

EVEN-VE: Eyes Visibility Based Egocentric Navigation for Virtual Environments

Muhammad Raees*, Sehat Ullah

Department of Computer Science and IT, University of Malakand (Pakistan)

Received 25 May 2018 | Accepted 1 August 2018 | Published 10 August 2018



ABSTRACT

Navigation is one of the 3D interactions often needed to interact with a synthetic world. The latest advancements in image processing have made possible gesture based interaction with a virtual world. However, the speed with which a 3D virtual world responds to a user's gesture is far greater than posing of the gesture itself. To incorporate faster and natural postures in the realm of Virtual Environment (VE), this paper presents a novel eyes-based interaction technique for navigation and panning. Dynamic wavering and positioning of eyes are deemed as interaction instructions by the system. The opening of eyes preceded by closing for a distinct time-threshold, activates forward or backward navigation. Supporting 2-Degree of Freedom head's gestures (*Rolling* and *Pitching*) panning is performed over the xy-plane. The proposed technique was implemented in a case-study project; EWI (*Eyes Wavering based Interaction*). With EWI, real time detection and tracking of eyes are performed by the libraries of OpenCV at the backend. To interactively follow trajectory of both the eyes, dynamic mapping is performed in OpenGL. The technique was evaluated in two separate sessions by a total of 28 users to assess accuracy, speed and suitability of the system in Virtual Reality (VR). Using an ordinary camera, an average accuracy of 91% was achieved. However, assessment made by using a high quality camera testified that accuracy of the system could be raised to a higher level besides increase in navigation speed. Results of the unbiased statistical evaluations suggest/demonstrate applicability of the system in the emerging domains of virtual and augmented realities.

KEYWORDS

Gesture Recognition, Interactive Systems, Computer Interfaces, Virtual Reality.

DOI: 10.9781/ijimai.2018.08.002

I. INTRODUCTION

NAVIGATION is the user's locomotion inside a virtual environment to reach to and inspect an object of interest from close proximity. The two broad metaphors of navigation are Egocentric and Exocentric. Where the former is analogous to moving the world in reference to user's input, the latter is moving the user's viewpoint inside the virtual world. Navigation often becomes a preliminary step to perform other interactions like selection, scaling and rotation. Furthermore, a wide spread 3D scene cannot be viewed from a single static look of virtual camera. Navigation is, therefore, important to explore different parts and portion of a synthetic world. Though, a number of navigation techniques have been proposed based on mouse/keyboard, their traditional menu-commands based interface fails to bring intuition and naturalism in VE. Low-cost vision based tracking on the other hand provides an applicable platform to devise a flexible interface for Human Computer Interactions (HCI). Enormous Gestures based techniques in the literature of 3D interactions provides estimable degree of realism. The latency and fatigue involved in posing gestures, however, are the challenges yet to be covered. Particularly, the time lapse in posing of gesture, adds in imbalance bandwidth between human-computer dialogues which in turns badly affects immersion.

As eyes-based interaction is faster than speech and gestures based interactions [1] therefore, interaction via eyes postures can reduce

the speed mismatch. This paper is an attempt to present a convenient and cost-effective eyes' wavering based interaction. The technique; Eyes Visibility based Egocentric Navigation in Virtual Environment (*EVEN-VE*), follows gestures of eyes which are speedier than any other gesture made by any other part of the body. Though worthy research has been carried out on iris recognition but iris varies from person to person, therefore it cannot be used for interaction [2]. The proposed system merely considers wavering of eyes hence processing time is not wasted in tracing user's specific features. Only flickering of eyes and tilting of head are traced by the system to interact with the designed 3D virtual world. *Navigation* is activated by a slight flickering-gesture of both the eyes while *Panning* is performed by tilting of head along the look-vector. To make user aware about the beginning of any action, all interactions are initiated when both the eyes are opened. At the closing state of eyes, threshold times for forward and/or backward navigation are checked. Keeping both eyes closed for extended time greater than that of normal flickering, activates forward navigation. Upholding the closing state of eyes for another equal interval leads to activation of reverse navigation. Panning is performed based on eyes position in dynamic scanned frames. As eyes' positions change with tilting of head, the 2DoF head's movement; *Rolling* and *Pitching*, are traced to pan along z-axis or x-axis respectively. The case-study project EWI was designed using the open libraries to implement and evaluate the system. Based on algorithm of the technique, EWI supports multiple speed sectors for navigation. The system was twice evaluated in two separate sessions. The first session was to evaluate accuracy and learning effect of the system. In second evaluation session, the system performance was assessed in different navigation sectors using two

* Corresponding author.

E-mail address: visitrais@yahoo.com

different cameras. Results of the second evaluation session proved that with a high quality camera both speed and accuracy of the system can be improved. The proposed technique is applicable for all the basic navigation tasks; exploration, searching and inspection [3]. Moreover, due to the least muscular strive and strain involved in wavering of eyes and tilting of head, less fatigue is guaranteed.

The paper is organized into 7 sections. Related work for Navigational interaction is discussed in Section II. Section III elaborates the algorithm of the technique in detail, section IV is about implementation and evaluation details. In Section V, the system is compared with state-of-the-art navigation techniques. Section VI discusses applicability of the technique in VR. Conclusion and future work is discussed in the last section.

II. LITERATURE REVIEW

No doubt, the ceaseless progress in processing and storage of computer systems has made possible the emergence of virtual reality applications in every field. To make possible convenient dialogue with such applications, realistic way of interactions is required. Being the most commonly used interaction task, navigation is focused by most of today's research works to make exploration feasible and flexible. Most of the navigation techniques in the literature of virtual reality are either based on mouse/keyboard or need Head-Mounted Display (HMD) [4]. Where the former techniques lack immersion, the techniques based on HMD always remain a second choice because of the high cost. The bimanual Multi-Finger gestural navigation of [5] works on a constrained tabletop surface and is not suited for hand-held devices. Another hand gesture based navigation; NuNav3D [6] preliminarily needs whole body pose estimation. This makes the system not only difficult to use but also 79% slower than Joypad based navigation [6]. The technique of lee et al. [7] consumes a large amount of processing for finger action recognition as the system ought to pass from three hefty stages; skin-color detection, k-cosine based angle detection and contour's analysis each for finger's state, position and direction. Furthermore, due to variable finger's thickness, accuracy of the system varies from user to user.

With the Drag'n Go [8] the position of screen's cursor casts ray to the target for navigation. The technique is applicable only for navigation in large empty space while askew navigation is not supported. In another related approach [9] dragging mouse in a specific direction shifts the look-at vector of virtual camera. Thus for a lengthy travel the practice is needed to be repeated again and again. Moreover, in the way if the ray collides with an object, navigation is halted and the collided object is mistakenly inspected.

In the head-directed navigation [10], speed and direction are calculated from the pose and direction of the user's head. Though intuition of the system is commendable but unintentional and casual head movements are fallaciously treated as navigation commands. The FmF (Follow my Finger) [11] presents MC (Management Cabin) model for navigation. The system projects 3D view of a virtual world on a 2D table-top device but suffers from disorientation problem.

Trackers based gestures recognition [12] and [13] are the noteworthy navigation techniques which work on natural gestures of hands. Both of the systems treat virtual scene as big object that can be grabbed and moved with hands. The techniques are interesting but remain a rare option due to the hand-worn overload.

The walking in place of [14] correlates pace of the natural walking with navigation speed in synthetic environment. The approach cannot be used while setting on chair. Chest expansion and Tilting of spine are treated as commands for navigation in [15]. The system needs a complex setup of sensors implementable only inside a controlled environment. Similarly, the foot-based interface for navigation suggested in [16]

requires a cumbersome setup of waist-based magnetic trackers with conveyer belt for navigation.

Fiducial markers have also been effectively used for navigation. In the DeskCube system [17] different markers are glued on different faces of a cube. The system works fine when the cube is still but fails to cope with occlusion and blurring while moving the cube. Similarly navigation system based on QR codes necessitates the placement of navigation device closed to the marker [18].

Using the leap motion based system [19] for navigation, user needs to activate commands via buttons within a limited space. The fingers based locomotion; FWIP (Finger Walk In Place) [20] necessitates touch of the fingers on display screen for navigation. The system is applicable only for mobile based VEs. The pointing technique of [21] uses two fingers; one for viewing and the other for direction. Though it successfully avoids the mistakes of gaze-directed navigation but speed controlling is its main challenge.

As eyes tracking based interfaces ensure quick and effortless interactions [22], therefore eyes based interaction is becoming prominent in the domain of VR. The system of [23] traces gaze direction after pushing a button to reach to an object. The system needs HTC Vive with a Tobii Eye-tracker and is therefore applicable inside a lab environment. With EyeScout system [24] eye trackers are mounted over a rail system. The trackers are made aligned with user's lateral movement by a computational method. Based on 9-point polynomial algorithm, the HoloLens based system of [25] maps eyes position to a plane in which calibration points are displayed. The system needs a constrained setup containing wireless camera, eye tracker and IR (infrared) filter. The lazyNav system [26] supports other interactions besides navigation using HMD and trackers. Extensive research in the literature has been made to enhance efficiency of eyes tracking. The algorithm proposed by [27] for measuring focal distance in immersive CAVE (Cave Automated Virtual Environment) is to improve precision of eyes tracking. Moreover, as eyes-tracking based interaction is supposed to be the VR's next frontier [28], models for accurate detection of eyes using next generation devices have also been suggested [29].

III. EVEN-VE: THE PROPOSED SYSTEM

Interactions via gestures of eyes are becoming an effective alternative due to the availability of portable head-mounted eye trackers. Nevertheless, the clumsy setup of wires, as shown in Fig. 1 restricts the use of such systems only inside the premises of a lab.

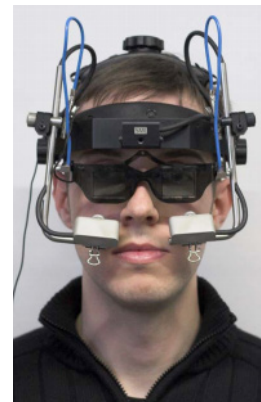


Fig. 1. The complex setup of eyes trackers [20].

Aim behind this research was to introduce a cheaper and easy-to-use interaction technique. The proposed system tracks eyes' positions and orientations through an ordinary camera without the use of any trackers

or bunch graphing [30]. A slightly longer than the normal flickering of eyes is assumed to start forward-navigation. If navigation is already activated then the same gesture is used to halt it. Backward navigation is activated by keep closing both the eyes for a longer time than the forward navigation. After detecting both the eyes, Mid-Point (MP) is calculated to get central point between the eyes [31]. MP is calculated from the horizontal and vertical positions of the eyes as,

$$MP.x = (LeftEye.x + RightEye.x)/2 \quad (1)$$

$$MP.y = (LeftEye.y + RightEye.y)/2 \quad (2)$$

Where X_{Eye} are point variables representing mid position of the eyes (eyeball with sclera). X is representing both *Left* (eye) and *Right* (eye).

A straight virtual ray starting from MP is supposed to represent user's direction. The ray's hit position is represented by a pointer in the designed 3D world, see Fig. 2. Before starting navigation, the pointer can be used to set position of the virtual camera. By default, navigation is performed straight inside the look vector.

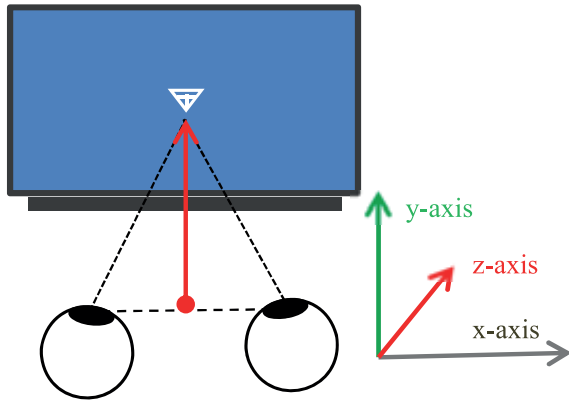


Fig. 2. Eyes with MP and virtual ray casted on screen.

Using 2D positions of the eyes in the initial five frames, Eyes Initial Average Area (*EIAA*) and Normal Speed Zone (*NSZ*) are defined. *NSZ* is a limited area where navigation with normal speed is performed whereas *EIAA* is calculated as,

$$EIAA = (LeftEye_AA + RightEye_AA) / 2 \quad (3)$$

Where X_AA is calculated from eyes' area in the initial five frames as,

$$X_AA = \frac{\sum_{i=1}^5 X_Area_i}{5} \quad (4)$$

Area of an eye, as shown in Fig. 3, is calculated from the length and width of eye (eyeball including iris and sclera) as,

$$X_Area = X.length \times X.width \quad (5)$$

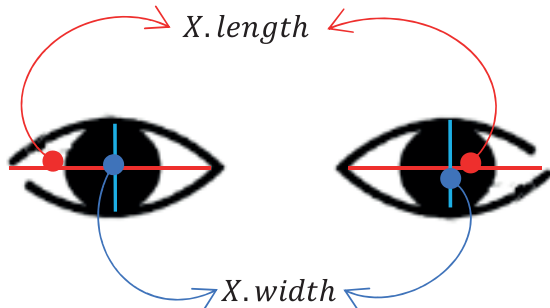


Fig. 3. Length and width of the eyes.

Navigation speed can be increased or decreased based on areas of the eyes. Moving head forward along *z*-axis results into increase in areas of eyes which in turn makes navigation speedier and vice versa.

Panning is performed by gentle tilts of head towards left or right along *z*-axis (*Rolling*). As conceivable, a slight bending towards left activates left panning whereas right panning is accomplished by bending head towards right. *Up panning* (flying) and *Down panning* (landing) are performed by *Pitching* gesture; raising and bowing head along *y*-axis respectively. With both *Rolling* and *Pitching* gestures of head, positions of eyes change. Calculating the dynamic positions of eyes against MP, horizontal or vertical panning is performed.

A. The System Architecture

The system designed for the implementation starts with a virtual world at front end with different 3D objects. At the backend, a region of interest (ROI) based on skin color is extracted from the scanned image. At the detection of eyes, MP is calculated and coordinates mapping is performed to set virtual camera position in the environment. As long as eyes are detected, the virtual pointer can move freely with the movement of eyes. If the eyes closing time extends the normal blinking duration, navigation is activated. Tilting of head, alters the eyes' vertical positions from the central MP, activates panning. Schematic of the proposed system is shown in Fig. 4.

B. Face Segmentation

In order to reduce processing cost and to avoid possibility of false detection first a ROI image (ROI-Image) is extracted. ROI-Image is supposed to contain both the eyes. This segmentation is carried out on the basis of skin color. As the YCbCr space thresholding provides best skin color segmentation [32], hence the YCbCr model is used for face extraction. Each scanned RGB frame is thus converted to YCbCr space (Frame_YCbCr) as,

$$\text{Frame_YCbCr} \begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65.1 & 128 & 24.8 \\ -37.3 & -74 & 110 \\ 110 & -93.2 & -18.2 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (6)$$

From the binary image of Frame_YCbCr, see Fig. 5, *ROI-Image* is extracted with rows $\langle m \rangle$ and columns $\langle n \rangle$ using our designed algorithm [33] as,

$$\text{ROI_Image}(m, n) = \begin{pmatrix} \bigcup_{r=Dm}^{\text{Frame_YCbCr.Row}(0)} (\text{Frame_YCbCr}) \\ \bigcup_{c=Lm}^{\text{Frame_YCbCr.Column}(Rm)} (\text{Frame_YCbCr}) \end{pmatrix} \quad (7)$$

Where Lm , Rm and Dm represents *Left-most*, *Right-most* and *Down-most* skin pixels.



Fig. 5. (a) RGB and (b) YCbCr models of a scanned image.

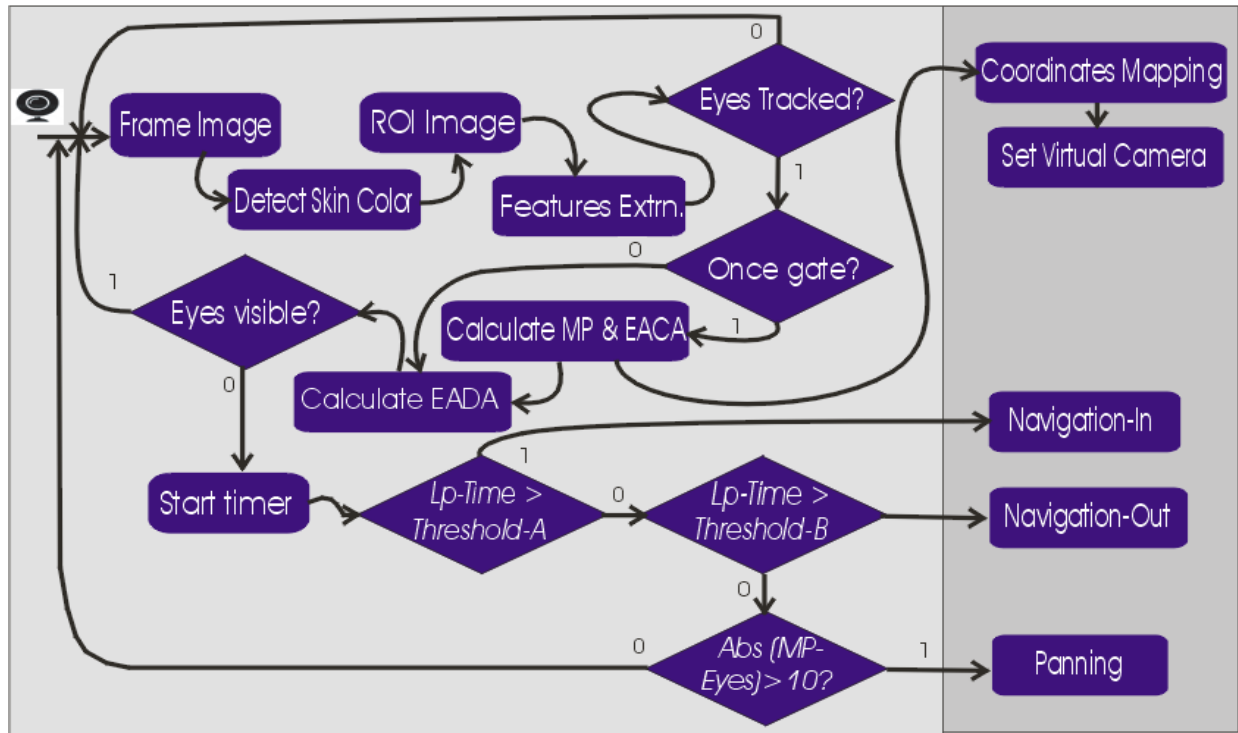


Fig. 4. Schematic of the proposed system.

C. Eyes Detection

The most accurate cascade classifier for eyes [34] is used for the detection and tracking of eyes. The algorithm detects eyes rapidly in live video and is more accurate than LBP [35]. With Haar-cascade, first the system is trained by several numbers of positive images and negative images and then rectangular features are selected by Ada-boost. Each time window of the eyes are moved over the scanned image and for each subsection, the Haar-like features are calculated. Greater the matching of positive features, higher will be the eyes recognition.

D. Coordinates Mapping

The notable challenge in implementation of the technique was to locate and move pointer of the virtual environment through pixels position of MP. The image frame of OpenCV starts from top left (0,0) while in OpenGL, (0,0) lies at the center of clipping area. To harmonize the dissimilar coordinate systems, we devise our four mapping functions m_1 , m_2 , m_3 and m_4 [36], as described in equations (8), (9), (10) and (11). The image frame is virtually split into four regions R_1 to R_4 as shown in Fig.6, where mapping for a region R_n is made by the corresponding m_n taking x and y of a pixel of R_n as independent variables.

$$m_1(x, y) = ((MP.x - (Tc/2))/(Tc/2), (Tr/2 - MP.y)/(Tr/2)) \quad (8)$$

$$m_2(x, y) = ((MP.x/Tc), (Tr/2 - MP.y)/(Tr/2)) \quad (9)$$

$$m_3(x, y) = ((MP.x - (Tc/2))/(Tc/2), (MP.y/Tr)) \quad (10)$$

$$m_4(x, y) = ((MP.x - (Tc/2))/Tc/2, (MP.y/Tr)) \quad (11)$$

In the above functions, $MP.x$ and $MP.y$ represent x and y positions of the traced pixel. Tr and Tc represent total rows and total columns respectively.

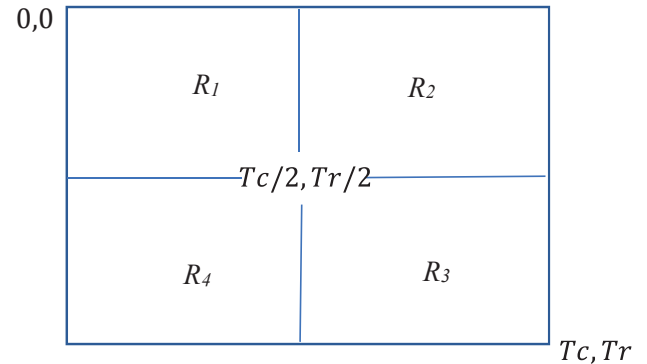


Fig. 6. Virtual division of the image frame.

E. Navigation

Navigation is the moving of a virtual camera in 3D VE towards or away from a look-at point. In the proposed system, navigation is activated/deactivated by a single prolonged blink of eyes. The normal blinking rate is approximately 1/3 of a second in which eyes remain closed for about 300-350 milliseconds. To avoid detection of normal unintentional blinking the closed state of eyes for less than 400 milliseconds are omitted. When navigation state is set enabled, user is informed by a beep and actual movement is started when eyes are opened. Keeping eyes closed after the first beep will activate reverse navigation. Backward navigation is distinguished by a lengthy beep sound. Textual information about each state is also displayed on the top of the virtual environment. By default navigation is performed in a straight line but user can change direction of navigation dynamically through eyes movement. The viewable scene is divided into 180 degrees as shown in Fig. 7. Moving the pointer thirty pixels towards right will result into a decrease of thirty degrees while moving thirty pixels towards left will make an increase of thirty degrees. The 30-pixels to 30-degree ratio can be decreased to divide the scene into further navigation paths. The virtual division of scene based on the position of the pointer (a 3D wire cone) is shown in Fig. 7.

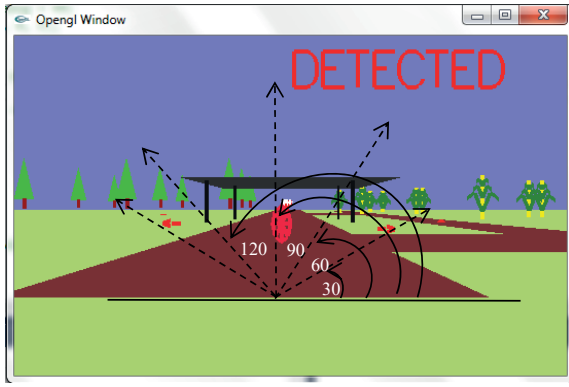


Fig. 7. Virtual division of the scene based on the pointer.

Navigation Speed

One of the main problems with VE navigation is speed control, the absence of which may either lead to disorientation or slow velocity [37]. Based on the quality of camera and distance of eye from it, various finite speed sectors ' S_n ' can be defined. Initially, the system was implemented and tested with the default three sectors, NSZ, one ahead and one behind the NSZ, see Fig. 8 and Fig. 9. Entrance into a particular sector S_n is determined from the Eyes Average Dynamic Area ($EADA$). Dynamic Areas (DA) of the eyes are calculated on the fly as,

$$\text{LeftEye_DA} = \text{LeftEye.Height} * \text{LeftEye.Width} \quad (12)$$

$$\text{RightEye_DA} = \text{RightEye.Height} * \text{RightEye.Width} \quad (13)$$

$$\text{EADA} = (\text{LeftEye_DA} + \text{RightEye_DA}) / 2 \quad (14)$$

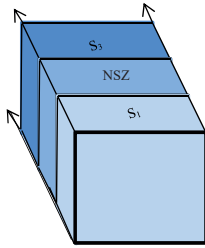


Fig. 8. First three different speed sectors for navigation.

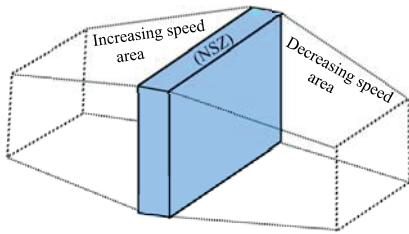


Fig. 9. NSZ and speeding areas over the look-vector.

With forward head's movement ahead of NSZ, $EADA$ increases and hence speed is increased. Similarly, by moving head behind the NSZ, $EADA$ decreases which results into decrease in speed. The '10' pixels extension in area is to avoid erroneous/unintentional increase or decrease. The pseudo-code for detecting variation in $EADA$ and accordingly setting navigation speed is as follow,

```

if (EADA >= EIAA + 10)
    Navigation_Speed += 0.5
    EIAA = EADA
if (EADA <= EIAA - 10)
    Navigation_Speed -= 0.5
    EIAA = EADA

```

When the condition of forward or backward head's movement meets, current area of eyes ($EADA$) becomes previous area ($EIAA$); see second statement of the condition in the pseudo-code. With this, algorithm of the system supports ' n ' sectors beyond and behind the NSZ provided that eyes are detectable in a sector S_i | $i \in \{1, 2, 3, \dots, n\}$. Eyes recognition and the extraction of area of eyes depend on the quality of camera and distance of user from camera, see Fig. 10.

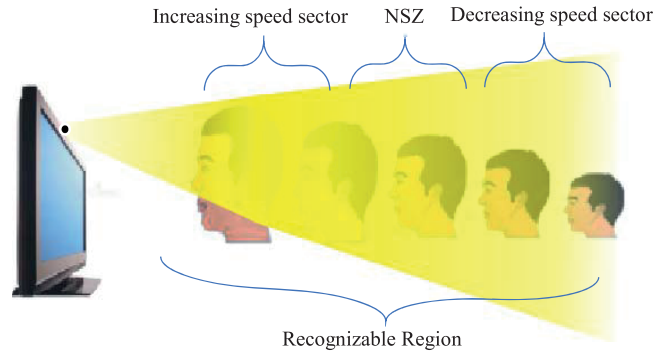


Fig. 10. The detectable region of camera with different speed sectors.

F. Panning

Panning is the translation of the camera eye and the look-at points horizontally or vertically. To avoid the possibility of disorientation, panning is enabled only inside NSZ. Furthermore, panning is performed with a constant moderate speed, independent from navigation's speed. Horizontal Panning is performed by slightly bending the head over the z-axis (Rolling gestures of head). With rolling gesture of head, one of the eye's position on y-axis increases while that of the other decreases. Similarly, with vertical movement of head (pitching), x and y positions of both the eyes are changed. The changes in positions of eyes are traced dynamically to perform up or down panning by changing y-coordinate of virtual camera. All this is measured against the vertical and horizontal position of MP as shown in Fig. 11, Fig. 12 and Fig. 13.

The pseudo-code for left and right panning is given as,

```

if (abs(LeftEye.y - MP.y) > 10 || abs(RightEye.y - MP.y) > 10)
    if (LeftEye.y < MP.y && RightEye.y > MP.y)
        Panning along +ve x_axis
    if (LeftEye.y > MP.y && RightEye.y < MP.y)
        Panning along -ve x_axis

if (abs(LeftEye.y - MP.y) > 20 && abs(RightEye.y - MP.y) > 20)
    if (LeftEye.y > MP.y && RightEye.y > MP.y)
        Panning along +ve y_axis
    if (LeftEye.y < MP.y && RightEye.y < MP.y)
        Panning along -ve y_axis

```

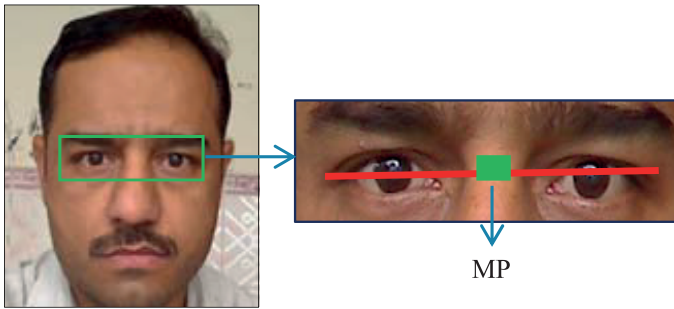


Fig. 11. Eyes tracking and setting of MP between the eyes.

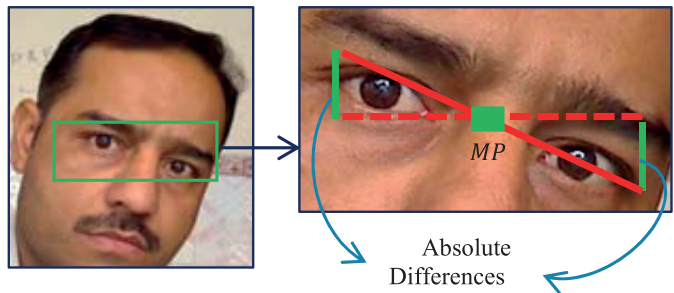


Fig. 12. Tilting of head towards Right (rolling) for horizontal panning.

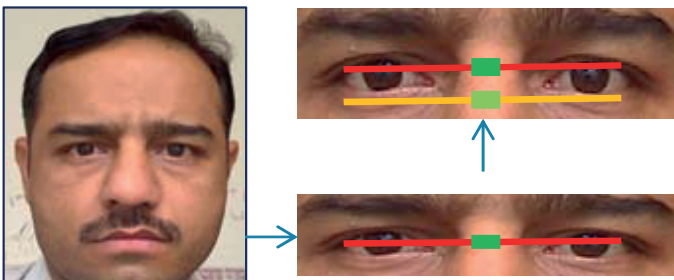


Fig. 13. Head's up gesture (pitching) for vertical panning.

IV. SYSTEM IMPLEMENTATION AND EVALUATION

The proposed system was implemented in a Visual Studio project; EWI (Eyes' Wavering based Interaction). The frontend virtual scene of the EWI was designed in OpenGL. Activation of the system is conditioned to visibility of both the eyes. User is constantly informed about activation of the system by the text "Activated" displayed in upper part of the scene. Similarly, about left and right panning user is informed by the "Turning" text at the respective side of the scene as shown in Fig. 14. The system was tested by sixteen male participants. Two trials were performed by each participant for each of the four pre-defined tasks.

A. Testing Environment

The 3D environment designed for the evaluation contained four routes as shown in Fig. 13 where a robot made of cubes represents the user's position in the environment. Symbolizing finishing point of the scene is marked by a board with text "Stop". Each of the routes leads to the one Stop-board. To immerse users and to have a perception of navigation and panning, the scene contained different 3D objects at different positions.

Route-1: Straight pathway leading to Stop-board.

Route-2: Right-Straight-Left pathway.

Route-3: Left-Straight-Right pathway.

Route-4: Up-Straight-Down pathway to have a flying effect to navigate over the bandstand. Although, this route is not rendered in the

scene, to avoid complication, but user can use it. A 2D model of all the routes is shown in Fig. 15.

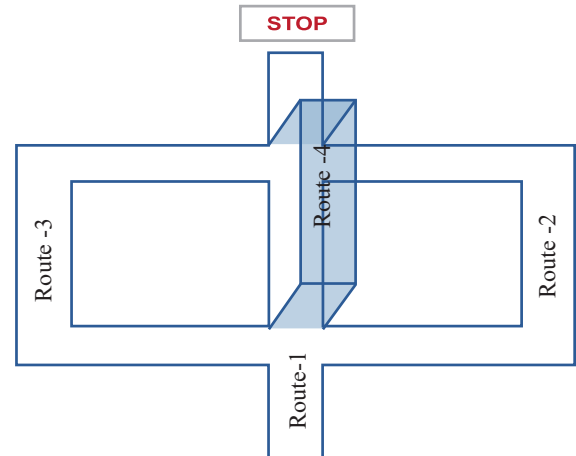


Fig. 15. 2D Model of the routes.

B. Evaluation Procedure

Each one of the participants was briefed about the environment and the interactions. They were guided by a demonstration about how to properly perform navigation and panning. Before actual trials, pre-trials were performed by each participant both for navigation and panning. For better detection of eyes, those wearing spectacles (two of the participants) were requested to remove glasses. Users were guided to hit the 'r' keyboard key to restart a current incomplete task or to perform a new task. At the completion of a trial, Enter key was to be pressed to reset the entire system of EWI for a new participant. False detection and wrong turns were counted as errors. All the trials were performed inside the university IT lab in a moderate lighting condition.

C. Evaluation Tasks

Participants were asked to perform the following four interaction tasks in the designed 3D environment.

Task-1: Touching the Stop-board using Route-1 and then back to starting point following the same route.

Task-2: Touching the Stop-board using Route-2.

Task-3: Touching the Stop-board using Route-3.

Task-4: Touching the Stop-board using Route-4 and then back to starting point using the same route.

D. Results of the First Evaluation Session

Navigation-In is the forward while Navigation-Out is the backward movements inside the VE. Left/Right panning is turning, with shift of virtual camera towards the respective direction accordingly. Task-1 tests Navigation-In and Navigation-Out. Task-2 and Task-3 are to evaluate Navigation-In, Right panning and Left panning. Task-4 assesses Up panning, Down panning and Navigation. Missed detection or false detection of the system after posing the required gestures were counted as errors. With this setup, overall accuracy rate for all the 448 interactions, as shown in Table I, was 91% with average Standard Deviation (SD) 0.5.

TABLE I. STATISTICS OF THE INTERACTION TASKS

Interaction	Correct	Total	%age
Navigation-In	123	128	96
Navigation-Out	121	128	94
Panning Left	61	64	95
Panning Right	60	64	93
Panning Up	28	32	87
Panning Down	27	32	84
Total	420	448	91

Mean of the %age accuracy with standard deviation of the interactions are shown in Fig. 16. Though accuracy varies slightly for the first three interactions ($M_{Navigation-In} = 96$, $M_{Navigation-Out} = 94.5$ and $M_{Left-Panning} = 95.3$), range of SD for all the interactions remained as low as 0.5. The less recognition of the last two interactions ($M_{Up-Panning} = 87.5$ and $M_{Down-Panning} = 84.3$) were due to the quicker move of eyes or of head. This implies that the system performance can be further improved with high efficiency camera.

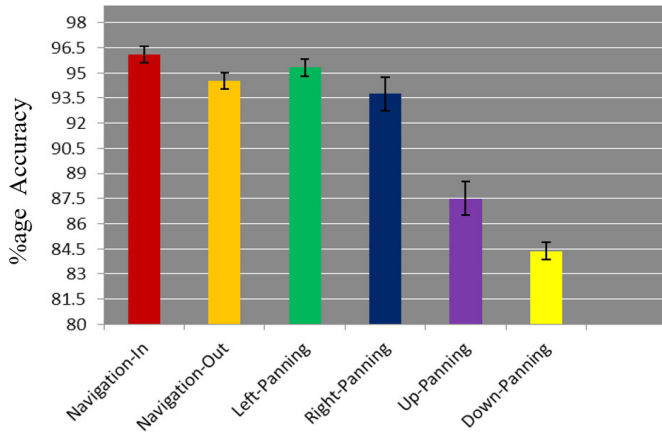


Fig. 16. %age accuracy with SD of the interactions.

E. Learning Effect

The learning effect was measured from the errors occurrence rate. Paired two sample *T*-test was used to analyze differences in means of the two trails. With null hypothesis (H_0) we assumed that mean difference (μ_d) is 0. The hypothesis was rejected as there was a significant difference between the outcomes of Trail-5 ($M=34.5, 21.3$) and Trail-6 ($35.5, 21.5$) conditions; ($t(5)=-3.8, p=0.0117$). The graph indicating this vivid decrease in error is shown in Fig. 17.

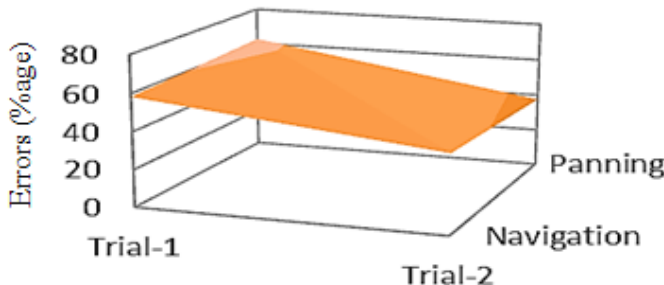


Fig. 17. The % age of errors occurred in the trials.

F. Assessment of Navigation Speed

In a separate evaluation session, performance of the system was assessed for different navigation speed in different sectors. Twelve participants, mean age 29 (SD=4.9, Range=17) familiar with image based HCI, were invited to the session. Two cameras, different in efficiency; HP Webcam and Logitech USB webcam, were used in the evaluation process. Participants were guided to perform the following tasks in a single trial using Route-1.

Task-5: Touch the Stop-board as quickly as possible.

Task-6: Get back to the starting point as slowly as possible.

The evaluation session was arranged in two phases with a short break in the middle. During the first phase, ordinary camera was used for tracking while in second phase the high quality, Logitech external camera. To properly evaluate individual trial, code of the system was updated to display number of the Highest Achieved Sector (*HAS*) and to stop detection when a trial completed. *HAS* is the highest sector for navigation in either direction and is obtained by incrementing the previous sector by 1.

$$HAS \in \{1, 2, 3, \dots, n\}$$

At the end of each trail, *HAS*, errors and task completion time were noted for statistical analysis. Results of the twelve trails of first phase are shown in Fig. 18.

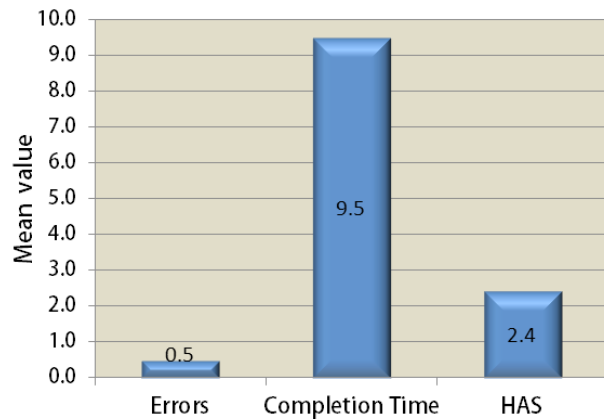


Fig. 18. Average of errors occurred, task completion time and *HAS* of the first phase trials.

By using the high quality camera, navigation in second phase was mostly performed in higher sectors (increase in *HAS*). Hence, the obtained average completion time was reduced 45%, see Fig. 19.

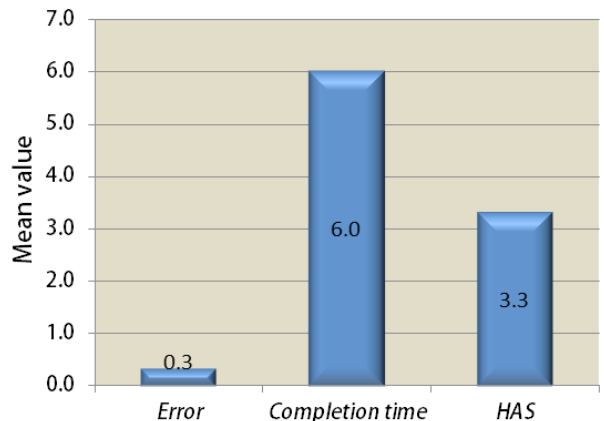


Fig. 19. Average of errors occurred, task completion time and *HAS* of the second phase trials.

As per the outcomes of test statistics, there exist a significant difference between mean completion times of tasks using ordinary camera (M=9.5,SD=S0.788) and quality camera (M=6.0, SD=S0.92), $t(11)=11.1$, $p=0.0001$. Similarly, a significant difference was reported between HAS achieved using ordinary Camera (M=2.14, SD=S0.45) and quality camera (M=3.3, SD=S0.6), $t(11)=-4.7$, $p=0.0006$.

Pearson’s correlation (r) was used to measure the relationship between *Errors* and *HAS*. A moderate correlation was demonstrated while using ordinary camera ($r=0.39$), see Fig. 20. A downhill (negative) correlation was obtained by using the quality camera ($r = -0.5$), see Fig. 21. The diversity of ‘ r ’ values therefore, testify the fact that error rate could be reduced with the use of a high quality camera. To avoid disorientation, turnings are made with a constant moderate speed independent of navigation speed.

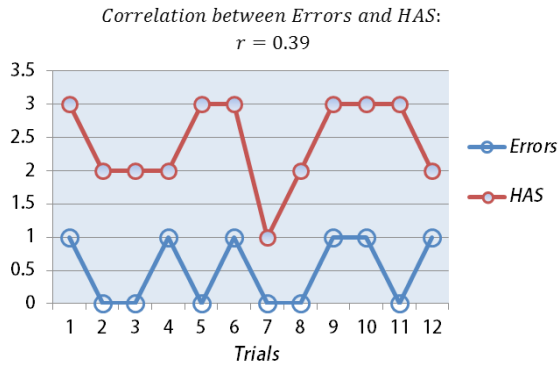


Fig. 20. The HAS and errors of the first phase trials.

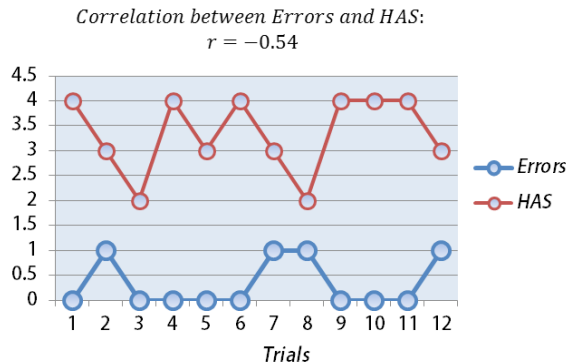


Fig. 21. The HAS and errors of the second phase trials.

G. Subjective Analysis

A questionnaire was presented to the user at the end of evaluation session to measure the four factors; Fatigue, Naturalism, Suitability in VEs and Ease of Use. The post-assessment questionnaire is shown in Table II.

TABLE II. THE POST-ASSESSMENT QUESTIONNAIRE ABOUT THE FOUR FACTORS

Question No.	Statement	Your response (tick one)?				
		Strongly agree	Agree	Indifferent	Disagree	Strongly disagree
1	I got tired after using the system					
2	The interactions performed were analogous to real world actions					
3	The technique is suitable for VR/AR applications.					
4	As a whole, the system was easy to use.					

Percentage of user’s response to the four factors are shown in Fig. 22.

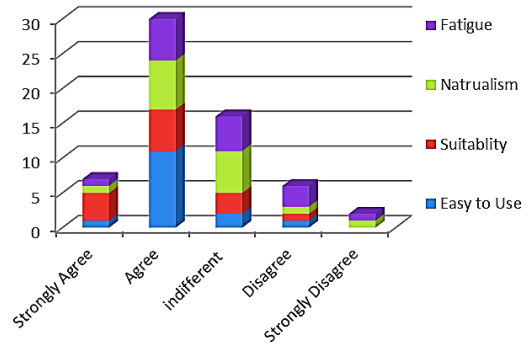


Fig. 22. Participants’ response about the proposed system.

V. COMPARATIVE STUDY

To systematically compare the approach with other techniques, we follow the standards presented by [38] and [39] for navigation in VE. Based on the information obtained from the literature, we assign a score (‘1’) to a technique (1-9) if a standard (S_i) is obeyed. The final score is calculated as,

$$Score = \sum_{i=1}^N S_i / N \tag{15}$$

Where N is the total number of the standard rules for navigation, see Table III.

TABLE III. THE BASIC FIVE STANDARDS OF NAVIGATION

Standards	Description
S_1	Keep user’s hands free
S_2	Useable while sitting
S_3	Simple and inexpensive
S_4	Support unrestricted flying
S_5	Wireless/no connection

Including the proposed technique, a total of nine state-of-the-art gesture based navigation techniques were selected for comparison. Details of the techniques including *Speed control*, *Dynamic turning* and *Possible challenges* are presented in Table IV.

After extracting information from the research works about the five standards, final score was computed, see Table V.

TABLE V. EVALUATION OF THE TECHNIQUES BASED ON THE STANDARDS

Technique No.	S_1	S_2	S_3	S_4	S_5	Score
1	0	1	0	1	0	0.4
2	1	1	0	1	0	0.6
3	1	0	1	1	0	0.6
4	1	1	0	1	0	0.6
5	1	1	1	1	0	0.8
6	1	1	1	1	1	1
7	1	1	0	0	1	0.6
8	0	1	1	0	1	0.6
9	1	1	1	1	1	1

TABLE IV. DETAILS THE SELECTED STATE-OF-THE-ART NAVIGATION TECHNIQUES

Technique No.	Authors	Year	Method	Device / Sensor used	Title	Dynamic Speed control	Dynamic turning	Possible Challenge(s)
1	Linn, Andreas [23]	2017	Gaze based	HTC Vive with Tobii Eyetracker	Gaze Teleportation in Virtual Reality	Yes (based on thumb button)	Yes (arc turning)	Highly Sensitive to calibration error
2	Ferracani et al. [40]	2016	Hand's gesture based	HMD, Kinect and Leap motion controller	Locomotion by Natural Gestures for Immersive Virtual Environments	No	No	WIP and Swim needs continuous legs and arms movement which may lead to tiredness
3	Kammergruber et al. [41]	2012	Arm's gesture based	Microsoft Kinect	Navigation in virtual reality using Microsoft Kinect	No	Yes	Lighting condition and background may reduce accuracy
4	Stellmach et al. [42]	2012	Gaze based	Tobii T60 eye tracker	Designing Gaze-based User Interfaces for Steering in Virtual Environments	Yes	Yes	Continuous gazing at destination point is required which may lead to asthenopia
5	Vultur et al.[43]	2016	Gesture based (upper body)	Microsoft Kinect	Real-time Gestural Interface for Navigation in Virtual Environment	Yes	Yes	Imprecise segment length of motion may reduce accuracy
6	Muhammad et al.[36]	2018	Finger's gesture (colored marker based)	Ordinary Camera	VEN-3DVE: vision based egocentric navigation for 3D virtual environments	Yes	Yes	Lighting condition and similar background object may reduce accuracy
7	X. Tong et al.[44]	2016	Gesture based (upper body)	Microsoft Kinect	Exploring Embodied Experience of Flying in a Virtual Reality Game with Kinect	No	Yes	The outstretched arm's gesture may cause hurting and quick weariness
8	Muhammad et al. [45]	2015	Fingers' gesture (Fiducial marker based)	Ordinary Camera	Steps Via Fingers: A New Navigation Technique for 3D Virtual Environments	Yes	Yes (at a constant 45o)	Quick to and fro motion may lead to unavoidable occlusion
9	M. Raees and S. Ullah	2018	Eyes flickering based	Ordinary Camera	The proposed technique	Yes	Yes	Enough lighting is required to trace eyes

VI. APPLICABILITY IN VIRTUAL REALITY APPLICATIONS

An immersive VR system should offer exploration of 3D VE. Most of the techniques investigated in the literature require at least one hand [46] for navigation. Such systems restrict users to perform other 3D interactions with one hand. Keeping both hands free during navigation guarantees natural interactions [46]. Keeping in view this basic rule [38], goal behind this research was to keep user's hands free during navigation. By this way users would have the option to perform other interactions (selection, scaling and rotation) with any gesture. As the technique reserves no hand's gesture for navigation, hence can be easily mingled with any gesture based VR system. Furthermore, as no computation for pupil or iris analysis is performed, therefore the technique is suitable for current VR application where speed matters. Results of the subjective evaluation also suggest that the technique is suitable for VR application as 75% of the users picked the fact.

VII. CONCLUSION AND FUTURE WORK

Navigation is often required in many 3D interactive virtual spaces. The emergence of virtual and augmented reality applications in different fields necessitates natural and simple way of navigation. With this contribution we propose a novel navigation technique which needs no extra device other than an ordinary camera. The intuitive eyes gestures for interaction not only ensure naturalism but also make it equally suitable for users with motor difficulties. Experimental results show that the proposed approach has reliable recognition and accuracy rates. Unlike other gesture based techniques [47], simple blinking of eyes are traced, hence the occlusion problems is minimized. Moreover, as the system remains indifferent to eyeglasses, therefore accuracy doesn't suffer from the size and color of eyeglasses. As the results suggest, errors due to false recognitions could possibly be reduced by using a high quality camera.

The proposed system is equally applicable in a wide spectrum of HCI including CAD, engineering, 3D gaming and simulation. In the emerging domain of Internet Of Things (IOT), the proposed system could be easily enhanced to interact with virtual object [48] and surgical navigation [49]. The work also covers the smooth integration of image processing and virtual environment which can lead to the design of more sophisticated virtual and augmented reality applications. This research work is a fraction of our aim of making possible interaction outside virtual reality labs. Although, we have succeeded for navigation, rotation and scaling are yet to be covered. In future, we are determined to enhance the system for rotation and scaling as well.

REFERENCES

- [1] Pfeiffer, Thies, M. E. Latoschik, and I. Wachsmuth, "Evaluation of binocular eye trackers and algorithms for 3D gaze interaction in virtual reality environments," *JVRB-Journal of Virtual Reality and Broadcasting*, vol. 5(16), (2008).
- [2] Hidalgo, V. Larrumbe, L. Martín García, and M. T. Lorenzo, "Iris recognition using the JAVAVis Library," *International Journal of Interactive Multimedia and Artificial Intelligence*, 1.1: 43-48, (2008).
- [3] L. Terziman, M. Marchal, M. Emily, F. Multon, B. Arnaldi, and A. L'ecuyer, "Shake-Your-Head: Revisiting Walking-in-Place for Desktop Virtual Reality," *Proc. ACM Symposium on Virtual Reality Software and Technology*, pp. 27-34, (2010).
- [4] D. S. Tan, G. G. Robertson, and M. Czerwinski, "Exploring 3D navigation: Combining speed-coupled flying with orbiting," In *Proceedings of the ACMCHI 2001 Conference on Human Factors in Computing Systems*, pp. 418-425, (2001).
- [5] Malik, Shahzad, A. Ranjan, and R. Balakrishnan, "Interacting with large displays from a distance with vision-tracked multi-finger gestural input," In *Proceedings of the 18th annual ACM symposium on User interface software and technology*, pp. 43-52, (2005).
- [6] Papadopoulos, Charilaos, D. Sugarman, and A. Kaufman, "Nunav3d: A touch-less, body-driven interface for 3d navigation," In *Virtual Reality Short Papers and Posters (VRW)*, pp. 67-68, (2012).
- [7] Lee, Daeho, and S. Lee, "Vision-based finger action recognition by angle detection and contour analysis," *ETRI journal* 33, no. 3, pp. 415-422, (2011).
- [8] Moerman, Clément, D. Marchal, and L. Grisoni, "Drag'n Go: Simple and fast navigation in virtual environment," In *3D User Interfaces (3DUI)*, IEEE Symposium, pp. 15-18, (2012).
- [9] Tan, Desney S., George G. Robertson, and Mary Czerwinski. "Exploring 3D navigation: combining speed-coupled flying with orbiting." In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 418-425. ACM, (2001).
- [10] Fuhrmann, Anton, D. Schmalstieg, and M. Gervautz, "Strolling through cyberspace with your hands in your pockets: Head directed navigation in virtual environments," pp.216-225, (1998).
- [11] Ajaj, Rami, F. Vernier, and C. Jacquemin, "Follow my finger navigation," *Human-Computer Interaction-INTERACT*, pp. 228-231.
- [12] Mine, Mark, Frederick P. Brooks, and C. H. Sequin, "Moving objects in space: exploiting proprioception in virtual-environment interaction," In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pp.19-26, ACM Press/Addison-Wesley Publishing Co., (1997).
- [13] Pierce, S. Jeffrey, Andrew, Forsberg, M. J. Conway, S. Hong, R. C. Zeleznik, and M.R. Mine, "Image plane interaction techniques in 3D immersive environments," In *Proceedings of the 1997 symposium on Interactive 3D graphics*, pp. 39-47, (1997).
- [14] Fuhrmann, Anton, D. Schmalstieg, and M. Gervautz, "Strolling through Cyberspace with Your Hands in Your Pockets: Head Directed Navigation in Virtual Environments," In *Virtual Environments Proceedings of the 4th EUROGRAPHICS Workshop on Virtual Environments*, Springer-Verlag, pp.216-227, (1998).
- [15] Davies, Char, and J.Harrison, "Towards Broadening the Aesthetics of Virtual Reality," *Compute Graphics* 30(4), pp. 25-28, (1996).
- [16] LaViola, Joseph, D. A. Feliz, D. F. Keefe, and R. C. Zeleznik, "Hands-free multi-scale navigation in virtual environments," In *Proceedings of the symposium on Interactive 3D graphics*, pp. 9-15, (2001).
- [17] Glueck, Michael, S. Anderson, and A. Khan, "DeskCube: using physical zones to select and control combinations of 3D navigation operations," In *Proceedings of the 2010 Spring Simulation Multiconference*, p. 200. Society for Computer Simulation International, (2010).
- [18] Montañés, J.A. Puértolas, A.M. Rodríguez and I.S. Prieto, "Smart indoor positioning/location and navigation: A lightweight approach," *International Journal of Interactive Multimedia and Artificial Intelligence*, 2.2, pp. 43-50, (2013).
- [19] Caputo, F. Marco, I. M. Ciortan, D. Corsi, M. De-Stefani, and A. Giachetti. "Gestural Interaction And Navigation Techniques For Virtual Museum Experiences," In *AVI CH*, pp.32-35, (2016).
- [20] Kim, Ji-Sun, D. Gračanin, K. Matković, and F. Quek, "Finger Walking in Place (FWIP): A traveling technique in virtual environments," In *International Symposium on Smart Graphics*, pp. 58-69. Springer, Berlin, Heidelberg, (2008).
- [21] Radkowski, Rafael, and C. Stritzke, "Interactive hand gesture-based assembly for augmented reality applications." *The Fifth International Conference on Advances in Computer-Human Interactions*, pp.303-308, (2012).
- [22] Piumsomboon, Thammathip, G. Lee, R. W. Lindeman, and M. Billingham. "Exploring natural eye-gaze-based interaction for immersive virtual reality." *Symposium on 3D User Interfaces (3DUI)*, pp. 36-39, (2017).
- [23] L. Andreas, "Gaze Teleportation in Virtual Reality," (2017).
- [24] Khamis, Mohamed, A. Hoesl, A. Klimczak, M. Reiss, F. Alt, and A. Bulling, "EyeScout: Active Eye Tracking for Position and Movement Independent Gaze Interaction with Large Public Displays." In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, pp. 155-166, (2017).
- [25] Kaaman, Albert, and Kalle Bornemark. "Gaze-supported Interaction with Smart Objects through an Augmented Reality User Interface." (2017).
- [26] Punpongsonon, Parinya, E. Guy, D. Iwai, K. Sato, and T. Boubekeur, "Extended LazyNav: Virtual 3D ground navigation for large displays and

- head-mounted displays," *IEEE transactions on visualization and computer graphics* 23, no. 8, pp.1952-1963, (2017).
- [27] Duguleană, Mihai, A. Nedelcu, and F. Bărbuceanu, "Measuring Eye Gaze Convergent Distance within Immersive Virtual Environments," *Procedia Engineering* 69, pp. 333-339, (2014).
- [28] J. Karlén (2017, September 6). Eye tracking is virtual reality's next frontier. Available: <https://venturebeat.com/2017/09/06/>
- [29] Padmanaban, Nitish, R. Konrad, T. Stramer, E. A. Cooper, and G. Wetzstein, "Optimizing virtual reality for all users through gaze-contingent and adaptive focus displays," *Proceedings of the National Academy of Sciences*: 201617251, (2017).
- [30] Dargham, J. Ahmad, A. Chekima and E.G. Moug, "Fusing facial features for face recognition," *Distributed Computing and Artificial Intelligence*. Springer, Berlin, Heidelberg, pp. 565-572, (2012).
- [31] Magdin, Martin, and F. Prikler, "Real time facial expression recognition using webcam and sdk affective," *International Journal of Interactive Multimedia and Artificial Intelligence*, In Press (2017).
- [32] Phung, S. Lam, A. Bouzerdoum, and D. Chai. "A novel skin color model in YCbCr color space and its application to human face detection," In *Proceeding of International Conference on Image Processing*, vol. 1, pp. I-I, (2002).
- [33] M. Raees, S. Ullah, S. U. Rahman, and I. Rabbi, "Image based recognition of Pakistan sign language," *Journal of Engineering Research* 4, no. 1, pp. 21-41, (2016).
- [34] Viola, Paul, and M. Jones, "Rapid object detection using a boosted cascade of simple features," In *Computer Vision and Pattern Recognition*, *Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1, pp. I-I, (2001).
- [35] Hansen, D. Witzner, and Qiang-Ji, "In the eye of the beholder: A survey of models for eyes and gaze," *IEEE Tran. Patt. Analysis and Mach. Intelligence*, 32 no. 3, pp.478-500, (2010).
- [36] Raees, Muhammad, Sehat Ullah, and Sami Ur Rahman. "VEN-3DVE: vision based egocentric navigation for 3D virtual environments." *International Journal on Interactive Design and Manufacturing (IJIDeM)*, pp.1-11, (2018).
- [37] H. Sayers, "Desktop virtual environments: a study of navigation and age." *Interacting with Computers*, 16, no. 5, pp. 939-956, (2004).
- [38] Zielasko, Daniel, S. Horn, S. Freitag, B. Weyers, and T. Kuhlen, "Evaluation of hands-free HMD-based navigation techniques for immersive data analysis." *IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 113-119, (2016).
- [39] Tapu, Ruxandra, B. Mocanu, and E. Tapu, "A survey on wearable devices used to assist the visual impaired user navigation in outdoor environments." *11th International Symposium on Electronics and Telecommunications (ISETC)*, pp. 1-4, (2014).
- [40] Ferracani, Andrea, D. Pezzatini, J. Bianchini, G. Biscini, and A. D. Bimbo, "Locomotion by natural gestures for immersive virtual environments," In *Proceedings of the 1st International Workshop on Multimedia Alternate Realities*, pp. 21-24, (2016).
- [41] Kammergruber, Florian, A. Ebner, and A. W. Günthner, "Navigation in virtual reality using Microsoft Kinect," In *Proceedings of 12th International Conference on Construction Applications of Virtual Reality*, (2012).
- [42] Stellmach, Sophie, and R. Dachsel, "Designing gaze-based user interfaces for steering in virtual environments." In *Proceedings of the Symposium on Eye Tracking Research and Applications*, pp. 131-138, (2012).
- [43] Vultur, Oana-Mihaela, S. Pentiu, and V. Lupu, "Real-time gestural interface for navigation in virtual environment," In *International Conference on Development and Application Systems (DAS)*, pp. 303-307, (2016).
- [44] Tong, Xin, A. Kitson, M. Salimi, D. Fracchia, D. Gromala, and B. E. Riecke, "Exploring embodied experience of flying in a virtual reality game with kinect," In *IEEE International Workshop on Mixed Reality Art (MRA)*, pp. 5-6, (2016).
- [45] M.Raees, S. ullah, and I. Rabbi, "Steps Via Fingers: A New Navigation Technique for 3D Virtual Environments," *Mehran University Research Journal of Engineering & Technology*, Pakistan, vol. 34, S1, pp.149-156, (2015).
- [46] Hilsendeger, Andre, S. Brandauer, J. Tolkendorf, and C. Fröhlich, "Navigation in virtual reality with the wii balance board." In *6th Workshop on Virtual and Augmented Reality*, (2009).
- [47] T. Pfeiffer, *Understanding Multimodal Deixis with Gaze and Gesture in Conversational Interfaces*. PhD thesis, Bielefeld University, Bielefeld, Germany, (2010).
- [48] Espada, J. Pascual, O.S. Martínez, B. Cristina, P.G. Bustelo and J. M. Lovelle, "Virtual objects on the internet of things," *International Journal of Interactive Multimedia and Artificial Intelligence*, 1.4: 23-29, (2011).
- [49] Guñales, Jorge, J.L. Carretero and M.B. García, "Virtual Planning and Intraoperative Navigation in Craniomaxillofacial Surgery," *International Journal of Interactive Multimedia and Artificial Intelligence*, 4.5: 65-67, (2017).



Muhammad Raees

Muhammad Raees is currently serving the department of Higher Education, KPK, Pakistan, as an Assistant Professor of Computer Science. He did M.Sc.(Computer Science) from Peshawar University, Pakistan in 2003. He completed M.Phil. with distinction in Virtual Reality Programming from University of Malakand in 2015. He is pursuing Ph.D. (Virtual Reality & Intelligent Systems) from University of Malakand. His research interests include Virtual Reality Programming, Computer Vision and Machine Learning.



Sehat Ullah

Sehat Ullah has been serving the Department of Computer Science and IT, University of Malakand Pakistan, since 2001. Currently he is Associate Professor of Computer Science and HOD of the Virtual Reality Department. He has Ph.D. degree in the field of VR and 3D Interactions from University of Évry, France. He has supervised more than eight Ph.D. scholars and has about 30 research publications to his credit.