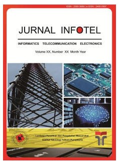

JURNAL INFOTEL
Informatics - Telecommunication - Electronics
Website Jurnal : http://ejournal.st3telkom.ac.id/index.php/infotel
ISSN : 2085-3688; e-ISSN : 2460-0997

Breast Cancer Detection using Residual Convolutional Neural Network and Weighted Loss

Samuel Aji Sena[1], Panca Mudjirahardjo[2], Sholeh Hadi Pramono[3]
[1,2,3] Jurusan Teknik Elektro, Fakultas Teknik, Universitas Brawijaya, Malang
Jl. M.T. Haryono No.167, Lowokwaru, 65145, Malang, Indonesia
Corresponding email:samuelsena@student.ub.ac.id

Received 23 May 2019, Revised 17 June 2019, Accepted 18 June 2019

Abstract — This research presents a breast cancer detection system using deep learning method. Breast cancer detection in a large slide of biopsy image is a hard task because it needs manual observation by a pathologist to find the malignant region. The deep learning model used in this research is made up of multiple layers of the residual convolutional neural network, and instead of using another type of classifier, a multilayer neural network was used as the classifier and stacked together and trained using end-to-end training approach. The system is trained using invasive ductal carcinoma dataset from the Hospital of the University of Pennsylvania and The Cancer Institute of New Jersey. From this dataset, 80% and 20% were randomly sampled and used as training and testing data respectively. Training a neural network on an imbalanced dataset is quite challenging. Weighted loss function was used as the objective function to tackle this problem. We achieve 78.26% and 78.03% for Recall and F1-Score metrics, respectively which are an improvement compared to the previous approach.

Keywords – breast cancer, invasive ductal carcinoma, residual convolutional neural networks, weighted loss, deep learning

I. INTRODUCTION

Breast cancer is one of the most deadly diseases, especially for women. Based on the data from the Ministry of Health Republic of Indonesia. In the year of 2013, 1,4 of 100 women were diagnosed with breast cancer. There were 347,000 cases of breast cancer in the year of 2013 [1]. Data from Badan Penyelenggara Jaminan Sosial (BPJS) shows that in the year of 2014 until 2015, the breast cancer cases were increased [1]. Invasive ductal carcinoma is the most common type of breast cancer because 80% of breast cancer case is invasive ductal carcinoma [2]. Early-stage breast cancer could be cured in most women. Thus, early detection of breast cancer cases is crucial.

Histopathology test is the most common method for breast cancer detection. This method involves an expert to analyze large slides of histopathology images. This is a time-consuming diagnosis since they need to separate malignant cells in a large area of benign cells. Computer-Assisted Diagnosis (CAD) is required to help pathologists with this task. This diagnosis is crucial to choose the best treatment based on the state of that particular breast cancer.

Researchers have been developing multiple algorithms to develop a CAD system for breast cancer. However, the majority of the algorithms are using sophisticated handcrafted features to represent the histopathology image before the classification step. To produce this type of feature, the process is very challenging which usually involving image analysis and multiple complicated pre-processing steps. In the past few years, deep learning is showing a good result in multiple computer vision tasks such as object detection, localization, and semantic segmentation. This deep learning framework is consists of convolutional neural network (CNN) and multilayer neural network. Instead of using handcrafted features, CNN can "learn" the most suitable features directly from the image data, which will be used later for classification process inside the multilayer neural network.

Previous researchers have worked to solve this problem using an in-depth learning approach. Cruz-Roa et al. [4] use only two layers of CNN and three layers neural network as the classifier. Andrew Janowczyk et al. [5] use pre-trained AlexNet CNN to classify the histopathology images. However, improvement is still required especially in the way the model trained on the unbalanced dataset to decrease the false-negative rate and at the same time, increasing the recall.

In this research, we use Residual CNN, which has better performance in image classification tasks compared to conventional CNN and weighted loss in the form of weighted cross-entropy loss as the objective function to solve the unbalanced dataset. We compare the result of this research with the previous work fairly, and it shows that there is an improvement compared to the previous work.

II. RESEARCH METHOD

A. Dataset

We used public IDC datasets from the Hospital of the University of Pennsylvania and The Cancer Institute of New Jersey. It was first introduced by Cruz-Roa et al. [3]. However, the dataset that publicly available is not the same as the original research. The public dataset consists of 277,525 patches of 50x50 pixels biopsy images in RGB color space. The class distribution in this dataset is not balanced. The number of the benign class is 198,738, and the malignant class is 78,786 (Fig.1).

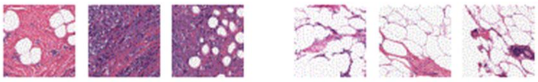

Fig.1. Benign and Malignant Sample in the Dataset

From this dataset, we randomly sampled the dataset into two parts. 80% for training and 20% for testing and do data augmentation in the form of random rotate on every malignant patch the training data and save it into a new file, so the number of data in each class is the same.

B. Data Pre-processing

After the split and augment the dataset, all samples inside it will be converted into YUV, HSV, YCbCr, and Lab color space. We trained our model on five different color spaces, including the original RGB color space. We used min-max feature scaling that scale the value of the input feature into 0 until 1 or -1 until one based on the color space used. The min-max feature scaling follows the equation

$$y = \frac{x_i - \min(x)}{\max(x) - \min(x)} \qquad (1)$$

y is the new feature value after scaled, x_i is the current feature value, $\max(x)$ and $\min(x)$ are the maximum and minimum value of the whole features that will be different for each color space used in the training process.

C. Model Architecture

The Residual CNN [5] introduced by Kaiming He et al. shows better performance in image classification tasks compared to conventional CNN. The residual block allows the use of more in-depth architecture that stacks multiple residual blocks, which consists of a convolution filter and a skip connection. This skip connection adds current features with the features from the previous layer. Based on the original paper, this method produces better performance compared to conventional CNN.

We designed two types of the residual block. The first type is made up of two convolution layers. In each layer, there are 32 units of 3x3 filter with a stride of 1. The residual block type II is a bit different. In this block, we put 2x2 max-pooling layer. Fig.2 shows the scheme of both residual blocks.

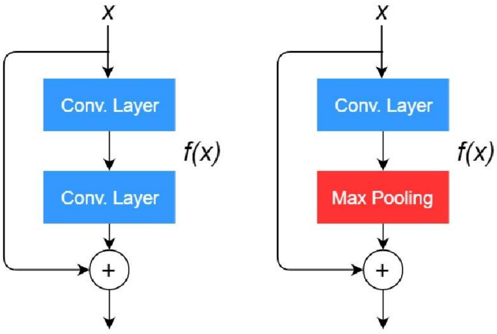

Fig.2. Residual Blocks

In this research, we stack the residual block in some way and use it as a feature extractor. The feature maps produced at the end of this feature extraction layer will be used as the input of the classification layer. Instead of using the *tanh* activation function as in the previous work by Cruz-Roa et al. We use ReLU [6] activation function on all convolution filters in this feature extraction layer. ReLU activation functions follow the equation

$$f(x_i) = \begin{cases} 0, if\ x_i \leq 0 \\ x_i, if\ x_i > 0 \end{cases} \qquad (2)$$

This activation function actually threshold the output to 0 to achieve the non-linearity.

We used a multilayer neural network as the classifier. The feature maps from the feature extraction layer will be flattened first to convert it into a vector form. There are three layers of the neural network inside the classification layer which are input, hidden and output layer. All neurons in input and hidden layer use ReLU activation function, and the output layer uses a sigmoid activation function, which outputs a probability of malignant cell as in this equation below. See Table 1 for the detailed model architecture.

Table 1. Model Architecture

Type	Layer	Features
Feature Extraction	Input	50x50x3
Feature Extraction	Convolution	48x48x32
Feature Extraction	Residual Block I	48x48x32
Feature Extraction	Residual Block II	24x24x32
Feature Extraction	Residual Block I	24x24x32
Feature Extraction	Residual Block I	24x24x32
Feature Extraction	Residual Block II	12x12x32
Feature Extraction	Residual Block I	12x12x32
Feature Extraction	Residual Block I	12x12x32
Feature Extraction	Residual Block II	6x6x32
Feature Extraction	Residual Block I	6x6x32
Classification	Input	1152
Classification	Hidden	256
Classification	Output	1

D. Weighted Loss

Our model was trained to classify benign and malignant class. We use only one neuron in the output layer. This type of output layer uses binary cross-entropy as the loss function. However, considering the class distribution on the dataset is unbalanced, the training process will be very challenging because the model will be biased towards the majority class inside the dataset, in this case, the benign class, which usually indicated by the high false-negative rate or relatively low recall.

To solve this problem, we proposed custom weighted loss in the form of weighted cross-entropy, which add a multiplicative coefficient (θ) into the standard cross-entropy loss. Both standard cross-entropy and weighted cross entropy presented in the following equations

$$CE = -y\log\acute{y} - (1-y)\log(1-\acute{y}) \qquad (4)$$

$$WCE = -y\,\theta\log\acute{y} - (1-y)\log(1-\acute{y}) \qquad (5)$$

y is the target output and $\acute{y}$ is the prediction from the classifier. Based on the equation of weighted cross-entropy, $\theta > 1$ will decrease the false-negative rate and increase recall. On the other hand, $\theta < 1$ will decrease the false positive rate and increase precision. This can be seen in the equation that the coefficient introduced in the positive target of the loss function. In the training process, we will find the optimal coefficient for this dataset.

E. Training Process

This section describes the whole training process. In this research, we split our training process into three phases as the following:

a) In the first phase, we trained our model using standard cross-entropy loss. The purpose of this training was to find the baseline result, which will be used later to find the optimal system.

b) In the second phase, we trained our model using weighted cross-entropy loss. We used 1.1, 1.2, 1.25, 1.3, 1.4, and 1.5 as the coefficient value. At the end of this phase, we observe the recall and F1-Score value for all coefficient. The coefficient that produces the highest F1-Score is the optimal value and will be used in the next training phase.

c) The purpose of the third phase was to find the best color space for this dataset. We trained our model using a weighted loss coefficient from the previous phase, but we convert the RGB dataset to HSV, YUV, YCbCr, and CIELab color before feeding the input images to our model. Same with the second process, we measure the recall and F1-Score. The best color space that produces the highest F1-Score is the final model.

Our training procedure uses a fixed configuration that will be used in all training phases. We use a mini-batch gradient descent method to speed up the training process [7], and for the optimizers we use adaptive Adam algorithm [8]. We used learning rate decay if there is no improvement after some epochs. See Table 2 for the detailed training parameters.

Table 2. Training Parameters

Parameter	Value
Batch Sizes	64
Epochs	32
Optimizers	Adam
Learning Rate	0.001
Learning Rate Decay	Patience = 3 epochs, Decay = 0.5, Minimum Learning Rate = 0.00001
Loss Function	Cross-Entropy (I) Weighted Cross-Entropy (II, III)

We trained our model using Nvidia Tesla K80 GPU. Training time for 32 epochs was more or less 110 minutes for all training phases.

III. RESULT

A. First Training Phase

The purpose of the first phase training was to find the baseline result. We trained our model using training parameters shown in Table 2, but we use standard cross-entropy loss in this phase. After 32 epochs, the training and testing results are shown in Table 3, and unnormalized confusion matrix is shown in Fig.3.

Table 3. First Training Phase Results

Metric	Value
Loss	0.309
Accuracy	87.24%
Precision	77.14%
Recall	71.36%

Metric	Value
Specificity	91.62%
F1-Score	0.7414

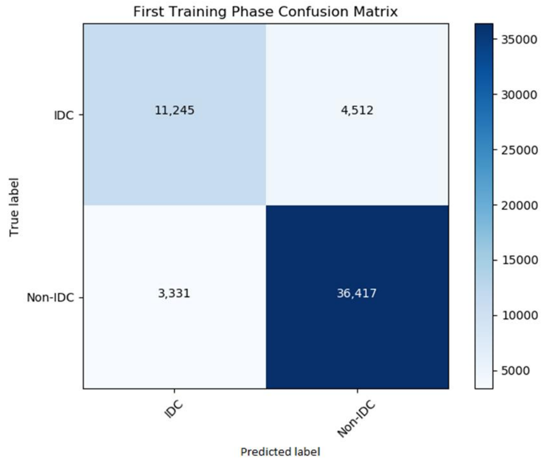

Fig.3. First Training Phase Confusion Matrix

B. *Second Training Phase*

In this phase, we trained our model with the same architecture as the previous training phase but using a different value of the weighted loss coefficient. It starts from 1.1, 1.2, 1.25, 1.3, 1.4 and 1.5. The purpose of this training was to find the best-weighted loss coefficient (θ) which produces the highest recall metric. The results from this training phase shown in Table 4.

Table 4. Second Training Phase Results

θ	Precision	Recall	Specificity	F1-Score
1	77.14%	71.36%	91.62%	0.7414
1.1	79.58%	72.50%	92.62%	0.7587
1.2	76.28%	75.15%	90.74%	0.7571
1.25	75.79%	80.00%	89.83%	0.7784
1.3	75.17%	78.52%	89.72%	0.7681
1.4	75.94%	76.35%	90.41%	0.7615
1.5	78.30%	74.04%	91.86%	0.7611
1.5	78.30%	74.04%	91.86%	0.7611

From this result, we conclude that 1.25 is the best value to use as the weighted loss coefficient, which produces best value of recall and F1-score. We prioritized finding the best F1-Score since the dataset is imbalanced. Then we consider the recall value which represents the model performance on predicting the positive sample. This coefficient will be used in the third training step.

C. *Third Training Phase*

The purpose of the third training phase was to find the best color space for this dataset. We trained our model with the same model architecture as the previous training phase and using 1.25 as the weighted loss coefficient. The color spaces used in this training phase are HSV, YUV, YCbCr, and CIELab. Table 5 shows the result of this training phase.

Table 5. Third Training Phase Results

Color	Precision	Recall	Specificity	F1-Score
RGB	75.79%	80.00%	89.87%	0.7784
HSV	70.28%	79.81%	86.62%	0.7474
YUV	77.65%	77.38%	91.17%	0.7752
YCbCr	77.79%	78.26%	91.14%	0.7803
CIELab	∞	0	1	∞

From this result, RGB still has the highest recall, but YCbCr has a better F1-Score, which is more important in this type of research that uses imbalanced dataset. On the other hand, our model unable to learn something using CIELab color space.

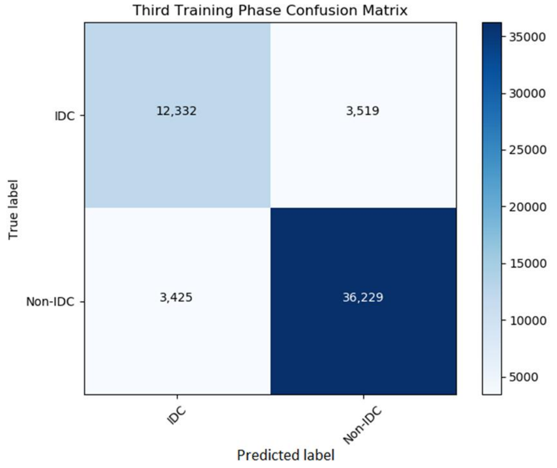

Fig.4. YCbCr Training Confusion Matrix

Figure 4 shows the confusion matrix from the YCbCr training result. There was an improvement compared to baseline results. The true positive rate was increased, and the false-negative also decreased. We chose this color space as the final model in this research, which yielded 78.26% recall and 0.7803 F1-Score. Since the dataset is imbalanced, a small improvement in F-1 Score is important.

Table 6 shows a comparison between recall and F1-Score measured with our method and previous work using the same dataset.

Table 6. Comparison with Previous Work

Method	Recall	F1-Score
Cruz-Roa et al [4]	68.69%	0.7294
Residual CNN	71.36%	0.7414
Residual CNN Weighted Loss	80.00%	0.7784
Residual CNN Weighted Loss (YCbCr)	78.26%	0.7803

IV. DISCUSSION

We gathered some insights from this research. First, our model improved the recall from 68.69% to 78.26% compared to previous work, which is important, considering that in the medical field we need to minimize the false-negative rate. We also find that there is an improvement in F1-Score, from 0.7294 to 0.7803.

Second, the use of YCbCr color space improved the F1-Score, and at the same time, the recall is decreased. By changing color space to YCbCr does not guarantee a significant effect on improving the model performance compared to RGB color space.

Third, our model was failed to learn from CIELab images because this color space has a negative range value, which will get zero output in all filters and neurons in our model if we use the ReLU activation function. Other ReLU variants like Leaky ReLU [9] will work for this type of feature.

V. CONCLUSION

We introduced a novel approach for IDC detection. Our model was made up of residual blocks and a multilayer neural network as the classifier. Our model was trained on imbalanced dataset using weighted cross-entropy loss. We saw an improvement in almost all metrics including precision, recall, and F1-Score. Future works will be focused on finding the best CNN architecture using the same weighted loss approach. Considering that the dataset is unbalanced, anomaly detection is better to approach for this kind of situation.

REFERENCES

[1] Kementerian Kesehatan Republik Indonesia (2017). Kementerian Kesehatan ajak Masyarakat cegah dan kendalikan kanker. Available: http://www.depkes.go.id/article/print/17020200002/kementerian-kesehatan-ajak-masyarakat-cegah-dan-kendalikan-kanker.html.

[2] BreastCancer.Org (2018), Invasive Ductal Carcinoma (IDC). Available: https://www.breastcancer.org/symptoms/types/idc.

[3] A. Cruz-Roa, A. Basavanhally, F. Gonzalez, H. Gilmore, M. Feldman, S. Ganesan, N. Shih, J. Tomaszewski, and A. Madabhushi, “Automatic detection of invasive ductal carcinoma in whole slide images with convolutional neural networks,” in Medical Imaging 2014: Digital Pathology. International Society for Optics and Photonics, 2014, vol. 9041, p. 904103.

[4] A. Janowczyk and A. Madabhushi, “Deep learning for digital pathology image analysis: A comprehensive tutorial with selected use cases,” Journal of pathology informatics, vol. 7, 2016.

[5] Vinod Nair and Geoffrey E. Hinton. “Rectified linear units improve restricted Boltzmann machines” ICML'10 Proceedings of the 27th International Conference on International Conference on Machine Learning. Pages 807-814.

[6] Y. Bengio, “Practical recommendations for gradient-based training of deep architectures,” in Neural networks: Tricks of the trade, pp. 437–478. Springer, 2012.

[7] P.D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” arXiv preprint arXiv:1412.6980, 2014.

[8] Andrew L. Maas, Awni Y. Hannun, Andrew Y. Ng. “Rectifier Nonlinearities Improve Neural Network Acoustic Models”.

[9] Duchi J, Hazan E, Singer Y. Adaptive subgradient methods for online learning and stochastic optimization. Journal of Machine Learning Research. 2011;12:2121-59.

[10] A. Krizhevsky, I. Sutskever, and E.G. Hinton, “Imagenet classification with deep convolutional neural networks,” in Advances in neural information processing systems, 2012, pp. 1097–1105.

[11] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[12] Doyle, S., Agner, S., Madabhushi, A., Feldman, M., and Tomaszewski, J. (2008). “Automated grading of breast cancer histopathology using spectral clustering with textural and architectural image features,”

[13] Dundar, M. M., Badve, S., Bilgin, G., Raykar, V., Jain, R., Sertel, O., and Gurcan, M. N., (2011). “Computerized classification of intraductal breast lesions using histopathological images,”

[14] Niwas, S. I., Palanisamy, P., Zhang, W., Mat Isa, N. A., and Chibbar, R., (2011). “Log-Gabor wavelets based breast carcinoma classification using least square support vector machine.”

[15] M. Rohmatillah, S. H. Pramono, Rahmadwati, H. Suyono, and S. A. Sena, "Automatic Cervical Cell Classification Using Features Extracted by Convolutional Neural Network," *2018 Electrical Power, Electronics, Communications, Controls, and Informatics Seminar (EECCIS)*, Batu, East Java, Indonesia, 2018, pp. 382-386.