

# Digital Image Processing to Determine Weight and Classification of Cow Weight with Deep Learning

Army Hudan Zhain\*, Mohamad Dional Guitara, Bambang Urip Hidayatulloh, Reza Adi Wibowo, Helmy Faisal Muttaqin

Informatics Department, Faculty of Engineering, Widyatama University, Bandung, Indonesia

\*zhain.hudan@widyatama.ac.id

## ABSTRACT

One of the cattle, cows, are the first commodity to produce meat for public's consumption. Both of the quality and quantity of the meat of a cow are affected by the weight of the cow itself. The more a cow weighs means a better indicator for economy in a cattle ranch. A cow's weight also involves in the production and reproduction process. Farmers are still using the conventional way in weighing the cows. Overcoming the problem, a solution that works as an alternative for the conventional cow weighing tool is needed. To be more practical, the AI (Artificial Intelligence) can be implied to help find the solution. The AI field to use is the Deep Learning with Convolutional Neural Network algorithms method. Based on the trial several scenarios, highest accuracy achieved on 0.89 by using EfficientNet without implementing Sobel and Canny.

## Keywords

Deep Learning, Convolutional Neural Network, Sobel, accuracy, Canny Edge, OpenCV, digital image processing

## Introduction

Cows are one of the cattle that are economically potential and quite promising [1]. The meat and dairy products are public's main consumption. To produce a great product, the cow itself has to be in a great condition, both quality and quantity. Both of the quality and quantity of the meat of a cow are affected by the weight of the cow itself. The size of a cow plays an important role in a cattle ranch. A cow's weight also involves in the production and reproduction process. The growth of cattle can be measured by the increase on the weight while the increase on the size of the body can be identified by measuring the height, length and chest [2].

Many farmers are still using the conventional way of weighing cattle. The body weight is also an indicator that signifies productivity value and the success of the ranch management [3]. The problem that often occurs in measuring the body weight of the cow is the unavailability of the corresponding measuring tools. The measurement has to be precise as it is crucial for both farmers and customers. It is important for both parties to know the weight of the cattle as the price refers to its size; the bigger the size the more expensive the price of the cattle. However, the consistency of the price is often blur to the actual size of the cow. This confusion basically happens because the

customer does not know the exact weight of the cow. To clear up the dishonesty, a solution to measure or at least estimate the weight of the cow in a more effective and practice way is needed. The AI field can be applied to overcome this problem.

Using the help of Digital Image Processing, the weight can be measured using image segmentation by separating the background image with the objects outside of the background (the image of the cow) that are classified as noise. Digital Image Processing can analyze the image and estimate the measurement of the weight. The application of Digital Image Processing can determine the weight of the cow by using Canny Edge Detection Method and Sobel Operator. This AI tool can also classify whether the cow is fat or thin by using Convolutional Neural Network Algorithms.

## Literature Review

### Digital Image Processing

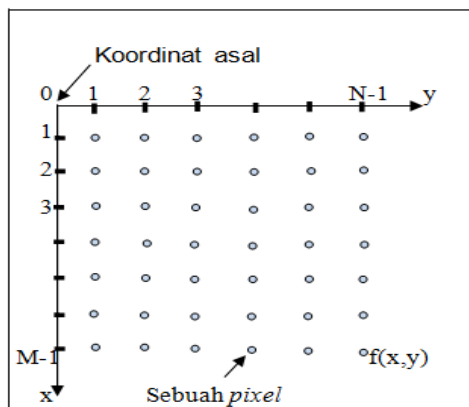
#### Definition

Images can be defined as a two-dimensional function,  $f(x, y)$ , where  $x$  and  $y$  are spatial coordinates with  $(x, y)$  value is often called intensity [4]. Digital image is an image of  $f(x, y)$

that has gone through digitalization both in terms of area coordinates and intensity values. A digital image consists of several elements, which each of the element has its own location and value. Each element is called picture elements or pixel [5]. Digital Image Processing is a field that studies image processing techniques. The contexts for ‘images’ here are photos and moving photos (the ones that come from webcam). While, digital means that the process is done digitally by utilizing computers [6].

**Digital image representation**

In computers, each pixel value is represented by two integers to indicate the location in the image plane. For example, coordinates (0,0) are used for the top left corner of the image and coordinates (m-1, n-1) are used for the bottom right corner in an image that is m x n pixels. In short, it is better known as ‘image’. A thing is called ‘image’ after it has been converted to digital. An example of an image is a file with \*.png or \*.jpg extension. The picture below shows the coordinate of digital image to two-dimensional (x, y) axis [5].



**Figure 1.** Coordinate of digital image

Mathematically, digital images can be written in the form of matrixes as seen below:

$$f(x, y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,M-1) \\ f(1,0) & \dots & \dots & f(1,M-1) \\ \dots & \dots & \dots & \dots \\ f(N-1,0) & f(N-1,1) & \dots & f(N-1,M-1) \end{bmatrix}$$

**Figure 2.** Mathematically, digital images in the form of matrixes

**Digital Image Format**

There are two files format that are often used in image processing; bitmap and vector images. The explanation for the two file formats as follows:

1. **Bitmap Image**

Bitmap image is also called raster image. Bitmap image saves the data of image codes digitally and thoroughly. Bitmap image stores its data in pixels. The representation of bitmap image is matrix or mapped in binary numbers or other number systems. This image has advantages in the manipulation of colors, but a shortage in modifying objects. Bitmap can visualize the fineness of gradations of shadows and colors of an image, but a zoom on the image will result in broken pixels. Therefore, bitmap is recommended for images that contain complex color gradations, such as photos and digital paintings. Bitmap images can be obtained using Scanner, Digital Camera, Video Capture, etc.

2. **Vector Image**

Vector image is an image generated from mathematical calculations that contain no pixels, a data stored in a form of position vector where the stored data are just the information of position vectors and the form of a function. Inversely proportional to bitmap image, vector image finds it difficult to modify colors, but an ease on forming objects by changing the value. Therefore, zooming in or out an image will result a relatively good quality image with minor-to-no changes. Vector images are generally generated using applications such as CorelDraw, Adobe Illustrator, Macromedia Freehand, AutoCAD, etc. [6].

**Types of Digital Image**

Based on the color value storage format, digital images are divided into three types, namely:

1. **Binary Image**

Binary image is an image that contains only two greyscale values: black and white. Pixels valued with 1 signify black color while pixels valued with 0 signify white color. Expressed in a function:  $f(x, y) \in \{0,1\}$

The example of binary image and the representation value of each pixel is shown at the Figure 1 [7]:

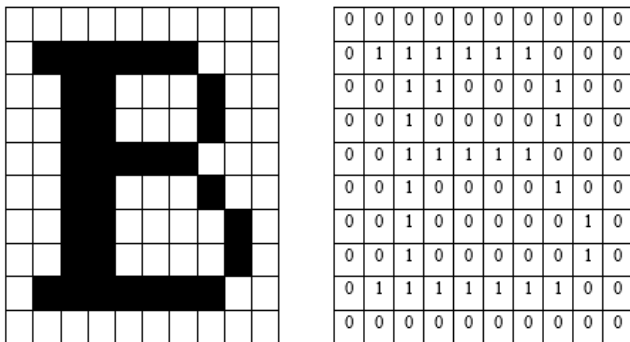


Figure 3. Binary image representation

2. Greyscale Image

An image that consists of one color of layer with certain greyscale degree. For most digital 8-bit images, the Grey-scale system is measured based on the brightness intensity valued from 0 – 255 where the solid black is 0 and the whitest is 255. Expressed in a function [7]:  $f(x, y) \sum [0 \dots 225]$

3. Color Image

Color image, or commonly called RGB image, is a type of image that presents colors in the form of components R (red), G (green), and B (blue). Each component of the color uses 8-bit (the value is ranging from 0 to 255). Thus, the possibility of the presence of the color is up to 255 x 255 x 255 or 16,581,375 colors. The following Table 1 shows the example and value of R, G, and B coloring [8].

Table 1. The colors and values of R, G, and B

Warna	R	G	B
Merah	255	0	0
Hijau	0	255	0
Biru	0	0	255
Hitam	0	0	0
Putih	255	255	255
Kuning	0	255	255

Image Resolution

Image resolution deals with the detail level of an image. The higher the resolution, the higher the detail level of the image. The unit in measuring image resolution can be a physical size in the form of the number of lines per mm/number of lines per inch or it can also be the size of the entire image with the number of lines in the image. The

resolution of an image can be measured in the following ways [9]:

1. Pixel Resolution
2. Spatial Resolution
3. Spectral Resolution
4. Temporal Resolution
5. Radiometric Resolution

Pixel Resolution

Pixel resolution is a calculation of the number of pixels in a digital image. A digital image with height of N pixel and width of M pixel has a resolution of M x N. Pixel resolution will give two integer numbers that consecutively represent the number of width pixels and the number of height pixels of the image.

Another definition of pixel resolution is the result of multiplying the number of pixels in width and height and then dividing them by 1 million. This kind of pixel resolution can usually be seen in Digital Cameras. An image with width of 2,048 pixels and height of 1,536 pixels will have a total of 2,048 x 1,536 = 3,145,728 or 3.1 mega pixels. Another calculation states the resolution in pixel per inch. This unit represents the number of pixels that are 1 line inch in the image [9].

Computer Vision

In fact, computer vision is trying to imitate humans' visual system. In the process of human vision, human sees objects using the sense of sight i.e. pair of eyes, then the image of the object is transmitted to the brain to be interpreted so that humans understand what objects that was seen. The result of the interpretation is later used to make decision [7].

Table 2. The Outline of Human Vision and Computer Vision

Human Vision	Computer Vision
Utilizing eyes and visual cortex in the brain.	Utilizing cameras that are connected to the computer systems.
Finding out from the image the object, the position, the movement, and the shape.	Automatically interprets images and tries to decipher the contents as in human sights.

Actually, computer vision is a technology that is capable of recognizing objects in observation. This capability comes from the combination of image processing and pattern recognition. Image processing is the first process of a computer vision to produce a better or easier-to-interpret images, while pattern recognition is an object identification process of an image. The processes of computer vision can be divided into:

1. Image Acquisition
2. Image Processing
3. Image Analysis
4. Image Understanding [10]

Computer vision is a combination of image processing and pattern recognition. Computer vision is the development of an explicit and blatant description of physical object of an image. The output of computer vision is a description or interpretation or some structural quantitative measurements in 3D scenes [11].

### Edge Detection

Edge is a sudden change in the intensity value of the degree of greyscale that happens in a short time. The aim of edge detection is to improve the presence of border line of an object in an image so the border between the two areas with different grey level can be set. The operators that can be operated in edge detection are [12]:

- a. Sobel
- b. Prewitt
- c. Canny
- d. Roberts
- e. Laplacian of Gaussian
- f. Zero Cross

### Canny Edge Detection

This method is one of the algorithms for edge detection. Canny Edge Detection was found by Marr and Hildreth who did a research on the modeling of human visual perception, that was later developed by John F. Canny in the 1986 using multi-phase algorithms to detect various edges in an image. The strengths of this method is the capability to reduce noise before counting edge detection which will result on more edges discovered. Canny edge detection basically operates as follows:

1. Image Refinement

In this phase, Gaussian Blur is commonly used.

2. Calculating the Potential Gradient

Image gradient is the operator that is the closest to the definition of an edge.

3. Median Blur

Median blur is an image refinement process by utilizing median filter. Median filter is really effective to remove noise but still maintaining the detail of the images because it does not depend on different values.

4. Sharp

Sharp is an image sharpening process by convoluting the original image by multiplying the original image's pixels with kernel sharp.

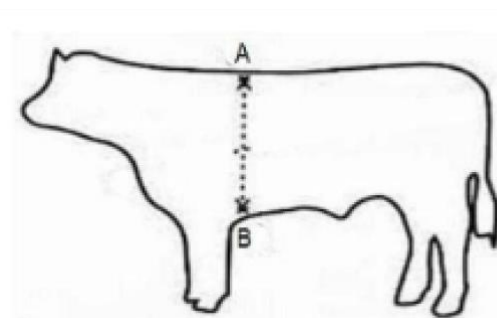
5. Cow Weight Estimation Formula

In determining the weight of a cow, the frequently used formula is Schrool Formula that is shown in the equation below:

$$\text{Berat Badan} = \frac{(LD+22)^2}{100}$$

**Figure 4.** Schrool formula

The data needed involves the chest size of the cow. Chest Circumference is measured in a circular position behind the bulge of the cow's shoulder at the top and back of the front leg [13].



**Figure 5.** The chest circumference of cattle cow

6. Non-maximal Suppression

Non-maximal suppression aims to remove gradient potential of a pixel from the edge candidates only if the pixel is not the local of the edge direction on that pixel position (this time the gradient direction is needed).

7. Hysteresis Thresholding

Hysteresis Thresholding is a classification using two values; High-threshold and Low-threshold.

8. Segmentation Process

Image segmentation is the reduction of an image into objects or regions, such as separating objects that are different from one to another.

### Convolutional Neural Network

Convolutional Neural Network is of the neural networks that is frequently used in image data and it can detect as well as recognizing the object of an image. Basically, CNN is similar to another neural networks. CNN consists of neurons that have weight, bias and activation function. However, the factor that makes it different is the architecture that is differentiated into two; Feature Extraction Layer and Fully-Connected Layer.

Feature Extraction Layer is a changing process of an image into features that is in the form of numbers that interpret the image. Feature Extraction Layer consists of two parts; convolutional layer and pooling layer.

Convolutional layer consists of neurons that are arranged in various ways that they form a filter with height and length (pixels). For example, the first layer of a feature extraction layer mostly is a convolutional layer with the size of 5x5x3. With the length of 5 pixels, height of 5 pixels, and width of 3 pixels due to the channel of the image. These three filters will be shifted to every part of the image. In every shifting, the “dot” operation will be executed between the input and value of the filter so it will result in an output that is commonly called activation map or feature map.

Whereas, pooling layer consists of a filter that has certain size and stride which will shift to every area of the feature map. The poolings that are commonly used are Max Pooling and Average Pooling. For example, if the Max Pooling in use is in the size of 2x2 with stride value of 2, the maximum filter shifting value in this 2x2 pixels area will be chosen while the Average Pooling will choose the average value of it.

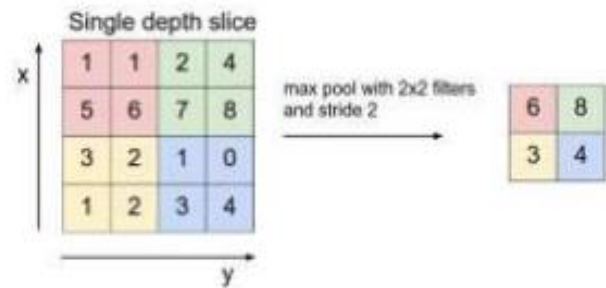


Figure 7. Max pooling

The purpose of using the pooling layer is to reduce the dimension of the feature map, thereby speeding up computation because fewer parameters need to be updated and resolving overfitting.

The next process, if the feature map (from the result of the feature extraction layer) is still in a form of multidimensional array, the feature map has to be reshaped into a vector so it can be used as an input of the fully connected layer. Fully connected layer is an MLP that has some hidden layers, activation function, output layers, and loss function.

The next important thing to know from CNN is Stride. Stride is a parameter that decides the total filter shifting needed. If the value of the stride is 1, it means convolutional filter will shift horizontally 1 pixel and consecutively vertically 1 pixel. The smaller the stride means the more detailed the information that we can gather from an input, but it requires more computation compared to bigger strides. Put in mind that using smaller stride will not always result in a more detailed information.

Padding or Zero is a parameter that determines the number of the pixels (contains 0 value) which will be added from every side of the inputs. This is

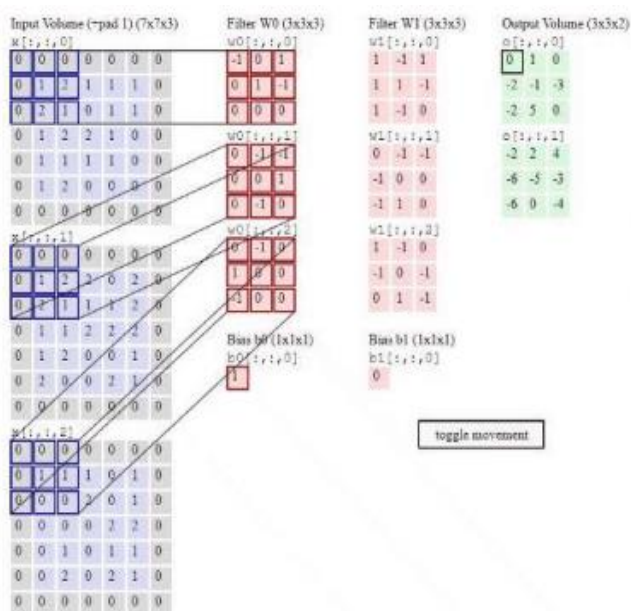


Figure 6. Feature map

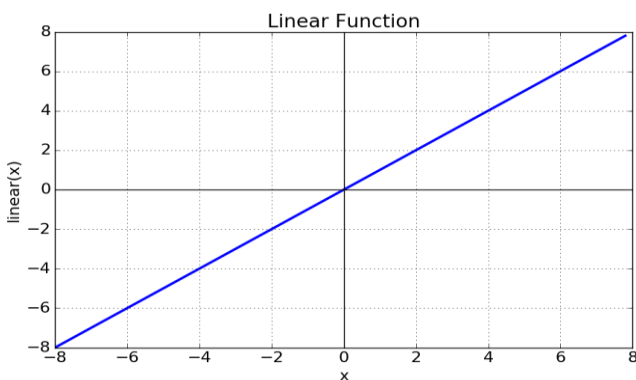
used to manipulate the dimension of the output of the convolutional layer (feature map) [14].

**Activation Function**

Activation function is a node that is added to the output of every neuron. This is also known as Transfer Function. This function can also be attached on between the two Neural Networks. Activation function is used to determine the output of the artificial neurons. Activation function makes a mapping of the value generated between 0 to 1 or -1 to 1 etc. (based on the function). Basically, action function can be divided into two types; linear activation function and non-linear activation function [15].

**Linear Activation Function**

Its function is for borders or linear. Therefore, the output of this function will be limited only to certain range.

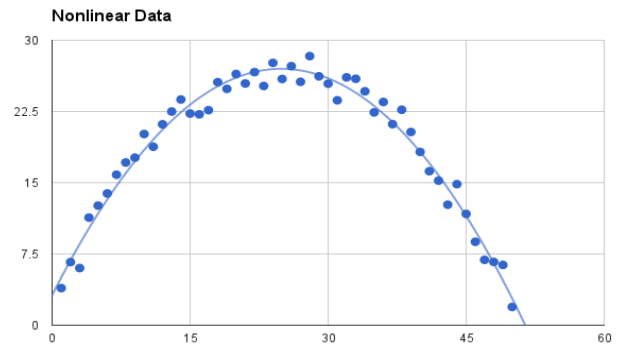


**Figure 8.** Linear function

The equation of activation function on linear function:  $f(x) = x$  with range:  $-\infty$  to  $\infty$ . Linear activation function does not help with the complexity or various basic data parameters that is being thrown to the artificial neural network [15].

**Non-Linear Activation Function**

This is the most frequently used activation function. The non-linear graphic can be seen on the figure below:

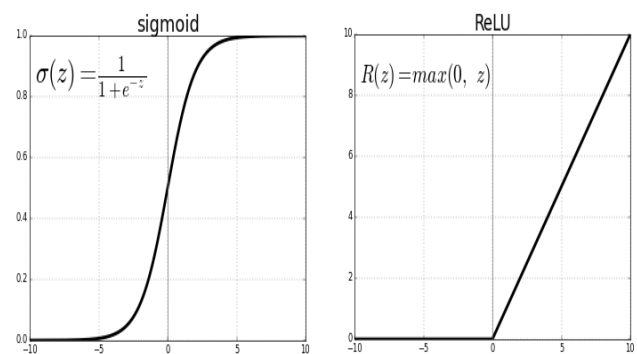


**Figure 9.** Non-linear activation function

Non-linear activation function puts the model for generalization or adaptation with various data on ease and differentiates between the outputs. The main terminology of this function for this function is Derivative or Differential and Monotonic. The function of Derivative or Differential: make changes on y-axis w.r.t. and x-axis. This is also known as tilting and Monotonic function is a function that is so varied that it never decreases or increases. Some of the activation function such as Sigmoid, Tanh, ReLU, and Leaky ReLU are included [15].

**A. ReLU (Rectified Linear Unit)**

ReLU is an activation function that is currently most used anywhere in the world because it is used in almost all convolutional or deep learning neural networks.



**Figure 10.** ReLU

As seen above, ReLU is half-repaired (starting from the bottom). If  $f(s)$  is zero,  $z$  is less than zero,  $f(z)$  is equal to  $z$ , and  $z$  is above or equal to zero. Range: (0 to infinity) both of the function and its derivation are monotonous. The actual problem is the sudden change of all negative values to zero which decreases the capability of the model to match properly. It means that every

negative input on the ReLU function activation suddenly becomes zero on the graphic, which in turn will affect the output of the graphic by not placing the negative value on the right place [15].

## Methodology

### System Overview

The classification system that is based on the weight of the cow that is developed using Convolutional Neural Network algorithms needs data inputs that are in a form of collection of images of cows. This system will apply the advanced Deep Learning technique. Therefore, a minimum of 30 dataset images is needed in the classification system. Those dataset images will go through the training phase which is the core process of deep learning technique. In this phase, the system will execute learning thoroughly until it meets the epoch or iteration which has been set beforehand. After the training phase, a model will be acquired which later will be used to classify the weight of the cow – whether it is fat or thin. Besides that, researchers, on this system, add implementation in a form of Graphic User Interface to maximize the visual or model from the cow weight classification system.

As for the estimation of the weight of the cow in this research, the images of the cows will be taken from a cattle ranch in Cirebon. The images will go through digital image processing process using Canny Edge Detection method and Sobel Operator. Thus, the measurement of the chest size

of the cow can be measured from the taken images which later will be added to the Schrool formula and the weight of the cow can be acquired.

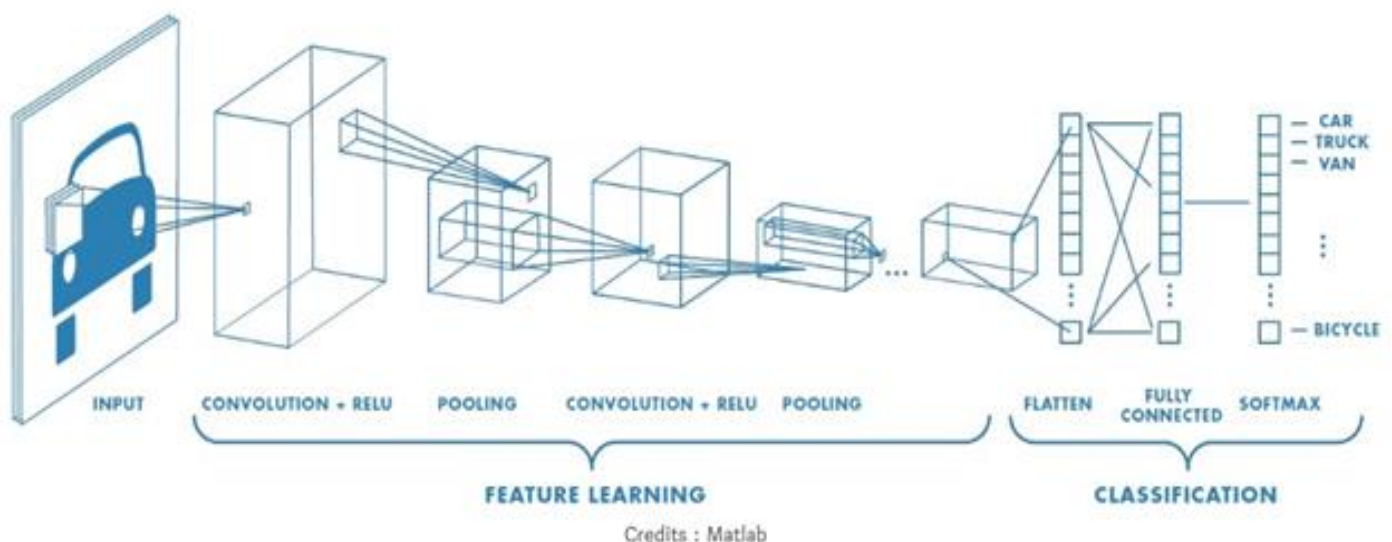
### System Planning

System planning is the main phase of this research. The planning used by researchers are divided into three; input, processing, and output.

The input is images with pixel size of 128x128 with a total of ~100 images with 80 images taken from the internet for dummies and 20 images for trial process. The processing that was done is divided into some steps; images input process, sliding window, area extraction and feature computation on areas using the convolutional neural network algorithms. The output of this planning is image classification acquired from the images whether the cow is fat or thin by displaying the percentage of accuracy in the form of Graphic User Interface.

### Computer Network Architecture

Convolutional Neural Network algorithms requires architecture design in the training process. This design consists of several network parts; image for detection, input neuron, convolution + activation (ReLU) + pooling layer, fully connected layer, classification and detection output are the requirements. The following image visualizes the design architecture of Convolutional Neural Network:



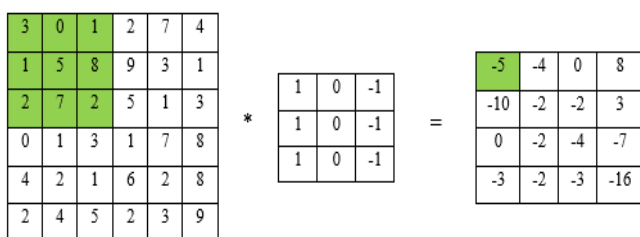
**Figure 11.** Architecture of Convolutional Neural Network

As seen above, we can identify that some parts of it are inputs of the image which later will be detected thoroughly. The picture above is used for a dummy with a size of 128x128 pixels and using RGB coloring as much as 3 channels hence the total calculated input neuron is 49,152 neurons from the calculation of 128x128x3. Each neuron has its own parameter value with each parameter on that networking is ranging from 0 to 255.

**Convolutional Layer**

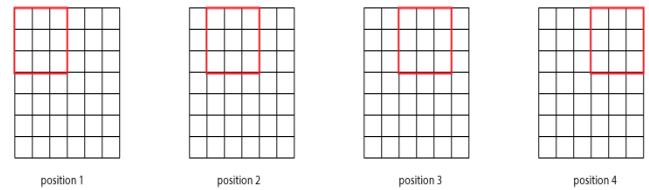
Convolution is used to fuse two sets of numbers which will result a new set of number, the third set. In this research, the two sets of numbers were taken from the inputs and kernel/filter while the third set or the new set of numbers was the output. Both input and kernel/filter are in a form of matrix. The set of numbers on the input were acquired based on the color level that are present in each pixel; the set of numbers on kernel/filter were taken according to the needs of the researchers. There are some types of kernel/filter that are commonly used; some names are identity operation, edge detection, sharpen, box blur, Gaussian blur, etc.

To get more understanding on the procedure of convolution process, researchers will use the sample of sets of numbers from the input because of the size limitation of 128x128 hence researchers use the sets of numbers with the size of 6x6 instead; the execution was done using kernel/filter for vertical edge detection operation on 3x3 size.



**Figure 12.** Convolution process

Stride, on the convolution calculation, with a value of 1 was applied on the 3x3 filter resulting on the visualization below:



**Figure 13.** The position of convolution process

The calculation on this 3x3 filter size convolution process began by doing sliding window on the top left corner to bottom left corner. The filter used was no more than a weight set; 3x3x3 = 9 + 1 bias = 10. On every position, the calculation formula for the pixels was  $\sum_i^n = W_i X_i + b$  which resulting in new values. A single filter will result in a filter with a volume of 4x4x1 as seen on Figure 8.

The size of the image produced by the convolution process becomes more and more shrinking consecutively. This is not a good thing as the size will also become really small. Besides that, it is also risky to utilize bigger-scale filters as the reduction of size will be faster. To prevent this problem, researchers will use the method on *step 1*.

**Activation Function**

The calculation using activation function to determine non-linear values from the result of convolution process is done in this step. The function used in this cow weight detection is ReLU. The formula of ReLU performed is  $f(x) = \max(x,0)$  with x acts as input neuron or node. The '0' on the formula means the linear unit that gets corrected if the input is below 0. Thus, if input is greater than 0, the output = input. The illustration on Figure 14 is an aid to get more understanding on ReLU.

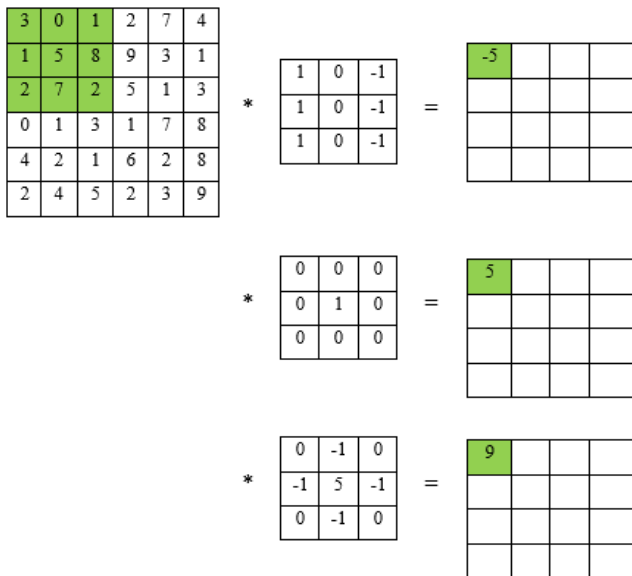


Figure 14. Activation function (ReLU)

The calculation on the first kernel  $(3 \times 1) + (0 \times 0) + (1 \times (-1)) + (1 \times 1) + (0 \times 5) + (8 \times (-1)) + (2 \times 1) + (7 \times 0) + (2 \times (-1)) = -5$ , second and third kernel consecutively,  $(3 \times 0) + (0 \times 0) + (1 \times 0) + (1 \times 0) + (5 \times 1) + (8 \times 0) + (2 \times 0) + (7 \times 0) + (2 \times 0) = 5$  and  $(3 \times 0) + (0 \times (-1)) + (1 \times 0) + (1 \times (-1)) + (5 \times 5) + (8 \times (-1)) + (2 \times 0) + (7 \times (-1)) + (2 \times 0) = 9$ . The greatest value from the three calculation will be used for the next output, thus it is 9 from the result of the third calculation. The following rows and columns follow the same rule.

**Pooling Layer**

This is a layer that has a function to reduce the size of the image. Pooling layer is commonly located right after convolutional layer. Generally, a pooling layer consists of a filter with certain sizes and strides which will shift to every area of the feature map.

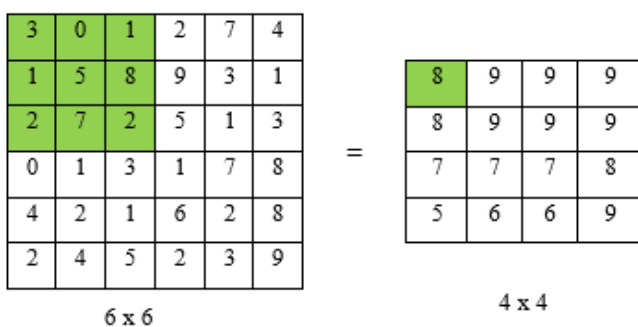


Figure 15. Pooling layer

The pooling layer on the Figure 15 is a layer that applies the Max Pooling method. On Figure 10, there is a layer with 6x6 size, if researchers use 3x3 size filter with stride value of 1, the result will be Max pooling with 4x4 size.

**Fully Connected Layer**

On the tip of convolution layer and pooling layer, network in general use a layer that is fully connected where every pixel is considered as a unique neuron; this is called as Fully Connected Layer. This layer is the last layer which is fully connected and contains countless neurons as a number of classes that has to be predicted.

On the procedure of fully connected layer, every pixel that is considered as a determinant whether the cow is fat or thin is combined. The output is the weight of the cow which consists of two labels; thin and fat cows.

**Softmax**

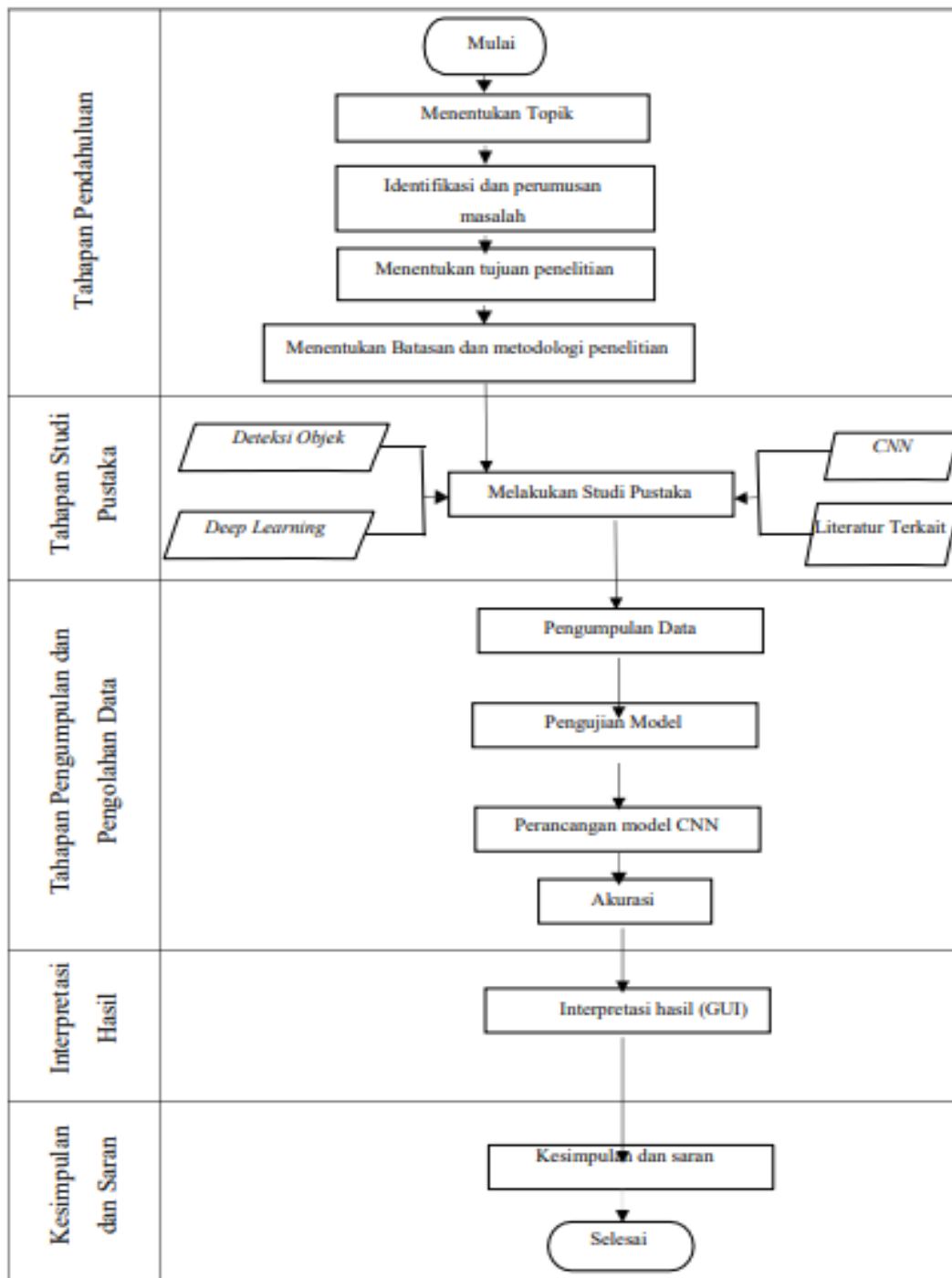
Softmax is a mathematical function that converts a number vector into a probability vector, with the probability of each value is proportional to the relative scale of each value in the vector.

**Detection Output**

Detection output is the result of the estimation of cow weight; whether they are fat or thin. In accordance with the output planning, researchers acquire the detection output with detailed information from the image that is highlighted in green and detailed image label with accuracy ranging from 1-99%.

**Research Steps/Phases**

There are also general descriptions of flowcharts and the explanation on each of the step/phase as follows:



**Figure 16.** The flowchart of general research process

1. Foreword

Researchers begin the initial planning by establishing some topics such as the topic of the research, problem identification, the aim and purpose of the research, and setting the scope and limitation.

2. Literature Review

Researchers get a deeper understanding that are related to the topic of the research such as the procedure of Convolutional Neural Network,

Deep Learning technique, and did a study on articles related to Machine Learning.

3. Data Collection and Processing

a. Data Collection

The data used in this research are in a form of images of fat cows and thin cows. The images were taken (download) from the internet; in detail, there are 44 images of fat cows, 44 images of thin cows, and some other images that are used in the testing process.

b. Data Processing

- Canny Edge Detection

This method detected all of the edges or lines that form the image object that also reaffirmed on those parts of the image. The purpose of this detection is making the object on the image to be easier to be recognized and to simplify the object from its previous form. The strength of this Canny method is the capability of reducing noise before calculating the edge detection which will result on more detection of the edges [16].

- Sobel Operator

This method is one of the operators that dodges that calculation of gradient on the interpolation point. This operator uses 3x3 size kernel filter for the calculation of the gradient so the estimation of the gradient is right in the middle of the window. For example, the scheme of pixels around the x, y pixel is:

$a_0$	$a_1$	$a_2$
$a_7$	(x, y)	$a_3$
$a_6$	$a_5$	$a_4$

Figure 17. The scheme of pixels on around

According to the scheme of the pixels around it, the amount of the gradient can be calculated using Sobel operator [17].

c. Planning of CNN Model

The analysis process in this research uses the aid of computer language; Python, Tensor flow, and hard (?). Some of the methods are done in this research, they are as follows:

- Convolutional Neural Network Backpropagation

This method is used to detect and recognize the object of an image and Backpropagation is used to re-adjust every weight and bias based on the errors on the execution of forward pass. In the application of CNN, to get the maximum result, a proper model and data proportion are needed so that the accuracy of the outcome will be precise. Thus, in this research, the researchers went through 12 trials to acquire one exact model and another 3 trials to acquire one exact dataset.

d. Model Testing and Accuracy

Data training in search for the best model was done in this testing phase. In this process, researchers used 2 different types of CNN architecture, EfficientNet and InceptionV3, which are known for their reliability. Some parameters

such as the total of the epoch, learning rate, optimizer have also been set. After the data training was done, the model went through evaluation due to the parameter such as checking for loss and accuracy to find out how good the formed model was. If the model still lacks of accuracy, it has to go through another training until the model has a small amount of loss and great level of accuracy.

e. Implementation Using Graphic User Interface (GUI)

Implementing the cow weight measurer system utilizing Graphic User Interface.

Results and Discussion

System Implementation

System implementation is the application of the planning that has previously been discussed into an error-free program such as errors on coding. System implementation consists of several parts such as hardware specification, software specification, and interface implementation.

Hardware Specification

The minimum specification requirement can be seen on Table 3:

**Table 3.** Hardware specification

No.	Types of Hardware	Description
1	Processor	Intel Core i3
2	Memory	4 GB
3	Hard Disk	500 GB

Software Specification

The minimum specification requirement can be seen on Table 4:

**Table 4.** Software specification

No.	Types of Software	Description
1	Operating System	Windows 10
2	Tools	Python 3.7

Model

The context of this research is to train the CNN architecture in various ways so that it can maximize the accuracy on the testing on cow

weight imageset. The architecture used is the one that is easy to understand. Besides that, this architecture can be run on personal computer so beginners will find it easy to operate deep learning and convolutional neural network even though it is their first time operating it.

First, researchers created a folder namely “Image Classification” as the main directory of the system. Inside the folder, there are some subfolders such as dataset that contains sets of data on the weight of the cows. Furthermore, there is a file namely ‘train.py’ that acts as the trainer for the data. Each training process consists of two main different architectures in use, EfficientNet and InceptionV3. Based on the architecture, we further analyze the impact of preset filters between none preset filters, Sobel and Canny.

**Table 5.** Dataset directory

Folder	Sub-folder	Content
Image Classification	example	14 images of heavy cows; 5 thin cows Testing
	dataset	44 images of heavy cows; 44 images of thin cows for training
	EfficientNet	arsitektur_inceptio
	InceptionV3	nv3.py
	TrainEfficientNet.py	Code for data training
	TrainInceptionV3.py	
	app (Graphical User Interface)	Code for GUI

**Image Pre-Processing**

The dataset or images was set on 128x128 dimensions so that it will be more specific in differentiating whether the cow is fat or thin were selected. The selection was done so that the result of the training will be more accurate. Furthermore, the training data was separated using 60% for training, 20% for validating and 20% for testing. Thus, for the total of 88 datasets, 52 datasets were used for training and the rest each 18 datasets were used for validating and testing.

**Training Process**

The dataset training of the cow weight is done in this step. Pay more attention to some of the parameters such as weights, Epochs, LR (learning rate), and BS (batch size). The Epochs that were used in this research is 100, in order to get the best model. The main concept of this epoch is that the more epochs done, the more chances of a better model will be acquired. Learning rate was set at  $10^{-3}$  and batch size was set at 32.

From this training process, the researchers underwent 6 types of scenarios in acquiring the best model based on the available dataset:

- 1) EfficientNet architecture with noisy-student weights, no pre filter
- 2) EfficientNet architecture with noisy-student weights, Sobel filter
- 3) EfficientNet architecture with noisy-student weights, Canny filter
- 4) InceptionV3 architecture with imagenet weights, no pre filter
- 5) InceptionV3 architecture with imagenet weights, Sobel filter
- 6) InceptionV3 architecture with imagenet weights, Canny filter

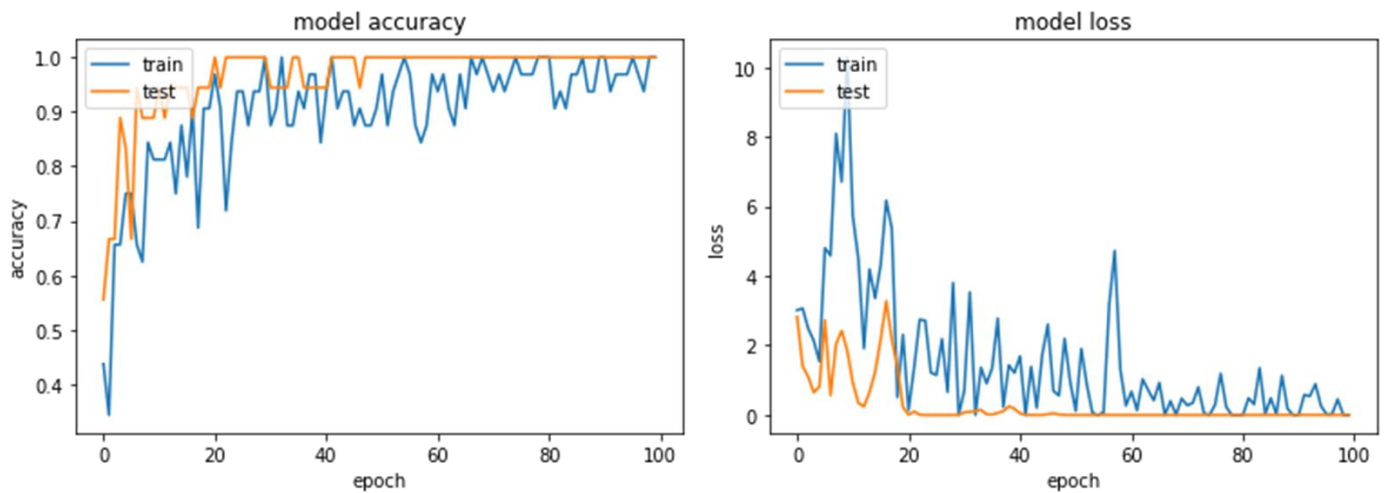


Figure 18. Model accuracy and loss for 1<sup>st</sup> scenario

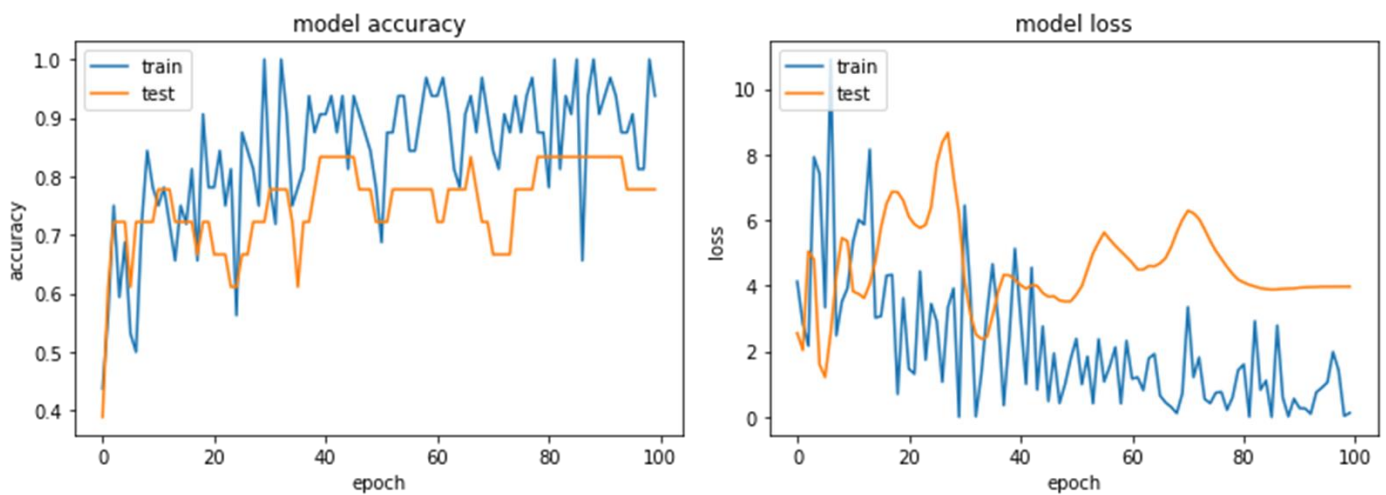


Figure 19. Model accuracy and loss for 2<sup>nd</sup> scenario

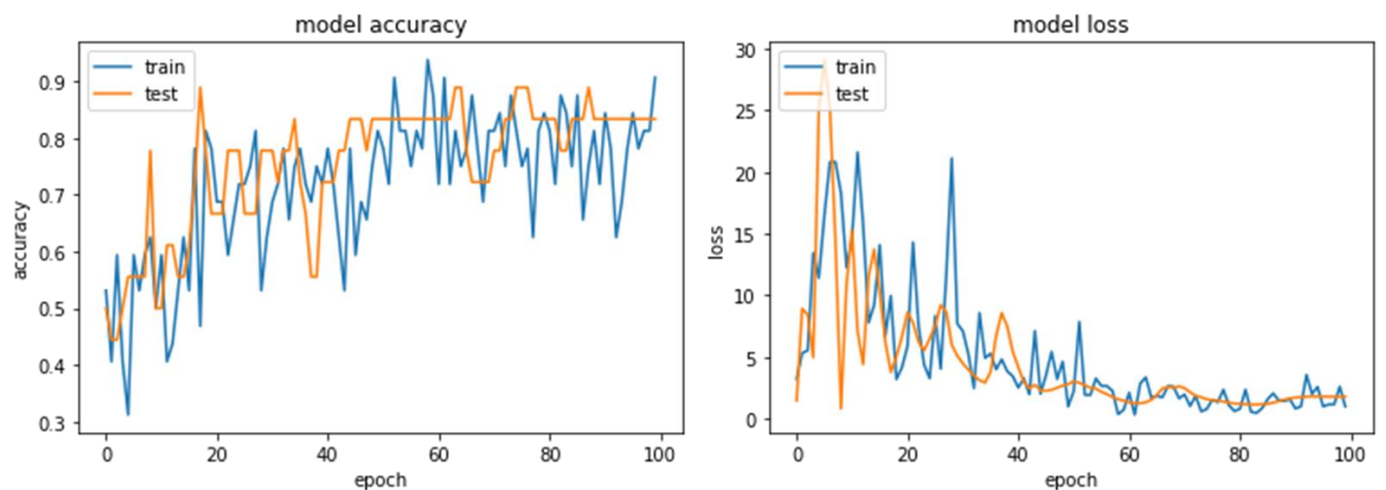


Figure 20. Model accuracy and loss for 3<sup>rd</sup> scenario

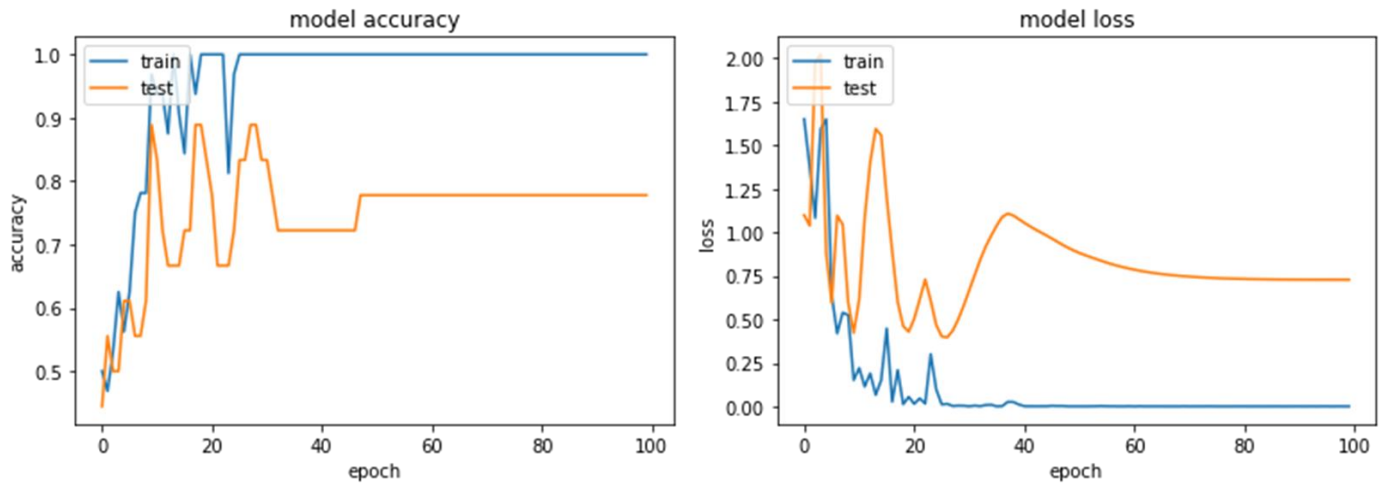


Figure 21. Model accuracy and loss for 4<sup>th</sup> scenario

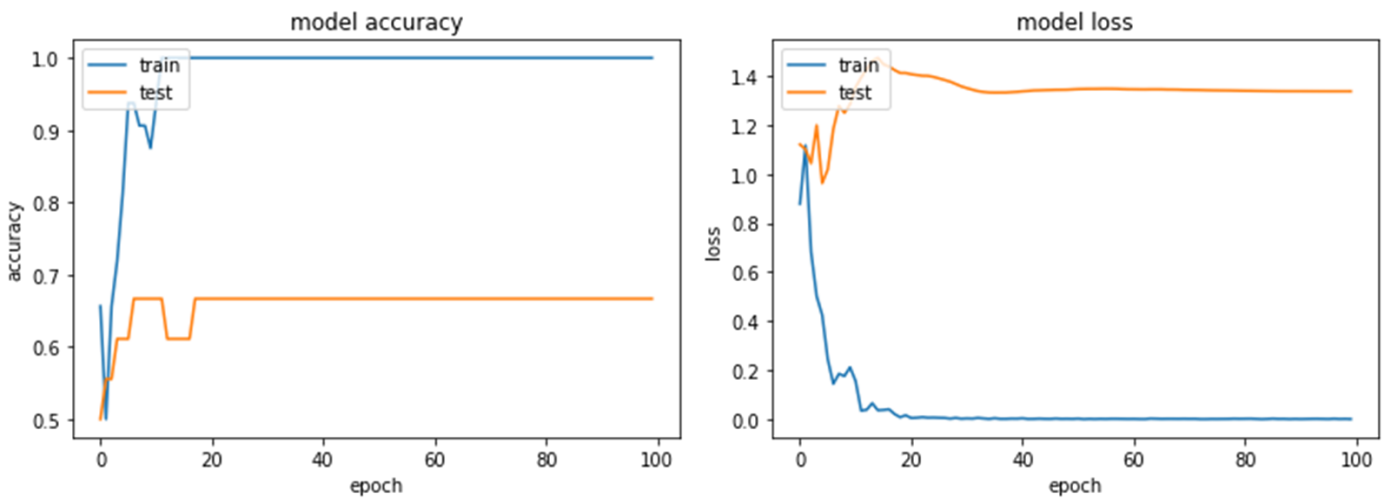


Figure 22. Model accuracy and loss for 5<sup>th</sup> scenario

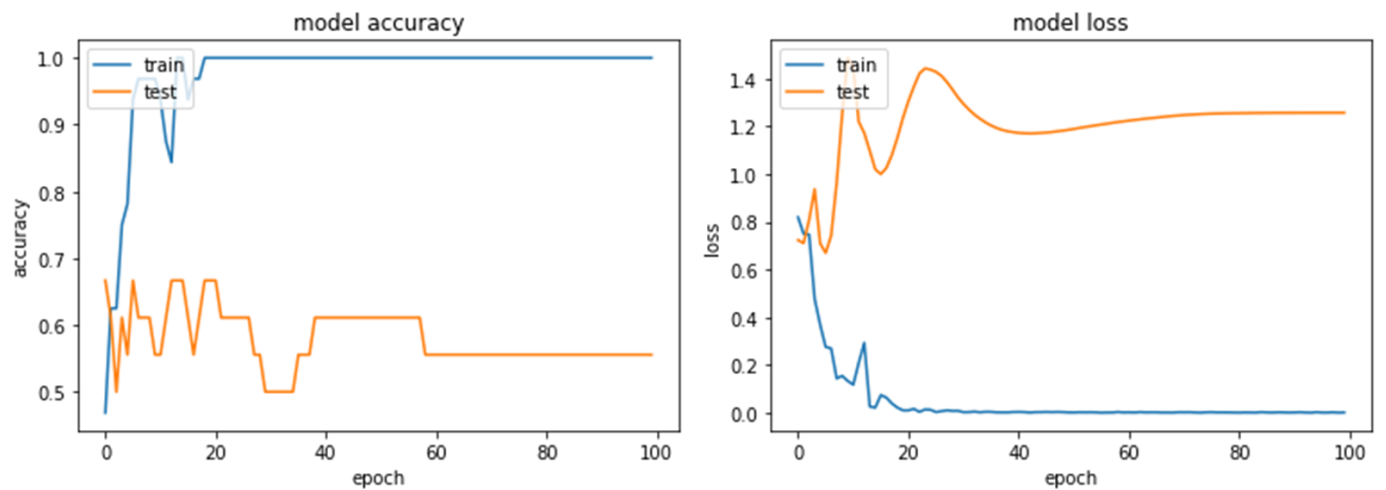


Figure 23. Model accuracy and loss for 6<sup>th</sup> scenario

From the training process, the result was acquired using the four parameters; accuracy, precision, recall and F1 score. Table 6 shows the result of the six scenarios executed.

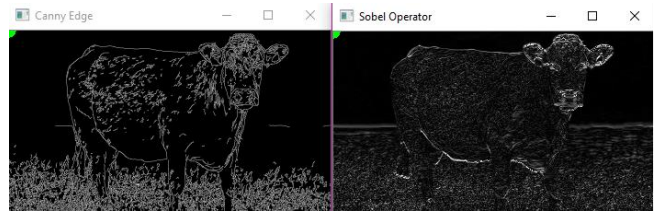
**Table 6. Model results**

No	Architecture	Pre filter	Accuracy	Precision	Recall	F1 score
1	EfficientNet	None	0.89	0.86	0.86	0.86
2		Sobel	0.83	0.85	0.92	0.88
3		Canny	0.72	1	0.5	0.67
4	InceptionV3	None	0.83	1	0.75	0.86
5		Sobel	0.72	0.6	0.86	0.71
6		Canny	0.83	0.7	1	0.82

**Testing**

Based on the result of the trial on the training process, the first scenario with EfficientNet architecture, noisy-student weights and no pre filter was the best model with a great level of accuracy 0,89 since using preset filters like Sobel and canny will remove some of the potential features that CNN could extract.

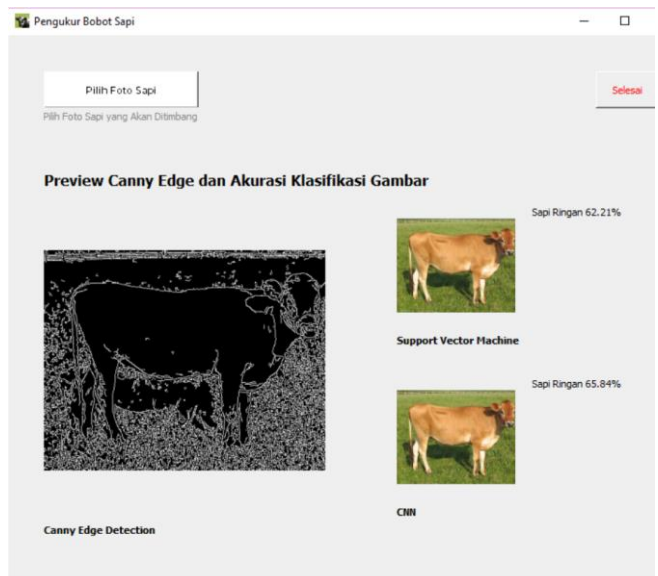
Therefore, this model is used for the classification of cow weight using the implementation of Graphic User Interface. And the best high accuracy model from InceptionV3 is also used for comparison purposes. The main display for the GUI testing classification image as follow:



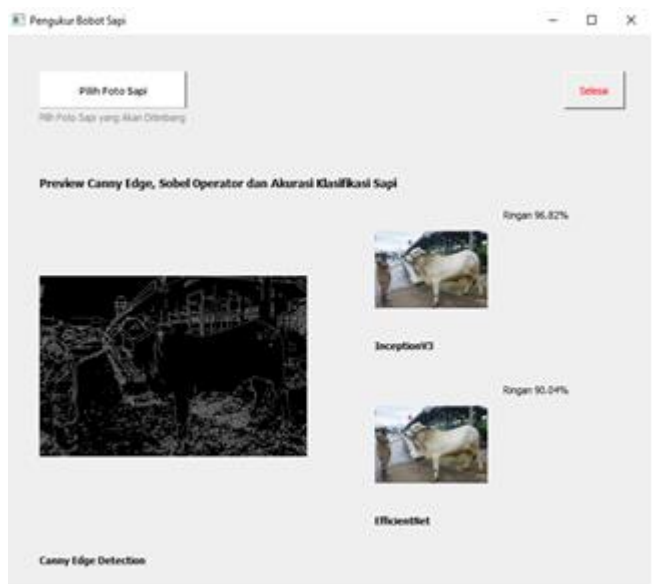
**Figure 25.** The comparison between Canny Edge and Sobel Operator

Two comparisons of edge detection (Canny Edge and Sobel Operator) will pop-up after choosing the image of the cow that will be measured.

the image of the cow that will be measured.



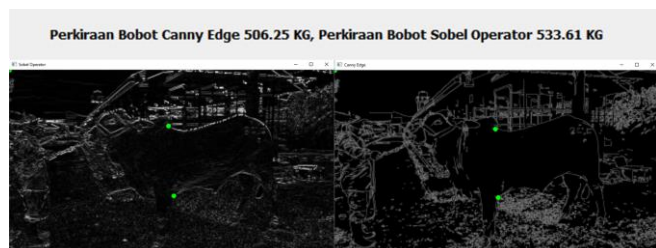
**Figure 26.** The classification of heavy cow



**Figure 27.** The classification of light cow

On the simple display above, a “choose image” button is present; if the button is clicked, an image can be chosen that will go through the testing for classification. Figure 25 shows some examples of the classification of the cow weight as well as the estimation of the cow weight (in kg) as a result of the calculation of Schrool formula.

The estimation of the cow weight is done by determining the cow back point and abdomen to determine the estimation of the cow chest size.



**Figure 28.** Cow weight estimation in kilograms

This measurement in Figure 28 is relatively close to the original value of the previously surveyed weight of the cow, which is 496 kg. But it should be mentioned that in this measurement, the point designation on the picture is very influential on the results obtained.

## Conclusion

### Conclusion

From this research on analysis of cow weight using deep learning method, some conclusions are as follows:

1. Deep learning method with Convolutional Neural Network algorithms can be implied into the classification of cow weight in a form of Graphical User Interface.
2. Based on the result of the testing of the scenarios in search of the best classification model, the best model was acquired from the testing that used EfficientNet architecture, noisy-student weights and no pre filter with the result of the evaluation shows a great level of accuracy 0,89.
3. Image Processing Application can determine the cow weight by utilizing Canny Edge Detection by making use of the chest size variable on the cow object.

### Recommendations

Some notes were taken to be put in mind on the next research on classification using deep learning method:

1. Do not worry on adding up the numbers of dataset that will later be used in the data training and testing because the more dataset for training and testing, the better the accuracy of the model that will be acquired.
2. Do the trials on various types of scenario? The other scenarios could be trying to use a

different set of parameters such as learning rate and batch size or any other use of the CNN architecture.

## References

- [1] I. P. Ningrum, M. A. Fahmy and J. Y. Sari, "Sistem Pakar Diagnosis Penyakit Hewan Sapi Dengan Metode Forward Chaining," Universitas Halu Oleo, Kendari, 2018.
- [2] M. Akbar, "Pendugaan Bobot Badan Sapi," Universitas Brawijaya, Malang, 2008.
- [3] R. Saladin, "Ilmu Tilik Ternak," Universitas Andalas, Padang, 1981.
- [4] R. E. Woods and R. E. Gonzales, Digital Image Processing. New Jersey: Pearson Education, 2008.
- [5] Rismiyati and Azhari SN M.T, "Implementasi Convolutional Neural Network Untuk Sortasi Mutu Salak Ekspor Berbasis Citra Digital," Universitas Gajah Mada, Yogyakarta, 2016.
- [6] T. Sutoyo, E. Mulyanto, V. Suhartono and O. Nurhayati, Teori Pengolahan Citra Digital, Yogyakarta: Andi, 2009.
- [7] R. Munir, Pengolahan Citra Digital dengan Pendekatan Algoritmik, Bandung: Informatika, 2004.
- [8] A. Kadir and A. Santoso, Teori dan Aplikasi Pengolahan Citra, Yogyakarta: Andi, 2012.
- [9] D. Putra, Pengolahan Citra Digital, Yogyakarta: Andi, 2010.
- [10] L. F. Basuki, "Implementasi Metode Histograms of Oriented Gradients Dengan Optimasi Algoritma Frei-chen Untuk Deteksi Citra Manusia," Universitas Komputer Indonesia, Bandung, 2016.
- [11] Wang, Z., Kieu, H., Nguyen, H., & Le, M. (2015). Digital image correlation in experimental mechanics and image registration in computer vision: Similarities, differences and complements. Optics and Lasers in Engineering, 65, 18-27.
- [12] N. Wakhidah, "Deteksi Plat Nomor Kendaraan Bermotor," Universitas Semarang, 2012.
- [13] Mustafa, N. B. A., Fuad, N. A., Ahmed, S. K., Abidin, A. A. Z., Ali, Z., Yit, W. B., & Sharrif, Z. A. M. (2008). Image processing of an agriculture produce: Determination of size and ripeness of a banana. IEEE International

- Symposium on Information Technology, pp. 1-7.
- [14] S. Sena, "Pengenalan Deep Learning Part 7: Convolutional Neural Network (CNN)," 2018. <https://medium.com/@samuelsena/pengenalan-deep-learning-part-7-convolutional-neural-network-cnn-b003b477dc94>.
- [15] Sharma, S., S. Sharma & A. Athaiya, (2017). Activation functions in neural networks. *International Journal of Engineering Applied Sciences and Technology*, 4(12), 310-316.
- [16] A. Faisal, "Deteksi Tepi Menggunakan Metode Canny Untuk Peningkatan Kualitas Citra Patah Tulang Tangan," Universitas Dian Nuswantoro, Semarang, 2017.
- [17] V Suhartono, E Mulyanto and T Sutoyo, "Teori Pengolahan Citra Digital," Andi, Yogyakarta, 2009.
- [18] T. Sutardi, "Pengaruh Kelamin dan Kondisi," Media Peternakan, 1983.
- [19] D. Purba, "Pengolahan Citra Digital," Yogyakarta, 2010.
- [20] K. Sembiring, "Tutorial penerapan Teknik Support Vector Machine untuk Pendeteksian Intrusi pada Jaringan," Institut Teknologi Bandung, 2008.