Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

Virtual Hippocampus: A Model for Augmenting the Learning Capability of Avatar in Virtual Environments

Muhammad Raees[1], Habib Ullah Khan[2], Member, IEEE , Inam Ur Rahman[3], and Hameed Ur Rehman[4]

[1,3] Department of Computer Science and IT, University of Malakand, Chakdara Pakistan
[2]Qater University
[4]Deptment of Computer Games Development, FCAI, Air University, Islamabad

Corresponding author: First A. Author (e-mail: author@ boulder.nist.gov).

This paragraph of the first footnote will contain support information, including sponsor and financial support acknowledgment. For example, "This work was supported in part by the U.S. Department of Commerce under Grant BS123456."

ABSTRACT The part of human brain; hippocampus, performs a critical role in the formation of memory about the objects and places traced during real-world navigation. In a Virtual Reality (VR) system, an avatar needs to have the ability to remember the objects it once discovers. To inculcate the human-like remembering capability in an Intelligent Virtual Agent (IVA), this paper presents a novel model for Virtual Hippocampus (VH). Without using the contemporary data-hungry Machine Learning classifiers, objects are learnt and recognized on the basis of distinguishing features. Moreover, the processing time is not consumed in training of trivial data; rather an IVA learns new objects and places as unfolded to it during explicit navigation. The VH is made of two dynamic data structures; *Distance with Direction* (DD) and Vector of Positions with Areas (VPA). The data structures are updated as new routes are followed or unknown objects are discovered. The IVA makes the use of VH for self-directed navigation and automated selection of any known object. The model is implemented and evaluated in a case-study project; Automated Scene Learning (ASL). The satisfactory outcome (83%) of the evaluation assures applicability of the model in intelligent VR systems.

INDEX TERMS Automation, intelligent agents, interactive systems, learning systems, Virtual Reality.

I. INTRODUCTION

Nature has gifted us with hippocampus; the curved formation in the human brain responsible for the creation of memories in general and about new places or sites in particular [1]. The places/sites once discovered are geo-tagged in the hippocampus. A complete image of the related geo-tags is then reinstated when navigation to a known place/site is required. Over the last two decades or so, VE has been setting new paradigms for man-machine interaction, training and entertainment. The rampant prevalence of VR in other domains necessitates the incorporation of human-like learning and recalling capabilities in an IVRS. The use of such intelligence-based interaction enhances the believability and realism of a synthetic world [1-3].

Avatar, as defined by Peterson, is the manifestation of self inside a VE [4]. An IVA, on the other hand, may intelligently respond to dynamic contents related queries besides its humanoid shape [5]. It is important that an avatar be intelligent like a human being and should recall whatsoever interaction it formerly accomplished. Such VR systems where IVA represents human users inside a VE are referred to as IVRS. It is the learning capability of the IVA that enhances immersivity of an IVRS [6].

This research work intends to present a model for VH, where an IVA learns the objects/sites in a VE as it moves inside a VE. A 2D frame image of the rendered scene is captured by using the glReadPixel routine [7] of the OpenGL library. In order to reduce computation, the capturing of a scene and related image processing are performed at each fifth step during navigation. To ensure this, a non-negative counter variable α is incremented with every forward step and is reset at each fifth step.

At $\alpha = 5$, the image of the rendered scene is masked into binary. After pre-processing, convex hulls around the rendered objects are drawn to accurately compute centroid and area of the objects. All the closed shaped objects in the initial rendered scene are learnt on the basis of their features; Area (A) and 3D Position (P^+). The mapping function σ transforms the 2D centroid (P) of an object into 3D position; P^+. A most-recently captured frame image Ω_i is compared with the currently captured frame Ω_{i+1} to identify new objects. The bitwise-XOR operation between Ω_i and Ω_{i+1} is performed to discover any new object in the currently rendered frame. If new objects are traced in the current frame image Ω_{i+1}, the VPA data structure is updated accordingly and the current frame is marked as most-recently; $\Omega_i = \Omega_{i+1}$. In case no new object is traced, the process is simply repeated. While reaching at the end of the scene; marked by a signboard, the VPA and DD structures are stored in the form of text files. Once learning of the scene is completed, the IVA may make the use of VH to access any of the known objects. To automatically trace a discovered object, contents of the DD and VPA structures are retrieved for the self-directed navigation and accessing of a known object.

II. LITERATURE REVIEW

No doubt, the goal of designing a fully immersive VR system is to mimic the real world too closely to be distinguished. Humans are gifted with the brain to learn and recall the details of objects while performing real-world interaction. Learning of objects on the basis of positions and shape is of vital importance in VR based health-care systems, training and education [8]. However, limited research work has been carried out about enhancing the intelligence of an IVA. The fewer systems proposed in the literature lacks the ability of autonomous learning capability. The work of [9] suggests Machine Learning (ML) algorithms to avoid avatars from bumping with the walls and virtual objects. By exploiting image processing, the algorithm in [10] steers a robot to follow the tracks marked by the special markers. The system needs proper lighting condition while performance of the algorithm depends on the size and shape of the markers. The offline navigation support approach of Jacko et al. [11] guides visitors via a virtual agent. Although the system supports dialogue based commands for interactions, the absence of learning necessitates explicit specifications of paths for navigation.

A path planning model is presented by Chaudhuri et al, [12]. The algorithm generates a central path inside a 3D object to learn paths and portions of the object. Similarly, the gaming interface of [13] incorporates entertaining features for searching and navigation based on learning. The approach utilizes data visualization elements and explicit learning of billboards for car navigation. A single agent architecture for imparting the reasoning and spoken capability in a VA is proposed in [14]. Similarly, the framework for assistance in pedagogy has also been proposed [15] where an animated agent provides information with believable actions. The use of face-to-face interactions and task-oriented dialogues enhances interactivity and believability in the domains of training and education [15]. However, the framework proposed for the agents make the use of pre-defined scripted knowledge and lacks the capability of automated learning [14], [16-19].

The task-independent cognitive architecture proposed by [20] is a valuable research for enhancing the intelligence of an agent. However, the framework does not support memory consolidation. Moreover, the agent lacks the capability to learn while interaction with the environment is going on.

The intelligence of an IVA depends on the ability to learn whereas for learning a dedicated memory architecture is required, hence this research.

III. MATERIALS AND METHODS

In the realm of VR, autonomous learning is a new paradigm with which interaction is made more engrossing and realistic. The ability of implicit learning may be used for various 3D interactions ranging from auto-selection to auto-navigation. Like the hippocampus of the human brain, an IVA ought to have a virtual hippocampus to learn the objects and routes of a VE. With this research work, we introduce the concept of the virtual hippocampus where a VE is deemed as a 2D grid. During explicit navigation of a user inside the environment, the IVA learns contents of the environment automatically. The dynamic data structures; DD and VPA of the VH are updated whenever a new route is followed or a new object is discovered.

A. PRE-PROCESSING

A rendered scene is packed into memory as a frame image (Ω) at run time by using the glStorePixels routine [21]. To obtain the corresponding contours image (C_I), adaptive thresholding [22] over the image Ω with rows $r = \{0,1,.,.,n\}$ and columns $c = \{0,1,.,.,n\}$ is performed. A frame pixel $Fp(px, py)$ at row px and column py in the Ω is replaced by a thresholded pixel $T_n(px, py)$ if the intensity of $T_n(px, py)$ is less than a constant τ. The value of τ is computed as,

$$\mu_b = \frac{\sum_{r=1}^{n}\sum_{c=1}^{m} Fp(r,c) \in Background}{r*c} \tag{1}$$

$$\mu_f = \frac{\sum_{r=1}^{n}\sum_{c=1}^{m} Fp(r,c) \in Foreground}{r*c} \tag{2}$$

$$\tau = \frac{\mu_b + \mu_f}{2} \tag{3}$$

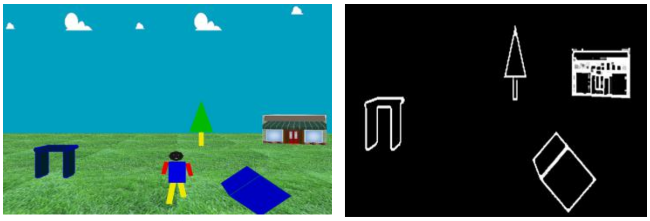
The contours image $C_I = \left\{T_p(px, py)\right\}_{px=0,py=0}^{px=r,py=c}$ from the frame image Ω is obtained as,

$$T_p(px,py)=\begin{cases}0 & Fp(px,py)<\tau\\ 1 & Otherwise\end{cases}$$

The morphological operation [23] is performed over the image C_I by a structuring element $\pi_{(5\times5)}$ to compact the image for onward processing.

$$C_I \circ \pi = (C_I \cdot \overline{\pi}) \oplus C_I \quad (4)$$

A frame image Ω and the corresponding C_I image are shown in Fig. 1.



(a) (b)

FIGURE 1. The (a) frame image and (b) the contours image.

B. OBJECT IDENTIFICATION

After finding out the contours of a rendered scene, closed objects are identified by convex hull [24]. This is not only to identify objects but to ignore the inner trivial parts or sub-objects. The set of pixels $\chi=\{p_1,p_2,..,p_n\}$ representing the points $\{p_i\}_{i=1}^{n}$ enclosed by the convex hull; ξ around an object is given as,

$$\xi=\left\{\sum_{i=1}^{|\chi|}\gamma_i p_i \mid \forall_i:\gamma_i\geq 0 \wedge \sum_{i=1}^{|\chi|}\gamma_i=1\right\} \quad (5)$$

where γ is the weight constant.

To obtain the feature set $S=\{(A_1,P^+_{1(x,y,z)}),(A_2,P^+_{2(x,y,z)}),..,(A_n,P^+_{n(x,y,z)})\}$ in a rendered frame, the convex hull $\{\xi_i\}_{i=1}^{n}$ of the n objects are computed in the contours image C_I . The two lightweight features; area (A) and the 2D position (P) are computed using the image moments [25-26].

$$A=\mu_{0,0}=\sum_{x=0}^{\xi.rows}\sum_{y=0}^{\xi.columns}\xi_{(x,y)} \quad (6)$$

$$\mu_{1,0}=\frac{\sum_{x=0}^{\xi.rows}x\xi_{(x,y)}}{\mu_{0,0}} \quad (7)$$

$$\mu_{1,0}=\frac{\sum_{y=0}^{\xi.columns}y\xi_{(x,y)}}{\mu_{0,0}} \quad (8)$$

$$P(x,y)=\left(\frac{\mu_{1,0}}{\mu_{0,0}},\frac{\mu_{0,1}}{\mu_{0,0}}\right) \quad (9)$$

C. MAPPING

The transformation the 2D position (P) of an object in C_I to its equivalent 3D position (P^+) in the OpenGL window is performed by a projection function σ . The identity matrix M is multiplied with the matrix representation of P ; Mp to get P^+ in OpenGL window having width W and height H . As the z-coordinate of all the objects in an image frame C_I is the same, therefore the z-position of the virtual camera (Z_{vc}) is assigned directly to the objects of the same frame; $P_z^+=Z_{vc}$.

$$M=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix} \quad (10)$$

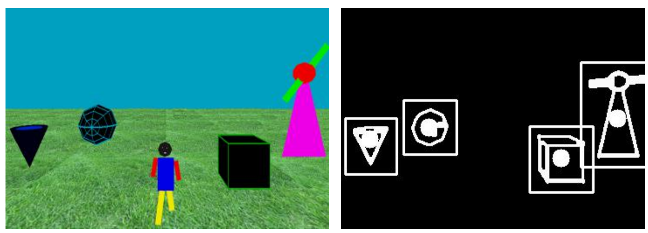
$$Mp=\begin{bmatrix}Px\\Py\\1\\0\end{bmatrix} \quad (11)$$

$$P^+=\sigma(Mp,M) \quad (12)$$

$$P^+=\left(W*\frac{\frac{\sigma_x}{\beta}+1}{2},H*\frac{\frac{\sigma_x}{\beta}+1}{2},P_Z^+\right) \quad (13)$$

were $\sigma_x=\sum_{j=0}^{3}(M_{0,j}*Mp_{0,j})$, $\sigma_y=\sum_{j=0}^{3}(M_{1,j}*Mp_{1,j})$ and $\beta=\sum_{j=0}^{3}(M_{3,j}*Mp_{3,j})$

The lightweight features $P^+_{(x,y,z)}$ and A are used to learn the 3D position and area of an object. As the centroid and area of the convex hull around an object are used, therefore the trivial spots within the objects are simply ignored. The convex hulls and centroids of the detected objects in the C_I is shown in Fig. 2. The features of the discovered objects are stored in the VPA structure with unique indexes.



(a) (b)

FIGURE 2. The (a) frame image and (b) the contours image with convex hulls and centroids.

D. THE VPA DATA STRUCTURE

The VPA structure is defined as a dynamic vector of vectors to store the features of traced objects, see Fig. 3. The structure is updated with the discovery of each new object. To avoid duplicate entries in the VPA, unique entries in the VPA is ensured before pushing in the data structure. The four features ($A_X,P^+_{X.x},P^+_{X.y},P^+_{X.z}$) of a discovered object X are checked sequentially with the existing entries of the VPA.

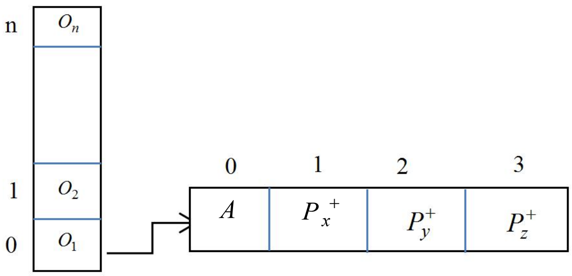


FIGURE 3. The VPA data structure.

If the features are not present in the VPA, then the VPA is resized to accommodate features of the newly discovered object. In case the features are already present, the object is ignored for learning and no pushing is performed in the VPA. The process is shown in Fig. 4.

E. THE DD GRAPH

The DD graph is made of nodes $N_d \in D_1$ and edges $E_g \in D_2$ where D_1 and D_2 are the direction and distance data structures. The DD graph is constructed during explicit exploration of the VE which is followed for automatic navigation to access objects of a VE. The graph is updated dynamically with forward navigation in the designed VE. It is assumed that a user follows the optimal and free from collision route during the navigation in a grid of voxels [27]. Practically the DD structure is implemented by two arrays; direction (D_1) and distance (D_2).

$$DD = (D_1, D_2) \tag{14}$$

While initiating a move in a particular direction (Straight, Left or Right), a character constant 'S', 'L' or 'R' is assigned respectively to the D_1 array.

$$D_1 = \{N_{d_1}, N_{d_2}, .,., N_{d_n}\} \tag{15}$$

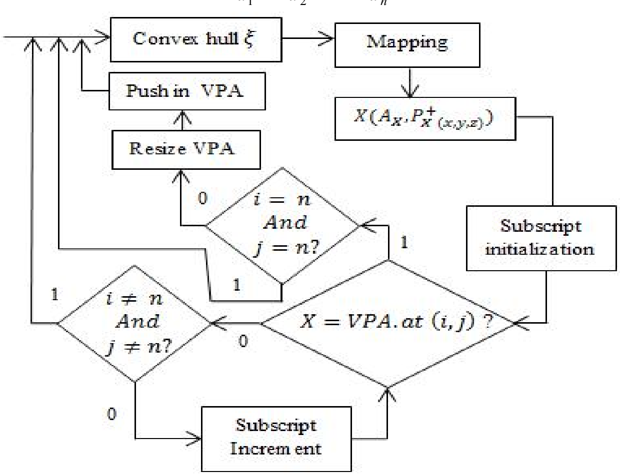


FIGURE 4. The process of updating VPA with unique entries.

The nodes represent direction of the following edge where an edge represents distance/length between a node N_{d_x} and the following node $N_{d_{x+1}}$.

If ' d ' is to represent Euclidian distance between the vertices, then the distance data structure is given as,

$$D_2 = \{d(N_{d_i}, N_{d_{i+1}}), .,., d(N_{d_{n-1}}, N_{d_n})\} \tag{16}$$

where d is the distance between a starting voxel (S_v) and a final voxel (F_v) and is given as,

$$d(S_v, F_v) = \sqrt{(F_v - S_v)^2} \tag{17}$$

Deeming the whole virtual scene as a grid of voxels; $V = \{v_1, v_2, .,., v_n\} \mid v_i \in R^3$, d is dynamically calculated when a turn is made or when the IVA reaches the endpoint. In such cases, the graph is updated by fixing distance of the previous edge $E_g \in D_2$ and adding a new node into D_1. The DD graph formed by going $S \to R \to S \to L \to S$ in a 3D grid; see Fig.5, is shown in Fig. 6.

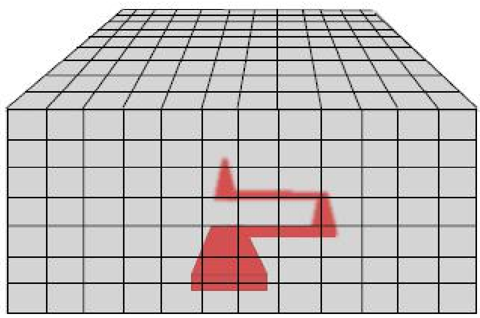


FIGURE 5. The path followed during explicit navigation in the grid of voxels.

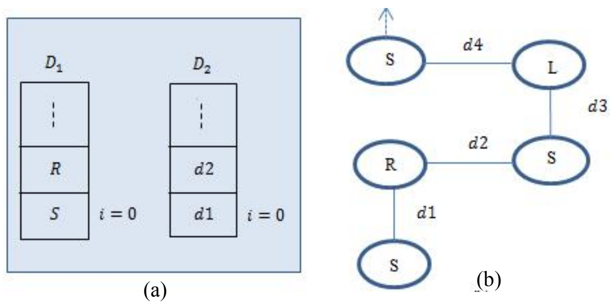


FIGURE 6. The (a) D1 and D2 arrays representing directions and distances and (b) the DD graph formed on the basis of D1 and D2 data structures.

The entire process of the model is shown in Fig. 7.

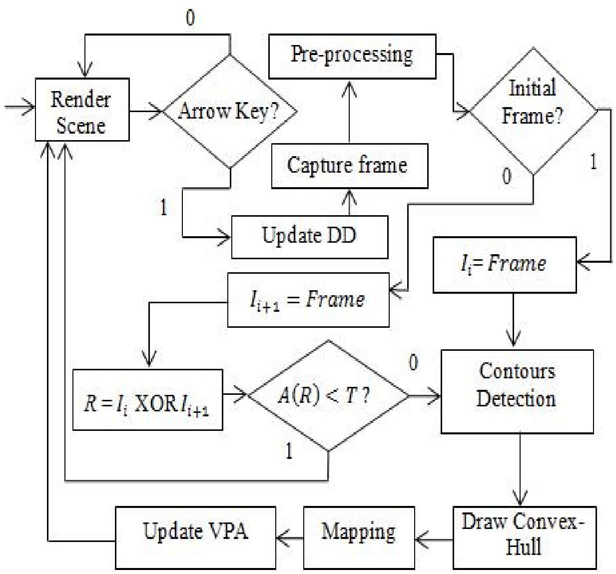


FIGURE 7. Schematic of the proposed model.

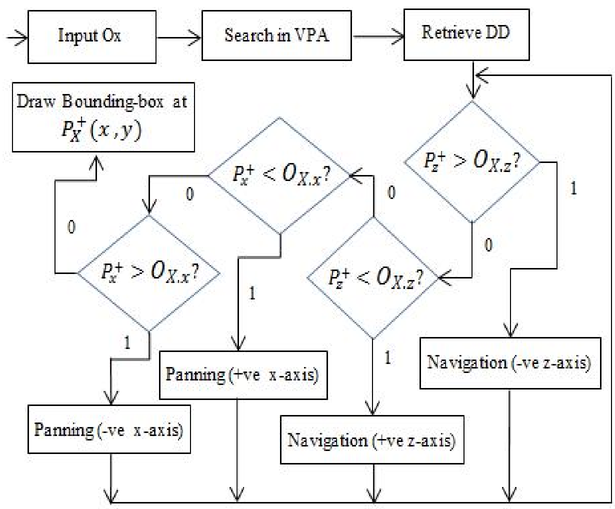


FIGURE 8. The process of automated navigation and panning till the location of a known object in the VE.

F. LOCATING A KNOWN OBJECT

Once the knowledge base is formed, an IVA can access any of the known objects by using its VH. Searching for the object in the VPA is performed on the basis of object name; O_i where $i = \{1,2,..,,O_n\}$. An input instance O_x is searched sequentially in the VPA till a matching entry $O_i \mid O_i = O_x$. At the successful matching, contents of the VPA; $(A_X, P^+_{X.x}, P^+_{X.y}, P^+_{X.z})$ are then followed for the autonomous navigation and selection. A known object X located at $P^+_{X.z}$ along the z-axis is accessed by performing inside navigation in a VE to the location $P^+_{X.z}$. All the frames from a point $P_t \mid P_t > P^+_{X.z} \vee P_t > P^+_{X.z}$ are displayed with a constant delay rate $\Psi = 400ms$. Moreover, horizontal panning is performed to bring the point $P^+_{X.x}$ to the clipping area. At $P^+_{X.z}$, the auto-navigation is stopped and the bounding box with the area A around the object and centroid $P^+_X(x,y)$, is selected. The process is schematically depicted in Fig. 8.

IV. IMPLEMENTATION AND EVALUATION

The VH model is implemented in a VS project; ASL using a Corei5 laptop with 2GB RAM and a built-in Radian graphics card. The immersive environment of ASL is designed in OpenGL. The OpenCV routines are used at the backend for image processing. The environment, as shown in Fig. 9, contains an avatar and various other 3D objects with different shape, size and textures. Besides navigation and panning [28], the arrow keys (left, right, up and down) are used to control movements of the IVA.

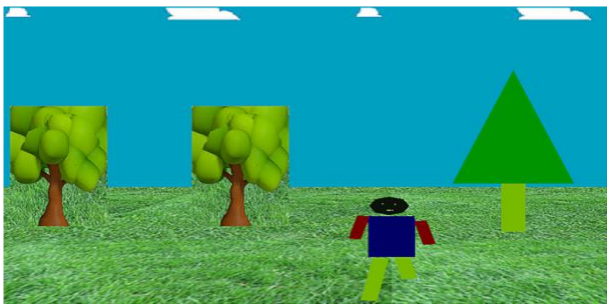


FIGURE 9. A scene of the VE with the avatar (IVA).

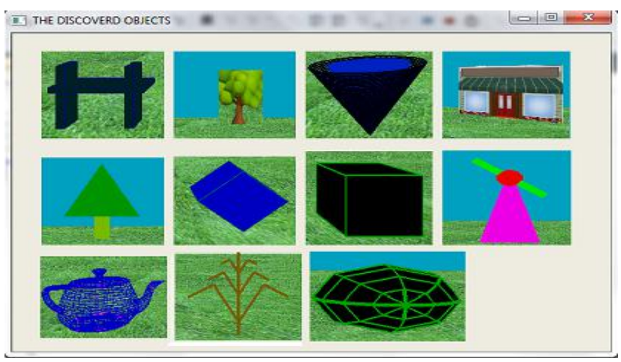


FIGURE 10. The discovered objects in a single image.

After every fifth step during the explicit navigation, the Trace_Objects module of the project is activated to learn objects in the newly rendered scene. After learning of the objects, an audio beep of 700MHz and 600ms delay is played to inform the user. The sections from the RGB frame image containing the objects are segmented out to make a list of known objects. To get input from the user about an object to be traced, the list of known objects is presented in an image form, see Fig. 10.

A user needs to click over the required object to trace the object in the VE. At the mouse-up event, the image presenting the list of known objects is disappeared while the appropriate object name (O_i) is searched in the VPA data structure. After getting the entries of O_i in the VPA, automated navigation towards the required object is initiated. Reaching at the position of the object in the VE, the required object is selected by highlighting a bounding box around the object. To search a new object or to reset the system to the beginning, the 'r' keyboard key needs to be pressed. With the pressing of the escape key, the system quits saving the DD and VPA in two separate text files.

The case-study project; ASL is designed with a total of 11 objects at different locations inside a 3D digital world. Six of the objects are solidly filled, two with surface textures while the remaining three objects are wired 3D objects. Six participants; ages between 21 and 42, all male evaluated the ASL in the university research lab. The participants were arranged into three groups, consisting of two users in each group. To assess facet-based learning of the objects from different perspectives, participants were asked to follow three different routes; Path-1, Path-2 and Path-3 see Fig. 11. While navigating on Path-1, the objects are learnt on the basis of frontal facets. By moving on Path-2 and Path-3, objects are learnt from the left and right facets respectively, see Fig. 12. All the paths meet at one point represented by a signboard displaying the text "Stop".

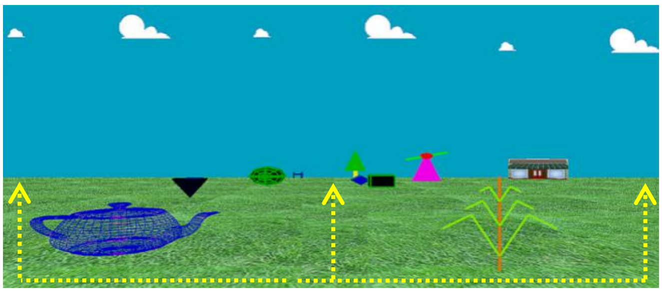
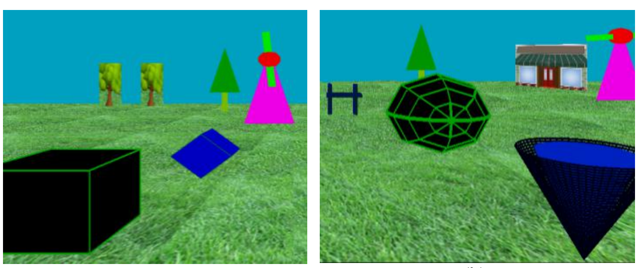


FIGURE 11. The three routes for explicit navigation in the VE.



(a) (b)

FIGURE 12. Views of the scene while going on (a) path-2 and (b) path-3.

By going through Path-1, all the objects are traced and recognized properly. However, as on Path-2 and Path-3, some objects occlude other objects partially or fully. Due to that reason, objects on Path-2 and Path-3 were not properly traced/recognized. Outcomes of the evaluation are shown in Table 1.

TABLE I
UNITS FOR MAGNETIC PROPERTIES OOUTCOMES OF THE EVALUATION WITH ACCURACY IN %AGE

	Objects Traced & Recognized	Not Traced	Traced but Not Recognized	Total	Accuracy (%)
Group-1	22	0	0	22	100
Group-2	17	3	2	22	77
Group-3	16	4	2	22	72
Total	55	7	4	66	83

V. DISCUSSION

The superiority of human intelligence over AI systems is mainly due to the human ability to remember the facts once perceived. The ability to remember the knowledge of the previously seen objects is invaluable in performing actions in present. The hippocampus of the brain is involved in storing and retrieving of scene-related information [29]. As an IVA facilitates interactions in a VE as a mentor or a guide [14], therefore it should have the human-like cognitive ability. However, for mimicking any intelligent behaviour, dedicated memory architecture is primarily required [30]. The goal of this research work is to imbue an IVA with hippocampus so that it may learn and remember the contents of a VE. The novel concept of VH is implemented in a VS project by making the use of the OpenCV and OpenGL libraries. In the designed VE, an IVA learns by itself whenever something new is sensed during exploration. After capturing an object for learning, the distinguishing lightweight features are stored for future use. As it is possible that the same object may be present at a different location, or different objects may be present at the same 2D location, therefore, all the four features are checked to avoid duplicate entries in the VH. Besides learning the objects, the paths followed during explicit navigation are also learnt for self-directed navigation. By giving an appropriate cue (name or number of an object), the past information in the VH are retrieved for interaction. In a nutshell, the cognitive process involved in learning and remembering is simulated in a systematic way. The model provides a baseline for intermingling intelligence in VR system, particularly for task automation inside a VE.

VI. CONCLUSION AND FUTURE WORK

We have presented a model to raise the learning capability of an IVA inside a VR system. During explicit navigation in a VE, an IVA learns new objects as unfolded to it. A detected unknown object is learned on the basis of its area and centroid. With two dynamic data structure, the VH of an avatar is augmented as a user navigates inside the VE. The formed VH is then used for autonomous navigation and selection of any known object in the VE. Without using any of the data-hungry ML classifiers, the model is implemented and evaluated in a case-study project. The satisfactory 83% average accuracy rate for autonomous selection and self-directed navigation testifies applicability of the model. The system, however, ignores objects inside an object for learning. With little effort, the model can be used in diverse application areas like the exploration of 3D medical images, simulations and decision making in VR games. The path-finding approach introduced can be followed for simplifying navigational tasks in complex virtual structures like that of brain and galaxies. To overcome the challenge of learning objects inside objects, we are determined to enhance the system so that nested objects with minute details can be learned and recognized. It is also planned to enhance the model such that an IVA may share its knowledge about a VE with other IVAs.

ACKNOWLEDGMENT

The authors are thankful to the chairman and staff of the Department of Computer Science and IT, University of Malakand, for extending all their possible support. We are also thankful to the students who earnestly participated in the evaluation process.

.

REFERENCES

[1] E. A. Maguire et al., “Navigation-related structural change in the hippocampi of taxi drivers,” in Proceedings of the National Academy of Sciences, vol. 97, no. 8, pp. 4398–4403, 2000. [Online]. Available: http://doi.org/10.1073/pnas.070039597

[2] J. F. Dunagan, “Neuro-futures: The brain, politics and power,” Journal of Futures Studies, vol. 9, no. 2, pp.1-18., 2004

[3] J. Lugrin, M. Cavazza, M. Palmer, S. Crooks, “AI-mediated interaction in virtual reality art," in International Conference on Intelligent Technologies for Interactive Entertainment, pp. 74-83, Springer, Berlin, Heidelberg, 2005. [Online]. Available: http://doi.org/10.1007/11590323_8

[4] M. Peterson, “Learning interaction in an avatar-based virtual environment: a preliminary study,” PacCALL Journal, vol. 1, no. 1, pp. 29–40, 2005.

[5] R. Aylett, M. Cavazza, “Intelligent Virtual Environments-A state-of-the-art report,” in Eurographics Conference, Manchester, UK, Sep 2001.

[6] J. Bates, “The role of emotion in believable agents,” Communications of the ACM, vol. 37, no. 7, pp.122-125, 1994. [Online]. Available: http://doi.org/10.1145/176789.176803

[7] J. G. Hagedorn, J. E. Terrill, A. P. Peskin, J. J. Filliben, “Methods for quantifying and characterizing errors in pixel-based 3D rendering,” Journal of research of the National Institute of Standards and Technology, vol. 113, no. 4, p.221, 2008. [Online]. Available: http://doi.org/10.6028/jres.113.017

[8] Q. Chen, A. S. M. M. Rahman, A. El-sawah, X. Shen, A. El Saddik, N. D. Georganas, “Access Learning Objects in Virtual Environment by Hand Gestures and Voice,” in 3rd Annual E-learning Conference on Intelligent Interactive Learning Object Repositories (I2LOR), Montreal, Quebec, 2006.

[9] M. Luck, R. Aylett, “Applying artificial intelligence to virtual reality: Intelligent virtual environments,” Applied Artificial Intelligence, vol. 14, no. 1, pp.3-32, 2000. [Online]. Available: http://doi.org/10.1080/088395100117142

[10] E. Rivas, K. Komagome, M. Mitobe, G. Capi, “Image-Based Navigation for the SnowEater Robot Using a Low-Resolution USB Camera,” Robotics, vol. 4, no. 2, pp.120-140, 2015. [Online]. Available: http://doi.org/10.3390/robotics4020120

[11] J. Jacko, J. van Luin, H. J. op den Akker, A. Sears, A. Nijholt, “A dialogue agent for navigation support in virtual reality,” in SIGCHI Conference on Human Factors in Computing Systems, Anyone, Anywhere, March 2001. [Online]. Available: http://doi.org/10.1145/634067.634138

[12] P. Chaudhuri, R. Khandekar, D. Sethi, P. Kalra, “An Efficient Central Path Algorithm for Virtual Navigation,” in Proceedings of Computer Graphics International, pp. 188-195, 2004.

[13] M. Rahman, A. El Saddik, “Eye & why: A prototype for learning objects visualization in virtual environment,” in Eighth IEEE International Symposium on Multimedia (ISM'06), pp. 787–789, 2006. [Online]. Available: http://doi.org/10.1109/ISM.2006.76

[14] J. Rickel, “Intelligent Virtual Agents for Education and Training: Opportunities and Challenges,” in International Workshop on Intelligent Virtual Agents, pp. 15–22, 2001. [Online]. Available: http://doi.org/10.1007/3-540-44812-8_2

[15] W. L. Johnson, J. W. Rickel, J. C. Lester, “Animated pedagogical agents: Face-to-face interaction in interactive learning environments,” International Journal of Artificial intelligence in education, vol. 11, no. 1, pp. 47–78, 2000.

[16] K. Ahmed, “Animated pedagogical agents in web-based intelligent learning environment,” Cingapura[sn]. Disponível em< http://www. sp. edu. sg/projects/tjournal/doc/kazi301102. pdf, 2005.

[17] C. Elliott, J. Brzezinski, “Autonomous Agents as Synthetic Characters,” AI magazine, vol. 19, no. 2, p. 13, 1998. [Online]. Available: https://doi.org/10.1609/aimag.v19i2

[18] C. Elliott, J. Rickel, J. Lester, “Lifelike Pedagogical Agents and Affective Computing: An Exploratory Synthesis,” in Artificial intelligence today, pp. 195–211, 1999. [Online]. Available: http://doi.org/10.1007/3-540-48317-9_8

[19] G. Falloon, “Using avatars and virtual environments in learning: What do they have to offer?,” British Journal of Educational Technology, vol. 41, no. 1, pp. 108–122, 2010. [Online]. Available: http://doi.org/10.1111/j.1467-8535.2009.00991.x

[20] M. Nuxoll, J. E. Laird, “Enhancing intelligent agents with episodic memory,” Cognitive Systems Research, vol. 17, pp. 34–48, 2012. [Online]. Available: http://doi.org/10.1016/j.cogsys.2011.10.002

[21] I. Elhassan, “Fast texture downloads and readbacks using pixel buffer objects in opengl,” nVidia Technical Brief., 2005.
[22] S. Ayas, H. Dogan, E. Gedikli, M. Ekinci, “A Novel Approach for Bi-Level Segmentation of Tuberculosis Bacilli Based on Meta-Heuristic Algorithms,” Advances in Electrical and Computer Engineering, vol. 18, no. 1, pp.113-121, 2018. [Online]. Available: http://doi.org/10.4316/AECE.2018.01014
[23] K. Sreedhar, B. Panlal, “Enhancement of images using morphological transformation,” arXiv preprint arXiv:1203.2514, 2012. doi.org/10.5121/ijcsit.2012.4103.
[24] G. Mei, N. Xu, “Cudapre3d: an alternative preprocessing algorithm for accelerating 3D convex hull computation on the GPU,” Advances in Electrical and Computer Engineering, vol. 15, no. 2, pp.35-44, 2015. [Online]. Available: http://doi.org/10.4316/AECE.2015.02005
[25] C. Stolojescu-CriŞan, S. Holban, “A comparison of X-ray image segmentation techniques,” Advances in Electrical and Computer Engineering, vol. 13, no. 3, 2013. [Online]. Available: http://doi.org/10.4316/AECE.2013.03014
[26] C. Elena-Gina, et al. "Novel interface for simulation of assembly operations in virtual environments." Advances in Electrical and Computer Engineering vol. 13, no.1, pp. 47-53, 2013. [Online]. Available: http://doi: 10.1145/1294211.1294238
[27] S. Denis, F. Novak, B. Zalik, "A Fast Method for the Alignment of the Displacement of Voxel Data." Advances in Electrical and Computer Engineering, vol. 12, no. 2, pp. 41-46, 2012. [Online]. Available: http://doi: 10.4316/AECE.2012.02007
[28] M. Raees, S. Ullah, “GIFT: Gesture-Based Interaction by Fingers Tracking, an Interaction Technique for Virtual Environment,” IJIMAI, vol. 5, no. 5, pp.115-125, 2019. [Online]. Available: http://doi.org/10.9781/ijimai.2019.01.002
[29] S. Larry et al., “Role of the hippocampus in remembering the past and imagining the future," in: Proceedings of the National Academy of Sciences, vol. 107, no. 44, pp. 19044-19048, 2010. [Online]. Available: http://doi: 10.1073/pnas.1014391107
[30] J. E. Wang, Y. Laird, Y. Wang, J. E. Laird, “Integrating Semantic Memory into a Cognitive Architecture,” Ann Arbor, MI: University of Michigan Center for Cognitive Architecture, 2006. [Online]. Available: http://doi=10.1.1.118.7022

Dr. MUHAMMAD RAEES is Associate Professor of Computer Science serving the Department of Higher Education KPK, Pakistan since 2007. His research interest includes Virtual Reality, Image Processing and Machine Learning.

Dr. HABIB ULLAH KHAN is working as a Professor of MIS in the Department of Accounting & Information Systems, College of Business and Economics, Qatar University, Qatar. He completed his Ph.D. degree in Management Information Systems from Leeds Beckett University, UK. He has nearly 25 years of industry, teaching and research experience. His research interests are in the area of IT Adoption, Social Media, Internet Addiction, Mobile Commerce, Computer Mediated Communication, IT Outsourcing, Big data, and IT Security.

Dr. INAM UR REHMAN is subject specialist in information technology at Elementary and Secondary Education Department, KP, Pakistan since 2014. He received his BS, M. Phil, and PhD degree in Computer Science from the department of Computer Science & IT, University of Malakand in 2006, 2016, and 2022, respectively. His research interests include computer graphics, virtual, augmented reality, and 3D HCI.

HAMEEDUR RAHMAN received the BS MS degree in software engineering with multimedia from the Limkokwing University of Creative Technology, Malaysia and the Ph.D. degree in Computer Science from the Faculty of Information Science and Technology, University of Kebangsaan Malaysia, Bangi, Malaysia. He has 10 years of Industrial experience in multinational companies. He is currently a Senior Assistant Professor at AIR University and senior Research Member with the Center for Artificial Intelligence Technology and Cluster Head of gaming and multimedia. He received some international awards including the Itex Bronze Medal of Innovation, Malaysia and the Winner of NASA Space App and Virtual Reality Arena. His research interests include the area of augmented reality, virtual reality, imagine processing, cryptography, Natural Language Processing, Data Mining, mobile technology, artificial intelligent, automation, and medical radiological technology.