

# Convolutional Neural Network on Tanned and Synthetic Leather Textures

Faadihilah Ahnaf Faiz<sup>1</sup>, Ahmad Azhari<sup>2,\*</sup>

*Department of Informatics, Universitas Ahmad Dahlan  
Yogyakarta, Indonesia*

*<sup>1</sup> fadhfaiz98@gmail.com; <sup>2</sup> ahmad.azhari@tif.uad.ac.id \**  
*\* corresponding author*

---

## ARTICLE INFO

### Article history:

Received 29 August 2020

Revised 02 December 2020

Accepted 22 December 2020

Published online 31 December 2020

---

### Keywords:

Tanned Leather

Synthetic Leather

Classification

Deep Learning

Convolutional Neural Network

## ABSTRACT

Tanned leather is an output from complex processes called tanning. Leather tanning is an important step that used to protect the fiber or protein structure of animal's skin. Another reason of tanning process is to prevent the animal's skin from any defect or rot. After the tanning is complete, the leather can be applied to produce a wide variety of leather products. Thus, the leather prices usually more expensive because it takes longer time in process. Another way to get cheaper price is make non-animal leather that usually known as synthetic or imitation leather. The purpose of this paper is to classify the tanned leather and synthetic leather by using Convolutional Neural Network (CNN). The tanned leather consist of cow, goat and sheep leathers. The proposed method will classify into four class, they are cow, goat, sheep and synthetic leathers. This research consist of 1280 training data with 448×448 pixels size as the input. With CNN method, this research shows a good result for the accuracy about 92.1%.

This is an open access article under the CC BY-SA license  
(<https://creativecommons.org/licenses/by-sa/4.0/>).

## I. Introduction

Leather can be produce various types of leather goods or handicraft products such as bags, shoes, wallets, belt, leather keychain, etc. Before becomes a leather goods product, the leather material have to get a process calls tanning. Leather tanning is an important step to carry out on a raw of animal's skin or hide [1]. The tanning process are divided into two methods. The first method is use chemicals process calls chrome tanning [2]. The other method is by using natural process that calls vegetable tanned leather [3]. The reason of animal's skin or hide have to be tanned are to make the leather pliable or soft to use, and as a preventive way to avoid it from defect or rot. The vegetable tanned process is also possible to coloring the leather into any colors. Tanned leather certainly has not a cheap price, it depends on the tanning method and done by the people who are experts in the tanning process [4]. The expensive price of the tanned leather become a reason for the production of synthetic or imitation leather. Normally, the synthetic leather made by poliuretan (PU) or polivinyl chloride (PVC) with the cheaper prices than the tanned leather from animal's skin [5]. The texture of synthetic leather that looks similar with the result of animal leather tanned process becomes another reason.

As the role of technology of the imitation leather developments, it is quite difficult to distinguish between real animal skin and imitation leather when only look it visually. It will be different if the two leather materials are still in sheet form. There will be a possibility to distinguish their authenticity, but if it already become goods or products it will be difficult to distinguish which one is made of genuine animal leather or imitation leather. One of the visible characteristics of the skin is the texture of the surface. Genuine leather tan surfaces tend to have natural micro-textures and random patterns so it holds a lot of information that can be analyse then [6]. Real animal skins also after going through the tanning process will have varying textures and colors [7]. Meanwhile, synthetic leather generally tends to have a repetitive surface pattern texture.

From previous researchers on texture leather classification [8], presents with Backpropagation Neural Network (BPNN) as classification method has been done to classify between animal skin texture and synthetic materials or fake leather with two segmentation models. Their study shows highest performance results for the animal skin, except the synthetic materials shows the poorest rate. In other research [9], classified leather surface defect with morphological processing and thresholding to improve performance of the segmentation. Based on previous literature and research, the purpose of this research is to see how well the Convolutional Neural Network (CNN) method is able to classify and find the similarities on the surface or textures of the skin, especially between genuine leather (animal skin) and synthetic leather.

## II. Methods

### A. Research Pseudocode

This research method consist of four main sections, the first section is collecting the datasets known as images acquisition using smartphone camera and take the leathers images then divided it into four categories or class such as cow, goat, sheep and synthetic leather. Next section, the raw images that has been taken in the images acquisition will preprocessed by resizing the images into 448×448 pixels and the images will changed from RGB into grayscale and gaussian images to be used as third section input data. Third section is the main of research, this architecture of Convolutional Neural Network as the method build and divided into two parts that is feature extraction layer and fully-connected layer. The last section is training the dataset and classify the results. After CNN shows the result and success to classify the leather types, then the Confussion Matrix will evaluate the method. The main block of this research can be seen at following convolutional neural network on tanned and synthetic leather textures pseudocode.

#### 1) Variable

```
var class_name, train_images_x, train_images_y : <list>
var train_labels: <list>
var images_size: <int>
var extend_train_images, extend_train_labels: <array>
def preprocess_grayscale(), def preprocess_gaussian()
```

#### 2) Algorithm

```
BEGIN
Image Acquisition
load images from drive THEN extract label from file_name

Image Preprocessing
IF file_name == "sheep" THEN
    return 0
ELSE IF file_name == "synthetic" THEN
    return 1
ELSE IF file_name == "goat" THEN
    return 2
ELSE
    return 3
ENDIF

DEF preprocess_grayscale(images_size, 448)
DEF preprocess_gaussian(images_size, 448)

Image Segmentation
FOR x in preprocess_grayscale()
    WRITE to train_images_x
ENDFOR

FOR x in preprocess_gaussian()
    WRITE to train_images_y
ENDFOR
```

```

%extend the data training
extend_train_images ← train_images_x
extend_train_images ← train_images_y

Training Process
Samples: 1280 images, 80% for training AND 20% for validation
Data training and validation : extend_train_images
Data training and validation labels : extend_train_labels

Build a Convolutional Neural Network architecture
1280 images with 448x448 pixel as input of CNN
Set 6 Convolutional layer and 6 Max-Pooling layer
Set ReLU as activation function each Convolutional layer
1 Output layer with 4 classes (neuron)
Set SoftMax activation function at the Output layer

Set the models
Learning_rate ← 0.0001
Number of epochs ← 259
Number of batch_size ← 64

% performs training
Classify and identify the results

Evaluate the data test with 32 different images in each class with Confussion Matrix.

END

```

### B. Image Acquisition and Preprocessing

The images acquisition processes in this research used a smartphone camera to capture the leather images with distances between the object and the camera lens about 10-20 centimeters and added external flash lamp for help increase the sharpness of leather images. The general purpose of image acquisition is to transform an optical image into an array of numerical data and also computer can manipulate it [10]. After the dataset captured, in order to simplify the computation process in the CNN architecture, the leather images will be preprocessed with resizing into 448×448 pixels images. This preprocessing step aims to shows how much the impact of dataset to the accuracy results [11]. This research collected about 640 primary datasets with 160 images each class and set the labels in each images. Some examples leather images of the dataset are shown in Figure 1. Where L1 are the image of Cow leathers, L2 images of Goat leathers, L3 images of Sheep leathers and L4 are the example images of synthetic leather.



Fig. 1. Leather images dataset

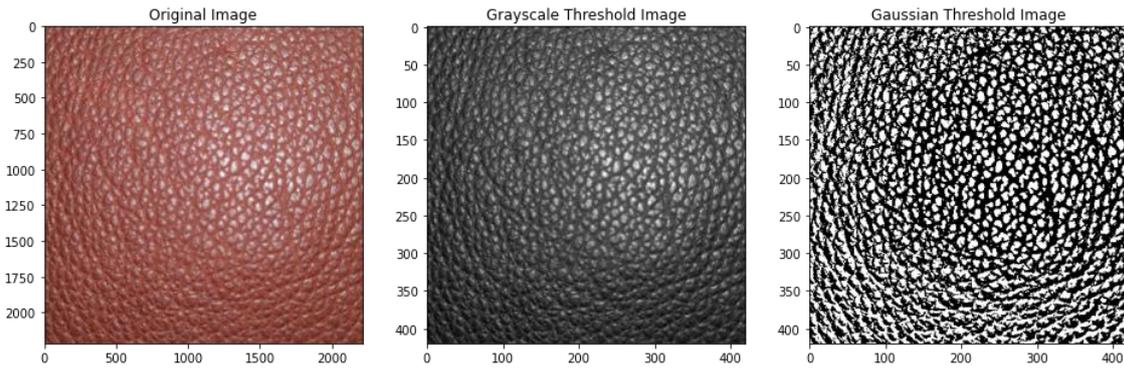


Fig. 2. Sample of original goat leather (left) and after segmentation; grayscale threshold image (center) and gaussian threshold image (right)

Table 1. Images datasets

Leather Types	Images Augmentation		Total
	Grayscale Threshold	Gaussian Threshold	
Cow Leather	160	160	320
Goat Leather	160	160	320
Sheep Leather	160	160	320
Synthetic Leather	160	160	320
Total	640	640	1280

### C. Image Segmentation

Image Segmentation in this research used to changes the RGB images into grayscale images. This method can also be used as a technique to increase the data training which aims to find out another informations that contained in the digital images. The main aim of segmentation process used to recognise the object in an images [12]. This research using two images segmentation methods, there are Grayscale Thresholding and Gaussian Thresholding. Both of them will be used as input data on the CNN models. By using both of segmentation methods, the datasets in this research will be increased into 1280 images leather, there are about 640 images with grayscale threshold and 640 images with gaussian threshold as the data training. Total of the datasets can be seen in Table 1 and the examples of the images segmentation result shown in Figure 2.

### D. Convolutional Neural Network

Convolutional Neural Network also known as ConvNet or CNN is one of class deep learning neural network classification method. It use the architectures of a Multi-Layer Perceptron (MLP) approach [13]. CNN use matrix of pixels from digital images as main object operations of the input data [14]. The training data in this method is a labeled dataset that calls Supervised Learning. CNN are widely used in computer vision technology [15]. In addition to digital images processing, CNN can also applied to the text dataset or Natural Language Processing [16][17][18] or video [19]. A Convolutional Neural Network architectures have several layers such as input layer, feature extraction layer that consist of Convolutional Layer followed by the activation function and some of cases followed by Subsampling or Pooling Layer [20]. The last layer is fully connected layer, in this layer the last feature map matrix will be flattened into a vector [21] and the classification results will be shown. CNN shows a good result to classify some cases such as health [13][14][22][23], social [24][25][26][27] and research sectors [20][28].

### E. Input Layer

The first layer in the architecture of Convolutional Neural Network is input layer, it represents the input of images into the CNN. In this research a grayscale image as the input data that has only one channel and the values between 0 and 255. It will be different if the input uses a RGB images. The RGB images as input will have three channels represents red, green and blue channel.

### F. Convolutional Layer

The Convolutional Layer is core of building blocks and computations in a Convolutional Neural Network architecture [29]. The first convolutional layer will extracting feature some generally motifs from the input layer as a matrix [30]. In this layer processing an operation named convolution that has a small matrix known as kernel. The kernel also called as kernel matrix refers to the matrix size that has width × height (kernel height equals with kernel width). Each convolutional operation always has an output called feature map or activation map, the output typically passed through the activation function. The activation function aims to enable the learning of nonlinear decision boundaries [31]. In this research will applies ReLU activation function on each convolutional layer. This activation function is used to speeds up and increase the performance of the model [32]. The convolutional operation can seen in Figure 3.

As example, a matrix with input size 5×5 pixels, then added the hyperparameter such as padding, kernel with size 3×3 and single stride (Stride = 1). Stride is a parameter that modifies over the movement of kernel on the matrix. The operation shows an output matrix with size 5×5, if added a padding it should use the Equation (1) and the output operation done by Equation (2). The size is not changed because it used a padding or also known as zero padding, it adding zero values in each matrix side [33]. In Figure 4 illustrated the kernel movements on the matrix pixels.

$$Padding = \frac{Kernel-1}{2} \tag{1}$$

$$Output = \frac{Input-Kernel+2*Padding}{Stride} + 1 \tag{2}$$

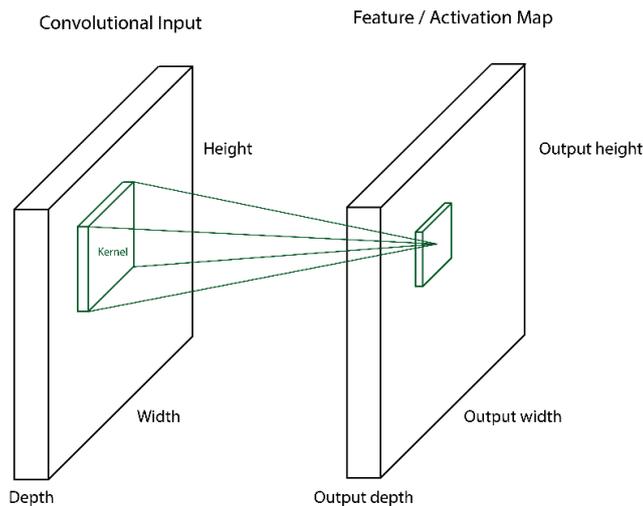


Fig. 3. Convolutional operation

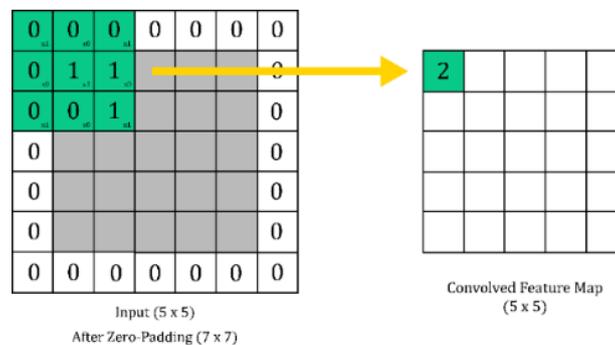


Fig. 4. Kernel movements on convolutional matrix with padding

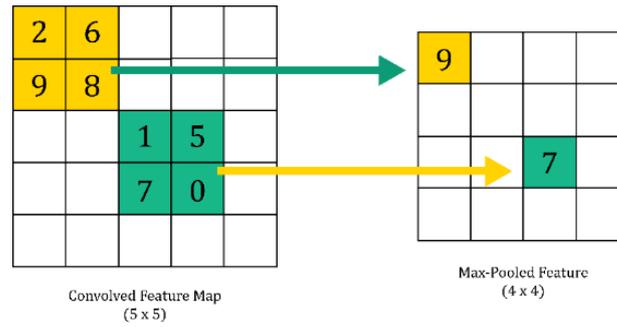


Fig. 5. Max pooling operations with single stride

### G. Pooling or Subsampling Layer

In order to reduce amount of the dimension [34] or parameter neurons from the feature map of convolutional layer [24], pooling operations should be applied [23]. This layers can also to reduce the overfitting. Although reduce the dimension sizes, pooling operations would not remove the informations values that saved from convolutional feature map. In the CNN architectures there are two general types of pooling layer, Max Pool and Average Pool [16][35]. Max pooling calculate the maximum values for each kernel from feature map, while average pooling calculate the average values for each kernel from feature map. This research use max pooling operations, Equation (3) show the formula of maximum pooling where  $b$  is the bias value that added into each elements  $C_n$  from feature maps.

$$Max\ Pooling = \begin{cases} \max(C_1 + b_1) \\ \max(C_2 + b_2) \\ \cdot \\ \max(C_n + b_n) \end{cases} \quad (3)$$

Figure 5 illustrated the max pooling operations where the input matrix is came from output of convolutional layer (feature maps) with  $5 \times 5$  size. Paramater in this pooling example used  $2 \times 2$  kernel size and single stride (Stride = 1). This pooling operations get output with  $4 \times 4$  size of pixels. The results of pooling layer uses as input matrix in the next convolution operation or it will flattened into a vector if it is the last pooling layer in the feature extraction section.

### H. Fully-Connected Layer

In this layer all of the last neurons from feature extractions layer flattened with Flatten function. Fully-connected layer are used at the end of the architecture of CNN after feature extractions layer [36]. Flatten purposed to converts the three-dimensional layer from previous section into one-dimensional layer as a vector. For example,  $6 \times 6 \times 192$  feature map output would be converted into 6912 of vector size. Sometimes, Dropout function will be applied to reduces overfitting and improve the training performance by randomly disabling of neurons in each layers [37]. It fine sholud be used between the fully-connected layer and after the last pooling layer. In order to classify the last image categories, it will use Softmax activation function. This activation commonly used to classify multiclass categorical data [38]. The final result on this research classify four categories of leather types.

## III. Results and Discussions

### A. Training, Validation and Testing Data Split

Splitting of the dataset in this research includes training data, validation data and testing data. The ratio of splitting data will be explained by the Table 2. About 1280 dataset images that was described in the Image Segmentation section above will used as training and validaton data with auto splitting ratio around 80% as training data and 20% as validation data. The testing data consist about 32 new images in each category that different and separated from 1280 of training dataset.

Table 2. Splitting dataset

Images Type	Training Data	Validation Data	Testing Data	Total
Cow Leather	263	57	32	352
Goat Leather	254	66	32	352
Sheep Leather	249	71	32	352
Synthetic Leather	258	62	32	352
Total	1024	256	128	1408

*B. Build a Convolutional Neural Network Architecture*

This research used a Convolutional Neural Network Sequential architecture that was builded and modified by trial and error until get the high accuracy result. Structure or architecture of CNN as we can see in in Figure 6 shows the feature extraction layer and Figure 7 shows the fully-connected layer. The architecture consist of an input layer, six convolutional layer with ReLu as the activation function, six max pooling layer, fully-connected layer with softmax activation function and output layer that have four class category.

Table 3 shows the detail or summary of both Figure 6 and Figure 7 from the input size, filters, activation function and output size in each layers. As the result this model architecture has 1.566.724 of parameters.

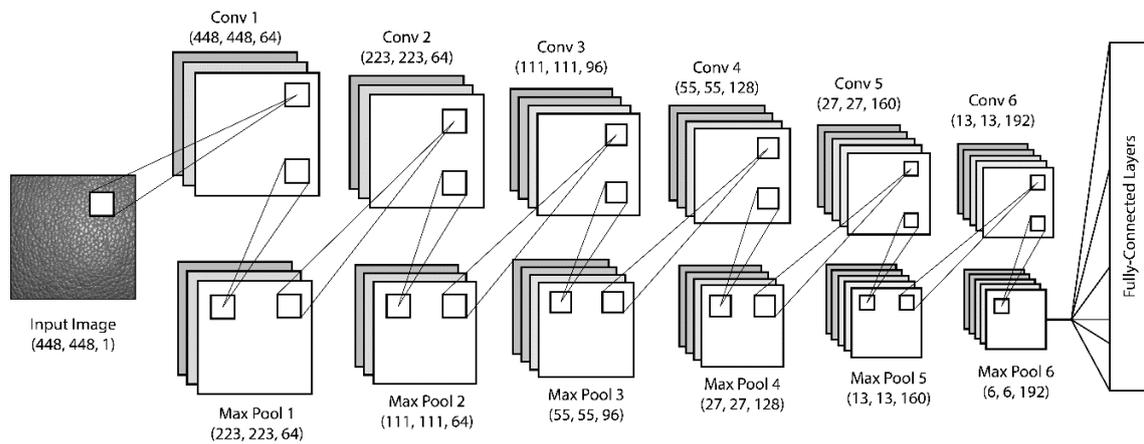


Fig. 6. Architecture of input layer and feature extractions layer

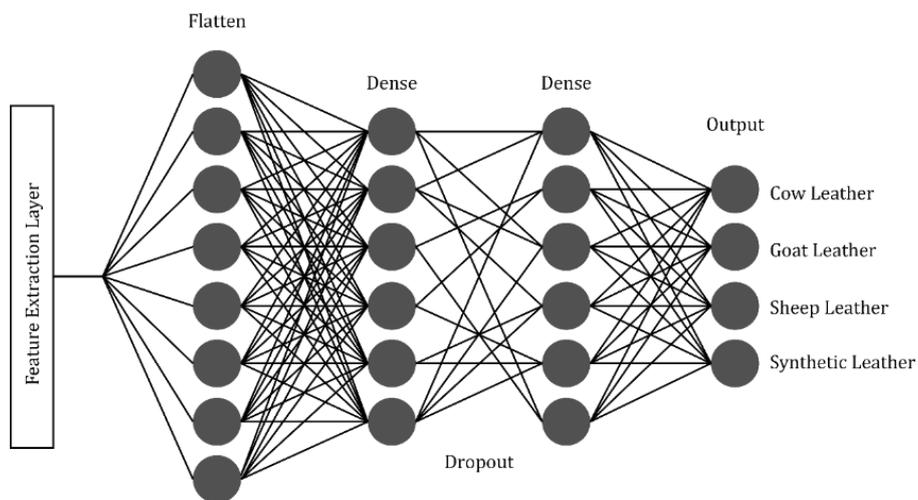


Fig. 7. Architecture of input layer and feature extractions layer

Table 3. Details of summary CNN architecture

Layer	Types	Filters	Kernel	Output Shape	Parameters
0	Input	1	3×3	(448, 448, 1)	0
1	Convolutional (Conv2D)	64	3×3	(448, 448, 64)	640
2	Pooling (MaxPooling2D)	64	3×3	(223, 223, 64)	0
3	Convolutional (Conv2D)	64	3×3	(223, 223, 64)	36.928
4	Pooling (MaxPooling2D)	64	3×3	(111, 111, 64)	0
5	Convolutional (Conv2D)	96	3×3	(111, 111, 96)	55.392
6	Pooling (MaxPooling2D)	96	3×3	(55, 55, 96)	0
7	Convolutional (Conv2D)	128	3×3	(55, 55, 128)	110.720
8	Pooling (MaxPooling2D)	128	3×3	(27, 27, 128)	0
9	Convolutional (Conv2D)	160	3×3	(27, 27, 160)	184.480
10	Pooling (MaxPooling2D)	160	3×3	(13, 13, 160)	0
11	Convolutional (Conv2D)	192	3×3	(13, 13, 192)	276.672
12	Pooling (MaxPooling2D)	192	3×3	(6, 6, 192)	0
13	Flatten	-	-	6912	0
14	Dense	-	-	128	884.864
15	Dropout	-	-	128	0
16	Dense	-	-	128	16.512
17	Output	-	-	4	516
Total					1.566.724

### C. Training the CNN Architecture Model

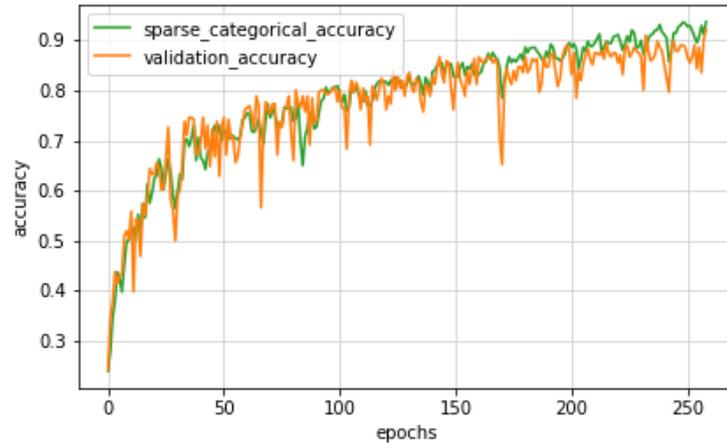
After builds and define the architecture of Convolutional Neural Network, continued by define the compile model such as the optimizers, loss function and metric accuracy. This research using Adam algorithm as optimizer because this optimizer method combines two popular optimization AdaGard and RMSProp [39]. Sparse Categorical Crossentropy for the loss function and Sparse Categorical Accuracy for accuracy metrics because our data are converted into categorical. Furthermore, to fit the model we have to add the Batch Size and Epochs value. Table 4 shows the summary of our model compile and summary of model fit before the training start. The training processes just completed and we are getting the validation accuracy result around 92.1% and the validation loss value around 0.82. It shown in Figure 8 for the accuracy and loss result. In addition to the plots Figure 8, orange lines define the validation data and the green lines define the training data results.

### D. Evaluate the Model

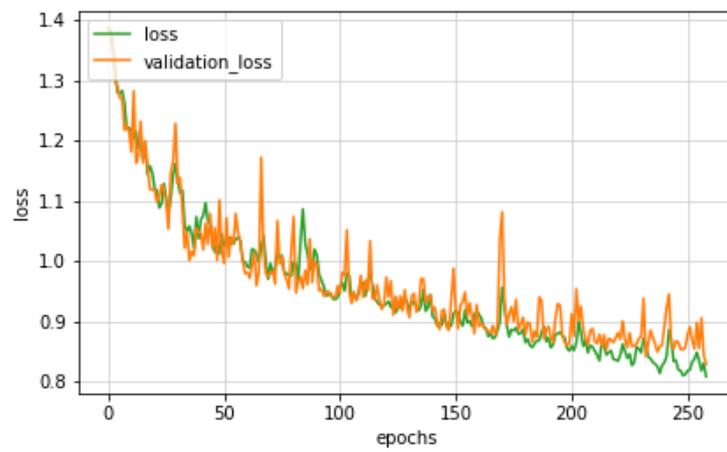
This research use Confusion Matrix as the performance measurement, it can be evaluated for classification problems where the output has two or more categories. Figure 9 shows the final measurement result from Confusion Matrix with validation data as the data evaluation. Form Figure 9, the Confusion Matrix result shows if the sheep leather, goat and synthetic leather from validation data has high matched score of accuracy between the actual values and the predicted values. For the data testing this research added 32 unlabeled images in each category, in example from the Figure 10, this research will testing a folder that contains 32 unlabeled images of goat leathers.

Table 4. Summary of model compile and model fit

Keras Attributes	Args	Values
Model Compile	Optimizers	Adam
	Loss Function	Sparse Categorical Crossentropy
	Metrics Accuracy	Sparse Categorical Accuracy
	Learning Rate	0.0001
Model Fit	Epoch	259
	Batch Size	64



(a)



(b)

Fig. 8. Validation result; (a) accuracy and (b) loss

	Sheep	Synthetic	Goat	Cow	
Sheep	0.90	0.03	0.06	0.01	True Label
Synthetic	0.00	0.97	0.02	0.02	
Goat	0.02	0.02	0.95	0.02	
Cow	0.05	0.00	0.09	0.86	
	Predict Label				

Fig. 9. Confusion matrix results from validation

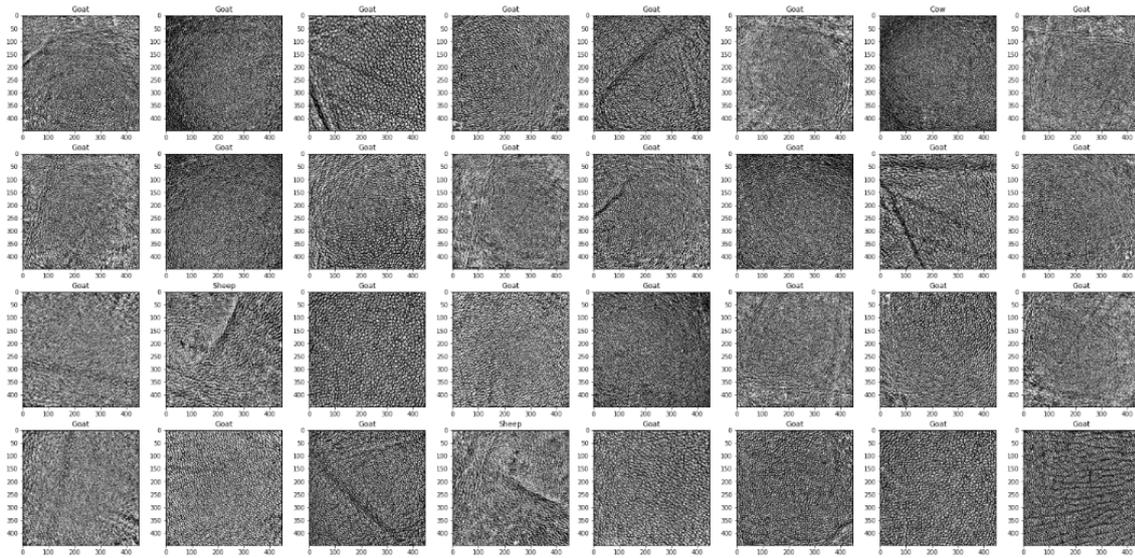


Fig. 10. Sample testing about 32 unlabeled images of goat leathers

The testing result shows 29 images are true detected as Goat leather, two images detected as sheep leather and one images as cow leather. Table 5 explained more the evaluation of data testing with Confussion Matrix.

As can be seen from Table 5, the accuracy calculation of data testing by sum the number of correctly predicted label values and divide it with the total number of data testing. The correctly classified images are located diagonally from the upper-left into the lower-right of Confussion Matrix tabel above and get the accuracy result from the data testing is about 89.0%. Then uploads the new image from data testing with loaded the weights sum from the data training that was saved with h5 model. Figure 11 will shows the comparison of the accuracy from the training, validation and testing result.

Table 5. Confussion matrix result for data testing

Confussion Matrix of Data Test		Predict Label				Total
		Sheep	Synthetic	Goat	Cow	
True Label	Sheep	31	0	0	1	32
	Synthetic	0	25	3	4	32
	Goat	2	0	29	1	32
	Cow	2	0	0	30	32
Total						128

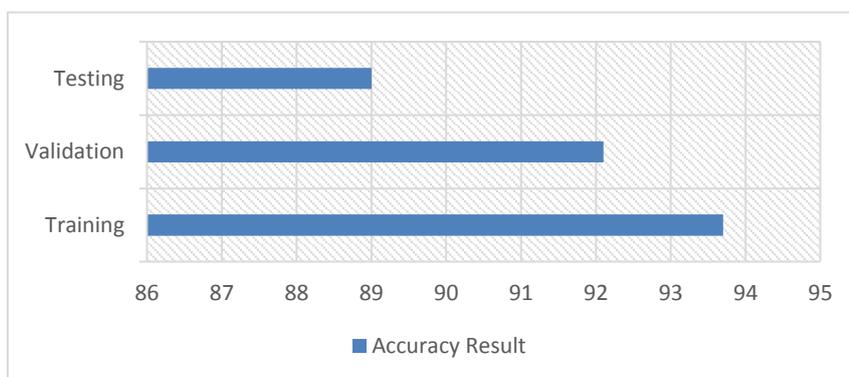


Fig. 11. Visualization accuracy result

## IV. Conclusion

Convolutional Neural Network method in this research shows a good performance and done classified between the genuine leather that consist of cow, goat, sheep and the synthetic leather. The final results as we can see from the Confussion Matrix, the goat leather slightly have a similarity with the sheep leather. Not only the goat or sheep that have a similarities result, but also several the synthetic leather are classified into the cow leather. Although the accuracy of the CNN models is excellent, but the loss value from the model is still high. In future research, hopefully this research can be an initiate to explore more of the dataset from the others leather types and increase the number of leathers dataset in each type.

## Declarations

### *Author contribution*

All authors contributed equally as the main contributor of this paper. All authors read and approved the final paper.

### *Funding statement*

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

### *Conflict of interest*

The authors declare no conflict of interest.

### *Additional information*

No additional information is available for this paper.

## References

- [1] L. Wang and C. Liu, "Tanning Leather Classification using an Improved Statistical Geometrical Feature Method," *2007 Int. Conf. Mach. Learn. Cyb.*, pp. 19–22, August 2007.
- [2] C. Wu, W. Zhang, X. Liao, Y. Zeng, and B. Shi, "Transposition of chrome tanning in leather making," *J. Am. Leather Chem. Assoc.*, vol. 109, no. 6, pp. 176–183, 2014.
- [3] H. Mahdi, K. Palmira, A. Gurshi, and D. Covington, "Potential of vegetable tanning materials and basic aluminum sulphate in Sudanese leather industry," *J. Eng. Sci. Technol.*, vol. 4, no. 1, pp. 20–31, 2009.
- [4] M. Jawahar, N. K. C. Babu, and K. Vani, "Leather texture classification using wavelet feature extraction technique," *2014 IEEE Int. Conf. Comput. Intell. Comput. Res. IEEE ICCIC 2014*, pp. 6–9, 2015.
- [5] N. Purwaningsih, "Penerapan Multilayer Perceptron untuk klasifikasi Jenis Kulit Sapi Tersamak," *J. TEKNOIF*, vol. 4, no. 1, pp. 1–7, 2016.
- [6] H. Chen, "The Research of Leather image segmentation Using Texture analysis Techniques Hong Chen," vol. 1032, pp. 1846–1850, 2014.
- [7] S. Winiarti, A. Prahara, Murinto, and D. P. Ismi, "Pre-trained convolutional neural network for classification of tanning leather image," *Int. J. Adv. Comput. Sci. Appl.*, vol. 9, no. 1, pp. 212–217, 2018.
- [8] S. A. M. Hashim, N. Jamaluddin, and A. Hasbullah, "Automatic Classification of Animal Skin for Leather Products Using Backpropagation Neural Network," *4<sup>th</sup> Natl. Conf. Res. Educ.*, no. May 2017, 2018.
- [9] C. Kwak, J. A. Ventura, and K. Tofang-Sazi, "Automated defect inspection and classification of leather fabric," *Intell. Data Anal.*, vol. 5, no. 4, pp. 355–370, 2001.
- [10] D. Sugimura, T. Mikami, H. Yamashita, and T. Hamamoto, "Enhancing Color Images of Extremely Low Light Scenes Based on RGB/NIR Images Acquisition with Different Exposure Times," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3586–3597, 2015.
- [11] K. S. Sudeep and K. K. Pal, "Preprocessing for image classification by convolutional neural networks," *2016 IEEE Int. Conf. Recent Trends Electron. Inf. Commun. Technol. RTEICT 2016 - Proc.*, pp. 1778–1781, 2017.
- [12] N. H. Dar and H.R. Ramya, "Image segmentation Techniques and its Applications," 2020.
- [13] K. Pai and A. Giridharan, "Convolutional Neural Networks for classifying skin lesions," *IEEE Reg. 10 Annu. Int. Conf. Proceedings/TENCON*, vol. 2019-October, pp. 1794–1796, 2019.
- [14] M. Anthimopoulos, S. Christodoulidis, L. Ebner, A. Christe, and S. Mougiakakou, "Lung Pattern Classification for Interstitial Lung Diseases Using a Deep Convolutional Neural Network," *IEEE Trans. Med. Imaging*, vol. 35, no. 5, pp. 1207–1216, 2016.
- [15] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *3<sup>rd</sup> Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–14, 2015.
- [16] A. Severyn and A. Moschitti, "Twitter Sentiment Analysis with Deep Neural Networks," *Proc. 38<sup>th</sup> Int. ACM SIGIR Conf. Res. Dev. Inf. Retr.*, no. November, pp. 959–962, 2016.
- [17] M. Alali, N. Mohd Sharef, M. A. Azmi Murad, H. Hamdan, and N. A. Husin, "Narrow Convolutional Neural Network for Arabic Dialects Polarity Classification," *IEEE Access*, vol. 7, pp. 96272–96283, 2019.

- [18] C. N. Dos Santos and M. Gatti, “Deep convolutional neural networks for sentiment analysis of short texts,” *COLING 2014 - 25<sup>th</sup> Int. Conf. Comput. Linguist. Proc. COLING 2014 Tech. Pap.*, pp. 69–78, 2014.
- [19] P. Wang, Y. Cao, C. Shen, L. Liu, and H. T. Shen, “Temporal Pyramid Pooling-Based Convolutional Neural Network for Action Recognition,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 12, pp. 2613–2622, 2017.
- [20] C. K. Dewa, A. L. Fadhilah, and A. Afiahayati, “Convolutional Neural Networks for Handwritten Javanese Character Recognition,” *IJCCS (Indonesian J. Comput. Cybern. Syst.)*, vol. 12, no. 1, p. 83, 2018.
- [21] J. Gu *et al.*, “Recent advances in convolutional neural networks,” *Pattern Recognit.*, vol. 77, pp. 354–377, 2018.
- [22] H. Chougrad, H. Zouaki, and O. Alheyane, “Convolutional Neural Networks for Breast Cancer Screening: Transfer Learning with Exponential Decay,” no. Nips, 2017.
- [23] Q. Li, W. Cai, X. Wang, Y. Zhou, D. D. Feng, and M. Chen, “Medical image classification with convolutional neural network,” *2014 13<sup>th</sup> Int. Conf. Control Autom. Robot. Vision, ICARCV 2014*, vol. 2014, no. December, pp. 844–848, 2014.
- [24] L. Pigou, S. Dieleman, P. J. Kindermans, and B. Schrauwen, “Sign language recognition using convolutional neural networks,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8925, pp. 572–578, 2015.
- [25] A. Verma, P. Singh, and J. S. Rani Alex, “Modified Convolutional Neural Network Architecture Analysis for Facial Emotion Recognition,” *Int. Conf. Syst. Signals, Image Process.*, vol. 2019-June, pp. 169–173, 2019.
- [26] K. Yanai and Y. Kawano, “Food Image Recognition using Deep Convolutional Network with Pre-Training and Fine-Tuning,” *Int. Conf. Multimed. Expo Work. . IEEE*, pp. 1–6, 2014.
- [27] G. Levi and T. Hassner, “Age and gender classification using convolutional neural networks,” *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.*, vol. 2015-October, pp. 34–42, 2015.
- [28] J. Zhou, D. Xiao, and M. Zhang, “Feature correlation loss in convolutional neural networks for image classification,” *Proc. 2019 IEEE 3<sup>rd</sup> Inf. Technol. Networking, Electron. Autom. Control Conf. ITNEC 2019*, no. Itmec, pp. 219–223, 2019.
- [29] T. Guo, J. Dong, H. Li, and Y. Gao, “Simple convolutional neural network on image classification,” *2017 IEEE 2<sup>nd</sup> International Conference on Big Data Analysis (ICBDA)*, Mar. 2017.
- [30] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, “A survey of the recent architectures of deep convolutional neural networks,” *Artif. Intell. Rev.*, pp. 1–70, 2020.
- [31] J. Kang, H. S. Choi, and H. Lee, “Deep recurrent convolutional networks for inferring user interests from social media,” *J. Intell. Inf. Syst.*, vol. 52, no. 1, pp. 191–209, 2019.
- [32] T. F. Gonzalez, “Handbook of approximation algorithms and metaheuristics,” *Handb. Approx. Algorithms Metaheuristics*, pp. 1–1432, 2007.
- [33] Y. Sun, B. Xue, M. Zhang, and G. G. Yen, “Evolving Deep Convolutional Neural Networks for Image Classification,” *IEEE Trans. Evol. Comput.*, vol. 24, no. 2, pp. 394–407, 2020.
- [34] L. Kang, P. Ye, Y. Li, and D. Doermann, “Convolutional neural networks for no-reference image quality assessment,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 1733–1740, 2014.
- [35] K. O’Shea and R. Nash, “An Introduction to Convolutional Neural Networks,” pp. 1–11, 2015.
- [36] J. Bernal *et al.*, “Deep convolutional neural networks for brain image analysis on magnetic resonance imaging: a review,” *Artif. Intell. Med.*, vol. 95, no. December, pp. 64–81, 2019.
- [37] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, “Improving neural networks by preventing co-adaptation of feature detectors,” pp. 1–18, 2012.
- [38] R. Hu, B. Tian, S. Yin, and S. Wei, “Efficient Hardware Architecture of Softmax Layer in Deep Neural Network,” *Int. Conf. Digit. Signal Process. DSP*, vol. 2018-Novem, pp. 323–326, 2019.
- [39] D. P. Kingma and J. L. Ba, “Adam: A method for stochastic optimization,” *3<sup>rd</sup> Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–15, 2015.