

A New Method for Head Direction Estimation based on Dlib Face Detection Method and Implementation of Sine Invers Function

Arqam M. Younis ^{1*}, Ghassan J. Mohammed ²

^{1,2}Department of Computer Science, College of Computer Science and Mathematics, University of Mosul, Mosul, Iraq

E-mail: ^{1*} arqam.csp35@student.uomosul.edu.iq, ² ghassan.j.m.aladool@uomosul.edu.iq

(Received July 29, 2021; Accepted September 14, 2021; Available online December 01, 2021)

DOI: [10.33899/edusj.2021.130962.1181](https://doi.org/10.33899/edusj.2021.130962.1181), © 2021, College of Education for Pure Science, University of Mosul.

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

Abstract

The detection and tracking of head movements have been such an active area of research during the past years. This area contributes highly to computer vision and has many applications of computer vision. Thus, several methods and algorithms of face detection have been proposed because they are required in most modern applications, in which they act as the cornerstone in many interactive projects. Implementation of the detected angles of the head direction is very useful in many fields, such as disabled people assistance, criminal behavior tracking, and other medical applications. In this paper, a new method is proposed to estimate the angles of head direction based on Dlib face detection algorithm that predicts 68 landmarks in the human face. The calculations are mainly based on the predicated landmarks to estimate three types of angles Yaw, Pitch and Roll. A python program has been designed to perform face detection and its direction. To ensure accurate estimation, the particular landmarks were selected, such that, they are not affected by the movement of the head, so, the calculated angles are approximately accurate. The experimental results showed high accuracy measures for the entire three angles according to real and predicted measures. The sample standard deviation results for each real and calculated angle were Yaw (0.0046), Pitch (0.0077), and Roll (0.0021), which confirm the accuracy of the proposed method compared with other studies. Moreover, the method performs faster which promotes accurate online tracking.

Keywords: Head direction, Dlib algorithm, Yaw, Pitch, Roll.

طريقة جديدة لتقدير اتجاه الرأس بالاعتماد على طريقة Dlib لاكتشاف الوجه وتوظيف دالة مقلوب الجيب

أرقم محمد يونس ^{1*}، غسان جاسم محمد ²

^{1,2} قسم علوم الحاسوب، كلية علوم الحاسوب والرياضيات، جامعة الموصل، الموصل، العراق

الخلاصة:

كان اكتشاف وتتبع حركات الرأس مجالاً نشطاً للبحث خلال السنوات الماضية، حيث يساهم هذا المجال بشكل كبير في الرؤية الحاسوبية وله العديد من التطبيقات. لذلك تم اقتراح عدة طرق وخوارزميات لاكتشاف الوجه، لأنها مطلوبة في معظم التطبيقات الحديثة، حيث تعمل كحجر أساس في العديد من المشاريع التفاعلية. ويعد تنفيذ الزوايا المكتشفة في اتجاه الرأس مهم جداً في العديد من المجالات، مثل مساعدة الأشخاص ذوي الإعاقة وتتبع السلوك الإجرامي والتطبيقات الطبية الأخرى. في هذا البحث اقترحنا طريقة جديدة لتقدير

زوايا اتجاه الرأس. وتعتمد على خوارزمية Dlib للكشف عن الوجه والتي تتنبأ بـ 68 معلمًا في وجه الإنسان. تعتمد حسابات الزوايا على بعض هذه المعالم لتقدير زوايا دوران الرأس الأفقية والإمالة العمودية والإمالة الجانبية في الرأس. تم تصميم النظام بواسطة برنامج بايثون لاكتشاف الوجه. لضمان التقدير الدقيق، حيث تم اختيار معالم مناسبة، بحيث لا تتأثر بحركة الرأس. لذلك فإن الزوايا المحسوبة دقيقة تقريبًا. أظهرت النتائج التجريبية قياسات عالية الدقة للزوايا الثلاث بأكملها وفقًا للمقاييس الحقيقية والمحسوبة. كانت نتائج الانحراف المعياري للعينة لكل زاوية حقيقية ومحسوبة هي Yaw (0.0046) ، و Pitch (0.0077) و Roll (0.0021) والتي تؤكد دقة الطريقة المقترحة مقارنة بالدراسات الأخرى. علاوة على ذلك ، تعمل الطريقة بشكل سريع في الوقت الحقيقي مما يعزز تتبع الدقيق عبر الإنترنت.

الكلمات المفتاحية: اتجاه الرأس، خوارزمية Dlib، دوران الرأس، الإمالة الامامية، الإمالة الجانبية.

1. Introduction

The estimation of head directions and tracking facial expressions had great importance in the past 25 years. Research has applied the concept in different fields such as self-driving technologies, machine-human interaction with users, 3D face reconstruction, and other topics that manipulate multimedia content. Historically, several major approaches had been proposed for face modeling such as discriminative/landmark-based approaches. These approaches were able to predict face landmarks on the human face [1].

Many state-of-the-art methods for face detection are based on landmarks-based approach. Such methods, such as Viola and Jones [2][3] are intended to detect faces in a given image or video frames. Viola-Jones method can localize the human face based on trained human face images by implementing Haar-Like features that are classified by Adaboost classifier. In fact, Viola-jones method has very reasonable results as an online face detector [3].

As commonly known, the Viola-Jones method and its enhancements are based on training images of the face pose to estimate whether it is up or down, or left or right. Although some of these methods used more training images for exact angles, but they are still inaccurate in detecting head directions [4]. Therefore, more improved methods had been proposed to make use predicated the landmarks of the human faces. A landmark is a point that is represented by a pixel on the image and refers to a certain position of the human face. In such way, Dlib face detection method had been proposed to predict 86 landmarks that localize the face in a given 2D image [5]. These landmarks can be used to estimate the head direction in degrees if they are implanted in a proper method. Some authors proposed a gaze estimation based on combination of head and eye directions by using Euler's Angles. This method depends on training images that simulate these angles to estimate the gaze angle. It rather calculates approximately accurate results [6].

Therefore, this paper proposes a new method for head direction estimation based on predicted landmarks being obtained from Dlib face detection. In the proposed method, no training images are used to estimate the head direction. But it can calculate the accurate direction of the human head by implementing a mathematical model of Trigonometric functions. Accordingly, would reduce computation time and provide accurate head direction tracking, especially in real time video processing. The paper is organized as: Section 2 illustrates the related works in the field. Section 3 introduces the estimation of head direction. Section 4 demonstrates the proposed method. Section 5 illustrates the experimental results. Finally, the conclusions are illustrated in section 6.

2. Related Works

During the last decade, many methods for head direction have been proposed relevant research. In [7], the authors proposed a method to estimate the head direction based on Dlib landmarks and ellipsoidal method. Several complex equations were used to calculate Yaw angle and then compare the

results with a dataset of images in which each Yaw angle defers in 5 degrees for each image. According to the selected landmarks, the authors calculated the Yaw angle according to the apparent sides of the face. In fact, they did not make the correct selection because using nose landmark and the last apparent area of the face is uncommon in all people. Despite that, they had no solution to estimate other angles. In [8], the authors proposed a 2D to 3D model transformation, in which the 2D image is converted to 3D model depending on a set of training images. The method compares the 2D given image with 2D image of the 3D model in the training images set. The 3D information of the matched 2D image would be considered to be used in head direction estimation based on a complex set of equations and matrixes. It also used Tan function which had no value in 0 degrees which is considered a big problem. This method assumes that the similarity between 2D images led to the same depth of the face features which is not a proper solution. In [6], the authors had proposed a Euler's angles-based method to estimate gaze by tracking the direction of the head based on the extracted face landmarks with Dilb method. The angles had been determined by comparing the captured frame with a set of training images. Each training image refers to a specific angle that would be used in comparison by implementing a neural network which is expensive in terms of time and processing resources. In [9], the authors proposed a neural network with three separated branches to estimate the three angles of the head pose. Average top-k regression was used to predict these angles via comparing the given image with the set of training images.

3. The Head Direction

Normally, the human head is located on the first vertebra of the spine known as the atlas vertebra, which is located above the axis vertebra where the head rotates as in Figure (1). Such movement arises from the contraction of the front and back muscles responsible for the movements of the head, These movements can be defined as the change in the angles of the direction of the head [10]. These angles are:

3.1 Yaw Angle

The head rotates around its vertical axis and turns left and right, which creates a rotation angle between (-75, 75) degrees. This type of movement allows rotation from zero to 150 degrees from right to left [11]. As shown in Figure.1.

3.2 Pitch Angle

The head tilts forward and flattens back, and forms a forward tilt angle of (-60, 60) degrees, that is, it allows the forward tilt with a movement ranging from zero to 120 degrees from bottom to top[11]. As shown in Figure.1.

3.3 Roll Angle

The head tilts to the right and left sides, and forms a lateral tilt angle between (-40, 40) degrees, that is, it allows a lateral tilt ranging from 0 to 80 degrees from right to left [11]. As shown in Figure.1.

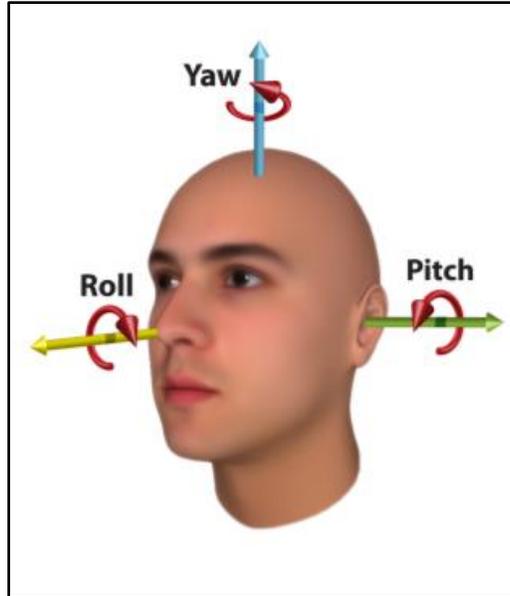


Figure 1. The main axes of head movement.

4. The Proposed Method

Mainly, this paper proposes three models to estimate the angles of the head direction. In which, Dlib predicts landmarks that have been utilized. Some of the face landmarks are useful to calculate Euclidian distance between two points or between a point and a line cross to other points. These distances would be used to calculate trigonometric functions that estimate the proper angles. In order to perform that, three contributions are proposed:

In order to estimate Yaw angle, the paper proposes a model based on two couples of points (17 with 21) and (22 with 26) as shown in Figure 2. the 2D distances are calculated between each couple points as shown in eq(1):

$$Euclidean\ Distance = \sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2} \dots (1)$$

Then, Yaw angle is calculated as in eq(2)

$$\theta = \begin{cases} 0 & \text{if } A = B \\ \sin^{-1}\left(1 - \frac{A}{B}\right) & \text{If } A < B \\ \sin^{-1}\left(1 - \frac{B}{A}\right) & \text{If } A > B \end{cases} \dots (2)$$

Where:

θ : is the Yaw angle

A: is the Euclidean distance between points (17 and 21)

B: is the Euclidean distance between points (22 and 26)

As shown in Figure 2 and eq.2, the distances are computed as in eq.1 to calculate the sine inverse to estimate the Yaw angle. The Figure 3 illustrates the relationships among these states.

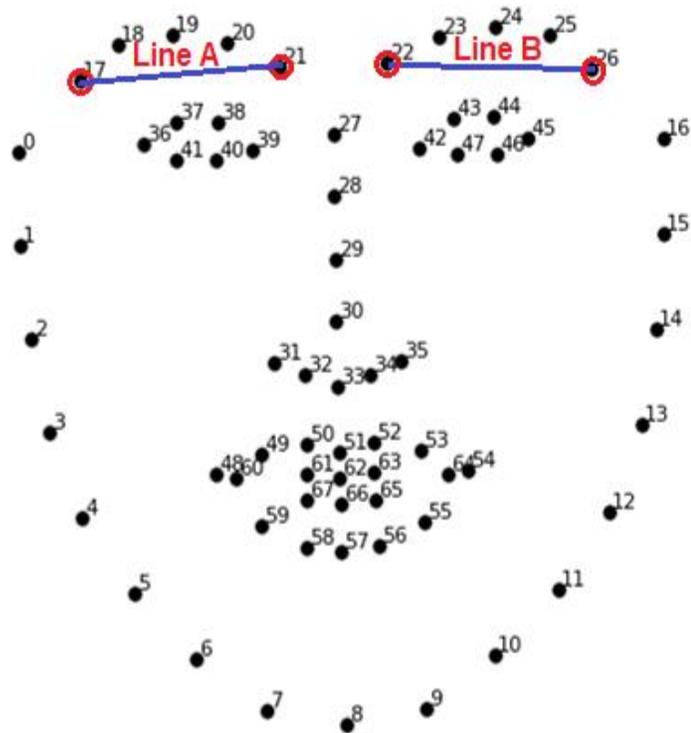


Figure2. Selected points for Yaw angle Calculation.

- (a) $A = B \Rightarrow \text{Yaw} = 0$ the head direction is straight.
- (b) $A < B \Rightarrow \text{Yaw} = A / B$ and the head direction to right.
- (c) $A > B \Rightarrow \text{Yaw} = B / A$ and the head direction to left.

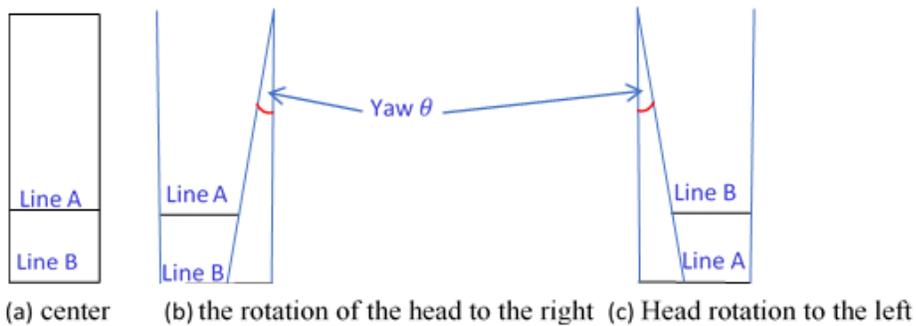


Figure3. The cases of distances between (17,21) and (22,26) landmarks.

The proposed model of Yaw angle estimation is applied to describe three types of Yaw angle rotation, center, right and left head rotation as in Figure 4.



Figure 4. Yaw angles estimation.

As related to pitch angle, another model is proposed. Three points (1,15 and 29) are specific to those forming triangle shapes are selected as in Figure 6. After that, the Euclidean distance is calculated between the line connected points (1,15) and point (29), as shown in python formula that calculates distance as in eq(3):

$$\text{Vertical Distance (VD)} = \text{np.cross}(p2 - p1, p3 - p1) / \text{np.linalg.norm}(p2 - p1) \dots(3)$$

Where:

Vertical Distance: is the shortest distance between p3 and line p1p2.

P1: is landmark (1).

P2: is landmark (15).

P3: is landmark (29).

Then, the Euclidean distances are calculated between the (1 and 29) landmarks and (15, 29) landmarks are calculated as in eq(1), θ_1 and θ_2 are calculated by sine inverse function as in eq(4) and eq(5):

$$\theta_1 = \text{Sin}^{-1}\left(\frac{VD}{A}\right) \dots(4)$$

$$\theta_2 = \text{Sin}^{-1}\left(\frac{VD}{B}\right) \dots(5)$$

Finally, the average of both angles (θ_1 and θ_2) represents the Pitch angle as shown in eq(6):

$$\text{Pitch Angle} = (\theta_1 + \theta_2)/2 \dots(6)$$

The selection of the points had been accurately performed, they are almost on the same line that passes through the center of the head and equals horizontal head axis. The selected points of the face landmarks are not affected by other head movements. The average of both (θ_1 and θ_2) manipulate pitch angle that may reduce one and decrease another of θ_1 and θ_2 . The sine inverse is used to estimate small angles because its value is incremented speedy in small angles that make deference in the angle's measure.



Figure 5. Pitch angles estimation.

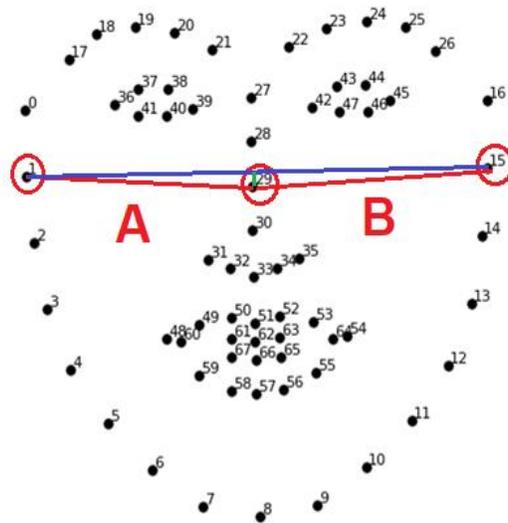


Figure 6. the selected landmarks for Pitch angle.

A third model is also proposed to estimate the Roll angle of head. For this purpose, points (1 and 15) of the face landmarks are selected as they represent the end points of the line connecting the two points located on the edge of the face as shown in Figure 8.

Finally, two distances are calculated based on Euclidean algorithm. The first is real distance (rd) and it is calculated between points (1 and 15). The second, is virtual distance. (vd) and it is calculated between points 1 and a virtual point having the x-axis of points 15 and Y axis or points 1. Usually both distances are calculated according to eq(1). A sine inverse function would be used to calculate the Roll angle as shown in eq(7):

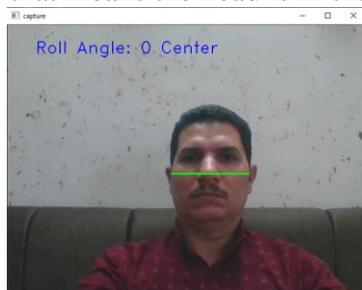
$$Roll = \begin{cases} 0 & \text{if } rd = vd \\ \sin^{-1}(rd/vd) & \text{if } rd \neq vd \end{cases} \dots (7)$$

Where:

rd: is distance between the two real points.

vd: is distance between one real and other virtual points.

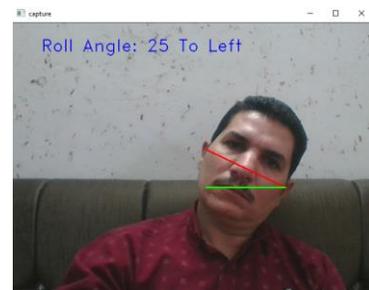
If the Roll has a positive value that means the head rolls to right. Otherwise, the head rolls left. But if it is 0 that means the head is in the center of the roll range.



(a) Center Roll angle



(b) Estimate Roll angle to right



(c) Estimate Roll angle to left

Figure 7. Roll angles estimate.

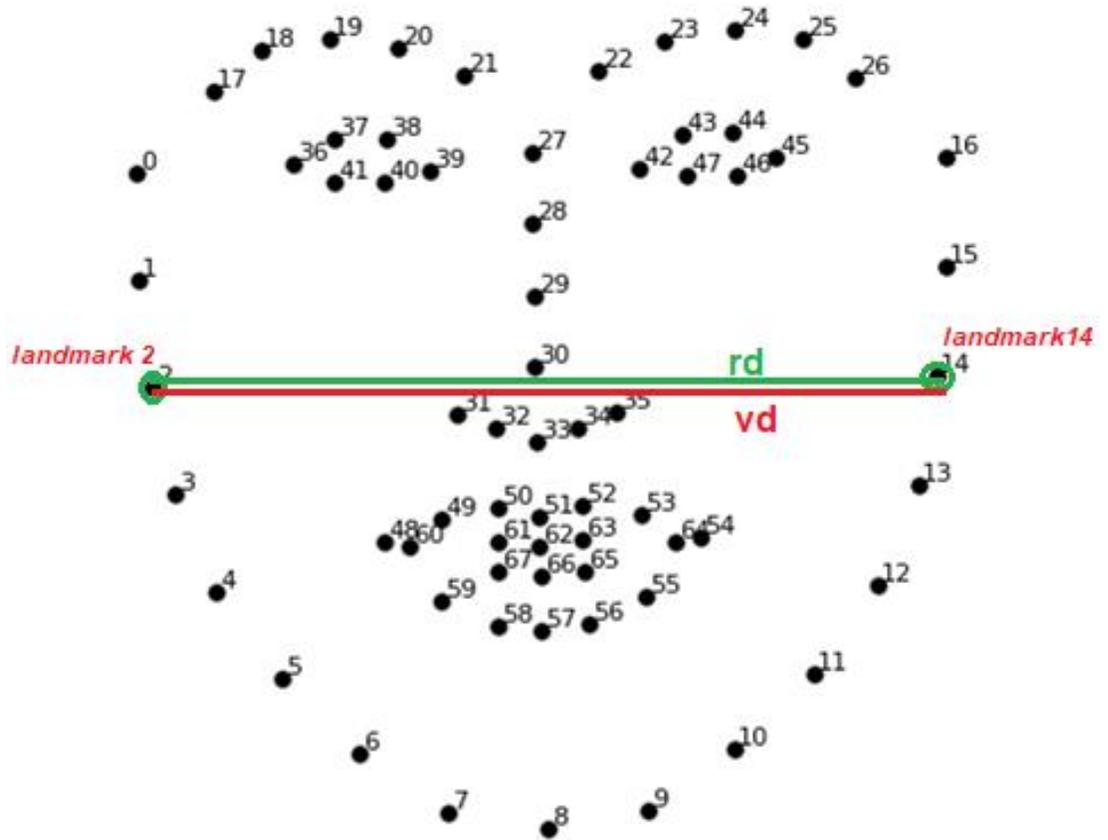


Figure 8. The selected landmarks for Roll angle.

The above contributions have been implemented in a script written in Python programming language. The code has been executed in a Lenovo core-i5 intel CPU with 4GB RAM and common web camera. The Dlib, Math, Numpy and time open-source libraries had been installed. The code was tested on a sample of 25 persons because of the known training dataset giving only 5 degrees difference between images.

5. Experimental Results

The proposed method was implemented on the samples by moving their heads to different angles and recording the real angles simultaneously with those measurements recorded by the calculated application obtained from the proposed method. The execution time of each of the videos and the Frame Per Second (FPS) were also recorded. The person is 70 cm away from the camera. The results of the sample (1) are listed in Table 1.

Table 1: The results of sample (1).

No.	Real Angles			Calculated Angles			Time Sec	FPS
	Yaw	Pitch	Roll	Yaw	Pitch	Roll		
1.	0	0	0	0	0.1	0	0.07	14
2.	2.5	2.5	2.5	2.6	2.45	2.6	0.07	14
3.	5	5	5	4.8	5.45	5.1	0.07	14
4.	7.5	7.5	7.5	7.4	7.4	7.62	0.07	14
5.	10	10	10	9.9	10.1	10.11	0.07	14
6.	12.5	12.5	12.5	12.4	12.6	12.3	0.07	14
7.	15	15	15	15.1	15.15	14.82	0.07	14
8.	17.5	17.5	17.5	17.4	17.46	17.45	0.07	14
9.	20	20	20	20.1	19.9	20.33	0.07	14
10.	22.5	22.5	22.5	22.4	22.45	22.4	0.07	14
11.	25	25	25	25.1	25.1	25.1	0.07	14
12.	27.5	27.5	27.5	27.4	27.55	27.6	0.07	14
13.	30	30	30	29.9	29.8	30.13	0.07	14
14.	32.5	32.5	32.5	32.4	-	32.4	0.07	14
15.	35	35	35	34.9	-	35.1	0.07	14
16.	37.5	37.5	37.5	37.4	-	36.95	0.07	14
17.	40	40	40	40.1	-	-	0.07	14
18.	42.5	42.5	42.5	42.6	-	-	0.07	14
19.	45	45	45	45.4	-	-	0.07	14

During the experiments every person was required to perform a set of head movement each with predetermined angle value (Real angles). Then the values of (Calculated Angles) were recorded and compared with the real angles.

Recall has been performed to calculate false predications as shown in eq. 8. After the results being analyzed the average of error between the real and calculated measures was (0.11) for Yaw angle, (0.12) for Pitch angle and (0.1) Roll angle. These results showed no more than 0.14 degrees of error in all calculated angles, which proves the accuracy of the proposed method. As related to the computation time, the average processing time for both face detection and angle measuring was 0.07 second which indicates high efficiency and allows to track head Direction in 14 frames per second. This is proven by calculated FPS. Figures 4, 5. and 7. showed the experiment of the three angles with sample (1).

$$\text{Recall} = \frac{TP}{TP+FN} \quad (8) [12]$$

The comparison results of the whole sample set are listed in Table.2.

Table 2: The whole sample results

Sample No.	Yaw Error	Pitch Error	Roll Error	Time	FPS
1.	0.112	0.121	0.101	0.07	14
2.	0.113	0.125	0.1	0.07	14
3.	0.110	0.120	0.102	0.07	14
4.	0.111	0.118	0.104	0.07	14
5.	0.114	0.115	0.105	0.07	14
6.	0.12	0.119	0.098	0.07	14
7.	0.115	0.122	0.099	0.07	14
8.	0.107	0.123	0.097	0.07	14
9.	0.109	0.119	0.102	0.07	14

10.	0.114	0.118	0.101	0.07	14
11.	0.103	0.122	0.1	0.07	14
12.	0.116	0.124	0.099	0.07	14
13.	0.120	0.121	0.098	0.07	14
14.	0.118	0.122	0.101	0.07	14
15.	0.117	0.120	0.102	0.07	14

The sample standard deviation has been calculated for the entire results of the sample for each angle by using the form in eq. 9. The Yaw has (0.0046), Pitch has (0.0077) and Roll (0.0021). These values showed no significant difference between the real and calculated measures of the given angles.

$$S = \sqrt{\frac{\sum(X_i - \bar{X})^2}{(N-1)}} \quad (9) [13]$$

Where:

X : The value of data distribution

\bar{X} : **The mean of** sample

N: Total number of the observations

Based on the above results analysis and calculated standard deviation values, the proposed method has performed high accuracy. Moreover, it performs faster than existing methods with 0.07 second processing time for 14 frames per second. As a major contribution of the proposed method, is the selection of right face landmarks to enhance the accuracy of angles estimation.

6. Conclusions

This paper proposed a new method for the estimation of the head direction implementing Dlib method and Trigonometric functions. The proposed method does not require to be trained more than one time to detect face. The estimation was based on predicted landmarks and mathematical calculation. The impressive point in this method was that the used landmarks were very successfully selected to represent the accurate angle estimation in comparison with previous works.

The Standard Deviation for the calculated angles with low error rate showed high performance that proves the efficiency of the method. Therefore, the proposed method may be implemented with more advanced face detection methods or used in other applications that require head direction estimation.

7. Acknowledgements

The authors would like to thank the University of Mosul / College of Computer Science and Mathematics for their cooperation in completing this work, and thank the sample members who participated in the research, and who helped us improve the quality of this work.

8. References

- [1] N. Ruiz, E. Chong, and J. Rehg, Fine-grained head pose estimation without keypoints. In Proceedings of the IEEE conference on computer vision and pattern recognition workshops (pp. 2074-2083), 2018.
- [2] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I-511. IEEE, 2001.

- [3] P. Viola and M.Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004.
- [4] D. Gayathri, K. Akhil, and V.Sairam, Haarcascade Classifier Based Examination System. *International Research Journal of Modernization in Engineering Technology and Science*, Vol:03, Issue:06, June-2021.
- [5] N. Boyko, O. Basystiuk and N. Shakhovska, Performance evaluation and comparison of software for face recognition, based on dlib and opencv library. In *2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP)* (pp. 478-482). IEEE, 2018.
- [6] E. Wong, S. Yean, Q. Hu, B. Lee, J. Liu, and R. Deepu, Gaze Estimation Using Residual Neural Network. *2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, 2019.
- [7] A. Narayanan, R. Kaimal, and K. Bijlani, Yaw estimation using cylindrical and ellipsoidal face models. *IEEE Transactions on intelligent transportation systems*, 15(5), 2308-2320, 2014.
- [8] S. Kong, and R. Mbouna, Head Pose Estimation From a 2D Face Image Using 3D Face Morphing With Depth Parameters. *IEEE Transactions on Image Processing*, 24(6), 1801–1808, 2015.
- [9] B. Huang, R. Chen, W. Xu, W, and Q. Zhou, Improving head pose estimation using two-stage ensembles with top-k regression. *Image and Vision Computing*, 93, 103827, 2020.
- [10] J. Forbes, and J. Das, *Anatomy, Head and Neck, Atlantoaxial Joint*, 2020.
- [11] A. Saeed, A. Al-Hamadi, and A. Ghoneim, Head pose estimation on top of haar-like face detection: A study using the kinect sensor. *Sensors*, 15(9), 20945-20966, 2015.
- [12] J. Miao & W. Zhu. Precision–recall curve (PRC) classification trees. *Evolutionary Intelligence*, 1-25, 2021.
- [13] J. Nicholls, *Standard Deviation: The Neglected Statistic*. 2020.