Method Development Through Landmark Point Extraction for Gesture Classification With Computer Vision and MediaPipe

Suherman Suherman [1], Adang Suhendra [2], Ernastuti Ernastuti [2]

[1]Department of Computer Science and Engineering, Universitas Serang Raya, Banten, Indonesia
[2]Department of Computer Science and Engineering, Universitas Gunadarma, Jakarta, Indonesia

***Abstract* - Examining the physical movements of students during their educational quests holds great significance as these nonverbal cues can exert a substantial influence on academic performance, and boost, learning outcomes, Consequently, numerous researchers are engaged in exploring the domain of gesture categorization employing machine learning techniques. Initially, we conducted an observation of students' movements in a virtual learning environment during face-to-face interactions with their teachers. This procedure yielded a roster of thirteen motion-based behaviors, encompassing actions such as tilting the head towards either direction, lowering and lifting the head, as well as gesturing with the right and left hand towards the head and neck area, and positioning the shoulders in a front and lateral direction. This research offers a technique for establishing a set of criteria for categorizing students' gesticulations in online learning by utilizing the comprehensive MediaPipe holistic library and OpenCV to detect, pose and extract salient landmarks. This endeavor culminated in the attainment of a percentage-based metric indicative of gesture identification efficacy pertaining to the aforementioned thirteen motion-based activities.**

DOI: 10.18421/TEM123-49
https://doi.org/10.18421/TEM123-49

Corresponding author: Suherman Suherman,
Department of Computer Science and Engineering Universitas Serang Raya, Banten, Indonesia
Email: suherman.unsera@gmail.com

Received: 16 March 2023.
Revised: 04 July 2023.
Accepted: 19 July 2023.
Published: 28 August 2023.

***Keyword* - Gesture, machine learning, online learning, MediaPipe.**

1. Introduction

The internet can serve as an efficacious pedagogical resource for both students and teachers alike, offering a virtual classroom that affords a superior alternative to conventional modes of instruction [1]. Virtual and traditional classes diverge primarily in their utilization of digital tools, as the former relies on internet-based platforms like Zoom, Google Meet, and comparable software applications to effectuate its educational objectives. Despite being pre-recorded, virtual classes are conducted in real-time between teachers and students [2]. The structure of a web-based/online learning environment permits the implementation of efficacious interaction between teachers and students, wherein learning is a cognitive process that can be experienced either through individual activity or via sensory input, such as through gestural expression or via the five senses. The endeavor to achieve favorable learning outcomes via the incorporation of gestural or sensory-based modalities into pedagogical frameworks is termed as "learning styles". Gestures, for instance, are a type of nonverbal communication that individuals employ in their unique manner while engaging in the process of learning [3], [4], [5]. These learning styles play a crucial role in determining academic success and can exert a positive influence on students' achievements [6]. Learning styles are found to be closely associated with and capable of influencing learning achievement by up to 52% [7]. Additionally, a number of other research works have noted the significant impact of learning styles on students' academic accomplishments, with effect sizes reaching as high as 80.8% [8]. Based on gestural expression, learning styles can be classified into three primary categories, including auditory, visual, and kinesthetic modes of learning [9], [10].

The kinesthetic learning style is primarily associated with physical behavior, involving a proclivity towards learning through physical activity, being attuned to expressions and nonverbal communication, possessing a physically-oriented disposition, and exhibiting a proclivity towards frequent movement [9]. In the context of learning, gestural behavior serves as a type of nonverbal communication that encompasses head, body, and hand gestures, including those of the hands, fingers, and shoulders [11], as well as the combination of these elements in tandem with facial expressions to effectively convey messages [12]. During class time, a student's posture and gestural behavior may involve gazing to the side, standing, bowing the head, writing, and reading, among other actions [13]. As depicted in Figure 1 below, such gestural behavior is a form of nonverbal communication that embodies a learning style and is typically exhibited through the movement of the head, shoulders, and hands.

Figure 1. Presents observation results of kinesthetic movements during learning: 1. Head up 2. Head down 3. Head turned to the right 4. Left hand to the head 5. Left hand to the face 6. Left hand to the neck 7. Shoulder to the right 8. Shoulder to the left 9. Shoulder to the front 10. Right hand to the head 11. Right hand to face 12. Right hand to neck 13. Head turned to the left

In the course of this study, the authors monitored students who were engaged in online learning sessions conducted via Zoom meetings. The observations gleaned from this research demonstrate that students tend to exhibit the types of movements detailed in Figure 1 during their learning activities. The outcome of this study establishes regulations from the detection, tracking, and extraction findings as a dataset to recognize student gestures performed during learning, specifically the movements made by the head, shoulders, and hands. These movements include the head turning to the left and right, looking up and down, shoulders moving to the left, right, and forward, and hands moving to the face, neck area and head. The outcomes of this research offer a valuable contribution towards future studies aimed at establishing a classification system for the learning styles of students.

2. Background of the Study

Object detection, tracking, and extraction using machine learning were carried out in this study by employing the Mediapipe model through a Python application. The Mediapipe framework is specifically developed for artificial intelligence applications and incorporates Tensor Flow, which enables acceleration of a device's GPU (Graphics Processing Unit) and CPU (Central Processing Unit) [14]. The BlazePose technique allows Mediapipe to detect and track 33 2D landmark points on the body in each RGB video frame, providing a comprehensive and precise view of the body's movements during learning [15]. The BlazePose technique identifies the key points on the identified body part, which produces x, y, and z axis coordinates as the detection result output [16]. Mediapipe has shown substantial advancements in solving problems related to image classification, object detection, object localization, and image segmentation. Hence, this model has gained immense popularity in the field of Computer Vision, particularly in the area of gesture detection [17]. Based on the 33 identified keypoints by mediapipe (as depicted in Figure 2), the proposed study focuses on detecting landmarks on the head, shoulders, and hands by concentrating on the position of 8 specific keypoints, as illustrated in Figure 3.

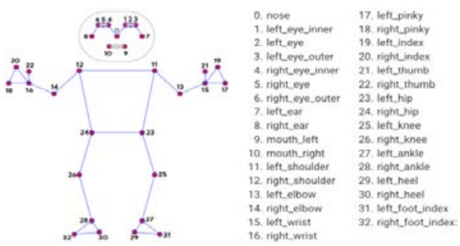

Figure 2. Landmark pose 33 keypoint

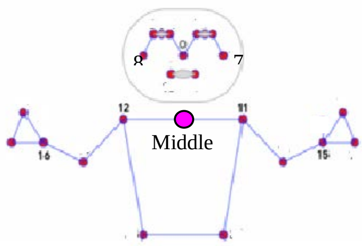

Figure 3. (8 keypoint detection

(MediaPipe, 2020)

The midpoint depicted in Figure 3 is generated by combining keypoint 11 and keypoint 12 through the equation:

xMiddle, yMiddle = int((x11+x12)/2), int((y11+y12)/2).

The resultant xMiddle and yMiddle coordinates refer to the middle keypoint, which corresponds to the pose keypoint of the neck.

In order to detect student gestures during learning, an extraction process is performed after the object is detected. This extraction process involves measuring the distance and comparing the coordinate points of a moving object per frame. The Euclidean distance formula is used to calculate the distance between these coordinate points. The Euclidean distance equation is a mathematical formula used to measure the distance between two points in Euclidean space, which is a space with a fixed number of dimensions where the Pythagorean theorem can be applied [18] The formula is widely used in mathematical and scientific applications to calculate distances in one or more dimensions, and is based on the principle of studying the relationship between angles and distances [19].

as a research proposal)

The equation for Euclidean distance is as follows :

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

Description :
d = Distance
x1 = initial x coordinate point value,
x2 = destination x coordinate point value
y1 = initial y coordinate point value,
y2 = destination y coordinate point value

3. Proposed Methods

The data used in this study was collected by recording videos of students participating in Zoom meetings during a 2 minute and 30 second learning session while sitting in a standard position. The standard sitting position involves sitting on a flat surface with a straight view, keeping the shoulders relaxed, the upper arms vertical, the forearms horizontal, and the lower legs vertical (Sritomo, 1995). The captured video is subsequently divided into segments, each representing an individual student (object), and treated as input data for further analysis. This process involves the following stages at figure 4 :

Detection and Tracking → Landmark Extraction → Methods for determining the classification of movement activities → Classification of Movement Activities

Figure 4. Stages of recognizing movement activities

In Figure 4, the first time the video input is read by OpenCV, the object will be captured in each frame by the OpenCV function to be acquired, processed, and made a decision. Before the keypoint is displayed, with the OpenCV function, the conversion process to RGB is required so that the color display can be displayed by the MediaPipe function correctly. The 8 specified keypoints are identified based on the location of the (x, y) coordinate points of the MediaPipe. The keypoints are identified based on the image width and image height to get the appropriate image size. To detect the 8 keypoints using the holistic MediaPipe library that will automatically integrate all 8 keypoints on the head, hands, and shoulders. After the 8 points are identified, they will be processed by the openCV function to be displayed on each Object frame according to the keypoint location on the MediaPipe with a bright green keypoint image as shown in Figure 5:

Figure 5. 8 Keypoint detected on the object

Explanation of figure 5, the head part is located on the nose (keypoint 0), left ear (keypoint 7), and right ear (keypoint 8). The shoulder is located on the left shoulder (keypoint 11) and right shoulder (keypoint 12). The hand is located at the left wrist (keypoint 15), right wrist (keypoint 16) and middle (keypoint between the left shoulder and right shoulder). Step 2 detection and tracking, objects commonly exhibit dynamic movements by carrying out a reaction [21].

The dynamic movement performed by the object is conducted in a stationary manner (i.e., sitting). The tracking of movement can be employed as material for the process of movement capture [20]. In this proposal for research, tracking refers to the procedure of transferring the x and y coordinate points from their initial positions to the next positions until they reach the destination x and y coordinate points. Each of the x and y coordinate points contain a respective value. As shown in the following Figure 6 :

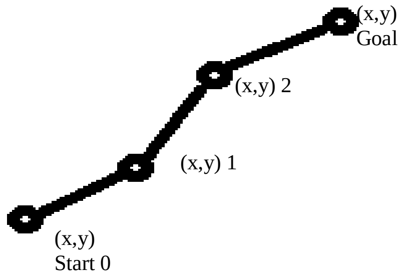

Figure 6. Tracking process

In Figure 6 shows, the procedure of tracking the starting coordinate point towards the goal involves passing through intermediate points on the way until the goal is reached. The distance covered throughout this tracking process will eventually approach zero.

Step 3 landmark extraction, extraction refers to the analysis of an object's movement, and the results of this analysis can be utilized as input for classification purposes. Specifically, extraction involves tracking analysis that yields the distance value by reducing the goal x, y coordinate point value with each displacement of the start x, y coordinate point value, until the value of the goal x, y coordinate point is reached. Thus, the distance between the start and goal coordinate points can be determined.

Step 4, the proposed method of determining the motion activity of the object is done by :

a. Involves the computation of the spatial separation between two distinct key points, identified as the starting point (x1,y1) and the goal point (x2,y2).

By using the Euclidean Distance equation, the distance value between 2 keypoints can be calculated. So obtained for each distance value between 2 keypoints as shown in Table 1 below:

Table 1. Distance equation between 2 keypoints

Equation	Function
$t0\_8 = \sqrt{((x0-x8)^2+(y0-y8)^2)}$	To determine the distance value of keypoint 0 and keypoint 8
$t0\_7 = \sqrt{((x0-x7)^2+(y0-y7)^2)}$	To determine the distance value of keypoint 0 and keypoint 7
$t0\_middle = \sqrt{((x0-xMiddle)^2+(y0-yMiddle)^2)}$	To determine the distance value of keypoint 0 and the Middle keypoint
$K\_Top\_Down = \sqrt{((x0-xMiddle)^2+(y0-yMiddle)^2)}$	To determine the distance set value between the 0 and Middle keypoint values when the object is initially detected
$t0\_16 = \sqrt{((x0-x16)^2+(y0-y16)^2)}$	To determine the distance value of keypoint 0 and keypoint 16
$t0\_15 = \sqrt{((x0-x15)^2+(y0-y15)^2)}$	To determine the distance value of keypoint 0 and keypoint 15
$tMiddle\_16 = \sqrt{((xMiddle-x16)^2+(yMiddle-y16)^2)}$	To determine the distance value of the Middle keypoint and keypoint 16
$tMiddle\_15 = \sqrt{((xMiddle-x15)^2+(yMiddle-y15)^2)}$	To determine the distance value of the Middle keypoint and keypoint 15
$t12\_Middle = \sqrt{((xMiddle-x11)^2+(yMiddle-y11)^2)}$	To determine the distance value of the Middle keypoint and keypoint 11
$t11\_Middle = \sqrt{((xMiddle-x12)^2+(yMiddle-y12)^2)}$	To determine the distance value of the Middle keypoint and keypoint 12
$tMiddle\_11 = \sqrt{((xMiddle-x11)^2+(yMiddle-y11)^2)}$	To determine the distance value of the Middle keypoint and keypoint 11
$BFront = \sqrt{((xMiddle-x11)^2+(yMiddle-y11)^2)}$	To determine the value of the distance set between the Middle and 11th keypoint values the first time the object is detected.

b. Comparing the distance values that have been calculated to determine the object's movement activity. The objective of juxtaposing the distance values documented in Table 2 is to gauge the degree of proximity between the two values and their effectiveness in identifying the direction of keypoint motion. Table 2 illustrates the outcome of the distance comparison :

Table 2. Distance comparison results

Comparison of distance value	Proximity of keypoint direction
t0_8 < t0_7	Keypoint 0 is moving towards keypoint 8
t0_7 < t0_8	Keypoint 0 is moving towards keypoint 7
t0_Middle < K_Top_Down	Keypoint 0 is moving towards the Middle keypoint
t0_Middle > K_Top_Down	Keypoint 0 is moving farther from the Middle keypoint.
t0_16 < tMiddle_16	Keypoint 16 is moving towards keypoint 0
tMiddle_16 < t0_16	Keypoint 16 is moving towards the Middle keypoint
t0_15 < tMiddle_15	Keypoint 15 is moving towards keypoint 0
tMiddle_15 < t0_15	Keypoint 15 is moving towards the Middle keypoint
t12_Middle < t11_Middle	Keypoint 12 is moving towards the Middle keypoint
t11_Middle < t12_Middle	Keypoint 12 is moving towards the Middle keypoint
tMiddle_11 > BFront	Keypoint 11 is moving farther from the BFront distance value

c. Determine the range of distance values to determine the range of threshold values. Comparison of distance values from Table 2 will produce a range of distance values as shown in Table 3. In Table 3, the distance value range data is generated from the training process using 5 detected objects, each object produces a different distance value range, so that the threshold value range will be determined from the smallest to the largest, as shown in Table 3 below :

Table 3. Determination of distance value range

Number	Keypoint comparison result	Object and Range distance value					Threshold value
		1	2	3	4	5	
1	t0_8 < t0_7	20.12-84.15	61.39-71.84	17.49-85.01	45.12-77.65	9.9-85.01	9.9 – 85.1
2	t0_7 < t0_8	51.71-75.24	10.2-81.02	51.04-85.38	25.18-85.15	45.79-83.93	10.2 – 85.38
3	t0_Middle < K_Top _Down	167.03-236.12	234.36-236.28	89.2-241.17	234.1-241.12	236.76-241.03	89.2 – 241.17
4	t0_Middle > K_Top_Down	394.67-427.55			389.02-464.47		389.02 – 464.47
5	tMiddle_16 < t0_16	9 – 147.18	144.4-262.84	178.68-243.4	257.61-261.08	82.97-257.61	9 – 262.84
6	tMiddle_16 < t0_16	120.49-206.7	257.3-276.05	210.91-262.3	244.9-260.69		120.49 – 276.05
7	t0_16 < tMiddle_16	349.36-411.1	399.82-413.66	15.12-375.71		362.89-405.6	15.12 – 413.66
8	tMiddle_15 < t0_15	68.88-217.75	148.49-253.55	39.2-221.38	166.09-205.41	221.29-264.33	39.2 – 264.33
9	tMiddle_15 < t0_15	234.1-272.9	84.96-153.93	233.14-252.77	248.52-266.05	55.97-249.73	84.96 – 272.9
10	t0_15 < tMiddle_15	342.12-473.84	302.22-391.2	363.23-396.43	20.71-474.67	300.91-456.21	20.71 – 474.67
11	t12_Middle < t11_Middle	195.92-256.5		151.92-258.12	256.35-257.33	197.88-258.03	151.92 – 258.12
12	t11_Middle < t12_Middle	20.59-249.06	246.33-258.8	251.1-257.05	185.95-215.57	219.06-238.0	20.59 – 258.8
13	tMiddle_11 > BFront	395.1-436.04	405.24-446.08	410.49-487.56	399.21-404.1	429.2-549.0	395.1 – 549.0

d. Determining the value of distance that is considered ideal and aligns with the specific requirements of motion activity.

Based on the range of threshold values from Table 3, the ideal distance value is then determined to determine the distance value for each keypoint comparison. The ideal distance value can be determined between 71% - 100% of the maximum threshold value. The ideal distance value is obtained from the following equation:

Ideal distance value = Percent Value * maximum Threshold value

From this equation, the results of the ideal distance value are shown in Table 4 below:

Table 4. Determination of the ideal distance value

Number	Keypoint comparison result	Threshold value	Percent Value	Ideal distance value
1	t0_8 < t0_7	9.9 – 85.1	72%	61,3
2	t0_7 < t0_8	10.2 – 85.38	71,8%	61,3
3	t0_Middle < K_Top_Down	89.2 – 241.17	82,8%	199,7
4	t0_Middle > K_Top_Down	389.02 – 464.47	92%	427,3
5	tMiddle_16 < t0_16	120.49 – 276.05	100%	276,05
6	t0_16 < tMiddle_16	15.12 – 413.66	100%	413,66
7	tMiddle_15 < t0_15	84.96 – 272.9	100%	272,9
8	t0_15 < tMiddle_15	20.71 – 474.67	100%	474,67
9	t12_Middle < t11_Middle	151.92 – 258.12	75,9%	195,97
10	t11_Middle < t12_Middle	20.59 – 258.8	75,7%	195,9
11	tMiddle_11 > BFront	395.1 – 549.0	72%	395,28

e. Rule for determining movement activity.

From the results obtained in Table 4, a rule is generated to be able to determine the classification of movement activities as in Table 5 below:

Table 5. Movement activity classification rule

Condition		*Then*
Distance comparison	Threshold values	Movement activity
If (t0_8 < t0_7)	and (t0_8 <= 61.3)	Head turn to the right
If (t0_7 < t0_8)	and (t0_7 <= 61.3)	Head turn to the left
If (t0_middle < K_Top _Down)	and (t0_middle <= 199.7)	Head down
If (t0_middle > K_Top _Down)	and (t0_middle >= 427.3)	Head up
If (t16_middle < t16_0)	and (t16_middle <= 276.05)	Right hand to the middle (neck/face)
If (t16_0 < t16_middle)	and (t16_0 <= 413.66)	Right hand to the head
If (t15_middle < 15_0)	and (t15_middle <= 272.9)	Left hand to the middle (neck/face)
If (t15_0 < 15_middle)	and (t15_0 <= 474.67)	Left hand to the head
If (t12_Middle < t11_Middle)	and (t12_Middle <= 195.97)	Shoulder to the left
If (t11_Middle < t12_Middle)	and (t11_Middle <= 195.9)	Shoulder to the right
If (tMiddle_11 > BFront)	and (middle_11 >= 395.28)	Shoulder forward

Table 5 explains, when the system is run, each distance value will be calculated and compared, so that it will reach the smallest distance value of the 2 distance values. When it has reached the smallest distance value, then the distance value is compared with the threshold value.

If it has entered the threshold value, the system will continue to calculate the distance value until the distance value comes out of the threshold value. After the distance value is out of the threshold value, then the system will display the resulting movement activity.

The rules from Table 5 above have been able to detect movement activities performed by students, whether the movement is done only 1 movement or by using a combination of movements, such as the right hand to the head and the head while turning to the left. This can be generated from rules that can recognize all 8 keypoint movements when the object is detected.

4. Experimental Results

During the extraction testing phase, we utilized the Python program in combination with the comprehensive MediaPipe library to identify and track movements of the head, shoulders, and hands. To accomplish this, we employed video input derived from a zoom meeting recording that possesses a frame rate of 25 fps, and dimensions of 1280 by 720 pixels.

The Table presented as Table 6 serves as an exemplar of the application of rules governing the detection, tracking, and extraction of landmarks within the context of objects engaged in motion. Specifically, the record provided in Table 6 pertains to the activity of the head turning rightward and tilting upwards.

Table 6. Records of detection, tracking, and extraction results of rule

Frame	Htl Htr BowL Rhh Rhmidd Lhh Lhmidd Sr Sl Sfrnt	Activities
1	138.92 : 127.42 : 360.73 : 1196.4 : 879.55 : 1244.43 : 913.77 : 359.03 : 374.37 : 366.49	
2	137.93 : 127.42 : 360.73 : 1194.52 : 877.82 : 1244.43 : 913.77 : 359.03 : 374.46 : 366.49	
3	137.93 : 128.41 : 360.73 : 1192.65 : 876.09 : 1245.34 : 914.62 : 359.03 : 374.46 : 366.49	
4	137.93 : 128.41 : 359.74 : 1191.71 : 876.09 : 1245.34 : 915.48 : 359.03 : 374.55 : 366.55	
5	138.1 : 128.57 : 358.74 : 1188.9 : 874.36 : 1244.43 : 915.48 : 359.03 : 374.55 : 366.55	
6	196.35 : 55.71 : 375.41 : 1158.86 : 869.84 : 1268.05 : 900.12 : 364.09 : 361.6 : 362.61	
7	209.27 : 35.81 : 385.07 : 1146.72 : 869.42 : 1271.98 : 889.41 : 365.11 : 339.71 : 352.69	
8	211.26 : 29.0 : 388.29 : 1145.45 : 870.79 : 1271.66 : 884.64 : 365.14 : 334.94 : 349.76	
9	212.25 : 26.63 : 389.78 : 1150.13 : 875.9 : 1272.23 : 883.25 : 365.2 : 333.17 : 348.9	
10	213.24 : 24.7 : 390.16 : 1163.07 : 887.34 : 1272.83 : 883.27 : 365.23 : 335.28 : 349.97	
11	213.24 : 23.54 : 390.54 : 1170.79 : 894.71 : 1277.01 : 886.92 : 365.27 : 335.39 : 350.04	
12	227.56 : 98.49 : 437.12 : 1090.47 : 840.65 : 1320.12 : 885.66 : 365.27 : 330.22 : 347.3	
	Head turn to the right	
13	226.06 : 133.84 : 463.48 : 1165.27 : 899.86 : 1400.24 : 940.94 : 360.01 : 315.98 : 337.08	
14	226.06 : 133.84 : 464.3 : 1165.27 : 899.01 : 1400.24 : 940.05 : 360.01 : 315.82 : 337.0	
15	226.06 : 133.84 : 464.3 : 1165.27 : 899.01 : 1400.24 : 940.05 : 360.01 : 315.82 : 337.0	
16	226.06 : 133.84 : 463.48 : 1165.27 : 899.86 : 1400.24 : 940.94 : 360.01 : 315.82 : 337.0	
17	226.06 : 133.84 : 463.48 : 1165.27 : 899.86 : 1400.24 : 940.94 : 360.01 : 315.82 : 337.0	
18	165.44 : 66.12 : 399.7 : 1275.4 : 950.77 : 1324.21 : 933.3 : 366.07 : 366.2 : 366.14	
	Head up	

Table 6 describes htl (head turned to the left), htr (head turned to the right), bowL (head bowed or head up), rhh (right hand to the head), rhmidd (right hand to the middle (neck/face)), Lhh (left hand to the head), Lhmidd (left hand to the middle (neck/face)), sr (shoulder to the right), sl (shoulder to the left), and sfrnt (shoulder to the front). The outcome of the record in frames 1 to 5 exhibit the distance metric of object activity in a stable state for each distinct motion. Frames 6 to 11, however, demonstrate a reduction in the distance metric of head turning rightward from 55.71 to 23.54, thus causing the distance metric to fall below the threshold value of 61.3 designated for this specific motion. The activity record of the motion is presented only when the pertinent distance metric exceeds the specified threshold value. Furthermore, in the case of the bowL activity record, frames 1 to 12 demonstrate that the distance metric under typical circumstances still remains below the threshold value designated for the head-up activity, which is 427.3.

Subsequently, within frames 13 to 17, the distance metric for the bowL activity surpasses the value of 427.3, falling within the range of 463.48 to 464.3, which indicates an activity of the head moving upwards. This particular instance of head-up activity is captured in the activity record during the distance measurement taken in frame 18.

The rules in Table 5 were tested on 8 objects, so that the number for each movement activity could be analyzed.

The results of the analysis obtained the number of each object as in Table 7:

Table 7. Detection of the number of motion activities

No	Object	Motion activity											Quantity
		1	2	3	4	5	6	7	8	9	10	11	
1	Object 1	7	4	3	3	0	0	1	2	1	0	0	21
2	Object 2	3	0	4	0	0	8	0	2	1	0	3	21
3	Object 3	1	4	0	3	5	6	2	5	0	0	2	28
4	Object 4	0	1	0	7	0	1	0	0	0	0	0	9
5	Object 5	2	2	1	0	1	0	0	1	2	1	1	11
6	Object 6	1	3	4	1	1	2	0	2	0	0	2	16
7	Object 7	0	2	0	3	1	1	6	1	0	0	2	16
8	Object 8	0	0	0	0	0	0	0	1	0	0	2	3

Description Motion activity :

1. Head turn to the right
2. Head turn to the left
3. Head up
4. Head bowed
5. Right hand to the head
6. Right hand to the middle (neck/face)
7. Left hand to the head
8. Left hand to the middle (neck/face)
9. Shoulder to the right
10. Shoulder to the left
11. Shoulder forward

As can be observed from Table 7, the enumeration of each object's movement activities has been established. Specifically, for the head turning to the right movement, there were 14 detections made per object, whereas for the head turning to the left motion, 16 detections were made per object. Similarly, for the head looking up movement, the number of detections was 12 for each object, and for the head down motion, 17 detections were made per object. The detection process for the objects' movement activities revealed that 8 instances of the right hand placed on the head were observed for each object. Additionally, the number of instances where the right hand was placed on the middle of the neck or face was 18 for each object.

Similarly, 9 occurrences of the left hand placed on the head were detected for each object, whereas 14 instances of the left hand placed on the middle of the neck or face were observed for each object. Moreover, four instances of the shoulder moving to the right were detected for each object, while only one instance of the left shoulder was observed for each object. Finally, 12 instances of the shoulder moving to the front were detected for each object.

Upon obtaining the number of identified movements, an analysis is carried out to determine the accuracy levels of each movement by calculating the undetected movements and detected but unsuitable movements. Table 8 presents the outcomes of this evaluation.

Table 8. Accuracy of each motion activity

No	Motion activity	Detected as conforming	Detected as non-conforming	Not detected	%
1	Head turn to the right	14	0	0	100
2	Head turn to the left	16	0	0	100
3	Head bowed	17	0	0	100
4	Head up	12	0	3	80
5	Right hand to the middle	18	5	0	78.26
6	Right hand to the head	8	0	0	100
7	Left hand to the middle	14	0	0	100
8	Left hand to the head	9	0	0	100
9	Right shoulder to the right	4	0	0	100
10	Left shoulder to the left	1	0	0	100
11	Shoulder forward	12	0	0	100

Table 8 provides the accuracy outcomes for each movement activity. The accuracy results for each movement activity are derived by utilizing the subsequent equation :

The obtained accuracy results for each movement activity were computed by employing the equation: Accuracy value = (Number of correct data / Total movement) x 100, as depicted in Table 8. The accuracy outcomes were revealed to be 100% for head movements turning to the right, 100% for head movements turning to the left, and 100% for head movements looking down. On the other hand, the accuracy level for head movements looking up was 80%, whereas for right hand movements to the middle, it was 78.26%. The accuracy level was found to be 100% for right hand movements to the head, left hand movements to the center, left hand movements to the head, left shoulder movements, right shoulder movements, and forward shoulder movements.

5. Conclusion

Based on the findings of the study, it can be inferred that :

1. The findings of the study reveal that the holistic MediaPipe library has the capability to identify and track the movements of eight different coordinate landmarks located on the head, including the nose (keypoint 0), left ear (keypoint 7), and right ear (keypoint 8); on the hand, including the left wrist (keypoint 15) and right wrist (keypoint 16); and on the shoulder, including the left shoulder (keypoint 11) and right shoulder (keypoint 12). This integration of key points is done automatically.
2. The process of obtaining the rule for movement activity begins with detecting and tracking moving objects, allowing for the determination of the coordinate point value at each position. Subsequently, extraction is carried out on each value to identify the smallest distance value by comparing the distances between two points. Further extraction is conducted to establish the range of distance values for each movement activity using the training data, resulting in the minimum and maximum threshold values. The determination of the threshold value used in the gesture classification rule is adjusted to the posture of the analyzed object, ranging between 71% and 100% of the maximum threshold value.
3. The findings of the study yielded a set of rule for classifying student movement activities during online learning. Using this rule, the accuracy levels of each movement activity were determined. Specifically, the accuracy of head rotation to the right was 100%, head rotation to the left was 100%, bowing of the head was 100%, head raising was 80%, right-hand movement to the middle was 78.26%, right-hand movement to the head was 100%, left-hand movement to the center was 100%, left-hand movement to the head was 100%, left shoulder movement was 100%, and forward shoulder movement had an accuracy of 100%.
4. The mean precision of identifying 11 student movements from the formulated rule is 96.21%.

References

[1]. YuFei, L., Saleh, S., Jiahui, H., & Abdullah, S.M.S. (2020). Review of the Application of Artificial Intelligence in Education. *International Journal of Innovation, Creativity and Change*, *12*(8), 548-555.

[2]. Rapanta, C., Botturi, L., Goodyear, P., Guàrdia, L., & Koole, M. (2020). Online University Teaching During and After the Covid-19 Crisis: Refocusing Teacher Presence and Learning Activity. *Postdigital Science and Education*, *2*(3), 923–945.

[3]. Sariyanidi, E., Gunes, H., & Cavallaro, A. (2015). Automatic analysis of facial affect: A survey of registration, representation, and recognition. *IEEE transactions on pattern analysis and machine intelligence*, *37*(6), 1113–1133.

[4]. Rose, Colin., & Nicholi, M. J. (2006). *Accelerated Learning for the 21st Century* (Ahimsa, Dedi, Translation). Bandung: Nuance Publisher.

[5]. Gorden, Dryden., & Vos, Jeanette. (2003). *The Learning Revoluton*. Bandung : Kaifa.

[6]. Prashig, B. (2007). *The Power of Learning Styles: Mendongkrak Anak Melejitkan Prestasi dengan Mengenali Gaya Belajarnya [Boosting Children's Achievement by Recognizing Their Learning Style*.] (Fauziah, Nina, Translation). Bandung: Kaifa.

[7]. Khoeron, I.R., Sumarna, N., & Permana, N.T. (2014). Pengaruh gaya belajar terhadap prestasi belajar peserta didik pada mata pelajaran produktif. [The Influence Of Learning Styles On Learning Achievement Students In Productive Subjects.] *Journal of Mechanical Engineering Education*, *1*(2), 291-297.

[8]. DePorter, Bobbi., & Hernacki, Mike. (2013). *Quantum Learning: Making Learning Comfortable and Fun*. Bandung: Kaifa.

[9]. Hasanah, I., Kantun, S., & Djaja, S. (2018). Pengaruh gaya belajar terhadap hasil belajar siswa kelas xi jurusan akuntansi pada kompetensi dasar jurnal khusus di smk negeri 1 jember semester genap tahun ajaran 2017/2018. [The Effect of Learning Style on Student Learning Outcomes.] *Scientific Journal of Education, Economic, and Social Science*, *12*(2), 277-282.

[10]. Mitra, S., & Acharya, T. (2007). Gesture Recognition: A Survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, *37*(3), 311-324.

[11]. Hendra. (2014). *Seni Bicara dan Bahasa Tubuh [Art of speech and body language]*. Yogyakarta: Mantra Books.

[12]. Priyadharshni, V., Anand, M. S., & Kumar, N. M. (2013). Hand Gesture Recognition System Using Hybrid Technology For Hard of Hearing Community. *International Journal of Engineering Mathematich & Computer Scince*, *1*(2), 50-63.

[13]. Nguyen, Q.T., Binh, H.T., Bui, T. D., & P. Dung, P.N.T. (2019). Student postures and gestures recognition system for adaptive learning improvement. *Conference on Information and Computer Science (NICS), IEEE*, 494–499.

[14]. Zhang, Y., Xiang, T., Hospedales, T. M., & Lu, H. (2018). Deep mutual learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* 4320-4328. Doi: 10.1109/CVPR.2018.00454,

[15]. Karem, S.R., Kanisetti, S.P., Soumya, K., Sri, J., Seelamanthula, G., & Kalivarapu, M. (2021). AI Body Language Decoder using MediaPipe and Python. *International Journal of Advance Research, Ideas and Innovations in Technology*, *7*(3), 2436–2439.

[16]. Anilkumar, A., K.T. Atulya., Sajan, S., & S. K.A. Sreeja. (2021). Pose Estimated Yoga Monitoring System. *International Conference on IoT Based Control Networks and Intelligent Systems*, 1-8.

[17]. Josyula, R., & Ostadabbas, S. (2021). A review on human pose estimation. *arXiv preprint arXiv:2110.06877*.

[18]. Firdaus, D., & R.P. Yusya. (2016). Perancangan Ikat Pinggang Elektronik Untuk Tunanetra Menggunakan Mikrokontroller Dan Global Positioning System (Gps) Pada Smartphone Android [Electronic Waistband Design for the Blind Using Microcontroller and Global Positioning System (Gps) on Android Smartphone]. *Journal of Electrical Engineering ITP*, *5*(2), 130-136.

[19]. Mustofa, Z & Suasana, I.S. (2018). Algoritma clustering K-medoids Pada e-government Bidang information and communication technology Dalam Penentuan status Edgi [K-Medoids Clustering Algorithm on E-Government in the Field of Information And Communication Technology in Determining Edgi Status]. *Journal of Information and Communication Technology*, *9*(1), 1-10.

[20]. Miki´c, I., Trivedi, M., Hunter, E., & Cosman, P. (2003). Human body model acquisition and tracking using voxel data, *International Journal of Computer Vision, 53*(3), 199–223

[21]. Wang, L., & Suter, D. (2007). Learning and matching of dynamic shape manifolds for human action recognition, *IEEE Transactions on Image Processing, 16*(6), 1646–1661.