Navigation System of 3D IIT Madras Campus Using a Car

Sweta Kumari (BT17D019)

INTRODUCTION

There are several studies which uses the virtual environments for navigation task and train the agent to get familiar with the environment using Deep reinforcement learning. Most of the studies uses simple geometric primitives: cubes and rectangles to make the walls. Péruch et al. proved that spatial information has been acquired through active exploration and that acquired spatial knowledge is superior to either the dynamic passive or dynamic static exploration. Witmer et al. showed that spatial knowledge learned in the virtual environment can be transferred to the real world. A cultural place of virtual educational field trip is perfect to engage real world teachers and students in a social group. Google DeepMind research on the learning navigation in complex environment has shown that 3D virtual environment is a best environment, where agent can interact and being trained by the deep reinforcement learning successfully. In self-driving cars, car is considered to be the navigating agent and it has to learn properly to navigate in the city in large scale region. Agent learns by performing a recursive task in the virtual environment. Agent first observes the current state of the environment and then takes the action based on the observation. After taking the action in the environment by the agent, agent gets a reward based on the difference between actual action and expected action at that particular state. This reward has been provided by the critic of the environment. Once the agent perform the action in the environment, the environment goes to the next state. To accomplish the better performance of the task of agent's learning to navigate in the environment, complexity of the environment needs to be increased in step by step. This work present the navigation system of a Indian Institute of Technology Madras (IITM) Campus using a Car, where we can do the study of making the car learn to navigate the campus without any input control, by setting a goal destination using Deep Reinforcement Learning Algorithms.

PROBLEM DEFINITION

There are many research disciplines and technology domains which are being very attracted towards the navigation task like, neuroscientist wants to crack the code of grid cells and place cells and also they want to model a mobile robots that can reach the given destination. Agent learns to navigate directly from the visual inputs has been shown to be possible in some domains such as Deep reinforcement learning approaches which learns from the reward. Some recent research has proved that reinforcement learning agents can learn to navigate house scenes (Zhu et al., 2017; Wu et al., 2018), mazes (e.g. Mirowski et al. 2016), and 3D games (e.g. Lample & Chaplot 2017). Despite of many successes, Deep reinforcement learning approaches are badly inefficient to the data and sensitive to the stimulus of the environment. Theses domains are well known for their successes in games and environments with perturbation than in real life applications. But from here the questions has been raised that whether these domain can be used for the large scale navigation based on visual real world images. Therefore it needs to be investigated in the future. So for that this work presents a model of "Navigation System of 3D Indian Institute of Technology Madras Campus using a Car".

METHODS

To model a navigation system, it is very important to make a perfect 3D map of the street and real looking building structures in the environment to give the agent visual inputs and cues to memories the area where the agent has been visited already.

Modelling the IITM Campus

To develop a navigation system with many stimulus so that agent can learn to navigate in the environment with only visual inputs, it is very important to make a very unique design. I used City Engine tool to model the buildings, streets and terrain of the IITM Campus. City Engine is a very advanced tool for modelling a 3D environments. This tool can make all manual works automated if we are going to model our imaginary environment. But if we want to model a copy of a real world environment using this tool then most of the works will become manual. Therefore to model the IITM campus, street maps has been simulated manually with proper roads as the original campus have. Detail information of the exact map is collected from the Open Street Map. The roads are having a sidewalks for the pedestrian so that agent or car should learn to not to hit the pedestrians walking on the sidewalks. Each and every building has been structured manually to give it the same outline structure of the buildings as it is in the real world IITM campus. Each building is having its name plate as well, to get the visual input for the learning agent by interacting in the environment. The following figure-1 and 2 shows the view of the IITM campus from the direction of Krishna and Taramani Gate with sky and without sky respectively.

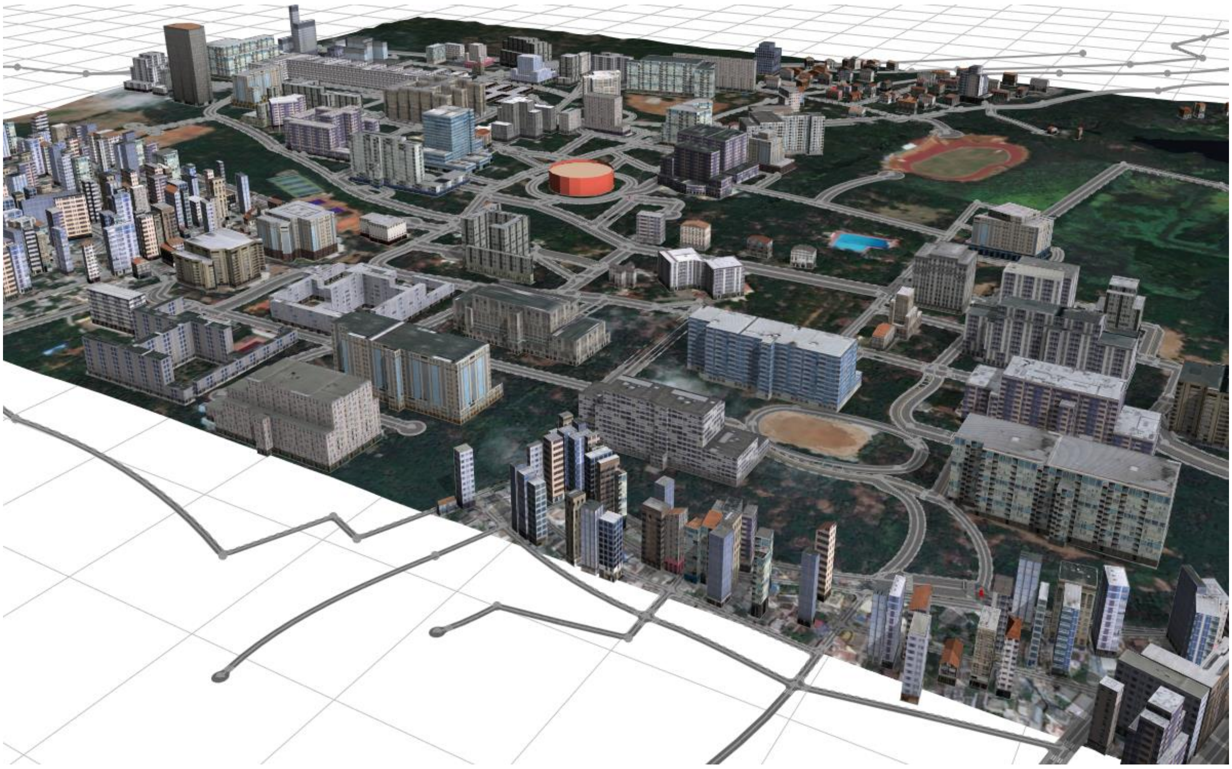

Figure-1: IITM Campus with all buildings, streets, sports field, and stadium without sky

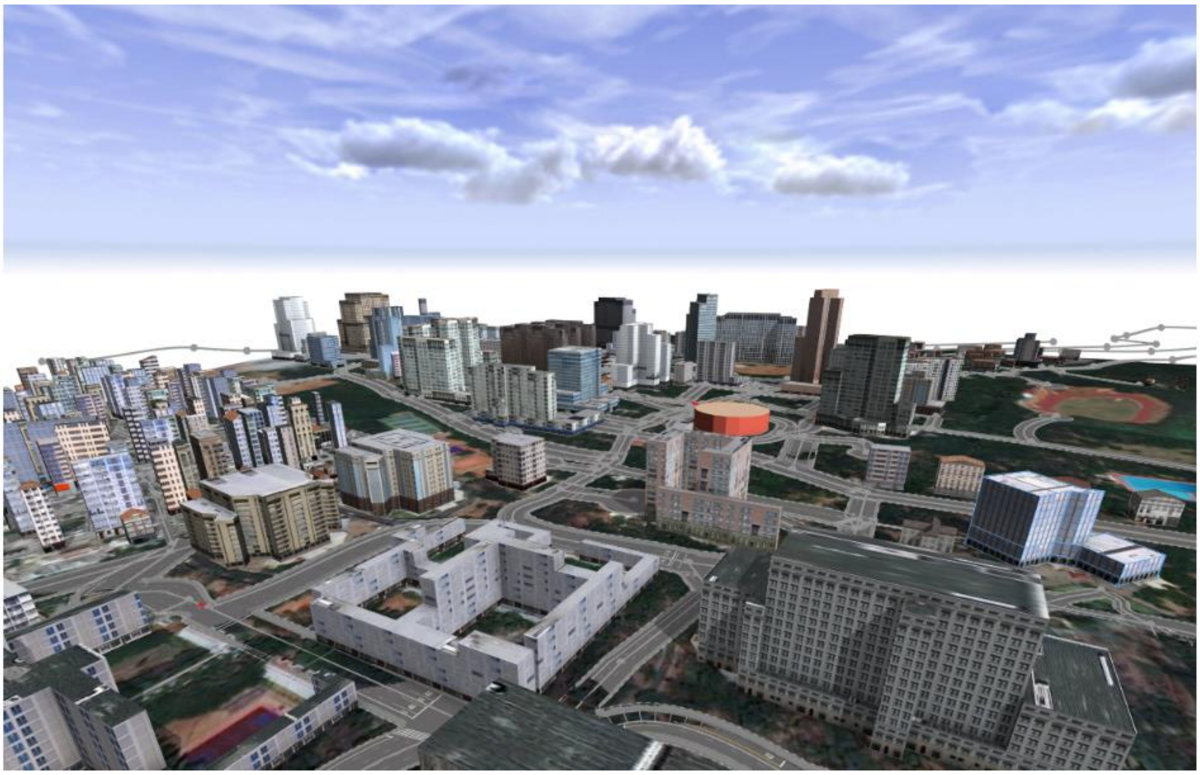

Figure-2: IITM Campus with all buildings, streets, sports field, and stadium with sky

So here we can see that almost exact structure of the streets, roads, outline of the buildings of hostels, departments, guest house, sports complex, open air theatre circle in red colour, and stadium with red boundary around it. Also we can see the greeneries over all of the campus.

Importing the modelled campus in Unity

After modelling the 3D IITM campus, this structure has been exported from City Engine to Unity. After this, to make the campus more realistic some deers, trees, cycles and students on cycling has been simulated in Unity. To model the trees, cycles with different varieties and students riding cycles, 3D paint has been used. The result is shown in figure-3.

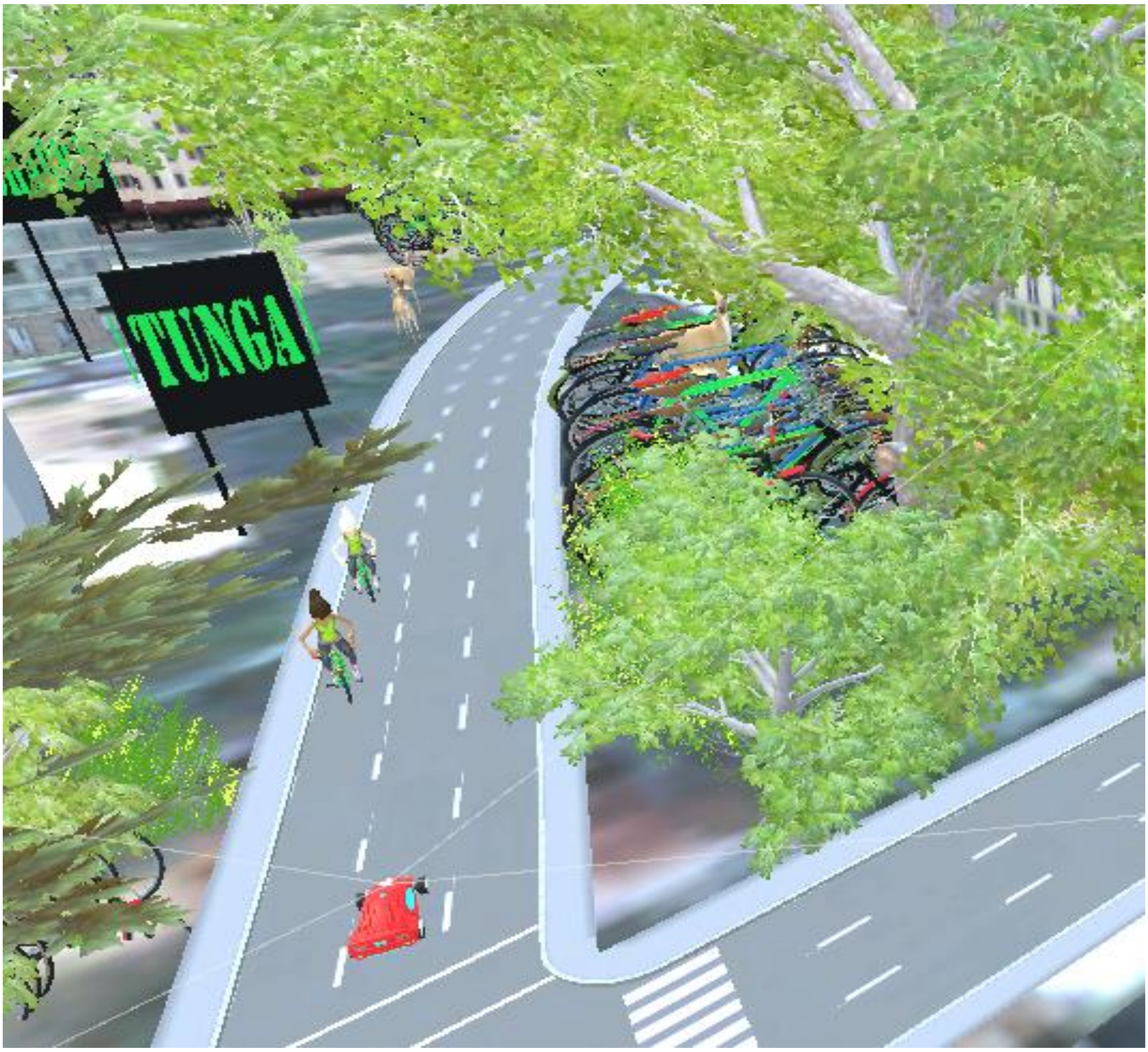

Figure-3: IITM Campus with trees, deers, cycles on the parking area, students riding the cycles and one red car in the road near the Tunga Hostel

Modelling the car in the campus with arrow keys control

I modelled a car in Unity and putted in the front of the Tunga Hostel in the campus. I wrote the C sharp script to control the car with four arrow keys in the keyboard from car user. When a user starts the car by pressing the forward arrow key then it plays one background music as well as ride the car. I have added one brake button (blue and white in colour) in the front lower middle area of the car (red colour), which is shown in figure-4.

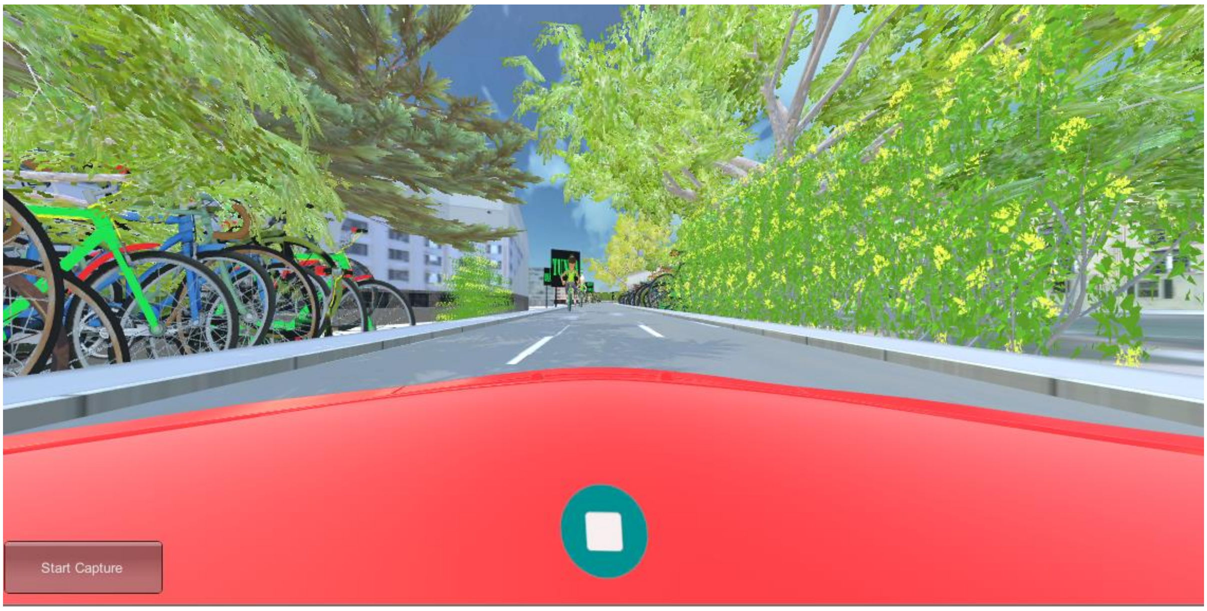

Figure-4: Driver's view from the car during driving the car with Brake button (green on the middle)

Colliders: Colliders are a component that allows the game object they're attached to react to other colliders provided that one of the game objects has a rigid body component attached. Colliders come in various shapes and types, and are denoted in the scene view by a green outline.

UI Canvas: The Canvas is the root component for rendering all UI objects in Unity.

UI Buttons: These buttons are layered over the top of the interface and draw attention to promoted or primary actions. They act as call to action buttons ("used for a promoted action"), meant to represent a single action users perform the most on that particular screen. I used UI buttons to perform a function in the environment.

Recording the video of driver's view from the car during driving

I assumed the camera as a driver and it is driving the car. I simulated the camera in such a way that it will be always in driver's seat of the car when user will ride the car from the arrow keys of the keyboard. To get a video of the complete ride of the car in running the scene from user, I used video capture with written code to control it. Thus, when user run the scene then it will ask the user to choose for the capture of the video, once user select this option then it will start to make a video recording mp4 file of the driver's view in a separate folder.

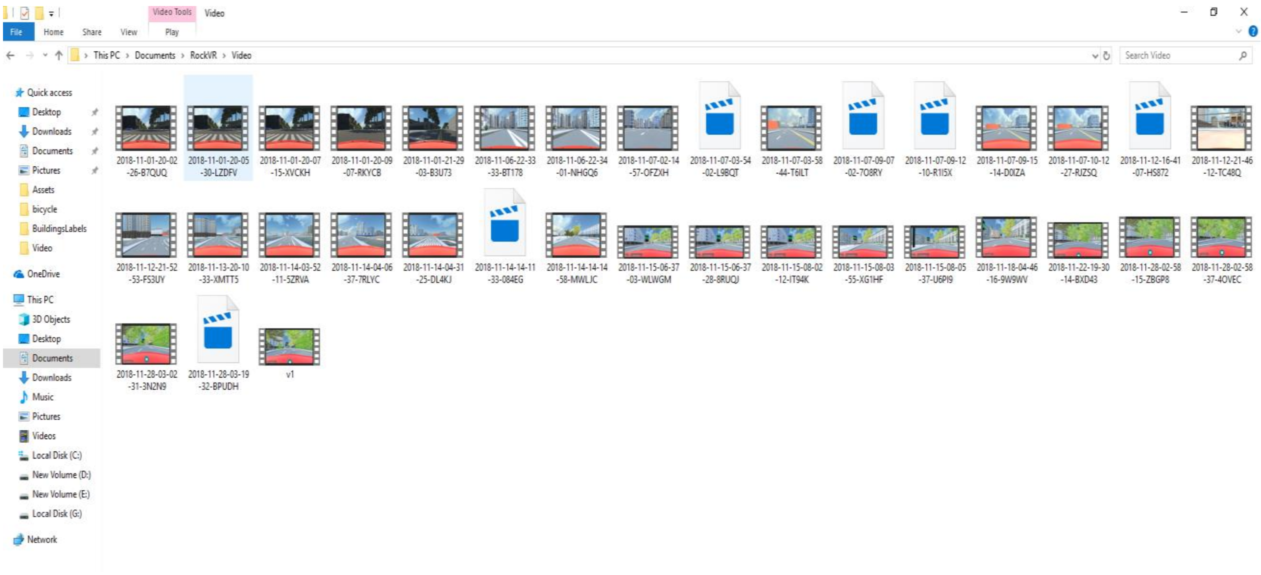
Video Tools
Video
File
Home
Share
View
Play
This PC › Documents › RockVR › Video
Search Video
Quick access
Desktop
Downloads
Documents
Pictures
Assets
bicycle
BuildingsLabels
Video
OneDrive
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
New Volume (D:)
New Volume (E:)
Local Disk (G:)
Network
2018-11-01-20-02-26-B7QUQ
2018-11-01-20-05-30-LZDFV
2018-11-01-20-07-15-XVCKH
2018-11-01-20-09-07-RKYCB
2018-11-01-21-29-03-B3U73
2018-11-06-22-33-33-BT178
2018-11-06-22-34-01-NHGQ6
2018-11-07-02-14-57-OFZXH
2018-11-07-03-54-02-L9BQT
2018-11-07-03-58-44-T6ILT
2018-11-07-09-07-02-7O8RY
2018-11-07-09-12-10-R1I5X
2018-11-07-09-15-14-D0IZA
2018-11-07-10-12-27-RJZSQ
2018-11-12-16-41-07-HS872
2018-11-12-21-46-12-TC48Q
2018-11-12-21-52-53-FS3UY
2018-11-13-20-10-33-XMTT5
2018-11-14-03-52-11-5ZRVA
2018-11-14-04-06-37-7RLYC
2018-11-14-04-31-25-DL4KJ
2018-11-14-14-11-33-084EG
2018-11-14-14-14-58-MWLJC
2018-11-15-06-37-03-WLWGM
2018-11-15-06-37-28-8RUQJ
2018-11-15-08-02-12-IT94K
2018-11-15-08-03-55-XG1HF
2018-11-15-08-05-37-U6PI9
2018-11-18-04-46-16-9W9WV
2018-11-22-19-30-14-BXD43
2018-11-28-02-58-15-ZBGP8
2018-11-28-02-58-37-4OVEC
2018-11-28-03-02-31-3N2N9
2018-11-28-03-19-32-BPUDH
v1

Figure-5: Directory where the video mp4 file is being generated and stored automatically in each execution of the game in Unity

Video Capture: Video Capture is a plugin that enables you, the Unity developer, to capture video and audio from your Unity application. It's great for recording video trailers, demos and in-app footage for your Unity-based game or app. It's fast, flexible and easy to use. When the video is recorded you decide how it's handled. Give your users complete freedom to share it, restrict it to playback from within your app, or anything in between.

Getting the trajectories of the car in a text file

I wrote a code for getting the information of the path which has been followed by the car to reach the destination. The followed path from the source to destination is called trajectory and for each point of the trajectory we have a state information of the video which is nothing but the frame at each time step. This text file contains the x, y (in case of slope somewhere in the road) and z coordinates of the position of the car in unity. Similarly, I recorded the coordinates of the four wheels of the car to get the orientation information of the car. After getting the coordinates of the car and four wheels, I wrote a python code to get 3D plot of the array of the coordinates. Therefore in each execution of the scene in the unity, we get the trajectory information of the car and its wheels in 3D plot.

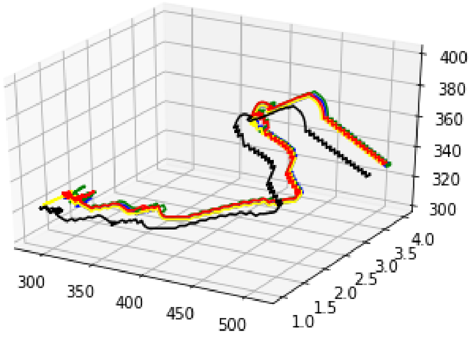

Figure-6: Trajectory of the car and its four wheels (Black: Car and Red, blue, green, yellow – four wheels)

Connected the scene with Head Mounted Display (Oculus)

To get the immersive feeling of the IITM campus, I connected the scene with the oculus.

RESULTS AND DISCUSSIONS

In this project, I successfully developed a model of the navigation system of 3D IITM campus using a car. This work provides us an immersive feelings of the campus tour, where we are driving the car as we drive in real world IITM campus. In this virtual reality environment of the campus, we can see the building of hostels, departments, sports complex, guest house, stadium with red colour boundary around it, admin block, central library, swimming pool, and gajendra circle, a large number of trees, deers, a large number of cycles of the students in the parking area, and also the students who is riding the cycles in the road. Apart from this, I also implemented a car with a driver who is driving the car inside the campus with a very nice control over the car with arrow keys and brake button to stop the car when the driver see any student or pedestrian coming towards the car.

Therefore this model gives the user a very immersive feelings, when the user drives the car by wearing the HMD over the head. For each ride of the car from the source to destination, it gives us a recorded video file of the driver's view during driving the car and the trajectory of the car and wheels in 3D plot using c sharp and python scripts. So that the frames of the video and trajectory of the car can be used for spatial navigation studies.

Use of the recorded videos for later studies

The frames of the video can be used as data inputs for several studies of Deep Learning such as traffic sign detection, lane detection, pedestrian detection and classification of the same. These datasets are not only useful for deep learning studies but also for reinforcement learning to make the agent (car) get familiar with the complete campus environment without giving any information (without arrow keys input) to the car where it has to go next to reach the destination place. So assume that the car is an actor or agent to take the action based on the iterative method of the value function of the network. So at time ‘t’ car will get an observation state s(t) from the value function of the network and then car will perform an action a(t) (one of forward, backward, left, and right) based on the observation state s(t) and then the state of the environment will become s(t+1), then car will get a reward r(t) from the value function based on the difference between actual action and the expected action, the same will be repeated for time t+1 and so on.

SUMMARY

I modelled the campus architecture with building and streets in City Engine and then exported it from City Engine to Unity. After this, I added name plates for the building, trees, deers, cycles in parking area, students riding the cycles in road in Unity. Then I modelled a car which is controlled by the arrow keys of the keyboard with the help of c sharp scripts. Then I wrote c sharp and python scripts to get a video of the driver’s view while riding the car and trajectory of the car from source to destination.

At the end of the project, IITM campus has been modelled in Unity to be used as the navigation system in my PhD research work. As my PhD research topic is self-driving car, where we are trying to build a bio-inspired attention model for traffic sign detection and classification using Deep Reinforcement Learning.

FUTURE WORK (write more)

- Some more objects can be added into the scene of the Unity to get more realistic feelings of the campus such as monkey roaming around the trees, ATM etc.
- Car can be controlled by some other inputs such as controller, speech
- Real world persons living in the campus can be imported
- This environment can be used for navigation task using Deep Reinforcement Learning

CHALLENGES

There were many challenges which I faced while doing this project:

- One of the toughest challenge was to find the best suitable tool to build a highly immersive IITM campus architecture. I got City Engine which is a very advanced software to make a city after struggling with many other software like Blender, 3D Max, and Maya etc.

- Modelling the assets of 3D structure of trees, deers, cycles and cycling students was also a tough task. Because available assets which were looking realistic was asking for a very high price. So I modelled those assets in 3D paint.
- Learning to use the City Engine software to model a city was also one of the toughest task. Then I followed some online tutorials to learn it.
- Controlling the car with arrow keys was little difficult to understand that how to write the code for this.
- Writing the code to recode the video of the driver’s view from the car while driving and getting the trajectory of the car and its four wheels was also one of the most challenging task. Which I solved it by trying it many times with debugging the code of c sharp and python.
- I did not have knowledge to operate with oculus so this was also a challenge for me

VIRTUAL REALITY IN THE PROJECT

This project is to model a virtual environment of the IITM campus where a user or driver wears an HMD or Oculus over his head and drive the car to reach the destination. This gives the user a highly immersive feelings to be in the IITM campus even the user is not in the campus in real world. Because the environment of the campus is same as it is in the real world.

ACKNOWLEDGEMENTS

I want to acknowledge the course Prof. Manivanan Sir and Prof. Srinivasa Chakravarthy Sir to give me a very helpful guidance for the project and also the Touch Lab and CNS Lab members.

REFENRECES

Mirowski, P., Grimes, M. K., Malinowski, M., Hermann, K. M., Anderson, K., Teplyashin, D., ... & Hadsell, R. (2018). Learning to Navigate in Cities Without a Map. *arXiv preprint arXiv:1804.00168*.

Mirowski, P., Pascanu, R., Viola, F., Soyer, H., Ballard, A. J., Banino, A., ... & Kumaran, D. (2016). Learning to navigate in complex environments. *arXiv preprint arXiv:1611.03673*.

Péruch, P., J. Vercher, and G.M. Gauthier, Acquisition of Spatial Knowledge Through Visual Exploration of Simulated Environments. Ecological Psychology, 1995. 7(1): p. 1.

Witmer, B.G., J.H. Bailey, and B.W. Knerr, Training Dismounted Soldiers in Virtual Environments: Route Learning and Transfer. 1995, U.S. Army Research Institute for the Behavioral and Social Sciences.

Zhu, Yuke, Mottaghi, Roozbeh, Kolve, Eric, Lim, Joseph J., Gupta, Abhinav, Fei-Fei, Li, and Farhadi, Ali. Targetdriven visual navigation in indoor scenes using deep reinforcement learning. In 2017 IEEE International Conference on Robotics and Automation, ICRA, pp. 3357–3364, 2017.

Wu, Yi, Wu, Yuxin, Gkioxari, Georgia, and Tian, Yuandong. Building generalizable agents with a realistic and rich 3d environment. arXiv preprint arXiv:1801.02209, 2018.

Mirowski, Piotr, Pascanu, Razvan, Viola, Fabio, Soyer, Hubert, Ballard, Andrew J, Banino, Andrea, Denil, Misha, Goroshin, Ross, Sifre, Laurent, Kavukcuoglu, Koray, et al. Learning to navigate in complex environments. arXiv preprint arXiv:1611.03673, 2016.

Lample, Guillaume and Chaplot, Devendra Singh. Playing FPS games with deep reinforcement learning. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, 2017.