**Jurnal Teknik Informatika (JUTIF)**
Vol. 3, No. 4, Agustus 2022, hlm. 1125-1130

DOI: https://doi.org/10.20884/1.jutif.2022.3.4.461
p-ISSN: 2723-3863
e-ISSN: 2723-3871

# *IDENTIFICATION OF BIOMETRIC DEEPFAKES USING FEATURE LEARNING DEEP LEARNING*

**Anita Sindar Sinaga[*1], Arjon Samuel Sitio[2], Sumitra Dewi[3]**

[1,3]Department Information Technology, STMIK Pelita Nusantara Medan, Indonesia
[2]Department Digital Business, STMIK Pelita Nusantara Medan, Indonesia
Email: [1]haito_ita@yahoo.com, [2]arjon@yahoo.com, [3]sumitar@yahoo.com

(Naskah masuk: 11 Juli 2022, Revisi : 20 Juli 2022, diterbitkan: 25 Agustus 2022)

***Abstract***

*Improved image quality on several frames extracted from video by manipulating image parameters by improving object edges and coloring segmentation to identify individual human biometric parts. Convolutional Neural Network (CNN) is designed to process two-dimensional data on images by classifying labeled data using the supervised learning method. The classification of fake or not fake images is done using the feature learning Deep Learning technique by forming a Machine Learning model. Video samples (testing and testing) are taken from YouTube randomly. Identifying the resemblance of one person's face to another's (real) face using deep learning. Identifying the resemblance of a person's face to another's face (real) on a genuine or fake label using CNN. Overall, the accuracy results models obtained the highest average accuracy on the face = 93.40%, mouth = 88.52%, eyes = 89.75 %. average accuracy = 90%.*

**Keywords**: *Classification, CNN, Deep Learning, Fake, Identification.*

## 1. INTRODUCTION

The emergence of the term deepfake begins with the development of artificial intelligence. Deepfake is a combination of the words deep learning and fake is used to place an image of a real person's face in a video into the target's face so that it is as if the target is doing or saying things that people do [1]. This technique includes identity falsification so that a tool is needed that can detect and identify engineered images in a video. Biometrics verifies a person's identity, by authenticating individual human beings through unique biological characteristics. The goal is to obtain biometric data from a person. The data obtained such as facial photos, voice recordings, and fingerprints [2].

Currently available editing tools with various advanced and compatible features. With editing tools, it is possible to edit an image or photo at any level for any purpose. The fact is that the average social media removes the metadata of images or photos uploaded on the platform to reduce their size. Reverse image search tools are used to identify the authenticity of an image or photo.

Deep Learning techniques are very effective in various fields such as image recognition and classification [3]. The Convolutional Neural Networks method is part of Deep Learning. By definition deep learning demonstrated from machine learning section that is used for high-level abstraction modeling of data based on algorithms using an implementation layer and using complex structures or vice versa, consisting of several non-linear transformations. The convolution Neural Network algorithm is a Multi-Layer Perceptron that is specially designed to identify two-dimensional images [4].

CNN imitates the way the human brain works to recognize the object it sees. With the help of CNN, computers can see and distinguish various objects. CNN is used to classify labeled data using the Supervised Learning method whose way it works is that there is data to be trained and there is a target variable so that the purpose of grouping data into existing data [5]. In processing complex data such as images and sound, feature extraction is generally carried out to convert the data into a form that can be understood by learning methods [6]. In this research, testing data and training data are sourced from original videos and engineering videos. The training process will run well when using a large number of image train data, this is because with so many images the model can learn to recognize these images better [7]. Each video is extracted into a number of image frames to be detected and labeled [8].

The next step is to form a video detection model. Image quality improvement by manipulating image parameters through object edge improvement and coloring. Image analysis utilizes feature extracts from the image through the segmentation stages [9]. In deep learning, the CNN method acquires high features in the image to the next layer to form a non-linear hypothesis that can increase the complexity of a mode [10].

## 2. METHODOLOGY

The Convolutional Neural Network is a machine learning method from the development of Multi-Layer Perceptron (MLP), research method, Figure 1.

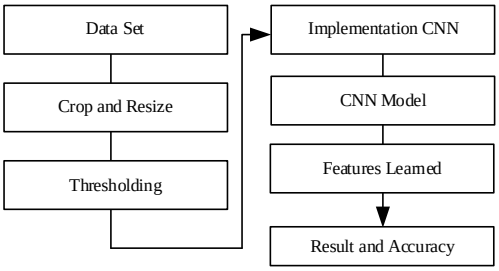



Figure 1. The Stage Research

The description of the research stages :

1. Data Set

The data collection in this study is sourced from digital video partitions by tracking videos of several png files. Digital video consists of a collection of photo frames. The series of frames is displayed on the screen at a rate depending on the frame rate entered (dframes/sec). If the frame rate is high enough, the human eye cannot capture the image frame by frame, but rather captures it as a continuous collection. Each frame consists of a digital form image. Each digital image is represented by a set of pixels represented by a matrix number in which each element represents an intensity value.

2. Crop and Resize

One of the processes to get Region of Interest is by cropping an image. Resizing is the process of resizing image becomes bigger or smaller from the size of the previous image with a size that is pre-determined [12].

3. Thresholding

Identify the image against the background using binarization [11]. The average color is grouped into two, if the color intensity starts from 0 to 255 then the middle value or threshold is taken, which is 128. If below 128 then the color will tend to be black and above 128 the color will tend to be white. Thresholding stage, the image is segmented based on object and background pixels to form and set:

$$g\,(x,y) = \begin{Bmatrix} 1\ if\ f(x,y) \geq T \\ 0\ iff(x,y)\ \leq T \end{Bmatrix} \quad (1)$$

4. Implementation of CNN

1) To start the feedforward process, it is necessary to have the number and size of layers to be formed, the size of the subsampling, the vector image obtained, the feedforward process works on the vector image through the convolution process and Max pooling to reduce the image size and increase the number of neuron [13].

2) Testing process, classification using weights and biases from the results of the training process. As with training processing, the difference is that there is no backpropagation process then feedforward. So that the final result of this process results in the accuracy of the classification carried out, the data that failed to be grouped, the number of images that failed to be grouped, and the network form formed from the feedforward process [14].

5. CNN Model

The stages of training on CNN consists of initialization, feedforward, backpropagation, and weight updates. An artificial neural network consists of several layers and several neurons in each layer. So it cannot be determined using definite rules and applies differently to different data. There are four primary types of CNN layers [15]. The Convolution Layer performs a convolution operation on the output of the previous layer. This layer is the main underlying the CNN. Filter-filter digeser keseluruh permukaan citra sehingga akan menghasilkan keluaran matriks feature map. Feature Map.

$$n_{out} = \left(\frac{n_{in} - k + 2p}{s}\right) + 1 \quad (2)$$

Information $n_{out} = size$ feature, $n_{in} =$ input matrix size, k = filter matrix size, p = size padding s: strid. convolution operation formula:

$$FM[i]_{j,k} = \sum m \sum n N_{[\![(j-]\!]_m, k-n)} F_{(m-n)} + bF \quad (3)$$

FM[i] : Matrix Feature Map ke-i, N : Input image matrix, F : Matriks filter konvolusi, bF : Refractive value on filter j,k : The position of the pixels on the input image matrix m,n : The position of the pixels in the convolution filter matrix After the convolution process is carried out, then activate the activation function using the Rectified Linear Unit (ReLU) function. Each pixel on the feature map will be inserted into the ReLU function, pixels that have a value of less than 0 will be converted in value to 0, with the formula f(x) = max(0,x). Output Layer, multiplying the values of the calculation results on the hidden layer by the weights that have been previously initialized and then added with bias values:

$$[\![y\_in]\!]\_1 = \sum_{j=1}^{m} Z_j * W_{j,i} + W_{0,i} \quad (4)$$

$y\_in_1 =$ the hidden node layer Z to i with the number of nodes m, $Z_j$ = node Z ke-j, $W_{j,i}$=bobot W untuk Zj dan node $Y_i$ bias W untuk $y\_in_1$.

If the stride value is 1, then the convolution filter will shift by 1 pixel horizontally and then vertically. The smaller the stride, the more detailed the information obtained from an input, but it requires more computation when compared to a large stride. A parameter that specifies the number of pixels (containing 0) to be added on each side of the

input to manipulate the output dimensions of the Feature Map. The dimensions of the input are 10x10, if convolution is done with a 3x3 filter and a stride of 2, a feature map with a size of 2x2 will be obtained. if 1 zero padding is added, then the resulting feature map is 3x3.

$$output = \frac{W - N + 2P}{S} + 1 \quad (5)$$

W = Input Length/Height, N = Filter Length/Height, P = Zero Padding, S = Stride, Max Pooling 2x2 with a stride of 2, then at each filter shift, the maximum value in the 2x2 pixel area will be selected.

6. Features Learned

Feature selection selects the optimal subset of features according to certain criteria. This method is aimed at eliminating irrelevant and redundant features, reducing the number of features in the model.

7. Result and Accuracy

The partitions that have been processed in the previous stage are identified from their shape and various sides using the Convolutional Neural Network. The results of the accuracy represent the performance used to assess the benchmarks for the success of the CNN model for classifying images, equations:

$$Accuracy = \frac{correct\ amount\ of\ data}{wrong\ amount\ of\ data}\ x\ 100\% \quad (6)$$

## 3. RESULT AND DISCUSSIONS

The source of the video is taken from a random youtube video, mp4 format. Duration 8 minutes 47 seconds. The result of tracking video becomes 1019 frames in png format, Figure 1.

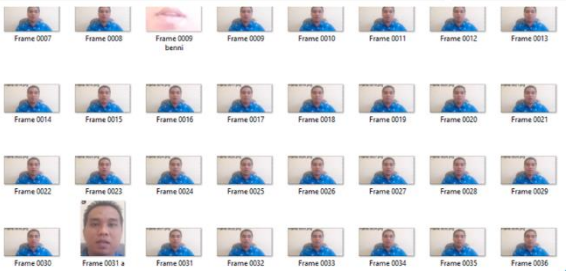


Figure 2. Result of Tracking Video

### 3.1. Data Set

Detection of moving objects using a background reduction algorithm based on a Gaussian mixed model. Morphological operations are applied to the resulting foreground mask to eliminate noise. Finally, blob analysis detects a cluster of connected pixels, which may be a procedure with moving objects. The delivery of detection to the same object is based only on motion. Every creaturehas a different shape, structure, and facial expression. Facial biometrics is the most accurate identification technology because the face is the most easily recognizable part of the human body. The face, eyes, and lips were cropped from the binary image of the training data, Table 1.

Table 1. Testing 1

| Biometric parth of faces | Crop and Resize | Thresholding Result | T Value |
|---|---|---|---|
| Eyes |  |  | 84 |
| Mouth |  |  | 121 |
| Faces |  |  | 74 |

### 3.2. Implementation CNN

a. Training Dataset

Convolutional layer consists of neurons arranged in such a way that it forms a filter with length and height (pixels). Layer first feature extraction layer on the convolution layer with a size of 10x10x3. The length is 10 pixels, the height is 10 pixels and the thickness/amount of 3 pieces corresponds to the channel of the image. These three filters will be shifted to all parts of the image. Each shift will be carried out a dot operation between the input and the value of the filter so as to produce an output or commonly referred to as an activation map or feature map. All features are taken and combined as the final result of the convolution layer, Figure 3.

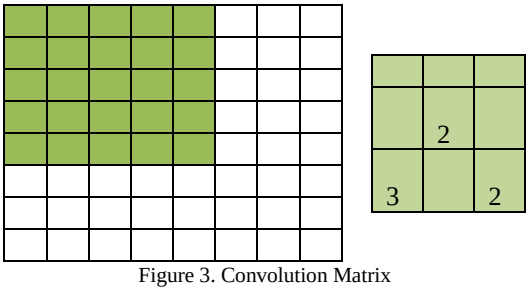



Figure 3. Convolution Matrix

Convolutions with a kernel size of 3x3 start from stride 1, kernel shift towards the 3x3 input matrix from left to right direction, Figure 4.

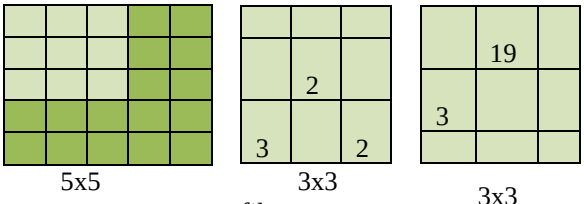



Figure 4. Matrix Convolution Multiplication

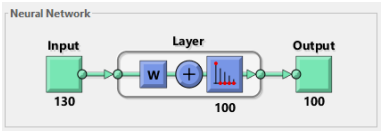
Pooling layers are used to reduce the size of the feature map, shift the window to the entire image surface, the window is used as a reference to select the maximum value in a certain area producing an output in the form of a feature matrix folder that contains the maximum values that are selected. to get the new matrix value sized 2x2 by taking the most maximum value from each window, Figure 5.

| | | |
|---|---|---|
| 5 | -19 | 5 |
| 13 | 3 | 4 |
| 5 | 2 | 6 |

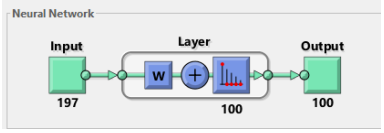
| | |
|---|---|
| 13 | 19 |
| 6 | 6 |

Figure 5. Pooling Layer Result

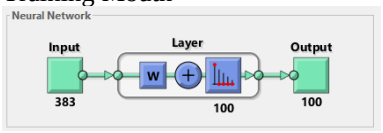
Inputs 'eyes005' is a 130x300 matrix, representing static data: 300 samples of 130 elements. Inputs 'mouth101' is a 197x300 matrix, representing static data: 300 samples of 197 elements. Inputs 'face031' is a 383x300 matrix, representing static data: 300 samples of 383 elements., Figure 6.




Training Eyes




Training Mouth




Training Faces

Figure 6. Neurol Network Training Model

Train the dataset to improve the quality of the results of the model to be built, the dataset will be trained through transform data in the form of train data will be resized to a smaller size. Extraction of input image size 47 × 100 to 60 x 45 by 2700 bytes. The data is divided by 80% as training data and 20% as testing data. The training process uses the parameters learning rate = 0.0001, mini-batch size = 10, MaxEpochs = 5.

Table 2. Eyes Training Result

| Epoch | Iterasi | Time (s) | Error | Accuracy |
|---|---|---|---|---|
| 1 | 1 | 37,05 | 1,764 | 27% |
| 2 | 50 | 43,57 | 0,865 | 79% |
| 3 | 75 | 36,53 | 0,753 | 87% |
| 4 | 90 | 82,02 | 0,562 | 95% |
| 5 | 125 | 372,43 | 0,348 | 100% |

Table 3. Mouth Training Result

| Epoch | Iterasi | Time (s) | Error | Accuracy |
|---|---|---|---|---|
| 1 | 1 | 29,04 | 0,987 | 29% |
| 2 | 50 | 49,03 | 0,654 | 73% |
| 3 | 75 | 53,02 | 0,783 | 79% |
| 4 | 90 | 93,43 | 0,450 | 88% |
| 5 | 125 | 484,43 | 0,263 | 100% |

Table 4. Faces Training Result

| Epoch | Iterasi | Time (s) | Error | Accuracy |
|---|---|---|---|---|
| 1 | 1 | 45,73 | 1,673 | 31% |
| 2 | 50 | 75.21 | 0,468 | 89% |
| 3 | 75 | 47,58 | 0,872 | 92% |
| 4 | 90 | 83,79 | 0,702 | 97% |
| 5 | 125 | 552,65 | 0,255 | 100% |

b. Testing Dataset

The data that has been grouped in the classification process will be matched with the existing dataset. If the data matches and has multiple matches in the dataset, it will match the data in the dataset. The testing process was carried out with several different scenarios at Testing 1, Testing 2, and Testing 3.

1. Testing with a data sharing ratio of 80:20

Table 5. Testing 1

| Biometrics | Percentage Identification | Accuracy Highest | Accuracy Low |
|---|---|---|---|
| Face | 90% | 92.14% | 36.87% |
| Mouth | 87% | 88.11% | 31.16% |
| Eyes | 85% | 87.73% | 33.05% |

2. Testing with a data sharing ratio of 70:30

Table 6. Testing 2

| Biometrics | Percentage Identification | Accuracy Highest | Accuracy Low |
|---|---|---|---|
| Face | 93.40% | 92.62% | 37.55% |
| Mouth | 87.92% | 88.52% | 41.42% |
| Eyes | 89.75% | 86.02% | 39.15% |

3. Testing with a data sharing ratio of 50:50

Table 7. Testing 3

| Biometrics | Percentage Identification | Accuracy Highest | Accuracy Low |
|---|---|---|---|
| Face | 90.21% | 91.42% | 37.26% |
| Mouth | 86.35% | 85.97% | 31.43% |
| Eyes | 87.00% | 85.10% | 30.65% |

### 3.3. Result

The results of the new testing data from the testing stage used 300 testing data, for each part of the facial biometrics. Each predict class and actual class of the biometric part of the face is determined by the mean, confusion matrix, Table 8.

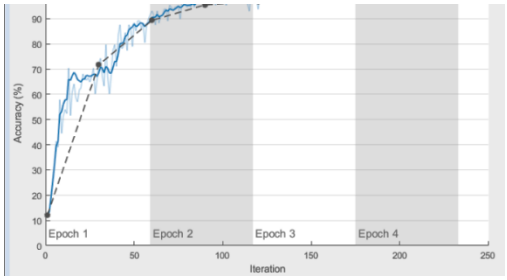
Table 8. Confusion Matrix

| Matrix | | Predict Class | |
|---|---|---|---|
| | | Original | Fake |
| Actual Class | Original | 270 | 0 |
| | Fake | 270 | 30 |

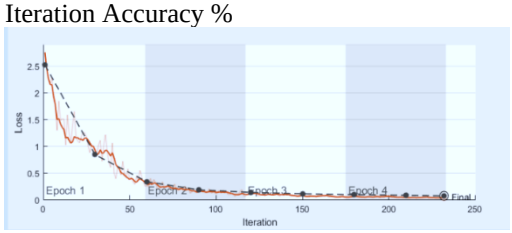
The classification of the original True video class prediction is 270 frames, the video class prediction is fake, there are 30 videos shown missing data in the fake class. Test network classify the validation data and calculate the classification accuracy.

```
YPred = classify(net,imdsValidation);
YValidation = imdsValidation.Labels;
```

```
accuracy = mean(YPred == YValidation)
accuracy = 0.90046
```




Iteration Accuracy %




Iteration Lost

Figure 7. Validation data and calculate the classification accuracy

## 4. CONCLUSION

The classification process determines that the selected parts of each pixel will be formed into the appropriate pattern or shape from the input image. The use of multiple convolution layers can increase the level of accuracy higher than by using 2 convolution layers, the more the use of convolution layers in the process of training the model. In a Neural Network one epoch is too big in the training process because all the data is included in the training process so it will take a while. To simplify and simplify the usual training process, satay data is divided per batch. The prediction results show that the prediction of the original or fake video from the modeling of the testing data on the image shows 90%, the video is identified as original and the fake video is as much as 10% according to the testing data. The higher the accuracy value of the test data set, the video can be identified as the original video, the value accuracy is about 90%-95%.

## ACKNOWLEDGMENTS

Thank you for the research funds support from LPPM STMIK Pelita Nusantara.

## DAFTAR PUSTAKA

[1] A. Ismail, M. Elpeltagy, M. S. Zaki, and K. Eldahshan, "A new deep learning-based methodology for video deepfake detection using xgboost," *Sensors*, vol. 21, no. 16, pp. 1–15, 2021, doi: 10.3390/s21165413.

[2] A. A. Kurniawan and M. Mustikasari, "Implementasi Deep Learning Menggunakan Metode CNN dan LSTM untuk Menentukan Berita Palsu dalam Bahasa Indonesia," *J. Inform. Univ. Pamulang*, vol. 5, no. 4, p. 544, 2021, doi: 10.32493/informatika.v5i4.6760.

[3] D. Wodajo and S. Atnafu, "Deepfake Video Detection Using Convolutional Vision Transformer," 2021, [Online]. Available: http://arxiv.org/abs/2102.11126

[4] N. Arif and F. S. Wahyuni, "Penggunaan Metode Machine Learning Untuk Pengenalan," vol. 5, pp. 6–7, 2016.

[5] T. T. Nguyen *et al.*, "Deep Learning for Deepfakes Creation and Detection: A Survey," *SSRN Electron. J.*, no. September, 2022, doi: 10.2139/ssrn.4030341.

[6] Y. Quiñonez, C. Lizarraga, J. Peraza, and O. Zatarain, "Image recognition in UAV videos using convolutional neural networks," *IET Softw.*, vol. 14, no. 2, pp. 176–181, 2020, doi: 10.1049/iet-sen.2019.0045.

[7] D. Zhang, C. Li, F. Lin, D. Zeng, and S. Ge, "Detecting Deepfake Videos with Temporal Dropout 3DCNN," *IJCAI Int. Jt. Conf. Artif. Intell.*, pp. 1288–1294, 2021, doi: 10.24963/ijcai.2021/178.

[8] S. T. Suganthi *et al.*, "Deep learning model for deep fake face recognition and detection," *PeerJ Comput. Sci.*, vol. 8, pp. 1–20, 2022, doi: 10.7717/PEERJ-CS.881.

[9] A. T. Putra, K. Usman, and S. Saidah, "WEBINAR STUDENT PRESENCE SYSTEM BASED ON REGIONAL CONVOLUTIONAL NEURAL NETWORK USING FACE RECOGNITION", J. Tek. Inform. (JUTIF), vol. 2, no. 2, pp. 109-118, Mar. 2021.

[10] Y. A. Hasma and W. Silfianti, "Implementasi Deep Learning Menggunakan Framework Tensorflow Dengan Metode Faster Regional Convolutional Neural Network Untuk Pendeteksian Jerawat," *J. Ilm. Teknol. dan Rekayasa*, vol. 23, no. 2, pp. 89–102, 2018, doi: 10.35760/tr.2018.v23i2.2459.

[11] H. S. Shad *et al.*, "Comparative Analysis of Deepfake Image Detection Method Using Convolutional Neural Network," *Comput. Intell. Neurosci.*, vol. 2021, 2021, doi: 10.1155/2021/3111676.

[12] C. Hu, H. Huang, M. Chen, S. Yang, and H. Chen, "Video object detection from one single image through opto-electronic neural network," *APL Photonics*, vol. 6, no. 4, 2021, doi: 10.1063/5.0040424.

[13] W. Wang and T. Li, "Fire Video Image Detection Based on a Convolutional Neural Network," *J. Phys. Conf. Ser.*, vol. 1453, no. 1, 2020, doi: 10.1088/1742-6596/1453/1/012161.

[14] Z. A. Fikriya, M. I. Irawan, and S. Soetrisno., "Implementasi Extreme Learning Machine

untuk Pengenalan Objek Citra Digital," *J. Sains dan Seni ITS*, vol. 6, no. 1, 2017, doi: 10.12962/j23373520.v6i1.21754.

[15] Intyanto, & Gramandha, W. (2021). Klasifikasi Citra Bunga dengan Menggunakan Deep Learning : CNN ( Convolution Neural Network ). Jurnal Arus Elektro Indonesia (JAEI), 7 (3), 80–83.