JITTER- Jurnal Ilmiah Teknologi dan Komputer Vol. 2, No. 3 Desember 2021

Classification of Sign Language Numbers Using the CNN Method

I Putu Iduar Perdana[a1], I Ketut Gede Darma Putra[a2], I Putu Arya Dharmaadi[a3]
[a]Department of Information Technology, Udayana University, Indonesia
Corresponding Author: [1]iduarperdana@gmail.com, [2]ikgdarmaputra@unud.ac.id,
[3]aryadharmaadi@unud.ac.id

Abstrak

Berkomunikasi merupakan kebutuhan semua individu karena setiap individu harus berkomunikasi dengan lingkungan. Berkomnikasi juga membuat seseorang mendapat informasi sehingga dapat dijadikan acuan untuk beradaptasi. penggunaan bahasa verbal dengan berbicara mengeluar suara adalah cara komunikasi individu, namun hal itu tidak dapat dilakukan saat berkomunikasi dengan individu yang memilki keterbatasan dalam mendengar. Keterbatasan tersebut membuat diperlukan cara komunikasi lain yaitu melalui bahasa isyarat. Bahasa isyarat banyak jenisnya salah satunya bahasa isyarat menggunakan tangan membentuk huruf atau angka. Bahasa isyarat terdapat standar, standar yang cukup terkenal adalah standar American Sign Language (ASL). Masih banyak yang sulit mengenal bahasa isyarat, maka solusinya adalah membuat sistem untuk klasifikasi bahasa isyarat. Penelitian ini akan membuat sistem machine learning untuk pengenalan angka bahasa isyarat standar *American Sign Language* (ASL) serta menerapkan *preprocessing* untuk optimalisasi hasil. Hasil penelitian ini adalah melakukan perbandingan metode *preproscessing* yang diterapkan pada sistem *Convlutional neural network* arsitektur mobilenetv2. Hasil akhir penelitian kombinasi metode *preprocessing Grayscale*, HSV, *Global Threshold* menghasilkan akuarasi pengenalan terbaik yaitu 97%.

Kata kunci: *American Sign Language*, Grayscale, HSV colourspace, Global Threshold*, adaptive Threshold, convolutional neural network, Mobilenetv2*

Abstract

Communicating is a need for all individuals because an individual must communicate with the environment. Communicating also enables someone to obtain information so that it can serve as a reference for adaptation. The use of spoken language while speaking out of a voice is an individual means of communication, but it cannot be applied when communicating with persons with hearing limitations. These limitations require another way of communication, namely through sign language. There are many kinds of ASL, one of which is ASL using hands to form letters or numbers. Standard popular Sign language is the American Sign Language (ASL) standard. Many still people difficult to recognize sign language, so a solution is to create a system for sign language classification. This research will create a machine learning system for number recognition in American standard sign language. Sign Language (ASL) as well as applying preprocessing to optimize results. The result of this research is to compare the recognition accuracy of the scenarios of different preprocessing methods applied in the Convolutional neural network system architecture MobileNetV2. The final result of this research is the combination of Grayscale, HSV, and Global Threshold preprocessing method yielding the best recognition accuracy of 97%.

***Key Word**: American Sign Language*, Grayscale, HSV colourspace, Global Threshold*, adaptive Threshold, convolutional neural network, Mobilenetv2*

1. Introduction

Communication is essential in the individual life of a person because a person must communicate with their environment. Communicate makes it easier for people to adapt to their environment because when they communicate, they get information for adaptation. Humans communicate by speaking with others, this how-to share information. The sound produced is a medium for communication between individuals, but it is different for people who have limitations in hearing, communicate using verbal language, namely through a voice it cannot do. These limitations require the solution is search media to replace the voice when speaking on communication. A solution for this problem is to replace the voice when talking with sign language or non-verbal language. Sign language or non-verbal language is a language created to help communicate with people who have disabilities. On the site, the European Union of the Deaf A Sign language is no universal sign language in the world. Their monitoring results found that in-country is possible to have more than one sign language. This condition can occur because sign language is a natural language that has linguistic characteristics as well as spoken language. Based on that statement, sign language or non-verbal are many types. Examples are facial expressions, mouth movements, hand movements, and body movements. The non-verbal language that is popular used to communicate with people who have limited hearing is the language that uses fingers as a medium of communication.

The sign language that uses hands is familiar used as a medium of communication in various countries. Different standards apply to each country. Example in America that uses a standard called American Sign Language (ASL). The ASL standard is a sign language standard adopted or developed from several other sign languages, namely French Sign Language, Martha's Vineyard Sign Language, and other sign languages. This sign language only uses one hand for its implementation. Using one hand is more efficient than using two hands [1].

Even an efficient standard is often misunderstood and does not know the meaning of the ASL standard sign language. This mistake is common for people who have never studied sign language. These problems need solutions that can help people recognize the ASL standard sign language. In this era of increasingly advanced technology, this problem can solve by creating a system that can recognize sign language. The system developed is a system that utilizes machine learning technology. Machine learning technology is very effective for developing object recognition or classification systems [2]–[6]. A recognition system for sign language using machine learning was carried out by [7], research is creating the recognition of ASL sign language letters system using a convolutional neural network. Refers to previous research, this research decided to use a convolutional neural network as the basis of the system. A Recognition system with a convolutional neural network requires an architecture, so the system developed will use the mobilenetv2. The dataset used is a sign language number data set in the form of a dataset collection of hand images that are ASL standard sign language number movements [8]. This research will focus on conducting trials to optimize the recognition system by applying a combination of preprocessing. Preprocessing applied to the system will be compared with the final result, namely recognition accuracy.

2. Research Method / Proposed Method

The research stages will be presented in the form of diagrams, the research stages consist of several stages. The research flow chart can be seen in Figure 1.

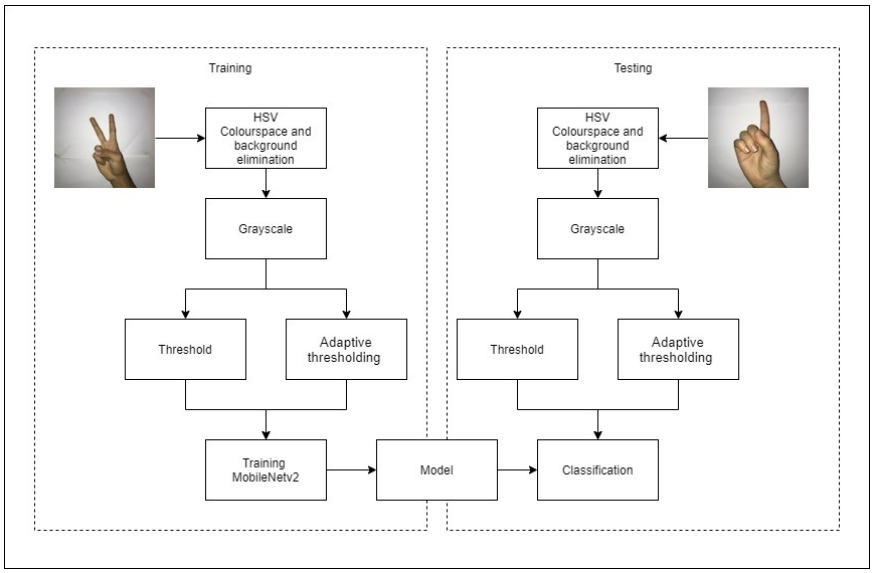


Figure 1. System workflow

Workflow the system developed consists of two modules is the training module and the test module. The training module consists of several processes. The process is dataset image input process, preprocessing, training with mobilenetv2, saving the model. The test module is image input process, preprocessing, object recognition, and display of results. The preprocessing process consists of several methods, namely Grayscale, HSV colorspace and background elimination, Global Threshold, and Adaptive Thresholding. All systems can be developing by the python language and the TensorFlow library.

2.1 Dataset Number ASL

The total ASL dataset is a collection of images from 218 volunteers performing 10 ASL figure movements. The dataset is a right-handed image of an individual. The specifications image is the image in color size 100x100 pixels (3-channel RGB) with a white background and the position of the hand in the center [8]. An example of an image from the ASL number dataset can see in Figure 2.

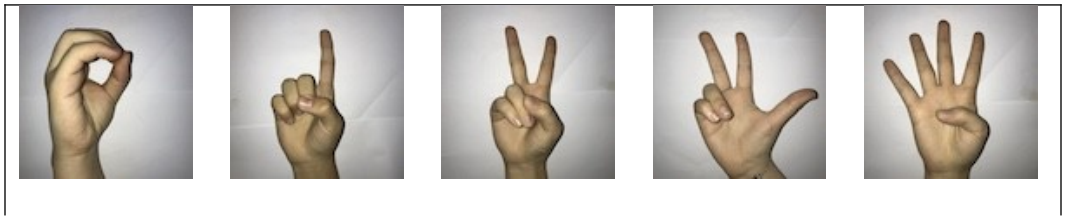


Figure 2. ASL Dataset

Figure 2 is an example of a dataset image of ASL numbers. The figure shown is an image of a hand that forms the numbers 0 to 9 in sign language. The shape of each number ASL on the data is different. Variationsshape of an image can apply as a reference for the recognition or classification system.

2.2 Machine Learning

Machine learning is a technique derived from artificial intelligence developed to infer data with a mathematical approach. Method machine learning imitates the way humans learn for the recognition or classification of objects. The essence of machine learning is a process that aims to create a (mathematical) model that describes patterns of something object used as references [9]. The application of machine learning requires an architecture for study the characteristics of an object. The architecture for application to machine learning techniques is the Convolutional Neural Network (CNN).

Convolutional Neural Network (CNN) is an architecture that can recognize information intended to predict an object. CNN's ability to recognize objects differs from the position of the input data. This ability makes Convolutional Neural Network (CNN) currently widely used in various fields [9]. CNN has components, namely the Convolutional layer, pooling layer, and fully connected layer [10], [11].

MobileNetV2 is a series of convolutional neural network (CNN) architectures developed to become the next generation MobileNetV1. Basis Mobilenetv2 architecture builds from Mobilenetv1 but has deferent is in mobilenetv2 performed simple retraining and requires no special operators to improve accuracy. The MobileNetV2 architecture has 32 convolutional layers and 19 residual bottleneck layers. The kernel used is the standard kernel used in many modern architectures, with a kernel size of 3x3. In addition, dropout and batch normalization. The components in the MobileNetV2 architecture are Depthwise separable convolution, Linear Bottlenecks, and Batch normalization [12].

2.3 Preprocessing

Preprocessing is a process carried out to improve image quality using different methods. The preprocessing process methods include the Grayscale, HSV colorspace and background elimination, Global Threshold, and Adaptive Gaussian Thresholding methods.

2.3.1 Grayscale

Grayscale is a method used to convert an RGB image into a grayscale image. This method has the concept of finding the average of the RGB image matrix [13]. The formula to get a Grayscale image can see in formula 1.

$$s=\frac{r+g+b}{3} \quad (2)$$

The S parameter is the result of image grayscale, while the RGB parameter is the image color value. The application of grayscale to the dataset image can see in Figure 3.

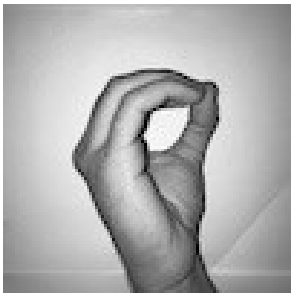


Figure 3. Grayscale Image

2.3.2 HSV colorspace and background elimination

HSV colorspace and background elimination is a preprocessing method that divides the color of the images into three separate parts, namely Hue, Saturation, and value. The HSV preprocessing method is a method that focuses on separating brightness from chromaticity. Components HSV are divided into three parts. The first is part Hue, which ranges from 0 to 179, the second part is Saturation starts from 0-255 and, the last part is value range from 0 to 255 [7]. The application of HSV colorspace and background elimination to the image dataset can see in Figure 4.

Figure 4. Image of HSV colorspace and background elimination

2.3.3 Global Threshold

Global Threshold is a method used to separate images with distributions of object intensity and different background pixels. The process of this method is changing the grayscale image into a binary image [13], [14]. Converting a grayscale image to binary can be done with formula 2.

$$g(x,y)=\begin{Bmatrix}1\,if\,(x,y)\geq T\\0\,if\,(x.y)<T\end{Bmatrix} \quad (2)$$

The formula shows that g (x, y) is a binary image generated from a grayscale image (x,y) and, T is a threshold value parameter. The application of Global Thresholding to the image dataset can see in Figure 5.

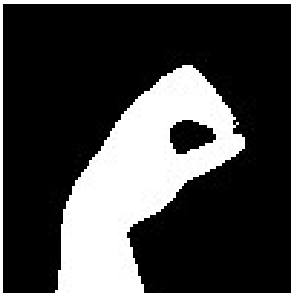


Figure 5. Global Threshold

2.3.4 Adaptive Thresholding

Adaptive Thresholding is a method of converting grayscale images into binary for condition images with varying contrast and lighting, a condition that makes it very difficult to separate pixels into background or foreground [15]. This method can apply by one of the following three formulas.

$$T = \frac{\sum_{(x,y)} \sum \in W f(x,y)}{N_W} - C \quad (3)$$

Or

$$T = median\{f(x,y),(x,y) \in W\} \quad (4)$$

$$T = \frac{max\{f(x,y),(x,y) \in W\} + min\{f(x,y),(x,y) \in W\}}{2} \quad (5)$$

Parameter W is the block for processed, parameter NW is the number of pixels in each block W, C is a constant that can be determined freely. If C = 0, it means that the threshold value is equal to the average value of each pixel in the block concerned. The application of Adaptive Thresholding to the image dataset can see in Figure 6.

Figure 6. AdaptiveThreshold

3. Result

The results and discussion explain the comparison between recognition without preprocessing and preprocessing methods for ASL sign language number classification using convolutional neural network architecture mobilenetv2. The test data used amounted to 100 images divided for each class. The detail in one class is ten images. The results of this research comparison can see in table 1.

Scenario	Epoch	Accuracy
Original image	50	88%
Grayscale+HSV+Global Threshold	50	95%
Grayscale+HSV+Adaptive Thresholding	50	97%

The results of the trials are resulted in 88% for a condition without applying Preprocessing. The second scenario is a scenario of applied Grayscale, HSV and, Global Threshold resulted in a classification accuracy of 97%. The last scenario is a scenario of applied Grayscale, HSV, and Adaptive Thresholding preprocessing obtained an accuracy of 95%. When viewed from the accuracy of the Grayscale, HSV, and Global Threshold scenarios, we get the best accuracy. Detailed results for the best scenario can see in the Confusion matrix in the image.

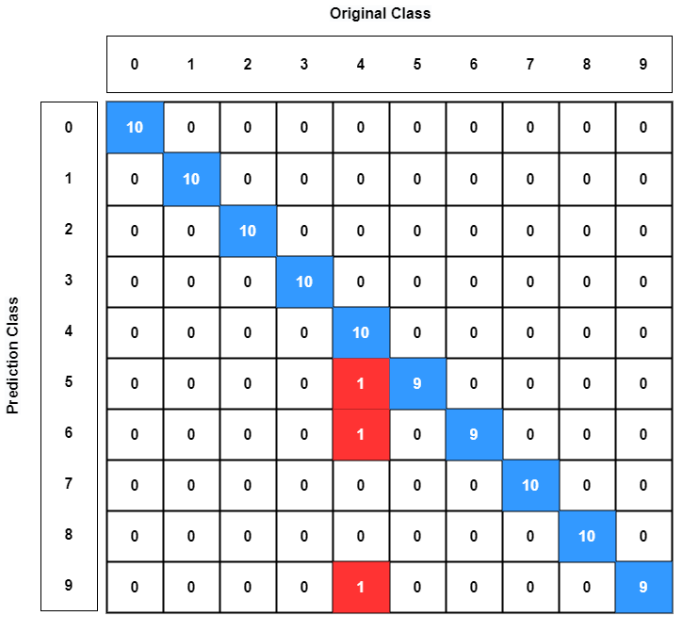
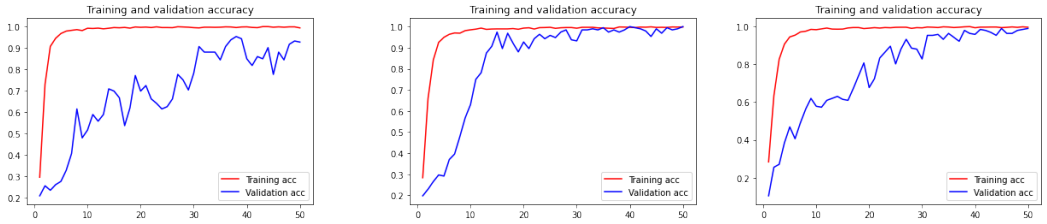


Figure 7. *Confusion matrix*

Figure 7 is a Confusion matrix for the results of the Grayscale, HSV, Global Threshold scenario. The results obtained are that there are three prediction errors by the system. The error is in the number five class with one wrong. An error resulted in the prediction number is four instead of five. The error is in the number six class with one wrong prediction. An error resulted in the prediction number is four instead of six. The error is in the number nine class with one error prediction. An error resulted in the prediction number is four instead of nine. The comparison of the graph of accuracy and validation accuracy can see in Figure 8. Figure 7 is a Confusion matrix for the results of the Grayscale, HSV, Global Threshold scenario. The results obtained is it found three mistake predictions from system. The first error at number five class with one mistake that is prediction system choose number four instead of number five. The second error at number six class with one mistake prediction that is prediction system choose number four instead of number six. The last error at number nine class with one mistake that is prediction system choose number four instead of number nine. The comparison of the graph of accuracy and validation accuracy can see in Figure 8.



(a) (b) (c)

Figure 8. Training accuracy and validation accuracy (a) Original Image, (b) Grayscale+HSV+Global Threshold, (c) Grayscale+HSV+Adaptive Thresholding

Figure 8 is a graph comparison of training accuracy and validation accuracy for the three scenarios. The original image scenario graph shows a stable training accuracy graph but unstable validation accuracy. The Grayscale, HSV, and Global Threshold scenario graphs show the stability of the training accuracy graph, and the validation accuracy graph line starts to stabilize from epoch 15. The Grayscale, HSV, Adaptive Threshold scenario graph shows a stable training accuracy graph, and the validation accuracy graph line starts to stabilize from epoch 35. Loss graph training and Loss validation can see in Figure 9.

(a) (b) (c)

Figure 9. Loss dan Loss validation (a) Original Image, (b) Grayscale+HSV+Global Threshold, (c) Grayscale+HSV+Adaptive Thresholding

Figure 9 is a comparison of loss training and loss validation graphs for the three scenarios. The original image scenario graph shows the loss training graph is stable decrease, but for validation accuracy, it is not stable from the beginning to the end. The Grayscale, HSV, Global Threshold scenario graph shows a stable loss training graph, and the loss validation graph line starts to stabilize and continues to decrease from epoch 15. The Grayscale, HSV, Adaptive Threshold graph shows a stable loss training graph, and the loss validation graph line starts to stabilize steadily decreasing from epoch 35.

4. Conclusion

This research has developed a system for the recognition or classification of a sign language number image. The recognition system developing is a system that uses a convolutional neural network with the Mobilenetv2 architecture as the basis. The basis of the system must also be supported by preprocessing to improve the results of recognition or classification accuracy. This reason makes this research focus on developing a combination of preprocessing that can improve accuracy results. This study decided to apply two scenario combinations of Preprocessing, namely a combination of Grayscale, HSV, and Global Threshold and a scenario combination of Grayscale, HSV, and adaptive Threshold. These two combinations will be compared for accuracy once applied to the system. This study uses a dataset with ASL standards. Total dataset of 2062 images divided into ten classes. The experiment this research using the number of test data, namely 100 images. The final result of this study succeeded in developing a system for sign language number recognition with a recognition accuracy of 97%. These results get with a developing system that applies the convolutional neural network architecture of mobilenetv2 by optimizing it with a combination of preprocessing. The preprocessing combination that gives the best improvement is the scenario combination of Grayscale, HSV, and adaptive Threshold preprocessing.

References

[1] Miskudin Taufik, “Bahasa Isyarat Menyatukan Dunia,” Oct. 13, 2020. https://itjen.kemdikbud.go.id/public/post/detail/bahasa-isyarat-menyatukan-dunia (accessed Jul. 26, 2021).
[2] X. Luo, X. Qin, Z. Wu, F. Yang, M. Wang, and J. Shang, “Sediment Classification of Small-Size Seabed Acoustic Images Using Convolutional Neural Networks,” *IEEE Access*, vol. 7, pp. 98331–98339, 2019, doi: 10.1109/ACCESS.2019.2927366.
[3] V. Borate, S. Patange, V. Vede, and O. Kale, “An Image Classification Based on CNN Approach For Plant Leaf Disease Detection,” vol. 4, no. 6, p. 3, 2018.
[4] S. Z. M. Zaki, M. Asyraf Zulkifley, M. Mohd Stofa, N. A. M. Kamari, and N. Ayuni Mohamed, “Classification of tomato leaf diseases using MobileNet v2,” *IJ-AI*, vol. 9, no. 2, p. 290, Jun. 2020, doi: 10.11591/ijai.v9.i2.pp290-296.
[5] I. K. G. Darma Putra, R. Fauzi, D. Witarsyah, and I. P. D. Jayantha Putra, “Classification of Tomato Plants Diseases Using Convolutional Neural Network,” *International Journal on Advanced Science, Engineering and Information Technology*, vol. 10, no. 5, p. 1821, Oct. 2020, doi: 10.18517/ijaseit.10.5.11665.
[6] Computer Science and Engineering Department, National Institute of Technology Manipur, Imphal, 795001, India, R. Meitram, and P. Choudhary, “Palm Vein Recognition Based on 2D Gabor Filter and Artificial Neural Network,” *JAIT*, vol. 9, no. 3, pp. 68–72, 2018, doi: 10.12720/jait.9.3.68-72.
[7] M. Hurroo and M. E. Walizad, “Sign Language Recognition System using Convolutional Neural Network and Computer,” *International Journal of Engineering Research*, vol. 9, no. 12, p. 6.
[8] A. Mavi, “A New Dataset and Proposed Convolutional Neural Network Architecture for Classification of American Sign Language Digits,” p. 5.
[9] J. W. Gotama Putra, *Pengenalan Konsep Pembelajaran Mesin dan Deep Learning*, 1.4. 2020. [Online]. Available: https://www.researchgate.net/publication/323700644_Pengenalan_Pembelajaran_Mesin_dan_Deep_Learning
[10] Md. M. Kabir, A. Q. Ohi, Md. S. Rahman, and M. F. Mridha, “An Evolution of CNN Object Classifiers on Low-Resolution Images,” in *2020 IEEE 17th International Conference on Smart Communities: Improving Quality of Life Using ICT, IoT and AI (HONET)*, Charlotte, NC, USA, Dec. 2020, pp. 209–213. doi: 10.1109/HONET50430.2020.9322661.
[11] F. Sultana, A. Sufian, and P. Dutta, “Advancements in Image Classification using Convolutional Neural Network,” *2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)*, pp. 122–129, Nov. 2018, doi: 10.1109/ICRCICN.2018.8718718.
[12] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted Residuals and Linear Bottlenecks,” *arXiv:1801.04381 [cs]*, Mar. 2019, Accessed: Feb. 02, 2021. [Online]. Available: http://arxiv.org/abs/1801.04381
[13] R. C. N. Santi, S. Pd, and M. Kom, “Mengubah Citra Berwarna Menjadi GrayScale dan Citra biner,” vol. 16, p. 6, 2011.
[14] K. Bhargavi and S. Jyothi, “A Survey on Threshold Based Segmentation Technique in Image Processing,” vol. 3, no. 12, p. 7, 2014.
[15] N. P. Sutramiani, Ik. G. Darmaputra, and M. Sudarma, “Local Adaptive Thresholding Pada Preprocessing Citra Lontar Aksara Bali,” *JTE*, vol. 14, no. 1, Jun. 2015, doi: 10.24843/MITE.2015.v14i01p06.