JIKO (Jurnal Informatika dan Komputer) Accredited KEMENDIKTI SAINSTEK, No.0173/C3/DT.05.00/2025
Vol. 8, No. 1, April 2025, pp. 56-61 p-ISSN: 2614-8897
DOI: 10.33387/jiko.v8i1.9519 e-ISSN: 2656-1948

CLASSIFICATION OF BEEF FRESHNESS LEVELS BASED ON IMAGE USING CONVOLUTIONAL NEURAL NETWORK

M.Subhan Anshori[1], Fatra Nonggala Putra[2*], Lestariningsih[3], Febbi Senna Lestari[4]

[1,2,3,4]Universitas Nahdlatul Ulama Blitar
Email: [1]subhananshori@gmail.com, [*2]fatranp@unublitar.ac.id, [3]lestariningsih@unublitar.ac.id, [4]ffebbisennal@unublitar.ac.id

(Received: 30 January 2025, Revised: 06 March 2025, Accepted: 09 April 2025)

Abstract

Beef is an essential food commodity with high economic value and a primary source of protein for society. The quality of beef affects consumer preferences, pricing, and market competitiveness. Quality assessment is generally conducted manually through visual inspection and smell, but this method is subjective and time-consuming and requires trained experts. This study aims to design and develop a beef quality classification system using a Convolutional Neural Network (CNN) model based on digital imagery. The dataset used consists of three beef quality categories: Grade 1 (fresh beef), Grade 2 (beef stored at room temperature for 7-14 hours), and Grade 3 (beef stored at room temperature for more than 14 hours). The dataset includes 180 images processed using cropping, resizing, and data augmentation techniques to enhance model variation and accuracy. The CNN architecture employs four convolutional layers with max pooling, followed by dropout and fully connected layers. The model was trained using the Adam optimizer, with a training-to-test data ratio of 80:20. Evaluation results demonstrated the model achieved an accuracy of 97.22%, with precision, recall, and f1-score values of 97.44%, 97.22%, and 97.22%, respectively. These findings suggest that the developed system has the potential to be used as an automatic tool for objective, fast, and accurate beef quality assessment.

Keywords: *Beef, freshness level, Classification, Convolutional Neural Network*

**Corresponding Author: Fatra Nonggala Putra*

1. INTRODUCTION

Beef is one of the food commodities with high economic value and is an important source of protein for humans. Beef quality is one of the main factors that determine the level of consumption, price, and competitiveness in the market [1], [2]. Beef quality assessment criteria include various physical aspects such as color, texture, and freshness. Traditional assessment of meat quality is generally done manually, either by experts or through visual inspection. However, this method is often inconsistent because it is subjective, takes a long time, and depends on the skill of the workers [3]. As technology evolves, artificial intelligence-based methods and digital image processing provide opportunities to improve accuracy and efficiency in the beef quality assessment process. One promising approach is using the Convolutional Neural Network (CNN) algorithm. CNN is known to be effective in image analysis, as it has the ability to extract important features in images automatically. With the CNN model, the process of grading or classifying beef quality can be done based on the physical attributes recorded in the digital images [3], [4], [5], [6], [7].

In this study, beef image data is categorized into three quality levels or grades, namely Grade 1 (fresh), Grade 2 (after 7-14 hours at room temperature), and Grade 3 (more than 14 hours at room temperature). The image data is then processed using preprocessing techniques such as cropping, resizing, and augmentation to increase data variation. The CNN model is designed to use multiple convolution layers and dense layers to optimize the classification of beef quality. This research aims to develop a CNN-based automated system that can accurately classify beef quality based on physical parameters. This system is expected to help related government agencies check the quality of beef to increase work efficiency and reduce dependence on manual methods that are less consistent.

2. RESEARCH METHODS

The stages in this research consist of 6 (five) stages of research shown in Figure 1, namely: 1) Literature Study, 2) Data Collection, 3) Data Preprocessing, 4) CNN Model Building, 5) CNN Classification, and 6) Testing. The research stages are explained as follows:

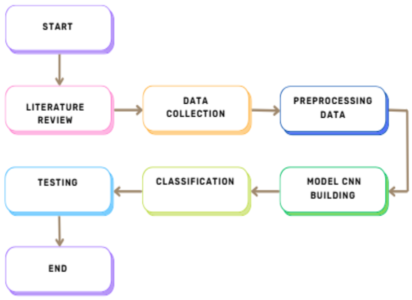

Figure 1. Research Stages

2.1. Data Collection

The population in this study is beef image data. The image data is obtained through direct shooting of beef categorized into three grades, namely Grade 1, Grade 2, and Grade 3. Grade 1 is photographed when the meat is fresh or freshly cut, Grade 2 is photographed when the meat has been at room temperature for 7-14 hours, and Grade 3 is photographed when the meat has been at room temperature for more than 14 hours. The dataset was generated using 12 pieces of meat, where each piece was photographed five times with different meat surfaces displayed. Therefore, each grade has 60 images of data, so the total data sampled in this study is 180 beef image data. The image used is an RGB (Red, Green, Blue) image with *.jpg file format. Data sampling is done randomly to divide the data into training data and test data with a ratio of 80:20, which is 80% training data and 20% test data validation. The amount of training data and test data is as follows.

Table 1. Equipment specifications

	Training Data	Testing Data	Total
Grade 1	48 data	12 data	60 data
Grade 2	48 data	12 data	60 data
Grade 3	48 data	12 data	60 data
Total	144 data	36 data	180 data

2.2. Preprocessing Data

In this research, the image data preprocessing stages are as follows.

a. Cropping, which is cropping or selecting the image area in the part that will be used [8].

b. Resize, which is changing the image resolution size so that it is different from the original image size [9].

c. Data Augmentation, which is a technique to generate a variety of image data through a transformation process on the original image [10]. This stage increases the variety of training data due to the limited number of datasets so that the model performance can be better. In this research, the data augmentation performed is rescaled, rotation range, zoom range, height shift range, width shift range, and flip.

2.3. CNN Model Building

The researchers compiled the architecture of the Convolutional Neural Network model using the layer arrangement shown in Table 2 and Figure 2.

Table 2. The architecture of the CNN model

Layers	Size
Convolution Layer (Conv2D)	16 filter size 3×3 dan activation ReLU
Pooling (Maxpooling)	Filter 2×2
Convolution Layer (Conv2D)	32 filter size 3×3 dan activation ReLU
Pooling (Maxpooling)	Filter 2×2
Convolution Layer (Conv2D)	64 filter size 3×3 dan activation ReLU
Pooling (Maxpooling)	Filter 2×2
Convolution Layer (Conv2D)	64 filter size 3×3 dan activation ReLU
Pooling (Maxpooling)	Filter 2×2
Dropout	0,2
Flatten	-
Dense	64 neuron and activation ReLU
Dense	32 neuron and activation ReLU
Dense	3 neuron and activation Softmax

Based on the layer arrangement, the Convolutional Neural Network model has four convolution layers with a filter size 3×3. The number of filters in this layer are 16, 32, and 64, arranged from the smallest to the largest number of filters. The arrangement of the number of filters is so that larger filters can extract image features from the previous smaller convolutions. In the arrangement of these layers, the Stride value is not specified, so a standard Stride value of 1 is used so that the filter moves one step or pixel when performing convolution operations. Every time one convolution layer is completed, Pooling is performed with the Maxpooling method of 2×2. After that, the process continues with Dropout 0.2, Flatten, and then a Fully Connected Layer. The Fully Connected Layer consists of three Dense layers with the number of neurons 64, 32, and 3. ReLU activation is used in each Convolution Layer and the first two Dense layers, while Softmax activation is used in the last Dense layer because the classification consists of three classes.

Figure 2. Visualization of CNN Architecture

2.4. Classification with CNN

In this research, the classification process is carried out using a Convolutional Neural Network (CNN) model created at the model-building stage. The model used is equipped with Adam optimization as an optimization algorithm, which improves the efficiency and effectiveness of the model training process. Experiments were conducted iteratively based on a predefined procedure by implementing variations in the batch size parameter and the number of epochs. The purpose of these variations is to explore their influence on model performance to obtain the best configuration that produces optimal classification performance.

2.5. Testing

Evaluation of the model's performance to be built in this study uses a confusion matrix to determine the number of correctly and incorrectly classified test data. From the confusion matrix, we can know the performance of the model by calculating the accuracy, precision, recall, and F1-score values. [3].

3. RESULT AND DISCUSSION

This section provides comprehensive information on the research results. The results can be presented in pictures, graphs, tables, and other elements that make it easier for readers to understand and refer to them in the manuscript. If the discussion is too long, sub-subheadings, such as the following example, can be created.

3.1. Preprocessing Data

The Preprocessing stages of beef image data before modeling are as follows.

a. Cropping: in this research, cropping is done on the input image to focus the required object and homogenize its size into a 1:1 ratio, with an example result shown in Figure 3.

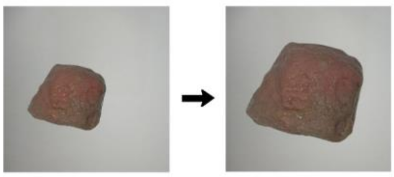

Figure 3. Example of Beef Image Cropping Result

b. Resize. In this research, each input image is resized to 256×256 pixels so that the size is uniform and the classification process is not too heavy. Figure 4 shows an example result.

Figure 4. Example of Beef Image Reize Result

c. Data augmentation consists of rescale, rotation range, zoom range, height shift range, width shift range, and flip. The data augmentation stages carried out in this research include:

1. Rescale is normalizing the image by dividing each pixel by 255 as the largest pixel so that the image pixel value is in the 0-1 range [11]. In this research, the rescale value is 1./255.
2. Rotation range, which rotates the image position randomly based on the specified rotation range [11]. In this research, the rotation range value is 20, indicating a rotation range of ±20°. This means that the image can be rotated up to 20° clockwise and 20° counterclockwise.
3. Zoom range, which enlarges the image display or vice versa based on the ratio range used from the original image [12]. In this research, the zoom range value is 0.2, which indicates a zoom-in and zoom-out range of ±20%. This means the image can be enlarged up to 120% and reduced to 80% of its original size.
4. Height shift range, which shifts the position of the image display vertically [13]. In this research, the height shift range value is 0.1, which indicates a shift range of ±10% of the image height.
5. Width shift range, which shifts the position of the image display horizontally [13]. In this study, the width shift range value is 0.1, which indicates a shift range of ± 10% of the image width.
6. Flip, which is flipping or reflecting the image both vertically and horizontally [14]. In this research, the flip is done horizontally.

In this stage, the fill mode is not included because it is sufficient to use the default fill mode, which is nearest. Fill mode is used to fill in gaps in image pixels caused by image rotation or shift [15]. Data augmentation is performed with "ImagesDataGenerator()" from Keras. In its application, "ImageDataGenerator()" performs data augmentation on the image randomly by combining predefined augmentation types, either partially or completely, with an example result shown in Figure 5.

Commented [fp1]: Adam Optimization

Figure 5. Example of Beef Image Augmentation Result

3.2. Model Building

In this research, the classification process is carried out using the built convolutional neural network model. The model is equipped with Adam optimization to optimize the performance of the training model. The batch size value used are 8, 16, 32 and the epoch value is set at 40,50, and100, which applies early stopping, a technique used to stop the training process when the model has reached a certain condition. In this modeling, the training process will stop when the val_accuracy value has not increased after 40 iterations of the highest val_accuracy.

First Convolution Layer

The first convolution layer with a filter count of 16 performs feature extraction from the input image, resulting in a feature map containing the original image's unique features. The feature map is then activated with ReLU to retain positive values and convert negative values to zero. The convolved feature map is smaller than the input image due to the effect of the filter size used. The size of the convolved feature map can be calculated using the formula where the first convolution becomes (256 - 3) + 1 = 254 so that the feature map is 254×254 pixels. Meanwhile, the number of parameters in the convolution layer can be calculated where the first convolution becomes ((3 × 3 × 3) + 1) × 16 = 448 parameters.

After activation, the process proceeds to the Max pooling operation to reduce the pixel size of the feature map. Max pooling 2×2 indicates that one of the largest pixel values from every four adjacent pixel values in the feature map is taken to retain the important information in the image. Therefore, the feature map resulting from Maxpooling has a pixel size of 50% of the previous size. In the first Maxpooling, the feature map size becomes 127×127 pixels.

Second Convolution Layer

The second convolution is performed with 32 filters to have as many as ((3 × 3 × 16) + 1) × 32 = 4,640 parameters. The feature map resulting from the second convolution has a size of (127 - 3) + 1 = 125 or 125×125 pixels. After activation with ReLU, the feature map undergoes a second Maxpooling so that its size becomes 62×62 pixels.

Third Convolution Layer

The process continues to the third convolution layer with 64 filters so that it has as many parameters as ((3 × 3 × 32) + 1) × 64 = 18,496 parameters. The feature map resulting from the third convolution has a size of (62 - 3) + 1 = 60 or 60×60 pixels. After ReLU activation, the feature map undergoes the third Maxpooling so that its size becomes 30×30 pixels.

Fourth Convolution Layer

The fourth or final convolution is performed with 64 filters to have as many as ((3 × 3 × 64) + 1) × 64 = 36,928 parameters. The feature map resulting from the fourth convolution has a size of (30 - 3) + 1 = 28 or 28×28 pixels. After ReLU activation, the feature map undergoes the fourth Maxpooling so that its size becomes 14×14 pixels.

Dropout

After completing four convolutions, the process continues to the Dropout stage, where some neurons are randomly removed. This stage reduces the dependency between neurons due to the complexity of the Hidden Layer, which can lead to overfitting. The value of 0.2 in Dropout indicates that 20% of neurons will be randomly discarded in each training iteration.

Flatten

Before heading to the Dense Layer (Fully Connected Layer), the feature map, which is a multidimensional array, goes through the Flatten stage to transform it into a one-dimensional array or vector. This stage is necessary because the Fully Connected Layer can only process one-dimensional arrays. The one-dimensional vector size can be calculated where (14 × 14 × 64) = 12,544.

Second Dense Layer

The Fully Connected Layer is the final layer of the CNN model that performs image classification. One-dimensional vectors are input to the first Dense Layer, which has 64 neurons. Each neuron in the layer is connected to all neurons in the input data so that the model can learn complex patterns. Each neuron is then activated with ReLU to produce a 64-dimensional vector. The number of parameters in this layer can be calculated as (12,544 × 64) + 64 = 802,880 parameters. After that, the vector becomes the input for the second Dense Layer, which has 32 neurons and is activated with ReLU. The number of parameters of this layer can be calculated as (64 × 32) + 32 = 2,080 parameters.

Third Dense Layer

Next, the vector becomes the input for the third Dense Layer, which has three neurons and is activated with Softmax. Softmax activation is chosen so that the output of this layer, which becomes the prediction result, is in the form of values 0, 1, and 2. The prediction value is adjusted to the initial case in the form of multiclass classification or classification with more than two classes. The number of parameters of this layer is (32 × 3) + 3 = 99. Thus, this model's total number of parameters that can be trained is 865,57. The summary of the defined model is shown in Figure 6. Graph of the movement of loss and accuracy values for training and validation data is shown in Figure 7.

Layer (type)	Output Shape	Param #
conv2d_4 (Conv2D)	(None, 254, 254, 16)	448
max_pooling2d_4 (MaxPooling2D)	(None, 127, 127, 16)	0
conv2d_5 (Conv2D)	(None, 125, 125, 32)	4,640
max_pooling2d_5 (MaxPooling2D)	(None, 62, 62, 32)	0
conv2d_6 (Conv2D)	(None, 60, 60, 64)	18,496
max_pooling2d_6 (MaxPooling2D)	(None, 30, 30, 64)	0
conv2d_7 (Conv2D)	(None, 28, 28, 64)	36,928
max_pooling2d_7 (MaxPooling2D)	(None, 14, 14, 64)	0
dropout_1 (Dropout)	(None, 14, 14, 64)	0
flatten_1 (Flatten)	(None, 12544)	0
dense_3 (Dense)	(None, 64)	802,880
dense_4 (Dense)	(None, 32)	2,080
dense_5 (Dense)	(None, 3)	99

Total params: 865,571 (3.30 MB)
Trainable params: 865,571 (3.30 MB)
Non-trainable params: 0 (0.00 B)

Figure 6. The summary of the defined model

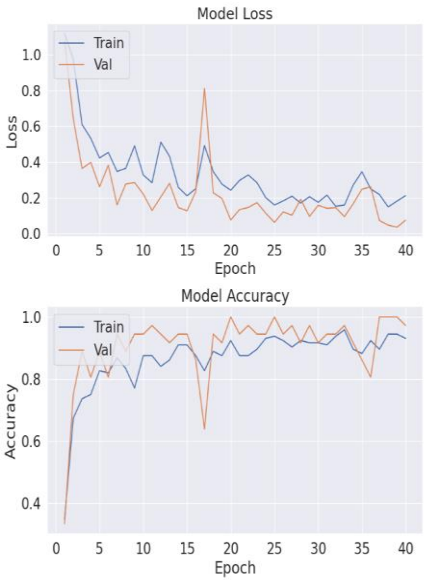

Figure 7. Line plots of Loss and Accuracy of CNN Model by Epoch

3.3. Testing

The testing or evaluation stage is carried out using the Confusion Matrix method, which measures the value of accuracy, precision, recall, and F1-score using testing data in the form of 36 beef images, each class totaling 12 Grade 1, Grade 2, and Grade 3 images. Based on the test results presented in Figure 8 and the confusion matrix below, it can be concluded that Grade 1 and Grade 2 images can all be predicted correctly; however, 1 data point of Grade 3 images is improperly predicted as Grade 2.

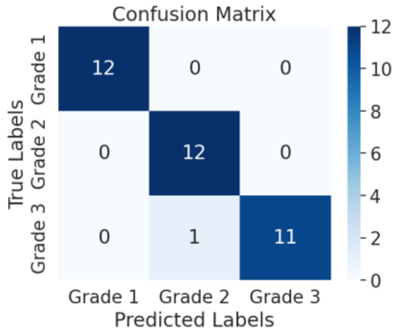

Figure 8. Confusion Matrix of Beef Classification Test Results

	precision	recall	f1-score	support
Grade 1	1.0000	1.0000	1.0000	12
Grade 2	0.9231	1.0000	0.9600	12
Grade 3	1.0000	0.9167	0.9565	12
accuracy			0.9722	36
macro avg	0.9744	0.9722	0.9722	36
weighted avg	0.9744	0.9722	0.9722	36

Figure 9. Classification Report Beef Testing

Based on the value calculation in Figure 9, the Classification report, it can be seen that the accuracy, precision, recall, and f1-score values have an accuracy value of 97.22%, precision of 97.44%, recall of 97.22%, and f1-score of 97.22%.

4. CONCLUSION

Beef freshness classification was conducted using self-shot beef image data with 180 images consisting of 144 training data and 36 testing data divided into three classes: Grade 1, Grade 2, and Grade 3. Each class has 36 training data and 12 testing data. Based on the test results, the model built has an accuracy value of 97.22%, precision of 97.44%, recall of 97.22%, and f1-score of 97.22%. These results prove that the model formed using CNN can work well in grading the freshness level of beef.

5. REFERENCES

[1] P. D. dan S. I. Pertanian, "Analisis Kinerja Perdagangan Daging Sapi," vol. 14, no. 1G, pp. 1–67, 2024.

[2] C. Felderhoff *et al.*, "Beef quality preferences: Factors driving consumer satisfaction," *Foods*, vol. 9, no. 3, pp. 1–22, 2020, doi: 10.3390/foods9030289.

[3] P. B. Asmoro and A. Solichin, "Penerapan Metode Convolutional Neural Network Untuk Klasifikasi Kualitas Daging Sapi Pada Aplikasi Berbasis Android," *Fakt. Exacta*, vol. 16, no. 4,

pp. 286–298, 2024, doi: 10.30998/faktorexacta.v16i4.19564.
[4] A. Rizky pratama, “Klasifikasi Daging Sapi Berdasarkan Ciri Warna Dengan Metode Otsu dan K-Nearest Neighbor,” *Techno Xplore J. Ilmu Komput. dan Teknol. Inf.*, vol. 6, no. 1, pp. 9–18, 2021, doi: 10.36805/technoxplore.v6i1.1239.
[5] S. Bagas Valentino, “Klasifikasi Kualitas Daging Marmer Berdasarkan Citra Warna Daging Menggunakan Metode Convolutional Neural Network,” *JATI (Jurnal Mhs. Tek. Inform.*, vol. 7, no. 1, pp. 125–129, 2023, doi: 10.36040/jati.v7i1.6128.
[6] T. Ekamila, F. Rahayu, A. Zuchriadi, and A. Octa Indarso, “Penerapan Deep Learning Untuk Klasifikasi Kesegaran Daging Sapi Berbasis Mobile Apps,” *Edu Komputika J.*, vol. 10, no. 1, pp. 10–16, 2023, doi: 10.15294/edukomputika.v10i1.68478.
[7] D. N. Gonçalves *et al.*, “Carcass image segmentation using CNN-based methods,” *Inf. Process. Agric.*, vol. 8, no. 4, pp. 560–572, 2021, doi: 10.1016/j.inpa.2020.11.004.
[8] A. Antoni, T. Rohana, and A. R. Pratama, “Implementasi Algoritma Convolutional Neural Network Untuk Klasifikasi Citra Kemasan Kardus Defect dan No Defect,” *Build. Informatics, Technol. Sci.*, vol. 4, no. 4, pp. 1941–1950, 2023, doi: 10.47065/bits.v4i4.3270.
[9] R. A. Pangestu, B. Rahmat, and F. T. Anggraeny, “Implementasi Algoritma CNN untuk Klasifikasi Citra Lahan dan Perhitungan Luas,” *J. Inform. dan Sist. Inf.*, vol. 1, no. 1, pp. 166–174, 2020.
[10] L. H. Ganda and H. Bunyamin, “Penggunaan Augmentasi Data pada Klasifikasi Jenis Kanker Payudara dengan Model Resnet-34,” *J. Strateg.*, vol. 3, no. 1, pp. 187–193, 2021.
[11] R. W. Wiratama, “Implementasi dan Klasifikasi Jenis-Jenis Batik Menggunakan Algoritma Convolutional Neural Network (CNN) dengan Model Arsitektur Resnet,” Politeknik Negeri Malang, 2023.
[12] M. F. Naufal and S. F. Kusuma, “Pendeteksi Citra Masker Wajah Menggunakan CNN dan Transfer Learning,” *J. Teknol. Inf. dan Ilmu Komput.*, vol. 8, no. 6, pp. 1293–1300, 2021, doi: 10.25126/jtiik.2021865201.
[13] R. Shinta, “Klasifikasi Citra Penyakit Daun Tanaman Padi Menggunakan CNN Dengan Arsitektur VGG-19,” Universitas Islam Negeri Sultan Syarif Kasim Riau, 2023.
[14] M. Toyib, T. D. K. Pratama, and I. Aqil, “Penerapan Algoritma CNN untuk Mendeteksi Tulisan Tangan Angka Romawi dengan Augmentasi Data,” *J. Mat. Ilmu Pengetah. Alam, Kebumian dan Angkasa*, vol. 2, no. 3, pp. 108–120, 2024.
[15] G. P. H. P. Gusti, E. Haerani, F. Syafria, F. Yanto, and S. K. Gusti, “Implementasi Algoritma Convolutional Neural Network (Resnet-50) untuk Klasifikasi Kanker Kulit Benign dan Malignant,” *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 4, no. 3, pp. 984–992, 2024, doi: 10.57152/malcom.v4i3.1398.