International Journal of Sciences: Basic and Applied Research (IJSBAR)

ISSN 2307-4531
(Print & Online)

http://gssrr.org/index.php?journal=JournalOfBasicAndApplied

Image Classification Modelling of Beef and Pork Using Convolutional Neural Network

Salsabila[a*], Anwar Fitrianto[b], Bagus Sartono[c]

[a,b,c]*Department of Statistics, IPB University, Bogor, 16680, Indonesia*

[a]*Email: salsaaay16@gmail.com*, [b]*Email: anwar_stat@gmail.com*, [c]*Email: bagusco@gmail.com*

Abstract

The high price of beef makes some people manipulate sales in markets or other shopping venues, such as mixing beef and pork. The difference between pork and beef is actually from the color and texture of the meat. However, many people do not understand these differences yet. One of the solutions is to create a technology that can recognize and differentiate pork and beef. That is what underlies this research to build a system that can classify the two types of meat. Since traditional machine learning such as Support Vector Machine (SVM) and K-Nearest Neighbor (KNN) uses manual feature extraction in pattern recognition, we use Convolutional Neural Network (CNN) that can extract the feature automatically through the convolution layer. CNN is one of the deep learning methods and the development of artificial intelligence science that can be applied to classify images. There is no research on using CNN for pork and beef classification. Several regularization techniques, including dropout, L2, and max-norm with several values in them are applied to the model and compared to get the best classification results and can predict new data accurately. The best accuracy of 97.56% and the lowest loss of 0.111 were obtained from the CNN model by applying the dropout technique using p=0.7 supported by hyperparameters such as two convolution layers, 128 neurons in the fully connected layer, relu activation function, and two fully connected layers. The results of this study are expected to be the basis for making beef and pork recognition applications.

Keywords: Beef and Pork; Model; Classification; CNN.

----

* Corresponding author.

1. Introduction

The high selling price of beef has resulted in the presence of some sellers to cheat by mixing pork that has a lower selling price with beef. This issue has been going on for several years in Indonesia. Hence, the government must consider the importance of safety and product quality assurance in the market. Food safety and assurance consist of several aspects, such as health, hygiene, and halal labeling [1]. Food products containing pork are strictly prohibited for consumption in countries with a majority Muslim and Jewish population [2]. The difference between pork and beef can be seen mainly in the color and texture. If the pork is paler in color with a smooth meat texture, the meat has a lighter color with a rough texture [3]. However, there are still many people who do not understand these differences. Nowadays, one of many solutions is to use a technique that might directly identify and distinguish the two types of meat. Image classification is one of the techniques in deep learning that can be used in distinguishing the two types of meat. In the development of the image classification method, Convolutional Neural Network (CNN) has been used by researchers as a deep learning classification method with good performance compared to traditional machine learning methods. Traditional Machine Learning algorithms such as SVM and KNN are becoming very popular methods in the classification field and in particular, have excellent performance especially on small-scale data [4]. Especially the KNN is inefficient when used on large-scale data [5]. Then, the feature extraction of the data requires a certain method. In other words, the feature extraction process is done manually before training the model. In contrast, CNN can automatically extract data features through the convolution process. Several studies prove CNN performs better than traditional machine learning methods. CNN produces the highest accuracy among the SVM [6,7] and ANN methods [6]. The development of CNN began with LeNet-05 architecture built by LeCun [8] in the classification and handwriting recognition of bank accounts. The LeNet-05 network is a very simple deep learning network. Deep Learning uses experimentally principles in finding the best hyperparameter in various cases. Hence, LeNet architecture may also be inappropriate in certain cases which results in poor performance of the model. The model is also vulnerable to overfitting, that is, the model was able to work well on training data but not for test data [9]. There are several methods of handling overfitting to produce more optimal model performance. Hence, this study aims to find out the best image classification model of pork and beef images by comparing several regularization methods to overcome the symptoms of overfitting. Image data in this study come from certain parts of beef and pork. Thus, the CNN model obtained may not be suitable for predicting data derived from meat images elsewhere. Therefore, further research could use more data on the complete part of pork and beef to build robust models.

2. Data and Methodology

2.1 Data

This study uses image data of beef and pork. The data were obtained from direct observation by taking pictures of the meat using a smartphone camera. The required data is the image of each type of meat (i.e beef and pork). Both types of meat were obtained from a traditional market in Bogor, the Surya Kencana Bogor market. As much as 90% of the data from both types of meat is the result of the shooting of 250 grams of pork and fresh beef tenderloin, which is divided into several pieces. From several pieces of meat, repeated shots were taken from different angles. While the other 10% of data comes from shooting for meat on the ribs and thighs without separating them into several pieces, but images were still taken repeatedly from different angles. From the shooting, 3000 image data were obtained for the total of both types of meat. Based on data, the researcher used 15 % of the total as test data. therefore, as many as 2550 data are training data, and 450 other data as test data.

2.2 Convolutional Neural Network

CNN is an implementation of a more special Artificial Neural Network (ANN) and is considered the best method for image recognition cases. Some research has proved CNN has a good performance on image classification cases. The research that using CNN for salak sortation has reached 81.45 % of accuracy [10]. Then, CNN has the best performance than SVM on 10 types of food classification [7]. CNN architecture consists of three layers, namely, the convolution layer, the pooling layer, and the fully-connected layer [12]. Figure 1 shows an architecture of CNN that uses two convolutions (C1, C2) and pooling (P1, P2) layers, and two fully connected (FC) layers. There are two main components of the CNN method, feature extraction and classifier. The feature extraction is at the convolution layer and the pooling layer, while the classifier is at the fully connected layer. Specific features of an object can be recognized in the feature extraction [13], while the classifier is used for model learning and find the corresponding label for every test image [14].

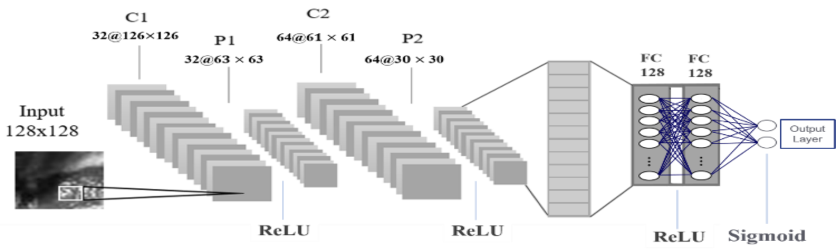


Figure 1: CNN architecture

Every classification case in machine learning is vulnerable to overfitting issues. The implementation of the final CNN model only fits with train data so that it is unable to predict new data. The following is an illustration of overfitting which is showed in the loss classification metric plot.

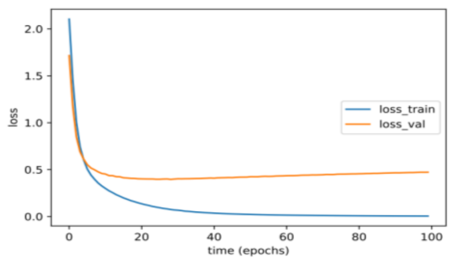


Figure 4: Illustration of overfitting shown in the loss classification metric plot [6]

Figure 4 shows that the loss from the train data decreases and then stabilizes, but the loss from the test data looks consistent away from the loss of train data. Furthermore, the regularization method is usually used in overfit model. There are several regularization methods, namely, dropout and max-norm.

2.3 Regularization Method

Regularization is one way of obtaining a robust model of the overfitting problem, in principle that limits the complexity of the parameters of the model and reduces the sensitivity of the effect on the training data. The most important concept of regularization is to find only useful features or characteristics and eliminate useless features from a network [15].

1. Dropout

The dropout technique can deal with overfitting [16] and is more effective for networks that use ReLU in the hidden layer. Dropout refers to eliminating neurons that are hidden or visible layers in the network. Each neuron will be given a p probability that is worth between 0 and 1. Dropout values in MLP hidden layers typically range from 0.5 to 0.8 [16].

2. Max-norm

One of the main forms of regularization is by limiting the vector of weights that enter into each layer hidden by constant c. The two previous regularizations are applied by adding a decayed form to the loss function. However, max-norm assigns the role by limiting the weights $|w|$ to always be below a constant value c.

2.4 Analysis Procedure

The data analysis steps carried out in this study are as follows:

1. Preprocessing by resizing the image to 128x128 pixels and changing the color channel to greyscale with the following calculation.

$$G_{Average} = \frac{R+G+B}{3} \tag{1}$$

R, G, and B are the values for each component of the Red, Green, and Blue colors in the image. The process of changing the image color channel to grayscale will produce an image that consists of an array and is in the value range 0 (black) - 1 (white) [17].

2. Partitioning data: 85 % train data; 15 % test data.
3. In the CNN model training process, the following is the process:

a. Convolution, i.e multiplication between input matrix and filter kernel matrix. The following is the equation in calculating the value of the feature map resulting from the convolution process if given the image $\in \mathbb{R}^{WxH}$, the convolution of the $f \in \mathbb{R}^{PXQ}$ kernel.

$$(\boldsymbol{X} * f)(i,j) = \sum_{m=0}^{P-1} \sum_{n=0}^{Q-1} \boldsymbol{X}(i+m, j+n) f(m,n) \ , \quad i = 0,..,H-1, j = 0,..,W-1 \tag{2}$$

Where :

$(X * f)\ (i,j)$ = feature map

X = input matrix

f = kernel matrix

m,n = kernel size

W,H = Width, Height (input size)

b. Activating the results of the convolution layer by applying the non-linear function ReLU to the values in the feature map.

$$g(x) = \max(0,x) \tag{3}$$

If the input is negative, the output of the neuron is expressed as 0. Whereas if the input is positive, then the output of the neuron is the input value itself.

c. Max-pooling to reduce input size.

d. Flattening. Get the last max-pooling result into one vector.

e. Apply deep neural network algorithm learning on the fully connected layer to get the best neurons that will be used in the classifier layer. There are two processes in the network, forward and backward propagation.

f. Apply the sigmoid function to get a probability of the classification result. This function converts the input value into a value range of 0 and 1 which are the values used to represent the output class for a binary classification
problem [16].

$$p(y=1|x;w) = g\left(\boldsymbol{wx^T}\right) = \frac{1}{1+e^{-\boldsymbol{wx^T}}} \tag{4}$$

where y is the class, x is the input image, w is weight, and g is the activation symbol of the function.

4. Build a CNN baseline model by inserting the required hyperparameter such as array batch size, optimizer type, number of iterations or epoch, initialization type, kernel filter type, and number of neurons in the fully connected layer. The baseline model is the CNN model without any regularization method applied.

5. Identifying overfitting.

6. Compare several models with regularization methods that can overcome the overfitting of the models.

a. The model uses Dropout regularization
b. The model uses L2 regularization
c. The model uses Dropout and L2 regularization
d. The model uses Max-Norm regularization
e. The model uses Dropout and Max-Norm regularization
f. The model uses L2 and Max-Norm regularization
g. The model uses Dropout, L2, and Max-Norm regularization

7. Evaluate the classification results of the baseline model and models at the sixth point.

A model with high accuracy and small error will be selected as the image classification model of pork and beef. The accuracy, F1-Score, and AUC scores of each model will be compared to get the best results.

2.5 Model Evaluation

During the training process using training data, the performance of the process is evaluated by test data. If the results obtained are not as expected, then an improvement is needed either by using different hyperparameters, different data partitions, or by changing the CNN structure using both data until an acceptable result is obtained. If the test data turns out to be unacceptable, repairs can be made again using training data [18].

1. Confusion Matrix

Confusion Matrix is a metric that can accurately evaluate classification models [18].

Table 1: Confusion matrix

		Actual Labels	
		0	1
Predicted Labels	0	*True Positive* (TP)	*False Positive* (FP)
	1	*False Negative* (FN)	*True Negative* (TN)

Table 1 shows the confusion matrix in the classification with two classes or binary numbers. True Positive (TP), namely the amount of data for the positive class, namely '0' and correctly classified according to the positive class. Meanwhile, True Negative (TN) is the amount of data for the negative class, namely '1' and is classified correctly according to the negative class. Meanwhile, False Positive and False Negative. False Positive is the amount of positive class data that is not classified correctly, while False Negative is the number of negative class data that is not classified correctly. Based on the value of the confusion matrix, there are several important values used to measure the goodness of the model. Some of them are the values for accuracy, sensitivity, specificity, precision, and F1.

- Accuracy

Accuracy in a confusion matrix is the probability for classification class values to be classified correctly with all the values from that class being classified correctly or not.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{5}$$

- Sensitivity and Precision

Accuracy and recall are two very important quantitative measures in a classification model.

$$\text{Precision (P)} = \frac{TP}{TP + FP} \tag{6}$$

$$\text{Recall (R)} = \frac{TP}{TP + FN} \tag{7}$$

The precision is for the calculations on the prediction of the class is positive, whereas the calculation of the actual class is positive. In general, precision describes any accuracy of the prediction result when the class prediction is positive. Meanwhile, considering the probability value for the acquisition of a positive class in the actual sample is accurately predicted as a positive class.

2. F1-Score

One quantitative measure that can combine the precision and recall or their harmonization is the F1 Score.

$$F1 - Score = \frac{2\,PR}{P + R} \tag{8}$$

or

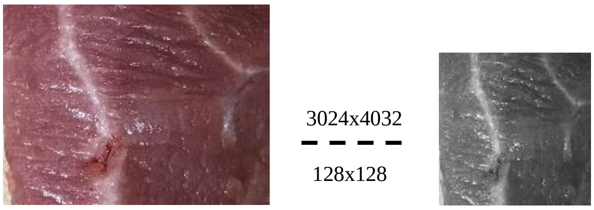
$$F1 - Score = \frac{2\,TP}{2TP + FP + FN} \tag{9}$$

The F1-score is in the range 0 and 1, if the F1-score is equal to 1, the classification obtained is perfect. In practice, the evaluation of the classification model idea using the F1 score is carried out on the test data and can be corrected until a satisfactory F1 score is achieved [18].

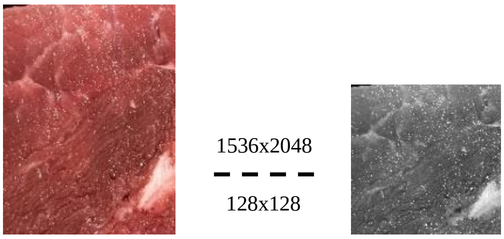
3. Result and Discussion

3.1 Preprocessing Result

The initial image size was uniformly changed to 128x128 pixels. Then, the converted image will have its color channel changed to greyscale. The white color of an image indicates that the pixel value is getting closer to 1. The following is an example of the preprocessing results of each image class.



(a)



(b)

Figure 5: (a) Conversion of a pork image to 128x128 pixels and greyscale;

(b) Conversion of a beef image to 128x128 pixels and greyscale

Figure 5 shows that the image that has been through the resizing process makes the images smaller because it is compressed and not because it is truncated. Therefore, the information of an image will not be lost even after preprocessing. Each image is a matrix that containing pixel values that will then be used as input in the CNN algorithm.

3.2 The result of Feature Extraction in Convolution Layer

The results at stage C1 on Figure 6 show that there are kernels capable of producing a feature map that almost depicts the input image, but there are also those that do not clearly show the pattern on the input image. Then, the results at stage P1 show that the image resolution is getting smaller and the visible pattern is also decreasing. The more convolution is carried out, the less information is generated as seen up to stage P2. Higher layer activation will show less input information. The more activations, the more specific information related to the target will be displayed. In the results of stages C2 and P2, it can be seen that there is a kernel that cannot work optimally so that the 'feature map' is not formed. This is normal for CNN networks, but it would be better if the filter was reduced [19].

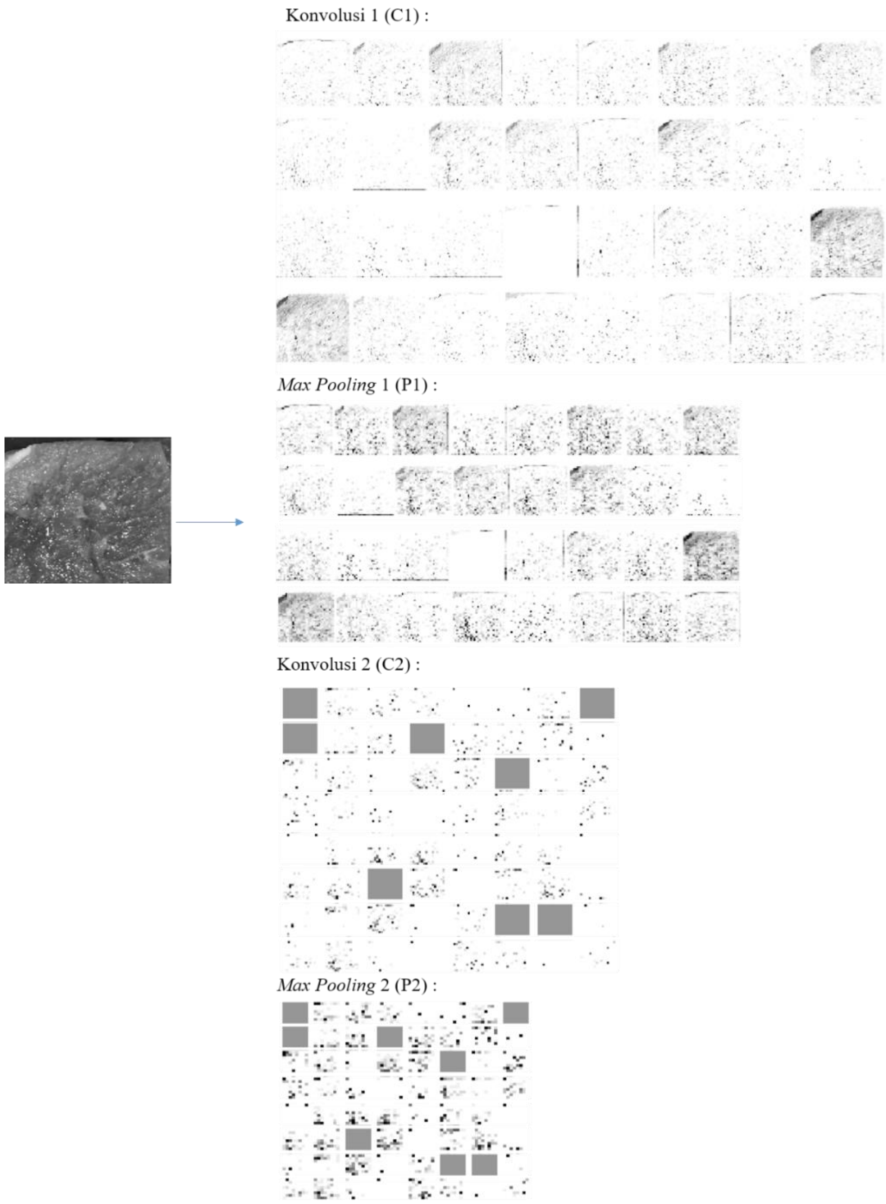


Figure 6: Convolution result of a beef image using 32 and 64 filters

3.3 Baseline Model

The baseline model in this study is a classification model that only uses CNN's main architecture. The following are the classification results using the Baseline model.

Table 2: Confusion matrix of image classification of pork and beef using the baseline model

Prediction	Reference	
	Pork	**Beef**
Pork	217	3
Beef	8	222

Table 2 shows the confusion matrix from the results of the classification of pork and beef images using the CNN Baseline model. The value of 217 represents the number of True Positives (TP) which is the amount of data for a positive class namely '0' or pork and is correctly classified as pork. In contrast, the value of 222 represents the number of True Negatives (TN) which is the amount of data for the negative class namely '1' or beef and is correctly classified as beef. Meanwhile, values 3 and 8 show the number of False Positive and False Negative, respectively. Figure 7 shows the presence of overfitting symptoms characterized by the greater the epoch the greater the distance between plot loss in the training data and the test data. This may result in the constructed classification model being unable to generalize test data. Overall, the loss value generated by this model is 0.216.

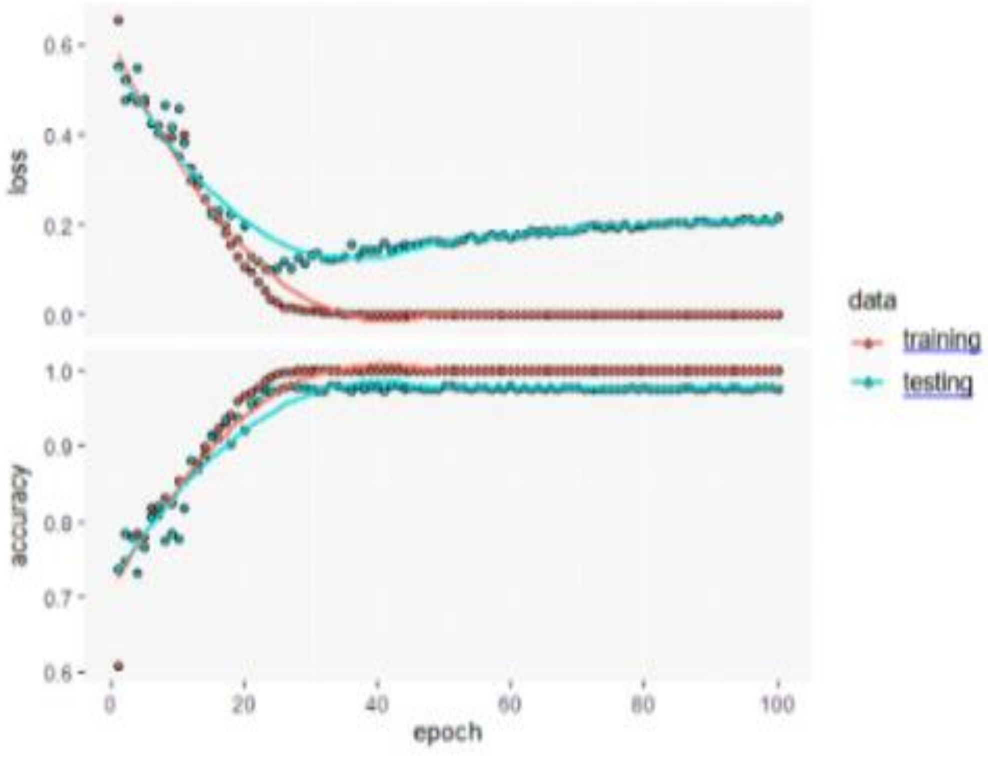


Figure 7: Plot loss and accuracy of the baseline model

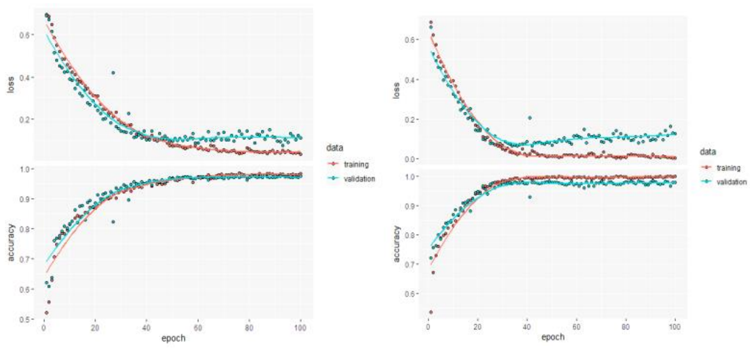
3.4 Summary of Classification Model Performance

Table 3: Model performance based on comparison of regularization in the models

Regularization		Model Goodness Measure			
Dropout ($\boldsymbol{p}$)	MaxNorm (c)	Accuracy	Loss	AUC	F1-Score
p=0,5		0,978	0,126	0,996	0,978
p=0,6		0,973	0,134	0,995	0,974
p=0,7		0,976	0,111	0,996	0,975
p=0,8		0,971	0,112	0,996	0,971
	c=3	0,973	0,136	0,994	0,973
	c=4	0,971	0,160	0,995	0,971
p=0,5	c=3	0,960	0,960	0,991	0,959
p=0,6	c=3	0,958	0,958	0,992	0,957
p=0,7	c=3	0,971	0,190	0,994	0,971
p=0,8	c=3	0,951	0,282	0,987	0,950

This study compared several regularization methods consisting of several values in each method. Based on the results in Table 3, several models can be the best model selection candidates, namely the dropout model (p = 0.5) and the dropout model (p = 0.7). The following is a summary of model performance based on several model comparisons.

Table 3 shows that model dropout ($p = 0.7$) has the lowest error rate. However, the dropout model ($p = 0.5$) produced higher accuracy and F1-Score than the dropout model ($p = 0.7$). Based on Figure 8, the dropout model ($p = 0.7$) shows a smaller gap between training and test losses than the dropout model ($p = 0.5$).



(a) (b)

Figure 8: Classification metric plot, (a) Dropout model ($p = 0.7$); (b) Dropout model ($p = 0.5$)

The lowest error rate and ideal classification metric are the main factors to be achieved in the image classification modeling of pork and beef in this study. This fact is because the model will be able to generalize to new data. Therefore, in this study, the dropout model (p = 0.7) is the best way to classify pork and beef images. The following confusion matrix for the classification result of pork and beef images using the dropout model (p= 0.7).

Table 4: Confusion matrix of image classification of pork and beef using dropout model ($p = 0.7$)

Prediction	**Reference**	
	Pork	**Beef**
Pork	218	4
Beef	7	221

Table 4 shows that the number of images classified correctly by their class 'pork' and 'beef' is 218 and 221 images, respectively. The model performance can be seen in the 97.6% accuracy obtained from the dropout model (p = 0.7). At the same time, the AUC and F1-Score values of the dropout model (p = 0.7) were 99.96% and 97.5%, respectively. Although the accuracy of the dropout model (p = 0.7) and the baseline model is not much different from F1-Score, it turns out that the error rate generated is small, 0.111.

4. Conclusion

The best classification result of pork and beef images was obtained from a model that used dropout regularization with $\boldsymbol{p} = 0.7$ because the best model goodness measure was produced and was able to overcome the overfitting symptoms that occurred in the baseline model. The CNN structure used is two convolution layers, three fully connected layers with hyperparameters such as two convolution layers, Adam optimizer, Glorot Uniform Initializer, ReLU activation function, Max-pooling, 128 neurons in the fully connected layer, 100 epoch, and two fully connected layers. Based on the model, from 450 test data, it is known that 218 images of pork were correctly classified and 221 images of beef were correctly classified. For further research, the researcher able to use more data, compare more types of meat using the CNN method and apply the dropout technique in the build of the CNN model to prevent overfitting.

References

[1]. Y. Erwanto, A. Mohammad Z, E.Y.P, Muslim Sugiyono, and A. Rohman. "Identification of pork contamination in meatballs of indonesia local market using polymerase chain reaction-restriction fragment length polymorphism analysis." Asian Australasian Jornal of Animal Science. vol 27, pp. 1487-1492, 2014.

[2]. S. Rahmati, N.M. Julkapli, W.A Yehye, and W.J Basirun. "Identification of meat origin in food products–a review." Food Control, vol. 68, 2016, pp. 379-390.

[3]. N. Lihayati, R.E Pawening, and M. Furqon. "Klasifikasi jenis daging berdasarkan tekstur menggunakan metode gray level coocurent matrix." in Proc. SENTIA 2016, 2016.

[4]. L. Zhu and P. Spachos. "Towards image classification with machine learning methodologies for smartphones." Machine Learning and Knowledge Extraction, vo.1, pp.1039-1057, 2019.

[5]. P. Dangeti. 2017. "K-Nearest neighbors and naive bayes." in Statistics for Machine Learning, Birmingham, UK: Packt Publishing Ltd, 2017, pp.186-187.

[6]. M. Hasan, S. Ullah, MJ. Khan, K. Khurshid. " Comparative analysis of SVM, ANN and CNN for classifying vegetation species using hyperspectral thermal infrared data." in ISPRS. vol.42, 2019.

[7]. H. Kagaya, K. Aizawa, and M. Ogawa. "Food Detection and Recognition Using Convolutional Neural Network." in Proceedings of the 22nd ACM international conference on Multimedia. pp. 1085–1088, Nov 2014.

[8]. Y. LeCun, K. Kavukcuoglu, and C. Farabet. "Convolutional networks and applications in vision." in ISCAS, 253–256, 2010.

[9]. S. Salman and X. Liu. "Overfitting mechanism and avoidance in deep neural networks." arXiv preprint arXiv:1901.06566, 2019.

[10]. Rismiyati. Implementasi convolutional neural network untuk sortasi mutu salak ekspor berbasis citra digital thesis, Universitas Gajah Mada, Yogyakarta, 2016.

[11]. M. S. Junayed *et al*. "AcneNet - a deep cnn based classification approach for acne classes." in *2019 12th International Conference on Information & Communication Tech, and System*, Surabaya, Indonesia, pp. 203-208, 2019.

[12]. J. Nagi, F. Ducatelle, G. A. Di Caro, D. Cireşan, U. Meier, A. Giusti, F. Nagi, J. Schmidhuber, and L. M. Gambardella. "Max-pooling convolutional neural networks for vision-based hand gesture recognition Signal and Image Processing Applications." in IEEE International Conference, pp. 342-347, 2011.

[13]. M. Alaslani, and L. Elrefaei. "Convolutional neural network based feature extraction for iris recognition." International Journal of Computer Science & Information Technology, vol. 10, pp. 65-78, April 2018.

[14]. I. Goodfellow, Y. Bengio, and A. Courville. "Feedforwards deep networks." in Deep learning, vol. 1, Cambridge, US : MIT press, 2016, pp.176-180.

[15]. Ying. "An overview of overfitting and its solutions." Journal of Physics, vol.1168, 2019, pp.022022.

[16]. N. Srivastava, G. Hinton, A. Krizhevsky, and I. Sutskever. "Dropout: a simple way to prevent neural networks from overfitting." Journal of Machine Learning Research, vol 15, pp.1929-1958, 2014.

[17]. Oles et al. "Package 'EBImage'." Internet: www.bioconductor.org/packages/release/ bioc/manuals/EBImage/man/EBImage.pdf., August. 2020 [Sept. 15, 2020].

[18]. H.H. Aghdam, and E.J. Heravi. "Convolutional Neural Network." In A Guide to Convolutional Neural Network (a practical application to traffic-sign detection and classification), Tarragona, SER: Springer, 2017, pp.106-118.

[19]. K. Cai, W. Cao, L. Aarniovuori, H. Pang, Y. Lin, and G. Li. "Classification of power quality disturbances using wigner-ville distributin and deep convolutional networks." IEEE Access, vol 7, pp.119099-119109. 2019.