyon dataset which is actually based on the image of the cancer itself. This can be seen from the performance in terms of time and accuracy in general (Table 13) . On the other hand, the transfer learning model based on cancer imagery is still limited. This is certainly an opportunity as well as a challenge for researchers in the field of computer vision for cancer images.

4. CONCLUSION

This research succeeded in building a CNN model with transfer learning techniques. This research found that the best hyperparameter value in this problem is the model with CNN DenseNet121 architecture, freeze rate 75%, initial weight ImageNet, learning rate 0.001 and optimizer RMSProp. The combination of these hyperparameter values produces a model with an accuracy of 0.98 with a training time of 19.5 seconds. These results surpass the performance of previous research models. In this study, analyzed the effect of freeze rate and initial weight on accuracy and training time as well. We found that a freeze rate of 75% produced the model with the best accuracy and training time. Analysis of the effect of the initial weight shows that using the weight of the model trained with the ImageNet dataset has better performance than the model trained with the Camelyon dataset.

ACKNOWLEDGEMENTS

This research was supported by grant scheme UI-UiTM Bilateral Strategic Alliance (BISA) Matching Grant year 2021-2022.

REFERENCES

[1] World Health Organization (WHO), "The top ten leading causes of death by broad income group," *Fact sheet*, vol. 8, no. 3, pp. 1–5, 2009.

[2] H. Sung *et al.*, "Global Cancer Statistics 2020: GLOBOCAN Estimates of Incidence and Mortality Worldwide for 36 Cancers in 185 Countries," *CA. Cancer J. Clin.*, vol. 71, no. 3, pp. 209–249, 2021, doi: 10.3322/caac.21660.

[3] J. S. Lin, L. A. Perdue, N. B. Henrikson, S. I. Bean, and P. R. Blasi, "Screening for Colorectal Cancer: Updated Evidence Report and Systematic Review for the US Preventive Services Task Force," *JAMA - J. Am. Med. Assoc.*, vol. 325, no. 19, pp. 1978–1997, 2021, doi: 10.1001/jama.2021.4417.

[4] B. Halalli and A. Makandar, "Computer Aided Diagnosis - Medical Image Analysis Techniques," in *Intech*, intechopen, 2018, pp. 85–109.

[5] M. N. Gurcan, L. E. Boucheron, A. Can, A. Madabhushi, N. M. Rajpoot, and B. Yener, "Histopathological Image Analysis: A Review," *IEEE Rev. Biomed. Eng.*, vol. 2, pp. 147–171, 2009.

[6] "Histopathology is ripe for automation," *Nature Biomedical Engineering*, vol. 1, no. 12, p. 925, 2017.

[7] A. Zotin, Y. Hamad, K. Simonov, and M. Kurako, "Lung boundary detection for chest X-ray images classification based on GLCM and probabilistic neural networks," *Procedia Comput. Sci.*, vol. 159, pp. 1439–1448, 2019, doi: 10.1016/j.procs.2019.09.314.

[8] T. Haryanto, I. Wasito, and T. Suhartanto, "Convolutional Neural Network (CNN) for Gland Images Classification," 2017, doi: 10.1109/ICTS.2017.8265646.

[9] A. Madabhushi and G. Lee, "Image analysis and machine learning in digital pathology: Challenges and opportunities," *Med. Image Anal.*, vol. 33, pp. 170–175, 2016, doi: 10.1016/j.media.2016.06.037.

[10] T. Haryanto, H. Suhartanto, A. M. Arymurthy, and K. Kusmardi, "Conditional sliding windows: An approach for handling data limitation in colorectal histopathology image classification," *Informatics Med. Unlocked*, vol. 23, p. 100565, 2021, doi: 10.1016/j.imu.2021.100565.

[11] C. Li *et al.*, "Transfer learning based classification of cervical cancer immunohistochemistry images," *PervasiveHealth Pervasive Comput. Technol. Healthc.*, pp. 102–106, 2019, doi: 10.1145/3364836.3364857.

[12] R. Kalaiarasan, K. Madhan Kumar, S. Sridhar, and M. Yuvarai, "Deep Learning-based Transfer Learning for Classification of Skin Cancer," *Sensors (Switzerland)*, vol. 21, no. 8124, pp. 450–454, 2022, doi: 10.1109/ICAAIC53929.2022.9792651.

[13] M. Humayun, R. Sujatha, S. N. Almuayqil, and N. Z. Jhanjhi, "A Transfer Learning Approach with a Convolutional Neural Network for the Classification of Lung Carcinoma," *Healthc.*, vol. 10, no. 6, pp. 1–15, 2022, doi: 10.3390/healthcare10061058.

[14] G. Ren *et al.*, "A Transfer Learning Framework for Deep Learning-Based CT-to-Perfusion Mapping on Lung Cancer Patients," *Front. Oncol.*, vol. 12, no. July, pp. 1–11, 2022, doi: 10.3389/fonc.2022.883516.

[15] K. Weiss and D. Khoshgoftaar, Taghi M. Wang, "A survey of transfer learning," *J. Big Data*, vol. 3, no. 9, p. 40, 2016, doi: DOI 10.1186/s40537-016-0043-6.

[16] S. Panigrahi, A. Nanda, and T. Swarnkar, "A Survey on Transfer Learning," *Smart Innov. Syst. Technol.*, vol. 194, pp. 781–789, 2021, doi: 10.1007/978-981-15-5971-6_83.

[17] F. Zhuang *et al.*, "A Comprehensive Survey on Transfer Learning," *Proc. IEEE*, vol. 109, no. 1, pp. 43–76, 2021, doi: 10.1109/JPROC.2020.3004555.

[18] K. Sirinukunwattana, D. R. J. Snead, and N. M. Rajpoot, "A Stochastic Polygons Model for Glandular Structures in Colon Histology Images," *IEEE Trans. Med. Imaging*, vol. 34, no. 11, pp. 2366–2378, 2015, doi: 10.1109/TMI.2015.2433900.

[19] K. Sirinukunwattana, S. E. A. Raza, Y. W. Tsang, D. R. J. Snead, I. A. Cree, and N. M. Rajpoot, "Locality Sensitive Deep Learning for Detection and Classification of Nuclei in Routine Colon Cancer Histology Images,"

IEEE Trans. Med. Imaging, vol. 35, no. 5, pp. 1196–1206, 2016, doi: 10.1109/TMI.2016.2525803.

[20] B. E. Bejnordi *et al.*, “Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer,” *JAMA - J. Am. Med. Assoc.*, vol. 318, no. 22, pp. 2199–2210, 2017, doi: 10.1001/jama.2017.14585.

[21] L. Pantanowitz, J. Szymas, Y. Yagi, and D. Wilbur, “Whole slide imaging for educational purposes,” *J. Pathol. Inform.*, vol. 3, no. 1, p. 46, 2012, doi: 10.4103/2153-3539.104908.

[22] T. Inoue and Y. Yagi, “Color standardization and optimization in whole slide imaging,” *Clin. Diagnostic Pathol.*, vol. 4, no. 1, pp. 1–22, 2020, doi: 10.15761/cdp.1000139.

[23] C. Shorten and T. M. Khoshgoftaar, “A survey on Image Data Augmentation for Deep Learning,” *J. Big Data*, vol. 6, no. 1, 2019, doi: 10.1186/s40537-019-0197-0.

[24] Y. Liu *et al.*, “Detecting Cancer Metastases on Gigapixel Pathology Images,” pp. 1–13, 2017, [Online]. Available: http://arxiv.org/abs/1703.02442.

[25] F. Hutter and L. Kotthoff, *84 Automated machine learning*. 2020.

BIOGRAPHIES OF AUTHORS

	Toto Haryanto Toto Haryanto receives Bachelor of Computer Science in 2006 from Bogor Agricultural University, Indonesia and a Master degree in 2011 from the same university. Currently he is PhD candidate at Faculty of Computer Science, Universitas Indonesia. At 2018 he joins visiting a student at Universitya of Pardubice, Czech Republic for three months funded by Erasmus+ Program. His research interest is in high-performance computing for medical images, machine learning and bioinformatics. He can be contacted at email: totoharyanto@apps.ipb.ac.id
	Heru Suhartanto Heru Suhartanto is Full Professor in Faculty of Computer Science, Universitas Indonesia (Fasilkom UI). He has been with Fasilkom UI since 1986. He graduated from undergraduate study at Department of Mathematics, UI in 1986. He receives M.Sc from Department of Computer Science, The University of Toronto, Canada in 1990. His Ph.D in Parallel Computing from Department of Mathematics, The University of Queensland in 1998. His main research interests are Cloud, Grid, High Performance and Numerical Computing. He has won some research grants; holds several software copyrights; published a number of books in Indonesian and international papers in proceeding and journal. He can be contacted at email: heru@cs.ui.ac.id.
	Aniati Murni Arymurthy Aniati Murni is Full Professor at Faculty of Computer Science Universitas Indonesia. She receives a BS in Electrical Engineering, from Universitas Indonesia in 1973. She graduated as Master of Science (MSc) in 1981 from Ohio State Univesity (OSU) in Computer and Information Science. In 1997 his PhD received from Universitas Indonesia with Sandwich Program in Pattern Recognition and Image Processing Lab (PRIP Lab) at Department Computer Science, Michigan State Universit (MSU). Professor Aniati has International more than hundreds international publication in image processing both in satellite images and medical images. Professor Aniati also IEEE member with research interest in image processing, image feature extraction and pattern recognition.. She can be contacted at email: aniati@cs.ui.ac.id
	Kusmardi Kusmardi is Lecturer and Researcher at Pathological Anatomy, Faculty of Medicine, Universitas Indonesia. He is Reviewer of International and National Journal. His research interest on Immunology, Pathology, Cancer Biology, Herbal Medicine, Biomedical Sciences, Oncology, Animal Model for Cancer Research, Biostatistic, and Research Methodology. He can be contacted at email: kusmardi.ms@ui.ac.id.