Mehran University Research Journal of Engineering and Technology
Vol. 40, No. 2, 358 - 370, April 2021
p-ISSN: 0254-7821, e-ISSN: 2413-7219
DOI: 10.22581/muet1982.2102.10

Thumb Inclination-Based Manipulation and Exploration, a Machine Learning Based Interaction Technique for Virtual Environments

Muhammad Raees[1a], Sehat Ullah[1b], Inam Ur Rehman[1c], Muhammad Azhar[1d]

RECEIVED ON 13.02.2019, ACCEPTED ON 03.05.2019

ABSTRACT

In the context of Virtual Reality (VR), interactions refer to the plausible actions in a Virtual Environment (VE). To have an engrossing interface, interactions by the gestures of hand are becoming prominent. With this research work, a novel interaction technique is proposed where interactions are performed on the basis of the position of thumb in dynamic image stream. The technique needs no expensive tracker but an ordinary camera to trace hand movements and position of thumb. The interaction tasks are enacted in distinct interaction states, where the Angle of Inclination (AOI) of thumb is used for state-to-state transition. The angle is computed dynamically between the tip-of-thumb and the base of the Region of Interest (ROI) of an input image. The technique works in two phases: learning phase and application phase. In the learning phase, user-defined fist-postures with distinct AOI are learnt. The Support Vector Machine (SVM) classifier is trained by the AOI of the postures. In the application phase, interactions are performed in distinct interaction states whereas a particular state is activated by posing the known posture. To follow the trajectory of thumb, dynamic mapping is performed to control a virtual hand by the position of thumb in the input image. The technique is implemented in a Visual Studio project called Thumb-Based Interaction for Virtual Environments (TIVE). The project was evaluated by a group of 15 users in a moderate lighting condition. The 89.7% average accuracy rate of the evaluation proves suitability of the technique in the wide range VR applications.

Keywords: 3D Interactions, Virtual Reality, Gesture Recognition, Machine Learning

1. INTRODUCTION

A VE is the computer generated emulation of the real world or of an imaginary space. In the context of VR, interaction is the man-machine dialogue inside a VE. With the privilege of interactions, a user sustains the belief of being there and takes a virtual world as a near-to-real world. By now, it has been proved that gesture-based interactions are suitable for 3D (Three Dimensional) interactions in a VE [1, 2]. 3D interaction can either be direct or indirect. The former is to select and/or manipulate objects through natural instinctive gestures while the latter is to generate commands using menus and button-clicks. Interactions via indirect techniques have a little consistency in a VE because of their difficulties and imperceptibility in the virtual space [1].

The interfaces based on the perceptive gestures ensure naturalism. Therefore, gestural interfaces are suitable for man-machine communication [3]. Although with such interfaces, the degree of realism of a VE is raised [4], due to the machine-side and user-side challenges [4, 5], hand gesture-based systems are comparatively more error-prone [5]. The said challenges are attainable if the efficacy of Machine Learning (ML) is

[1] Department of Computer Science and IT, University of Malakand, Chakdara, KPK, Pakistan.
Email: [a]visitrais@yahoo.com (Corresponding Author), [b]sehatullah@hotmail.com, [c]inam.btk@gmail.com, [d]azharitteacher@gmail.com

properly explored in the realm of VE. This may lead to the designing of a reliable, faster and feasible gesture based 3D interface.

With this research work we report the design, implementation and evaluation of Thumb Inclination Based Manipulation and Exploration (TIME); an interaction technique for the VE. Unlike other interaction techniques, interaction tasks are performed by the position of thumb. To reduce computational cost, instead of storing the database of gestures, different fist-postures for different interactions are learnt and recognized dynamically. Moreover, the technique needs no explicit programming for features tracking. During the learning phase, different fist-postures are learnt on the basis of distinct AOI. A dialogue-box with the basic interaction tasks: translation, selection and scaling is displayed whenever a still fist-posture is traced for about 500 ms. A user can associate a posture with any interaction by clicking over an interaction task in the learning phase. In case of inappropriate posture, the system can be reverted by availing the appropriate option. In the application phase, switching to an interaction state is performed by posing a known fist-posture making the appropriate AOI. At the detection of a fist-posture, the AOI is forwarded to the SVM classifier for accurate classification. After an interaction state S_k is activated, the movements of the hand along the x, y and/or z-axis are traced to perform interaction I_k about the respective axis. Linking the libraries of OpenGL and OpenCV, the TIME technique is implemented in the case-study project; TIVE. A satisfactory accuracy rate of 89.7% was achieved for a total of 240 interaction attempts.

Rest of the paper is organized into 6 sections. Previous work is discussed in Section 2. The TIME technique is presented in Section 3. Section 4 is about the implementation of the technique for interactions in a VE. Evaluation of the technique without the use ML is discussed in Section 5. Discussion about the interaction technique is covered in Section 6. Finally, Section 7 presents conclusion and future work.

2. RELATED WORK

By dint of interactions, a user sustains the belief to be an active actor of the virtual environment. To ensure the naturalness of a VE, the interface of a VR application should be consistent and coordinated [6]. Interactions by the traditional interactive tools like keyboard and mouse are insufficient and lag behind to completely involve the users [7-8] in interactive 3D VE. As hand gestures are adaptable and flexible, hence suitable for interactions in a VE [2,9].

In the literature of VR, several gesture-based techniques have been proposed so far to ensure feasible 3D interactions. The earlier systems based on magnetic [10] and mechanical [11] tools are becoming inadequate due to their intrinsic limitations and cumbersome setups. Nevertheless, the recent advancements in image processing have paved the ways for more efficient and natural interfaces. The systems proposed by [12-14] utilize static hand postures for interaction. However, the static posture based systems suffer from the orientation obstruction [15]. For dynamic gestural interfaces, colored markers are proposed for hand gestures recognition [12, 16-17]. Although, different sections of hand are easily identified by the use of markers, such systems are highly light sensitive [18]. Systems based on inertial sensors have also been proposed to overcome the issue of light dependency [19, 20]. The tangible 3D system of [21] supports distinct hand gestures. The use of multiple interfacing devices like black light source, projectors and polarized glasses make the system a rare choice. Similarly, the Infrared (IR) based tracking system of [22] requires a high cost Helmet-Mounted Display with an array of IR cameras. Interactions by the Fiducial markers have also been proposed for VR and Augmented Reality (AR) systems [23-25]. The systems proposed by [26, 27] utilize markers in AR applications. However, these systems offer no facility for object manipulation. Moreover, Fiducial marker are good for short range applications only [28]. Frameworks about the use of Microsoft Kinect; the contemporary tracking device, have also been presented for VR interactions [29]. While Kinect is suitable for tracing the body gestures [30], the device cannot distinguish each finger individually [31]. With the help of Leap Motion Controller (LMC), the bimanual interaction technique [32] detects small finger movements. The LMC based systems provide promising results besides a larger workspace if used

with HMD. However, for desktop-based applications, the applicability of such systems is less because of the restrained working space of the LMC.

3. TIME: THE PROPOSED TECHNIQUE

This research work intends to incorporate the ML classifier in the designing of a gesture-based direct interface. Engine of the system comprises three main modules; Fist-Detector (FD), Fist-Learner (FL) and Fist-Handler (FH). The modules perform detection, learning and handling of the traced fist-postures respectively. The first two modules; FD and FL are invoked in the learning phase. On the basis of the single feature (AOI), the FL module learns different fist-postures.

To properly trace the position of thumb, segmentation [33, 34] of a dynamic scanned image is performed on the basis of skin-color. Pre-processing for the binarization of the input images is performed after capturing the video frames. In the application phase, the FD traces the postures and forwards that to the PH module. Moreover, coordinates mapping is performed to control the positioning of the Virtual Hand (VH) by the position of thumb in video streams. Keeping into account the persistence of frame data, the FD module detects the postures. After the learning phase, the FH is invoked by every traced posture to classify the posture and feed an appropriate interaction command to the VE. In the learning phase, Thumb Template (TT) is identified besides computing the Thumb Area (TA) and Mid of Thumb (MT).

The Time-Tick procedure of the system checks the stillness of a fist-posture for calculating the AOI. The AOI is calculated if a known fist-posture is posed for about 500ms. A user needs to click on an interaction task (displayed in the form of a dialogue box) to reserve a fist-posture for an interaction task. The AOI and the interaction task selected are forwarded to the SVM classifier for learning. In the application phase, coordinates mapping is performed for the real-time interaction with the VE. Based on the AOI, the SVM classifier recognizes the posture and an interaction task is enacted. After initiating an interaction task, dynamic position of the thumb is traced to interact with the VE. To cancel an interaction task and to switch the system back to the default state, the fist-posture with thumb pointing downward is needed to be posed. The entire process is shown in Fig. 1.

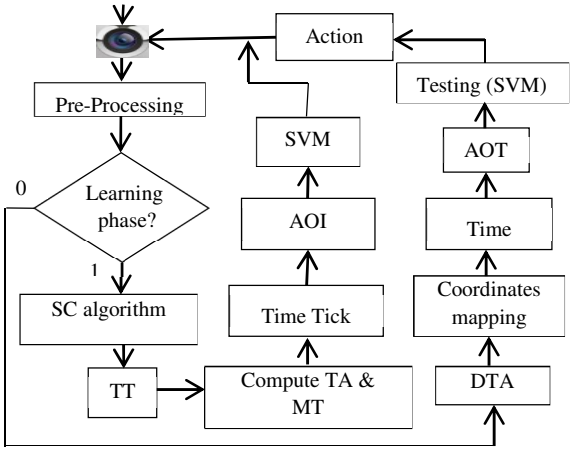


Fig.1: Schematic of the Time Technique

3.1 Pre Processing

Both in the learning and application phases, a scanned frame image; $\mathrm{Fr}_{\mathrm{img}}$ is converted to YCbCr color space to get the $\mathrm{YC}_{\mathrm{img}}$. As proved by [35], the YCbCr is the optimal model to separate skin color from the non-skin colors. The YCbCr model is, therefore, followed for the segmentation of hand.

$$\mathrm{YC}_{\mathrm{img}}\begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \frac{1}{256}\begin{bmatrix} 65.6 & 129 & 24.8 \\ -37.8 & -74.5 & 112.3 \\ 111.9 & -93.6 & -18.5 \end{bmatrix}\begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1)$$

where $\mathrm{YC}_{\mathrm{img}}$ is the YCbCr model based representation of the $\mathrm{Fr}_{\mathrm{img}}$. The $\mathrm{YC}_{\mathrm{img}}$ is then thresholded with the most optimal chrominance range [36] to get the binary version ($\mathrm{B}_{\mathrm{img}}$) of the $\mathrm{YC}_{\mathrm{img}}$.

$$\mathrm{B}_{\mathrm{img}} = \begin{cases} 1, & \text{if } 77 < \mathrm{YC}_{\mathrm{img}}.\mathrm{Cb} < 127 \\ & \quad \Lambda\ 133 < \mathrm{YC}_{\mathrm{img}}.\mathrm{Cr} < 173 \\ 0, & \text{Otherwise} \end{cases} \quad (2)$$

3.2 Extraction by Sliding Scan

The ROI is extracted by using our designed sliding scan algorithm [37]. The algorithm traces the first-most white pixels at top, left and right. The region enclosed by the boundary pixels; Top-most (T_m), Left-most (L_m) and Right-most (R_m) is extracted as the ROI, see Fig. 2.

$$ROI = \begin{pmatrix} \cup_{r=D_m}^{Tm}(B_{img}), \\ \cup_{c=L_m}^{Rm}(B_{img}) \end{pmatrix} \quad (3)$$

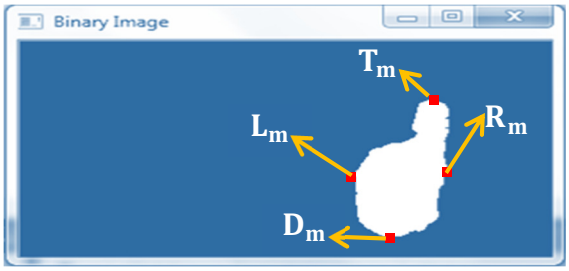


Fig. 2: The B_{img} with the boundary pixels

The same algorithm is followed to trace the TT. To accurately trace the TT, a user needs to pose a fist-posture with thumb up in the initial frame. In order to avoid false detection of thumb, the skin region (white) with 5 non-skin pixels at the top, left and right are extracted as the TT. Moreover an empirical constant ξ is added with the thumb Top-most pixel (TT_m) to scan enough region of the thumb.

$$TD_m = TT_m + \xi \quad (4)$$

The region (white/skin) enclosed by the thumb boundary pixels TL_m, TR_m, TT_m and TD_m pixels, as shown in Fig. 3, is treated as the TT for onward processing. Once the TT is extracted, template matching [38-39] is performed to locate the thumb position in the dynamic image frames. The area of thumb (TA) is calculated using the zero order moments [40], whereas the mid-point of thumb; MT is obtained by the 1st and 2nd order moments [40].
Using the zeroth moment, the area (TA) of the TT is calculated as,

$$TA = \sum_{x=0}^{TT_{rows}} \sum_{y=0}^{TT_{columns}} TT_{(x,y)} \quad (5)$$

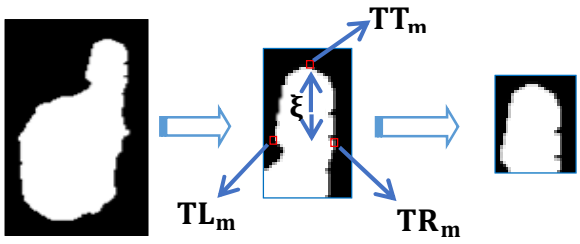


Fig. 3: Extraction of the TT from the ROI

where *x* is the row and *y* the column position of a skin pixel in the image; TT. The extraction of ROI and TT from an input image is shown in Fig. 4.

Fig. 4: (a) The FR_{img} , (b) The ROI Image and (c) The Thumb Template (TT).

3.3 Coordinates Mapping

Parallel processing is performed in the TIVE project to capture real time video streams and to perform interaction in the VE at same time. However, the coordinate systems of the OpenCV and OpenGL are quite different, hence coordinates mapping is required for the seamless interaction. The Origin $O(0,0,0)$ of the OpenGL rendering frame lies at the middle of the clipping area, see Fig. 5(a). Unlike OpenGL, the image frame origin; $F_o(0,0)$ lies at the top left, see Fig. 5(b).

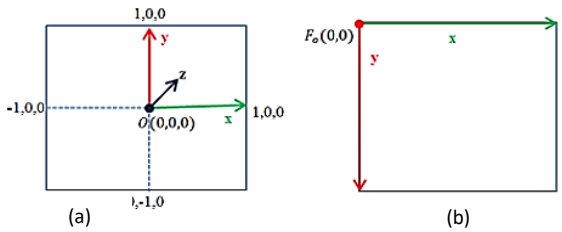


Fig. 5: The (a) OpenGL and (b) OpenCV Coordinates

In the proposed system, the mapping function; w [37] transforms the MT $\in \mathbb{R}^2$ to the OpenGL coordinates. During the application phase, let $MT_f \in \mathbb{R}^2$ denotes the MT position in the first frame and let $MT_d \in \mathbb{R}^2$

be the dynamic position of the MT in any following input image frame. Based on the mentioned assumption, the position of VH; VHP : VHP $\in \mathbb{R}^3$ is computed as,

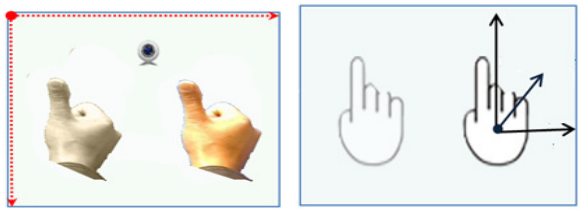
$$\mathrm{VHP}(x,y) = \mathrm{w}(\mathrm{MT_f}, \mathrm{MT_d}) \quad (6)$$

To keep the VH visible during navigation, the look-at vector value of the Virtual Camera (VC) is constantly assigned as the z-axis to the VHP.

$$\mathrm{VHP.z} = \mathrm{VC.z} \quad (7)$$
$$\mathrm{w}(x;y) = ((\Delta \mathrm{Px}/\mathrm{Tc});\ (\Delta \mathrm{Py}/\mathrm{Tr})) \quad (8)$$
$$\Delta \mathrm{Px} = (\mathrm{MT_{d.x}} - \mathrm{MT_{f.x}}) \quad (9)$$
$$\Delta \mathrm{Py} = (\mathrm{MT_{d.y}} - \mathrm{MT_{f.y}}) \quad (10)$$

where 'Tc' and 'Tr' represent the total number of columns and rows respectively, see Fig. 6. The mapping function is controlled by a Boolean variable η as,

$$\eta = \begin{cases} 0 & \text{Learning Phase} \\ 1 & \text{Application Phase} \end{cases}$$



(a) (b)

Fig. 6: The Mapping between (a) OpenCV and (b) OpenGL

3.4 Fist Detection

The FD module of the system traces the static fist-postures. A posture is traced for learning (in learning phase) or for testing (in application phase) if posed for about 500ms. To avoid the gesture-spotting issue [41], the extraction of the feature AOI is performed after the expiry of the time-slice. To ensure this, the time-tick module of the system measures the dynamic variation between any two successive image frames; Next Frame (NF) and Previous Frame (PF). The absolute bitwise difference between NF and PF is checked by a background stop-watch. A slight hand and/or thumb movement resets the stop-watch. Detecting no variation for approximately 500ms, the posture is assumed to be posed properly. At the beginning of making a posture; $t_0 = 0$, the first NF is set as PF. With each following tick; $t_n \mid t_n > t_0$, a scanned NF is compared with the PF. If the difference is high, the stop watch is made reset, otherwise is incremented by 1. At $t_n = 500$, the skin-color based detection from the last NF is performed. The process is shown in Fig. 7.

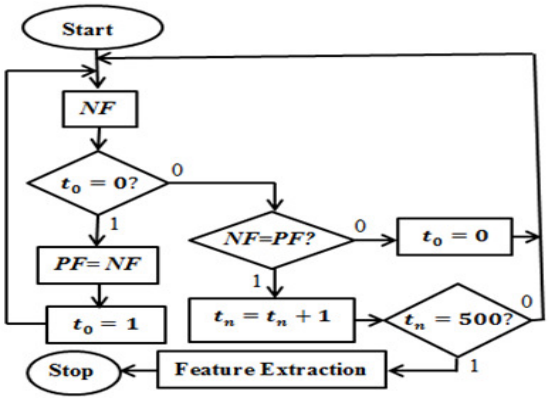


Fig. 7. Schematic of the Time Tick Module

3.5 Fist Learning

The FL module learns a scanned posture on the basis of the feature; AOI. In order to allow the user to set a captured posture for an interaction, a list of interaction tasks is displayed over the posture image. The list contains the basic interactions, except navigation which is accomplished by the perceptive movement of hand in the default state.

From the displayed interaction tasks, a user may select one interaction at a time to associate it with a posture. By selecting an interaction task from the list, a posture is made reserved for that particular interaction task. The position; Bottom Mid (BM) of the $\mathrm{Fr_{img}}$ as origin (see Fig. 8), the single lightweight feature; AOI is calculated as,

$$\mathrm{AOI} = \arccos\left(\frac{(\mathrm{MT_x})(\mathrm{BM_x})+(\mathrm{MT_y})(\mathrm{BM_y})}{\sqrt{(\mathrm{MT_x^2}+\mathrm{MT_y^2}).(\mathrm{BM_x^2}+\mathrm{BM_y^2})}}\right) \quad (11)$$

A unique Fist ID ($\mathrm{F_{id}}$) is assigned to a posture. The $\mathrm{F_{id}}$ is used as a label of the class representing the feature vector. After selecting an interaction task for a traced

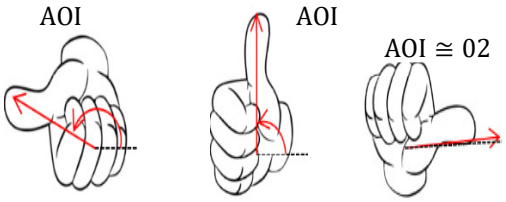


Fig. 8: The Three Fist Postures with the approximate AOI

posture, the vector containing the single AOI feature is forwarded to the SVM classifier for training (in learning phase) or for classification (in application phase). The effective ML classifier; SVM [42, 43] is used to learn distinct fist-postures on the basis of AOI. The classifier is designed to learn features $F_i \in \mathbb{R}$ and set Y for class labels $y_i \mid y_i \in Y$ where $i = \{1, 2, ., ., n\}$. By using the feature vector, SVM builds an optimal hyperplane. The hyperplane is used to predict a class label y_x for feature vector F_x using the set S of features and class lables; $S = \{(F_1, y_1), ., ., ., (F_n, y_n)\}$. If most of the features belonging to y_x are on one side of the hyperplane as $y_i \in \{+1, -1\}$. In the proposed technique, the classifier learns the features $AOI_i \in \mathbb{R}$ and class labels y_i for n numbers of distinct fist-postures; $i = \{1, 2, ., ., n\}$. Hence, the set S is given as, $S = \{(AOI_1, y_1), ., ., ., (AOI_n, y_n)\}$. The inner product space $X: X \subseteq \mathbb{R}$ is computed to get the scoring function f between $AOI_i \in X$ and $y_i \in Y = \{1,2, ., ., n\}$.

The function f measuring the similarity of an input instance $AOI_i \in X$ in the defined prototype space D is given as,

$f : X \times D \rightarrow \mathbb{R}$

During the learning phase, a unique natural number; F_{id} is assigned dynamically to a fist-posture when a user opts for associating the posture for an interaction task. The same F_{id} is used as a class label for the detected feature (AOI) of the fist-posture. Therefore, with the dataset D the SVM classifier associates F_{id} with its feature vector AOI_i,

$$D = \{ (AOI_i, F_{id}) | \, AOI_i \in \mathbb{R} \}_{i=1}^{F_{id_n}} \quad (12)$$

3.6 Fist Handling

After learning the postures for different interactions, the FH module identifies a known posture by performing the One-Versus-Rest (OVR) approach instead of the One-Versus-One [44]. To obtain label y_x with the OVR approach, F_{id_n} classes are compared with $F_{id_n} - 1$ classes for an unknown extracted feature

$$AOI_x \text{ as}, y_x = \text{argmax}_{i=1,2,.,.,F_{id_n}} (w_i . \gamma(AOI_x) + b_i) \quad (13)$$

where γ is the decision function, w the weight vector and b the slope intercept of the hyperplane [45]. The predicted class label; y_x is used as F_{id} to get an associated State-ID (SID). The process of performing an interaction task T_x bearing SID_x by posing a known fist-posture Fid_x is given as,

$\gamma(AOI_x) \rightarrow y_x$
$y_x \rightarrow Fid_x$
$Fid_x \rightarrow SID_x$

The process of initiating an interaction (task) from a fist-posture is shown in Fig. 9.

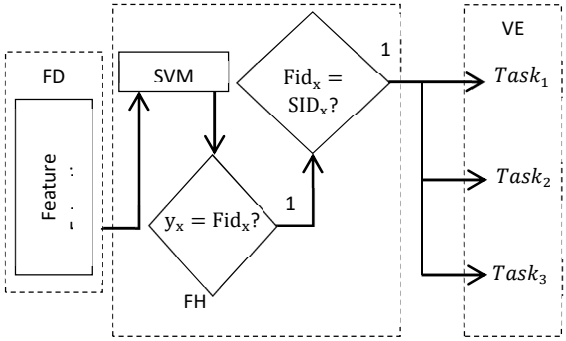


Fig. 9: Identifying an Interaction from a known Fist-Posture

3.7 Interactions by the Movements of Thumb

The basic VR interactions; translation, selection, scaling and navigation are performed by the perceptive movements of thumb. At an attained state $SID_i \mid SID_i \neq SID_0$, interactions are performed by the dynamic 2D position of the MT.

Each time, the position of the MT in a preceding frame (MT_P) is checked against the MT in the following frame (MT_f). Tracing a change between the MT_p and MT_f about an axis, appropriate interaction is performed along that particular axis. An object is

selected for manipulation as the VH enters into the aura of the object.

3.7.1 Navigation

Navigation refers to the insight of locomotion inside a VE. For exploring a VE, navigation is supported in the default state. As conceivable, the inside (forward) navigation is performed by the forward hand movement, see Figs. 10-11. The reverse (backward) navigation is carried out by the movement of hand away from the camera. To deduce forward or backward hand movement in a scanned 2D image, the initial TA is compared with the Dynamic Thumb Area (DTA); DTA=(y)(TA), for y > 1. To prevent the possibility of unintentional movement of hand, an increase or decrease by 8 units is omitted as clear from the following pseudo-code.

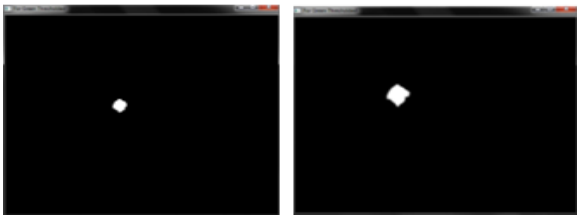
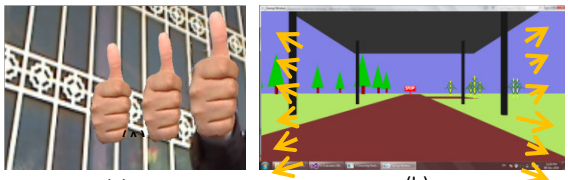


Fig. 10: (a) TA in an Initial FR_{img} and (b) DTA after moving the hand towards the camera (Forward Movement)



(a) (b)

Fig. 11(a): Forward Hand Movement and (b) Forward Navigation.

if (DTA $>$ TA + 8)
Forward Navigation
if (DTA $<$ TA $-$ 8)
Backward Navigation

3.7.2 Selection

At the time of posing a fist-posture learnt by the SVM for selection, an object behind the VH is selected. For accurate selection of an object; O_b, 0.2 units (OpenGL) is added with the center of a 3D object. The extended box is treated as the object's aura. The pseudo-code of the selection process is given as,

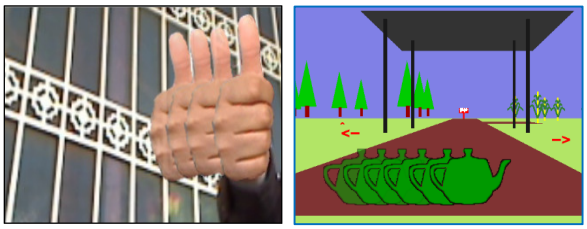
if (SID_x = Selection)
if $\begin{pmatrix}(\text{abs}(\text{VHP}(x) - O_b(x) < 0.2\ ) \text{ AND} \\ (\text{abs}(\text{VHP}(x) - O_b(x) < 0.2)\end{pmatrix}$
Selection of the O_b

3.7.3 Translation

Translation is the interaction used to change the position of a virtual object about an axis in a VE. After switching to the translation state by posing the appropriate fist posture, horizontal movement of hand translates an object along the x-axis. Translation along the y-axis is performed by the vertical hand movement. For translation about the z-axis is performed by the forward and backward hand movement along the look-vector.

if (abs($MT_{p.y}$-$MT_{f.y}$)<8 AND abs($MT_{p.x}$- $MT_{f.x}$)>8)
if ($MT_{f.x}$ > $MT_{p.x}$)
Translation along the +ve x-axis
if ($MT_{f.x}$ < $MP_{p.x}$)
Translation along the -ve x-axis
if (abs($MT_{p.x}$-$MT_{f.x}$)<8 AND abs($MT_{p.y}$- $MT_{f.y}$)>8)
if ($MP_{f.y}$ < $MP_{p.y}$)
Translation along the -ve y-axis

The movement of hand along the x-axis for translation along the x-axis is shown in Fig. 12.



(a) (b)

Fig. 12(a): Movement of hand (b) Translated the object along x-axis.

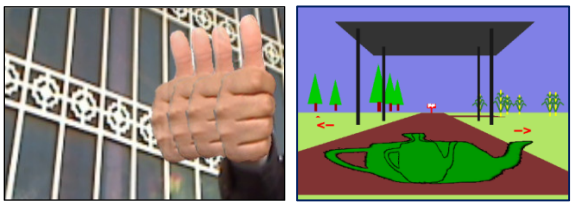
3.7.4 Scaling

Scaling is to increase (scale-up) or decrease (scale down) the size of an object. Scaling (scale up) about the x and y axis is performed by the hand movements along the +ve x and y-axis respectively. As conceivable, down scaling is carried out by the hand

movements along the –ve x or y axis. The forward hand movement scales up while the backward movement of hand scales down an object about the z-axis respectively.

if (DTA $>$ TA + 8)
Scale about the z-axis
if (DTA $<$ TA − 8)
Down scale about the z-axis

Scaling about the x-axis is shown in Fig. 13.



(a) (b)

Fig. 13(a): Horizontal Movement (b) Scaling about the x-axis

4. IMPLEMENTATION AND EVALUATION

The TIME technique is implemented in the case-study application; TIVE using a Corei5 laptop with 2.60 GHz processor and 4GB DDR. In a Visual Studio project, the OpenGL library was used for the front-end VE. At the back-end image processing was performed by the OpenCV library. Offering a first person's view, the VH represents the user's position in the VE, see Fig. 14. Different 3D objects are rendered at different points of the VE so that to engross the users during interaction. The system provides both textual and audio (beef) signal whenever a user initiates an interaction. With the *'r'* key-press event, the entire system is reset where the VC eye is set to look at the origin of the scene; O(0,0,0).

Fig. 14: The Virtual Scene for Evaluation of the time

Fifteen participants, all male, ages 22-44, 14 right-handed and 1 left-handed performed the four tasks. The users were familiarized to the system by demonstrating how to interact and make the postures. Moreover, all the participants performed pre-trials for the basic interaction tasks. They were guided to press the *Enter* key to reset the system for a new trial. All the experiments were performed in the University IT lab in an average lighting condition with illumination level approximately 110 lux [46]. A user interacting with the system with his thumb is shown in Fig. 15.

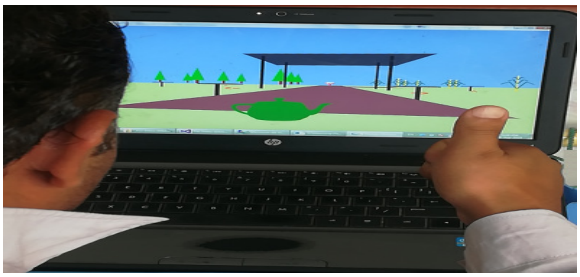


Fig. 15: A User Performs Interaction by Dynamic Movement of Thumb

4.1 The Interaction Tasks

Participants were asked to perform the following four tasks in the designed 3D environment.
The tasks are arranged in order to assess the basic interactions; Selection, Scaling, Navigation and Translation. In the mid of the VE, a Teapot is rendered to be picked (selected) and manipulated. Each of the users performed two trials of the following tasks.

Task-1: Navigate (forward) to the end point of the VE.
Task-2: Navigate (backward) to the starting point.
Task-3: Select the teapot and translate it till the far middle table.
Task-4: Select the teapot and Scale it about the x, y and/or z-axis.

In a single trial, scaling is assessed three times, selection and navigation are evaluated two times while translation is evaluated one time. False detection and inappropriate interactions were deemed as errors. Overall accuracy achieved for the 240 interaction attempts, as shown in Table 1, was 89.7%. Mean of the accuracy rate (in %age) of the two trials are shown in Fig. 16.

Table 1: Figures and facts of the evaluation				
Interaction Task	Correct	False	Total	Accuracy (%)
Selection	57	3	60	95
Scaling	83	7	90	92.2
Translation	25	5	30	83.3
Navigation	53	7	60	88.3
	218	22	240	89.7

4.2 Learning Effect

Outcomes of the evaluation revealed that performance of the users improves with practice. The learning effect was measured from the errors occurrence rate. To analyze differences in the means of the two trails,

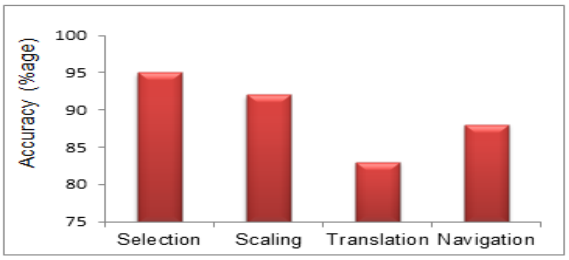


Fig. 16: Mean of the Percentage Accuracy of the two trials

a paired two sample T-test was used. It was assumed that the means were the same; (H_0: $\mu_d = 0$). The hypothesis; H_0 was rejected after getting a significant difference between the outcomes of Trail-1 (M=63.03, SD=5.4) and Trail-2 (39.92, SD=5.9) conditions; (t(6)=-9.08, p=0.009).

The graph showing the errors (%age) of Trail-1 and Trial-2 is shown in Fig. 17. During translation and navigation, the MT was wrongly traced and hence, comparatively more errors were counted as shown in the Fig.

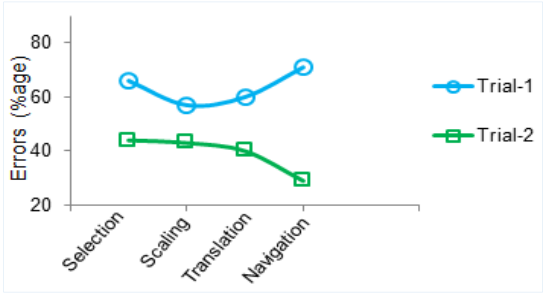


Fig. 17: The Percentage of errors occurred in Trial-1 and Tiral-2 interactions.

4.3 Subjective Analysis

At the end of the evaluation, a questionnaire was presented to the participants to measure the three factors; *Ease of Use*, *Fatigue* and *Suitability in VE*. Most of the participants are opted in favor of the technique. The percentage of the user's response acquired by the questionnaire is shown in Fig. 18.

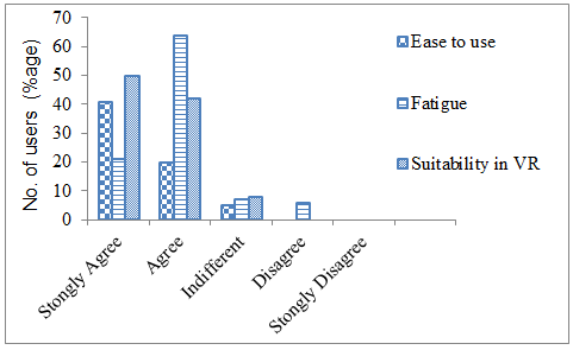


Fig. 18: Subjective analysis of the three factors

5. RECOGNITION BY EXPLICIT PROGRAMMING

To evaluate the recognition of fist-postures without the use of ML, a separate project; TIVE-2 was designed by modifying code of the TIVE project. As no learning is involved in TIVE-2, the FL and FH modules were replaced by a single module; FR (Fist Recognizer). Using a nested if-else structure, explicit programming was performed to recognize different fist-postures. The algorithms for dynamic interaction by the movements of thumb (as discussed under subsection 3.7) were kept unchanged. With TIVE-2, the same tasks were performed by twelve participants in the same environment (the university IT lab). Each of the users performed two trials of the tasks. An average accuracy rate of 82.8% was achieved for 192 interaction attempts (see Table 2).

Table 2: Outcomes of the second evaluation				
Interaction Task	Correct	False	Total	Accuracy (%)
Selection	39	9	48	81.3
Scaling	60	12	72	83.3
Translation	19	5	24	79.2
Navigation	42	6	48	87.5
	162	**30**	**192**	**82.8**

As the AOI for different fist-postures were explicitly set, some postures were not correctly identified. The reason behind low accuracy was variation in the size (length and width) of the users' hand and/or thumb. Moreover, some of the participants faced difficulties in posing the exact fist-postures with the required AOI.

6. DISCUSSION

According to the recent research works carried out in VR interaction, it has been proved that gesture-based interactions are suitable for VE. However, it is also a fact that such systems are difficult to design. As the hand size and length of fingers vary from individual to individual, therefore interactions by the whole hand gestures are susceptible to false recognition.

With the TIME technique a novel interaction approach is proposed where users will be able to interact with a 3D environment by the simple postures of thumb. With the inclusion of ML classifier (SVM), the system is made intelligent to associate a fist-posture for an interaction task at run time. To analyze outcomes of the technique with and without ML classifier, two projects; TIVE and TIVE-2 were designed. With TIVE, a user trains the system with different fist-postures at run time. In the TIVE-2 project, AOI for the postures are pre-defined during coding. As a user trains and tests the system with his/her own fist-posture, therefore, comparatively high accuracy rate was achieved for the TIVE project. However, due to dissimilarities in hand and thumb size of the users, low accuracy rate was reported for TIVE-2.

The distinguishing feature of the proposed technique is that it frees a VR user to remember the gestures set by others. Once an interaction state is activated, the perceptive horizontal and vertical movements of hand are traced for translation, selection, navigation and scaling. Unlike the costlier and complex setup of data-gloves [47] and armbands [48-49], an ordinary camera is used for the detection of hand and thumb. Outcomes of the technique support applicability of the technique in the VR domain. It is pertinent to add that during the evaluation it was observed that most of the errors were due to the quicker movement of the user's hand. In such cases an ordinary camera misses some of the required frame data. The challenge of quicker hand movement can be resolved by using a high quality camera. Moreover, with the use of a high speed processor, faster frame rate and timely extraction of frame data is possible. In short, accuracy rate of the system can be raised with the use of a high speed processor and a quality camera.

7. CONCLUSION

To cope with the rampant pace of the VR developments, a simple and natural interface is needed for intuitive 3D interactions. With this contribution, we propose a ML based interaction technique where interactions are performed by simple movement of hand. Based on the positions of thumb, different fist-postures are learnt and recognized based on the lightweight feature; AOI. To increase accuracy of the technique, the SVM classifier may be trained with some additional features as well. For instance, the angle of declination of thumb may be used to unambiguously specify the position of thumb in input stream. Similarly, image-based features after adaptive tiling may be used to improve the accuracy rate. However, the single lightweight-feature; AOI is used to ensure quick processing. The technique is twice evaluated; with ML and without ML classifier. An average accuracy of 89.7% was achieved for the TIVE project where the ML algorithm is used to recognize the fist-postures. In TIVE-2 project, postures are identified without the use of ML classifier. In a separate evaluation session, comparatively low accuracy (82.8%) was reported for TIVE-2. It is pertinent to note that the TIVE-2 project was evaluated by 12 users. By increasing the number of users, probability of dissimilarities among hand/thumb size would increase. Hence, the possibility of low accuracy will also increase in case of using the technique without ML classifier.

As a whole, outcomes of the evaluations suggest suitability of the technique in a wide spectrum of man-machine interactions particularly in 3D gaming, robotics virtual prototyping and simulation. The work also presents the integration of image processing, ML and VR. With less efforts, the technique can be made implementable on other sensing platforms. As our future strategy, we are determined to enhance the system for the collaborative VE.

ACKNOWLEDGEMENT

The authors whole-heartedly acknowledge the support and assistance provided by the staff of the Department of Computer Science and Information Technology, University of Malakand, Pakistan.

REFERENCES

1. Rautaray S.S., Agrawal A., "Vision based hand gesture recognition for human computer interaction: a survey", *Artificial Intelligence Review*, Vol. 43, No. 1, pp.1-54, 2015.
2. Gallud, J. A., Villanueva, P. G., Tesoriero, R., Sebastian, G., Molina, S., and Navarrete, A., "Gesture-based interaction: Concept map and application scenarios", *Proceedings of 3rd International Conference on Advances in Human-Oriented and Personalized Mechanisms, Technologies and Services*, pp. 28-33, Nice, France, 22-27 August 2010.
3. Caputo F.M., "*Gestural interaction in Virtual Environments: user studies and applications*", Doctoral dissertation, University of Verona, 2019.
4. Vanacken D., Beznosyk A., Coninx K., "Help systems for gestural interfaces and their effect on collaboration and communication", In *Workshop on Gesture-Based Interaction Design: Communication and Cognition*, 2014.
5. Benko H., "Beyond flat surface computing: challenges of depth-aware and curved interfaces", *Proceedings of the 17th ACM International Conference Multimedia,* pp. 935–944, Vancouver, British Columbia, Canada, 19-24 October 2009.
6. Cashion J., Wingrave C., LaViola J. J., "Dense and dynamic 3D selection for game-based virtual environments", *IEEE Transaction on Visualization and Computer Graphics*, Vol. 18, No.4, pp.634–642, 2012.
7. Rautaray S. S., "Real Time Hand Gesture Recognition System for Dynamic Applications", *International Journal of UbiCom*p, Vol.. 3, No. 1, pp. 21–31, 2012.
8. Hassan A., Shafi M., Khattak, M.I., "Multi-touch collaborative gesture recognition based user interfaces as behavioral interventions for children with Autistic spectrum disorder: A review", *Mehran University Research Journal of Engineering and Technology*, Vol. 35, No. 4, pp.543-560, 2016.
9. Mitra S., Acharya T., "Gesture recognition: A survey", *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, Vol. 37, No. 3, pp.311-324, 2007.
10. Kiyokawa K., Takemura H., Katayama Y., Iwasa H., Yokoya N., "Vlego: A simple two-handed modeling environment based on toy blocks", *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, pp. 27-34, Hong Kong, 1-4 July, 1996.
11. Hua J., Qin H., "Haptic sculpting of volumetric implicit functions", *Proceedings of the 9th Pacific Conference on Computer Graphics and Application*, pp.254–264, Tokyo, Japan, 16-18 October 2001.
12. Cui Y., Weng J., "Appearance-based hand sign recognition from intensity image sequences", *Computer Vision and Image Understanding*, Vol. 78, No. 2, pp. 157–176, 2000.
13. Kelly D., McDonald J., Markham C., "A person independent system for recognition of hand postures used in sign language", *Pattern Recognition Letters*, Vol. 31, No. 11, pp. 1359–1368, 2010.
14. Wang R. Y., Popović J. "Real-time hand-tracking with a color glove", *ACM Transactions on Graphics*, Vol. 28, No. 3, p. 63, 2009.
15. Kaur H., Rani J., "A review: Study of various techniques of Hand gesture recognition", *Proceedings of the 1st IEEE International Conference on Power Electronics, Intelligent Control and Energy Systems*, pp. 1–5, Delhi Technological University, Delhi India, 4–6 July 2016.
16. Kolsch M., Turk M, "Fast 2d hand tracking with flocks of features and multi-cue integration", *Proceedings of the Computer Vision and Pattern Recognition Workshop*, pp. 158–158, Washington, DC, USA, June 27 - July 02 2004.
17. Kisacanin B., Pavlovic V., Huang T. S., "*Real-*

Time Vision for Human-Computer Interaction", Springer Science & Business Media, 2005.

18. Lamberti L., Camastra F., "Real-time hand gesture recognition using a color glove", *Proceedings of the International Conference on Image Analysis and Processing*, pp. 365-373, Ravenna, Italy, September 14–16 2011.
19. Kratz L., Smith M., Lee F. J., "Wiizards: 3D gesture recognition for game play input", *Proceedings of the 2007 Conference on Future Play*, pp. 209-212, Toronto, Canada, November 14–17, 2007.
20. Neto P., Pires J. N., Moreira A. P., "Accelerometer-based control of an industrial robotic arm", *Proceedings of the 18th IEEE International Symposium on Robot and Human Interactive Communication*, pp. 1192–1197, Toyama, Japan, 2009.
21. Kim H., Albuquerque G., Havemann S., Fellner D.W., "Tangible 3D: Hand Gesture Interaction for Immersive 3D Modeling", *Proceedings of the 11th Eurographics Conference on Virtual Environments*, pp. 191–199, Aalborg, Denmark, 06 – 07 October 2005.
22. Wolf M.T., Assad C., Stoica A., You K., Jethani,H., Vernacchia M.T., Fromm J., Iwashita, Y., "Decoding static and dynamic arm and hand gestures from the jpl biosleeve", *IEEE Aerospace Conference*, pp.1–9, Big Sky, MT, USA, 2-9 March 2013.
23. Chun J., Lee B., "Dynamic Manipulation of a Virtual Object in Marker-less AR system Based on Both Human Hands", *KSII Transactions on Internet and Information Systems*, Vol. 4, No. 4, 2010.
24. Buchmann V., Violich S., Billinghurst M., Cockburn A., "FingARtips: gesture based direct manipulation in Augmented Reality", *Proceedings of the 2nd International Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia*, pp. 212-221, Singapore, 15-18 June 2004.
25. Raees M., Ullah S., Rabbi I., "Steps Via Fingers: A New Navigation Technique for 3D Virtual Environments," *Mehran University Research Journal of Engineering and Technology*, Vol. 34, No. S1, pp. 149-156, August 2015.
26. Bergig O., Hagbi N., El-Sana J., Billinghurst M., "In-place 3D sketching for authoring and augmenting mechanical systems", *Proceedings of the 8th IEEE International Symposium on Mixed and Augmented Reality*, pp. 87–94, Florida, USA, October 19 -22 2009.
27. Hagbi N., Bergig O., El-Sana J., Billinghurst M., "Shape recognition and pose estimation for mobile augmented reality", *IEEE Transactions on Visualization and Computer Graphics*, Vol. 17, No. 10, 1369–1379, 2010.
28. Song P., Goh W.B., Hutama W., Fu C.W., Liu, X., "A handle bar metaphor for virtual object manipulation with mid-air interaction", *Proceedings of the SIGCHI Conference on human factors in Computing Systems*, pp. 1297–1306, Austin, Texas, USA , 5-10 May 2012.
29. Oprisescu S. Barth E., "3D hand gesture recognition using the hough transform", *Advances in Electrical and Computer Engineering*, Vol. 13, No. 3, pp.71–76, 2013.
30. Frank W., Bachmann D., Rudak B. Fisseler D., "Analysis of the accuracy and robustness of the leap motion controller", *Sensors*, Vol. 13, No. 5, pp. 6380–6393, 2013.
31. Nabiyouni M., Laha B., Bowman D. A., "Poster: Designing effective travel techniques with bare-hand interaction", *IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 139-140, Minneapolis, MN, USA, 29-30 March 2014.
32. Jin H., Chen Q., Chen Z., Hu Y., Zhang J., "Multi-LeapMotion sensor based demonstration for robotic refine tabletop object manipulation task", *CAAI Transaction on Intelligent Technology*, Vol. 1, No. 1, pp. 104–113, 2016.
33. Bari M., Ahmed A., Naveed S., "Lungs Cancer Detection Using Digital Image Processing Techniques: A Review", *Mehran University Research Journal of Engineering and Technology*, Vol. 38, No.2, pp. 351-360, 2019.
34. Unar S., Jalbani A.H., Jawaid M.M., Shaikh M. Chandio A.A., "Artificial Urdu text detection and localization from individual video frames", *Mehran University Research Journal of*

Engineering and Technology, Vol. 37, No.2, pp.429-438, 2018.

35. hung S. L., Bouzerdoum A., Chai, D., "A novel skin color model in ycbcr color space and its application to human face detection", *Proceedings of the International Conference on Image Processing*, Vol. 1, pp. I-I, Rochester, NY, USA, 22-25 September 2012.
36. Maheswari S., Korah R., "Enhanced skin tone detection using heuristic thresholding", *Biomedical Research*, Vol. 28, No. 9, pp. 29-35, 2017.
37. Raees M., Ullah S., Rahman S. U., "VEN-3DVE: vision based egocentric navigation for 3D virtual environments", *International Journal on Interactive Design and Manufacturing*, pp. 1-11, 2018.
38. Hashemi Nazanin S., Roya B. A., Atieh S., Bayat G., Parastoo F., "Template Matching Advances and Applications in Image Analysis", *arXiv preprint arXiv:1610.07231*, 2016.
39. Varma V., Nathan-Roberts D., "Gestural Interaction with Three-Dimensional Interfaces; Current Research and Recommendations", *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, Vol. 61, No. 1, pp. 537-541, Los Angeles, CA , 28 September 2017.
40. Rocha L., Luiz V., Paulo C. P., "Image moments-based structuring and tracking of objects", *Proceedings of the XV Brazilian Symposium on Computer Graphics and Image Processing*, pp. 99-105, Fortaleza-CE, Brazil, 10-10 October 2002.
41. Neto P., Pereira D., Pires J.N., Moreira A.P., "Real-time and continuous hand gesture spotting: An approach based on artificial neural networks", *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 178-183, Karlsruhe, Germany, 6-10 May 2013.
42. Lu D., Weng Q., "A survey of image classification methods and techniques for improving classification performance", *International Journal of Remote Sensing*, Vol. 28, No. 5, pp. 823-870, 2007.
43. Rafique A., Malik K., Nawaz Z., Bukhari F., Jalbani A.H., "Sentiment Analysis for Roman Urdu", *Mehran University Research Journal of Engineering and Technology*, Vol. 38, No. 2, pp.463-470, 2019.
44. Anthony G., Gregg H., Tshilidzi M., "Image classification using SVMs: one-against-one vs one-against-all", *arXiv preprint arXiv:0711.2914*, 2007.
45. Sassano M., "Virtual examples for text classification with support vector machines", *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 208-215, Association for Computational Linguistics, 2003.
46. Osterhaus W.K., "Office lighting: a review of 80 years of standards and recommendations", *Proceedings of the Conference Record of the IEEE Industry Applications Conference Twenty-Eighth IAS Annual Meeting*, pp. 2365-2374, Toronto, Ontario, Canada, 2-8 October 1993.
47. Kerber F., Puhl M., Kruger A., "User-independent real-time hand gesture recognition based on surface electromyography," *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services*, pp.36, Vienna, Austria, 4-7 September 2017.
48. Weissmann J., Salomon R. "Gesture recognition for virtual reality applications using data gloves and neural networks", *Proceedings of the International Joint Conference on Neural Networks*, pp. 2043–2046, Washington, DC, USA, 10-16 July 1999.
49. Ramalingam B., Veerajagadheswar P., Ilyas M., Elara M.R., Manimuthu, A., "Vision-Based Dirt Detection and Adaptive Tiling Scheme for Selective Area Coverage", *Journal of Sensors*, 2018.