MITOR: Jurnal Teknik Elektro

Implementation of the Viola-Jones Algorithm for a Hand Sign Language Translation System

Atik Novianti*, Siti Nurul Muthiah, Asep Mulyana
Jurusan Teknik Elektro – Politeknik Negeri Malang
Malang, Indonesia
*atiknovianti@polinema.ac.id

Abstract – *People who speech impaired that uses body movements to deliver information. In everyday life, it is not uncommon for people with disabilities to communicate with normal people who do not understand the meaning of what is delivered. One example is when a speech-impaired person wants to order food at drive-thru service fast food restaurant, the seller must have special skills to understand what is being delivered. In this paper, we will discuss the application of the Viola Jones algorithm for hand gesture detection systems in drive-thru service fast food restaurant. The Viola Jones algorithm was proposed because the algorithm is the most famous and best choice of gesture classifier for detecting hands in real time based on the accuracy speed. The system is able to translate the given gestures, data enters Firebase as a real time database, sellers can find out the total shopping through the cashier application. The system cannot translate the gesture if the given hand is not completely in the specified area of the box. Based on testing on 10 different respondents, the system can detect hand gesture with an average system success of 92,01%.*

Keywords – *Hand gesture recognition; Viola-Jones algorithm; Speech-impaired communication; Real-time detection; Drive-thru service.*

I. Introduction

COMMUNICATION is one of the essential aspects of social life. Language serves as a medium to facilitate smooth communication. However, not everyone can communicate verbally due to certain limitations, one of which is speech impairment. Speech impairment is a condition where an individual has difficulty communicating due to speech disorders, either in language or sound [1]. Individuals with speech impairments use sign language, which employs body movements to convey information. In everyday life, it is not uncommon for people with speech impairments to communicate with normal individuals who may not understand the meaning of the signs being conveyed.

One example is when a speech-impaired person wants to order food at a drive-thru service in a fast-food restaurant. The seller must have special skills to understand what is being conveyed. In an era where technology is rapidly advancing, an automatic hand sign language translator can be a practical solution. This system can be realized in various ways, one of which is through image processing.

Image processing is the application of computational transformations to images, such as sharpening, contrast adjustment, and more [2]. Many studies utilize image processing to solve various problems, including: detecting diseases in banana trees [3], detecting surface cracks in concrete structures [4], identifying types of brain tumors [5], detecting fire [6], recognizing emotions based on facial recognition [7], verifying handwritten signatures [8], detecting mask usage [9], identifying industrial machine errors [10], and many others. On the other hand, numerous studies have discussed hand sign language translators for various purposes, but variables such as background, camera angles, and lighting still pose challenges.

Hanwen Huang et al. researched a fast and robust method for hand sign recognition based on RGB video by detecting skin color, extracting contours, and segmenting hand regions [11]. Additionally, previous research presented a real-time hand sign recognition system based on near-infrared devices, which directly analyzed infrared images to infer static and dynamic gestures without using skeletal information [12]. Hand sign recognition is also used as an essential part of

The manuscript was received on January 12, 2024, revised on July 1, 2024, and published online on July 26, 2024. Emitor is a Journal of Electrical Engineering at Universitas Muhammadiyah Surakarta with ISSN (Print) 1411 – 8890 and ISSN (Online) 2541 – 4518, holding Sinta 3 accreditation. It is accessible at https://journals2.ums.ac.id/index.php/emitor/index.

Human-Computer Interaction, providing an approach for two-way interaction between computers and users [13].

Based on the above description, this paper discusses the implementation of the Viola-Jones algorithm for a hand sign language translation system in drive-thru fast food restaurants. The Viola-Jones algorithm, introduced by Paul Viola and Michael Jones [14], is proposed because it is not only used for face detection but also the most famous gesture classifier [15]. The algorithm is the best choice for detecting hand signs in real-time based on its accuracy and speed, taking no more than 40 ms [16] [17]. The system works by translating hand gestures into data using the Viola-Jones algorithm when a speech-impaired person orders food. After the hand sign is translated, the data enters Firebase as a real-time database, and the seller can determine the total purchase through a cashier application. The system's output is a printed receipt, facilitating transactions for speech-impaired individuals.

The paper is presented in several sections: introduction, research methods, research results and discussion, and conclusion.

II. Research Methods

The Viola-Jones algorithm combines shapes and edges, features, template matching, and other statistics with AdaBoost. Haar-like features are used to calibrate features, and feature evaluation is accelerated by integral images. Adaboost filters the cascade classifier to eliminate non-object images and improve accuracy.

i. Haar-like Features

Haar-like features are classified into three categories: edge features, line features, and diagonal features, which are combined into feature templates. There are white and black matrices in the feature template, and the eigenvalue of the template is defined as the subtraction of the white rectangle pixels and the black rectangle pixels. Haar eigenvalues reflect changes in the grayscale of an image.

ii. Integral Image

Haar feature calculations are performed by summing all pixels in a rectangular region. The Viola-Jones detection algorithm uses the concept of integral images. The sum of pixels in all regions of the image can be obtained in a single image pass, significantly increasing the computational efficiency of eigenvalue calculations. Figure 1 shows an integral image where the value at the first point is the sum of pixels from A, while the third point is the sum of pixels from A and C, and so on.

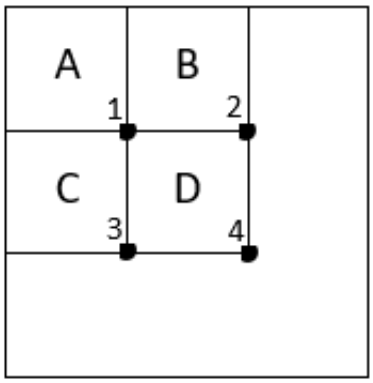

Figure 1: Integral image

iii. Adaboost

Adaboost is an algorithm capable of performing feature selection and classification simultaneously. The Adaboost algorithm is iterative and can eliminate some unnecessary training data features, activating only the primary training data.

iv. Cascade Classifier

In the cascade classifier architecture, all rectangular features are divided into several groups, each containing some rectangular features used at each stage of the cascade classifier. Each stage of the cascade classifier determines whether the area is the desired object or not. If yes, it proceeds to the next, more complex classifier stage [18].

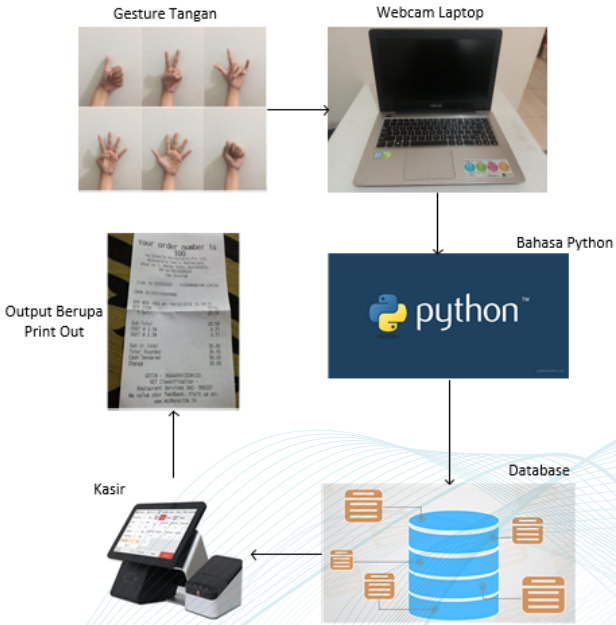

Figure 2: System model

The system model for translating hand signs for speech-impaired individuals at drive-thru fast food restaurants is shown in Figure 2. When a speech-impaired individual arrives to order food, they show hand gestures to the camera. The system can read six hand gestures, which include: the first gesture means burger, the second means chicken, the third means a 5-piece chicken meal, the fourth means rice, the fifth means ice cream, and the sixth means next. The next gesture functions to send the order data to the database. If someone wants to order a burger and chicken, they must give the burger gesture, then the next gesture, followed by the chicken gesture, and the next gesture. The system records the ordered menu as data and forwards it to a real-time database. Once the data is entered, the ordered menu will be processed in the cashier application and generate the total payment, which can be seen by the admin at the next counter. The admin prints out the receipt and gives it to the customer for payment.

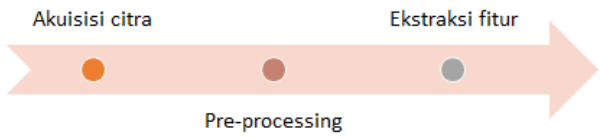

Figure 3: Main image processing steps

Figure 3 shows the image processing steps in the system, consisting of three main processes: image acquisition, pre-processing, and feature extraction [19]. Image acquisition is the process of capturing hand images in real-time. The next step is pre-processing using the Viola-Jones algorithm. The output of pre-processing is the classification of the image as either a hand or not. If the classification output is a hand image, the next step is feature extraction, aiming to recognize hand characteristics by extracting hand features.

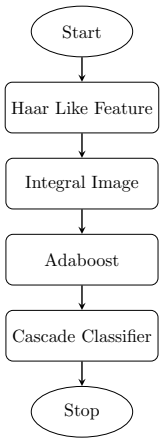

Figure 4: Pre-processing steps

The pre-processing steps for hand detection in detail are shown in Figure 4. The initial stage in the Viola-Jones algorithm is converting the image color from RGB to black and white, referred to as Haar-like Features [20]. The hand image is converted to white, while the background is converted to black, as illustrated in the program snippet in Figure 5. The lower_skin is intended to convert the background color to black, while the upper_skin converts the hand skin color to white.

```
hsv = cv2.cvtColor(roi, cv2.COLOR_BGR2HSV)

lower_skin = np.array([0,20,70], dtype=np.uint8)
upper_skin = np.array([20,255,255], dtype=np.uint8)
```

Figure 5: Haar-like Feature program snippet

The next stage is Integral Image to accelerate the calculation process, given that many features exist in an image. After the Integral Image process, Adaboost is used for the hand and background in the input image. The final part is the Cascade Classifier, which classifies the input image as a hand or not. The pre-processing also includes dilation using an elliptical kernel and blurring to help reduce noise. Figure 6 shows the image during the pre-processing stage.

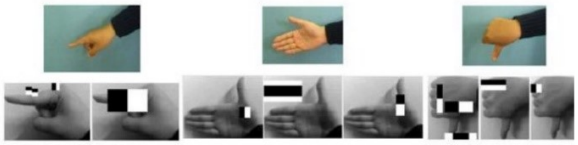

Figure 6: Pre-processing stage image

The feature extraction stage involves extracting information from an image. The feature extraction flow is shown in Figure 7. First, the object's edges are determined and defined as hand contours, illustrated in the program snippet in Figure 8. Next, the fingertips are determined, represented by Hull points found using the Convex Hull algorithm. Additionally, finger descriptions are represented by defect points, found using Convexity Defects. Convexity Defects returns an array where each row contains values such as start point, end point, and farthest point.

The next step is to calculate the finger length using equations as shown in the program snippet in Figure 9. In addition to calculating finger length, the feature extraction also includes calculating the distance between fingertips, referenced in Equation 1.

$$d = \frac{2 \cdot ar}{a} \tag{1}$$

The calculation continues with determining the angle between fingers using the variables a, b, and

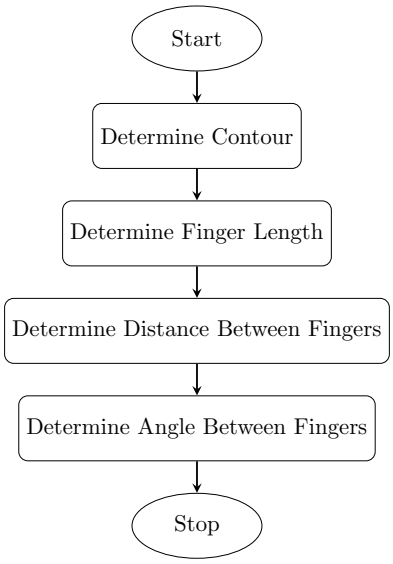

Figure 7: Feature extraction flow

```
contours,hierarchy= cv2.findContours(mask,cv2.RETR_TREE,cv2.CHAIN_APPROX_SIMPLE)

cnt = max(contours, key = lambda x: cv2.contourArea(x))

epsilon = 0.0005*cv2.arcLength(cnt,True)
approx= cv2.approxPolyDP(cnt,epsilon,True)
```

Figure 8: Contour program snippet

```
a = math.sqrt((end[0] - start[0])**2 + (end[1] - start[1])**2)
b = math.sqrt((far[0] - start[0])**2 + (far[1] - start[1])**2)
c = math.sqrt((end[0] - far[0])**2 + (end[1] - far[1])**2)
s = (a+b+c)/2
ar = math.sqrt(s*(s-a)*(s-b)*(s-c))
```

Figure 9: Finger length program snippet

c obtained while calculating the finger length. The relationship between the variables a, b, and c can be explained in Figure 10.

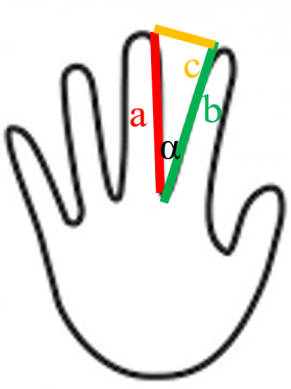

Figure 10: Relationship of variables a, b, and c

In trigonometry, the cosine law relates the lengths of the sides of a triangle to the cosine of one of its angles. Based on Figure 10, the cosine law states thatα is the angle between sides a and b, opposite side c. Therefore, to determine the length of side c, refer to Equation (2). Equation (3) is used to determine the angle α.

$$c = \sqrt{a^2 + b^2 - 2ab \cdot \cos(\alpha)} \tag{2}$$

$$\alpha = \cos^{-1}\left(\frac{a^2 + b^2 - c^2}{2ab}\right) \tag{3}$$

The angle between fingers is determined by referencing Equation 3 and the program snippet shown in Equation (4).

$$\text{angle} = \cos\left(\frac{b^2 + c^2 - a^2}{2 \times b \times c}\right) \times 57 \tag{4}$$

Once the hand sign has been determined, the next step is to use the if command to translate the hand sign into the ordered menu. Each menu has JSON data containing the menu, price, and quantity. Each piece of data will be stored in the `tmp_menu` variable. When the program detects the next sign, it will automatically send the `tmp_menu` variable to the send function. If no menu is selected before the next sign, no menu will be sent to the database, where the real-time database used is Firebase.

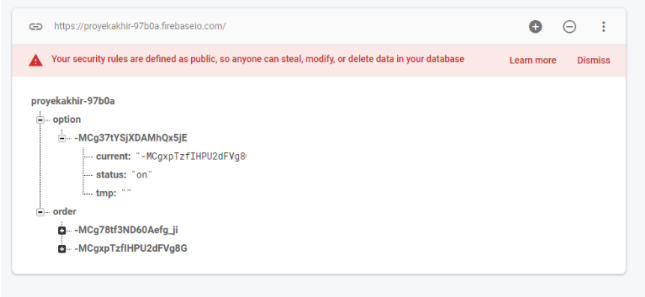

Figure 11: Firebase display

Based on Figure 11, the current data shows the active order data. If the cashier application has already saved the data, the status will change to off. When a new order is placed, the current data will change according to the new order. The current data contains a randomly generated code. The tmp contains the data currently being recorded. The order data contains a collection of previously completed orders.

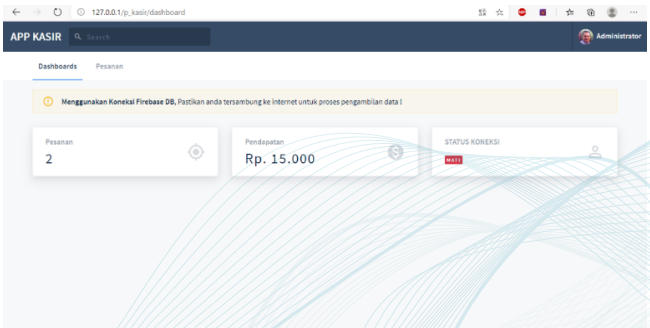

Figure 12: Dashboard menu display

In the cashier application, there are two menus, namely dashboard and orders, as shown in Figure 12. The dashboard menu contains the total orders that have been placed and the total revenue. Figure 13 shows the menu used to print the summed order receipt, making it easier for speech-impaired individuals to make payments.

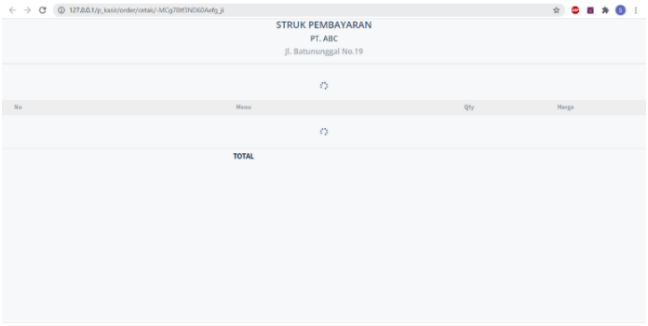

Figure 13: Data to be printed

III. Results and Discussion

The implementation of the hand sign language translator in drive-thru fast food restaurants was tested by simulating the order process, where the input is hand signs through a camera and the final output is a printed receipt. For example, if a customer wants to order chicken and a burger, the process occurring in the system sequentially is shown in Figure 14.

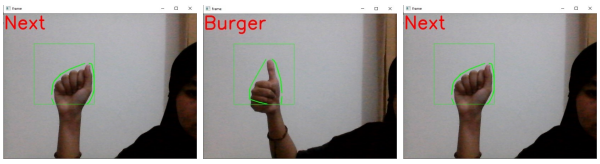

Figure 14: Input is the "next" hand sign/gesture to continue the order (left), Input is burger hand sign/gesture (middle), Input is the "next" hand sign/gesture to send data to the database (right)

i. Testing Hand Signs Against Light Intensity Changes

The purpose of the test is to determine the effect of light intensity when giving hand signs on the translation results. The test was conducted using two scenarios. The first scenario was carried out with a light intensity of 55 lux, while the second scenario was conducted with a light intensity of 71 lux, with data collection for each gesture 30 times. The test results are shown in Figure 15. Based on the displayed graph, it can be seen that changes in light intensity affect the translation results, evidenced by the difference in success rates between scenario 1 (55 lux) and scenario 2 (71 lux). From the two test scenarios, all hand signs have a success rate of above 90%. The average success rate for scenario 1 is 93.33% and for scenario 2 is 95.56%. Therefore, it is concluded that light sources with intensities of 55 lux and 71 lux can be used for hand sign translation. The minimum and maximum light intensity values are not discussed in this paper due to the limited variation of light sources that can be used.

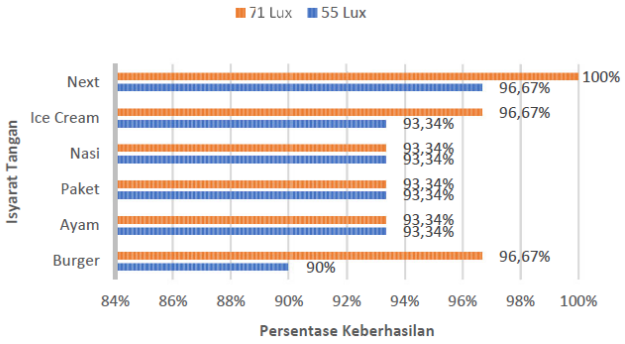

Figure 15: Test results for light intensity changes

ii. Testing Hand Signs Against Distance Changes

The test was conducted to determine the effect of distance changes between the hand and the camera on the system's ability to translate signs/gestures. As before, the test was conducted for two different conditions, namely distances of 50 cm and 60 cm. Data collection for each gesture was conducted 30 times. Figure 16 shows the test results of hand signs against distance changes, where each gesture has a relatively similar success rate except for the ice cream gesture. When the test distance is 50 cm, the average success rate for translating the ice cream gesture drops sharply to 16.67%. The average success rate for a distance of 50 cm is 67.88% and for a distance of 60 cm is 96.11%. Therefore, it is concluded that at a distance of 60 cm, hand sign translation can be carried out. The minimum and maximum distance values are not discussed in this paper.

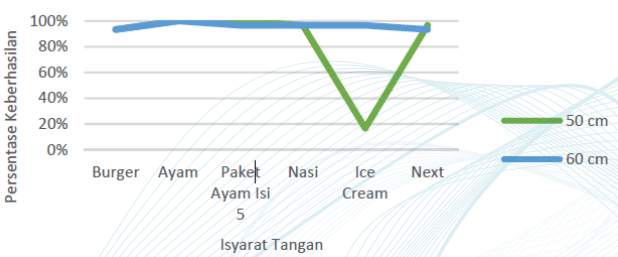

Figure 16: Test results for distance changes

iii. Testing Hand Signs Against Angle Changes

The test of hand signs against angle changes was carried out using a light intensity of 71 lux, a distance of 60 cm, a white background, and with angles of 0°, 45°, and 90°. The angle change referred to is the angle formed between the camera position and the hand, where the hand rotates to the right and/or left. For each hand sign, data was collected 30 times, resulting in the data shown in Table 1. The test results show that the highest average success rate is when the angle formed is 0°, which is 96.11%, followed by 45° and 90°. This is because the hand object is directly facing the camera, making the detection and translation process easier. Conversely, when the angle formed between the camera and the object is 90°, the detection and translation process becomes more difficult, with an average success rate of 74.44%.

Table 1: Test results for angle changes

Hand Sign	0°	45°	90°
Burger	96.67%	90%	16.67%
Chicken	96.67%	96.67%	96.67%
5-piece Chicken Meal	100%	83.34%	86.67%
Rice	93.34%	86.67%	86.67%
Ice Cream	96.67%	76.67%	66.67%
Next	93.34%	70%	93.34%
Average	96.11%	83.89%	74.44%

iv. Testing Hand Signs with Incomplete Objects

The purpose of this test is to determine whether the system can still translate hand signs when the object is partially cut off. The test of hand signs with partially and completely intact objects was conducted using the best conditions from previously tested parameters, with data collection for each gesture 30 times. Table 2 shows the test results for gestures as partially cut off and complete objects. The test results show that when the object is partially cut off, the hand sign detection and translation process cannot be performed. An object is considered cut off when the given hand sign is not entirely within the designated box area. This causes the system to incorrectly translate the given hand sign. An example of testing with a partially cut-off object can be seen in Figure 17.

v. Testing Hand Signs with Different Respondents

The purpose of this test is to determine the reliability of the system that has been created by varying the respondents giving hand signs. The number of respondents tested was 10 people. The test of hand signs with

Table 2: Test results for incomplete objects

Gesture	Partially Cut Off	Complete Object
Burger	10%	96.67%
Chicken	0%	93.34%
5-piece Chicken Meal	0%	93.34%
Rice	0%	93.34%
Ice Cream	0%	93.34%
Next	0%	100%
Average	1.66%	95%

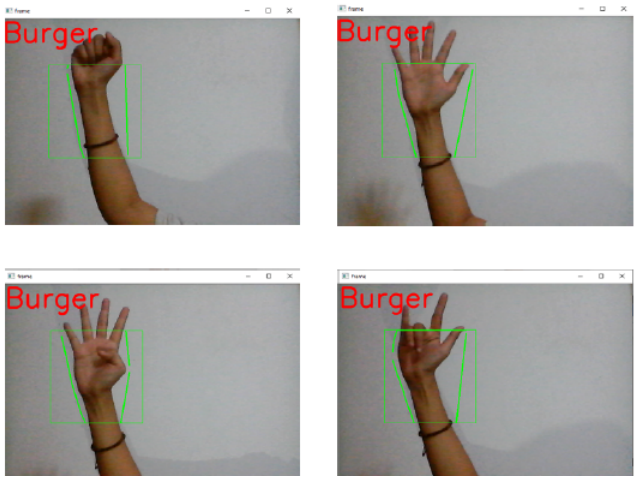

Figure 17: Example of testing with a partially cut-off object

different respondents used the best conditions from previously tested parameters, with data collection for each hand sign for each respondent 30 times. The test results are written in Table 3, where the average success rate of the system for different respondents is 92.01%.

IV. Conclusion

The development of a hand sign language translation system for speech-impaired individuals in drive-thru fast food restaurants has been successfully realized. The system is capable of translating the given gestures, with the data entering Firebase as a real-time database, and the seller can determine the total purchase through the cashier application. The system output is a printed receipt, facilitating transactions for speech-impaired individuals. The detection process uses the Viola-Jones algorithm. Based on the test results, it is known that the system can work more optimally when using higher light intensity and a distance between the camera and the hand in the range of 50-60 cm, with a white background as it can facilitate the detection process. The hand tilt position affects the translation result, with the 0° angle having the highest average accuracy of 96.11%. The system cannot translate gestures if the given hand sign is not entirely within the designated box area. Based on tests with 10 different respondents,

Table 3: Test results with different respondents

Hand Sign	1	2	3	4	5	6	7	8	9	10	Average
Burger	96.66%	93.33%	93.33%	86.66%	93.33%	96.66%	93.33%	83.33%	93.33%	90%	91.99%
Chicken	93.33%	96.66%	93.33%	93.33%	93.33%	90%	93.33%	96.66%	83.33%	90%	92.33%
5-piece Chicken Meal	90%	93.33%	86.66%	93.33%	93.33%	90%	93.33%	93.33%	90%	90%	91.33%
Rice	96.66%	93.33%	96.66%	90%	96.66%	93.33%	83.33%	93.33%	90%	86.66%	91.99%
Ice Cream	93.33%	90%	90%	93.33%	93.33%	96.66%	90%	93.33%	83.33%	93.33%	91.47%
Next	93.33%	96.66%	93.33%	93.33%	93.33%	90%	96.66%	93.33%	86.66%	93.33%	92.95%
Average Success Rate											92.01%

the hand sign translator system can detect the given gestures with an average system success rate of 92.01%.

References

[1] H. Purwanto, *Ortopedagogik Umum*, 1998.

[2] V. Wiley and T. Lucas, "Computer vision and image processing: A paper review," *International Journal Of Artificial Intelligence Research*, vol. 2, no. 1, pp. 28–36, 2018. [Online]. Available: https://doi.org/10.29099/ijair.v2i1.42

[3] N. Anasta, F. X. A. Setyawan, and H. Fitriawan, "Disease detection in banana trees using an image processing-based thermal camera," in *IOP Conference Series: Earth and Environmental Science*, 2021, pp. 1–8. [Online]. Available: https://doi.org/10.1088/1755-1315/739/1/012088

[4] A. Mohan and S. Poobal, "Crack detection using image processing: A critical review and analysis," *Alexandria Engineering Journal*, vol. 57, no. 2, pp. 787–798, 2018. [Online]. Available: https://doi.org/10.1016/j.aej.2017.01.020

[5] P. Afshar, A. Mohammadi, and K. N. Plataniotis, "Brain tumor type classification via capsule networks," in *International Conference on Image Processing (ICIP)*. IEEE, 2018, pp. 3129–3133.

[6] G. Marbach, M. Loepfe, and T. Brupbacher, "An image processing technique for fire detection in video images," *Fire Safety Journal*, vol. 41, pp. 285–289, 2006. [Online]. Available: https://doi.org/10.1016/j.firesaf.2006.02.001

[7] D. Yang, A. Alsadoon, P. W. C. Prasad, A. K. Singh, and A. Elchouemi, "An emotion recognition model based on facial recognition in virtual learning environment," in *International Conference on Smart Computing and Communications*. Elsevier B.V., 2018, pp. 2–10. [Online]. Available: https://doi.org/10.1016/j.procs.2017.12.003

[8] E. Alajrami, B. A. M. Ashqar, B. S. Abu-nasser, A. J. Khalil, M. M. Musleh, and A. M. Barhoom, "Handwritten signature verification using deep learning," *International Journal of Academic Multidisciplinary Research*, vol. 3, no. 12, pp. 39–44, 2019.

[9] A. N. Kadum, "Face detection method with mask by improved yolov5," *Journal of Image Processing and Intelligent Remote Sensing*, vol. 4, pp. 9–19, 2023. [Online]. Available: https://doi.org/10.55529/jipirs.41.9.19

[10] M. R. Rasyid, Z. Tahir, and N. Syafaruddin, "Digital image processing for detecting industrial machine work failure with quantization vector learning method," *Journal Pekommas*, vol. 4, no. 2, p. 131, 2019. [Online]. Available: https://doi.org/10.30818/jpkm.2019.2040203

[11] H. Huang, Y. Chong, C. Nie, and S. Pan, "Hand gesture recognition with skin detection and deep learning method," in *IOP Conference Series: Journal of Physics*, 2019, pp. 1–6. [Online]. Available: https://doi.org/10.1088/1742-6596/1213/2/022001

[12] T. Mantecon, C. R. Del-Blanco, F. Jaureguizar, and N. Garcia, "A real-time gesture recognition system using near-infrared imagery," *PLoS One*, pp. 1–17, 2019. [Online]. Available: https://doi.org/10.1371/journal.pone.0223320

[13] V. Mukthineni, R. Mukthineni, and O. Sharma, "Face authenticated hand gesture based human computer interaction for desktops," *Cybernetics and Information Technologies*, vol. 20, no. 4, pp. 74–89, 2020. [Online]. Available: https://doi.org/10.2478/cait-2020-0048

[14] A. Zarkasi, S. Nurmaini, D. Setiawan, A. Kuswandi, and S. D. Siswanti, "Implementation of facial feature extraction using viola-jones method for mobile robot system," in *Journal of Physics: Conference Series*. Institute of Physics Publishing, 2020. [Online]. Available: https://doi.org/10.1088/1742-6596/1500/1/012011

[15] D. M. Abdulhussien and L. J. Saud, "An evaluation study of face detection by viola-jones algorithm," *International Journal of Health Sciences (Qassim)*, pp. 4174–4182, 2022. [Online]. Available: https://doi.org/10.53730/ijhs.v6ns8.13127

[16] S. Antoshchuk, M. Kovalenko, and J. Sieck, "Gesture recognition-based human – computer interaction interface for multimedia applications," in *Digitisation of Culture: Namibian and International Perspectives*, 2018, pp. 269–286. [Online]. Available: https://doi.org/10.1007/978-981-10-7697-8

[17] S. Kudubayeya, N. Amangeldy, A. Sundetbayeva, and A. Sarinova, "The use of correlation analysis in the algorithm of dynamic gestures recognition in video sequence," in *International Conference on Engineering and MIS*, 2019, pp. 1–6.

[18] J. Huang, Y. Shang, and H. Chen, "Improved viola-jones face detection algorithm based on hololens," *EURASIP Journal on Image and Video Processing*, vol. 6, pp. 1–11, 2019. [Online]. Available: https://doi.org/10.1186/s13640-019-0419-1

[19] A. Jefiza *et al.*, "Klasifikasi wajah manusia menggunakan multi layer perceptron," *Jurnal Integrasi*, vol. 15, no. 2, pp. 142–148, 2023.

[20] A. Sinha and S. Barde, "Multi invariant face detection via viola jones algorithm," *European Chemical Bulletin*, pp. 24–32, 2023. [Online]. Available: https://doi.org/10.31838/ecb/2023.12.s1.003