AUGMENTED REALITY INTERIOR DESIGN

Tejas Kale[1], Ram Raut[2], Sushant Mahalle[3], Akshay Katta [4], N. F. Shaikh[5]

Dept. of Computer Engineering M.E.S College of Engineering (SPPU) Pune, India
tejasvijaykale@gmail.com, ramraut25@gmail.com, mahallesushant1998@gmail.com, akshaykatta11@gmail.com, nfshaikh@mescoepune.org

****

Abstract: Augmented Reality is a field of research which deals with combination of reality with computer related data. With advancement in computer vision and in mobile computing, AR has scaled across platforms, and has increased adoption in major products. Augmented Reality allows to robustly insert virtual objects in a sequence of scenes. Our system operates on three main augmented reality facilities i.e Motion tracking, environmental understanding, light estimation. In this paper, we propose a system that is capable of alternatively tracking horizontal and vertical planes, repainting planes, rendering 3D interior design assets using marker less detection.

Keywords: *Augmented reality, plane detection, motion tracking, light estimation*

I INTRODUCTION

Augmented reality is a technology that works on computer vision based recognition algorithm to augment three dimensional graphics, video, and imagery, sound. It is a good way to render real world information and present it in an interactive way so that virtual elements become part of the real world. Augmented Reality comes becomes possible due to various sensorial input data using on device sensors like the accelerometer, gyroscope, depth sensors. The input through sensors is processed by the computing device inorder to understand the environment based on the available amount of light which enables the user to place 3D models in real time. Proposed system of Interior design using Augmented Reality implements various terminologies, they are Position estimation: wherein current co-ordinates of a virtually placed object is calculated using the camera inorder to localize its current object in control. Instant Motion tracking: Motion analysis influenced by motion model for region tracking, planar target tracking, tracking motion in sixth degree of freedom. Simultaneous localization and mapping (SLAM) : Panaromic mapping technique, sensor fusion, place recognition for localization of virtual assets. Depth Estimation: Dense depth Map, key frame selection for enhancing the placements of interior design assets. [1] In this paper, we depict our work to enhance interior design through real time horizontal and vertical plane anchoring, vertical planes; wall repainting and augmenting various design aspects using various technologies such as Google's ARCore, computer vision.

II MOTIVATION

The degree of freedom for user to design their homes according to their own creativity and imagination is what motivates us the most to propose a system which is capable of improvising user experience. Helping a user to reconstruct their interior scenes, paints their walls virtually, place 3D objects through the system is our motive. To enable various professionals like architects, interior designers and common users to visualize various interior design patterns, implement their ideology in home decoration activities. Problems that occur in interior design is that user is not able to visualize and feel how actually an object may look in reality, which also restricts designers and architects to convey their design to their customers on an idea of full-fledged interior design. This gives us an opportunity to resolve relevant problems like this.

III RELATED WORK

Various previous researches have shown varying results. Instant Motion tracking consists of a motion analysis module, a region tracking module, and either a planar target tracking module for planar surfaces. Using solely the motion data produced by the motion analysis module, the region tracking algorithm tracks individual objects or regions while discriminating them from others. [2] To track an input region, we first crop the motion data to a corresponding dilated sub-region. Then, using iteratively reweighted least squares we fit a parametric model to the region's weighted motion vectors to determine the region's movement across consecutive frames.

Pose estimation is car knowing the correspondences between 2D measurements in the images and 3D features of the model. [3]Alternatively other approaches to pose estimation can be considered by ARCore uses an iterative search for the pose. Although markers were used. Also considering key points, these methods are also used for fast re-localization issue in environments that have been previously reconstructed using a SLAM approach.

While SLAM provides an inherent tracking solution, it does not provide any reference to a known, global location. Therefore information that is referenced to such a real location, for example through a GPS position, cannot easily be rendered in a purely SLAM-based system. Panoramic Mapping system is based on a simultaneous mapping and tracking approach, operating on cylindrical panoramic images. The algorithm is conceptually comparable to SLAM, however it create a 3D map of the environment, but limit the map to a 2D panorama. [4] This simplification works well for users who are standing still while turning the phone. The method creates a cylindrical map of the environment on the fly and simultaneously uses this map for tracking the camera orientation.

As the user navigates through their environment with a smartphone in hand, our pipeline starts by tracking 6DoF poses using the off-the-shelf VIO platform of ARCore. Our system could use any other VIO or SLAM platform at this stage. Once the6DoF tracking pipeline is initialized and given the latest available camera image, the first step towards computing a depth map consists of identifying a keyframe from the past image frames that is suitable to perform stereo matching. Next, the relative 6DoF pose between the keyframe and the current frame is used to perform polar rectification. depth estimation is based on stereo matching between the most recent image and a past keyframe. The choice of the keyframe is dependent on several disparate factors, each of which contribute to the potential matching quality of a candidate keyframe. [5] For instance, greater depth accuracy is gained by increasing the stereo baseline between the chosen keyframe and the present position, but such frames are also further back in time, which can introduce temporal inconsistencies.

Therefore as per, various related research it can be inferred that augmented reality essentially need support of various aspects like understanding its surrounding environment with respect to position of the user, amount of light necessarily needed to enable the system to place 3D objects in real space, keeping a track of motion with relative to user position, device movements, odometry of the assets, ground plane detection, occlusion of real world and virtual objects, realism of the assets, space coverage, etc.

In our study, we propose a system that uses the previous research effectively to solve the problem of vertical plane detection along with vertical plain repainting, augmenting interior design patterns vertically as well as horizontally, dedicating emphasis to all aspects of interior design through augmented reality.

IV IMPLEMENTATION

Augmented Reality for interior design requires proper communication between hardware and software components. Environment plays a major role in the system architecture. Accurate functioning of camera, sensors, types of software API's working together makes the system capable of augmenting virtual 3D objects on the display screen.

All relative terminologies for augmented reality are supported by Google's ARCore utility. ARCore is Google's platform for building augmented reality experiences. Using different APIs, ARCore enables your phone to sense its environment, understand the world and interact with information.

Fundamentally, ARCore is doing two things: tracking the position of the mobile device as it moves, and building its own understanding of the real world. ARCore's motion tracking technology uses the phone's camera to identify interesting points, called features, and tracks how those points move over time. With a combination of the movement of these points and readings from the phone's inertial sensors, ARCore determines both the position and orientation of the phone as it moves through space.In addition to identifying key points, ARCore can detect flat surfaces, like a table or the floor, and can also estimate the average lighting in the area around it. These capabilities combine to enable ARCore to build its own understanding of the world around it. ARCore's understanding of the real world lets you place objects, annotations, or other information in a way that integrates seamlessly with the real world. ARCore enhances our work through its enormous capabilities. [1]

A simple workflow for implementation is followed such that an AR utility enabled smartphone camera will sense the environment, process the area captured, anchor the free space so that objects can be placed in it. Selecting objects from an interactive menu is the next step. Scaling the object according to user convenience, rotating the object across x, y and z axis is a feature.

ARCore package com.google.ar.core.* provides a way to stack objects on device screen that use classes like Pointer- Drawable, ModelLoader, CameraConfig, CreateAnchor, Ses- sion, Pose classes. Crutial and necessary ones are included in the class diagram. Classes and their methods are such - ”Camera” class with methods ”getDisplayOriented- Pose()” , ”getPose()” , ”getProjectionMatrix()” and ”getTreckinState()”. Class ”Camera- Config” supports Camera class with methods like ”getFpsRange()” , ”getImageSize” , ”get- TectureSize()” . Other classes that are included in the package are ”Plane” with methods ”createAnchor()” , ”getAnchors()” , class ”Pose” with methods ”transformPoint()” , ”extractRotation()” , ”extract- Translation()” class ”LightEstimate” with methods ”getEnvi- ronmentalHdrMainLighrIntensity()”,”getEnvironmentalHdr- MainLiightDirection()” , ”getTimeStamp()”. Class ”Anchor” supports class ”Plane” with methods ”getCloudAnchor()” and ”getTrackingState()” . Class ”Session” with methods ”getAl- lAnchors()”, ”getAllTrackables()” , ”get-DisplayGeometry()” and ”Update()” is used to save and update the final scene time to time prepared by the user.

A. Wall Repainting

Inorder to implement the wall painting feature earlier it was necessary to define bounds of a wall, mark their con- tours inorder to detect the edges and boundaries of a wall. The OpenCV library was not used upto its full potential to implement computer vision algorithms to enhance the feature extraction of the vertical plane scene. Wall repainting phenomenon in the proposed system is implemented through computer vision OpenCV library along with using Python programming language as a tool. We enforce the system to detect a particular wall color, allow user to select their desired pattern or color to repaint the actual wall paint covered area with desired color. Contour detection of wall area will help in repainting the wall as like a fission reaction spills with similar cells. Selecting a predefined color will enable the system to recognize and save that color in a temporary storage facility. While the user selects their desired color the system will process the information to replace the color in database with the corresponding user desired color.

Vertical Plane recognition along with horizontal Plane recognition is a challenge. Implementing both of the aspects alter- natively in the system as separate features will allow user to specifically perform visualization of various features of interior design separately and depict a clean implementation. Features like placing furniture, changing wall paints, visualization various interior design plans, patterns, ceiling designs, light lamps and so on will help user enhance their understanding and design of their home structure.

Figure 1: Rendered 3D model

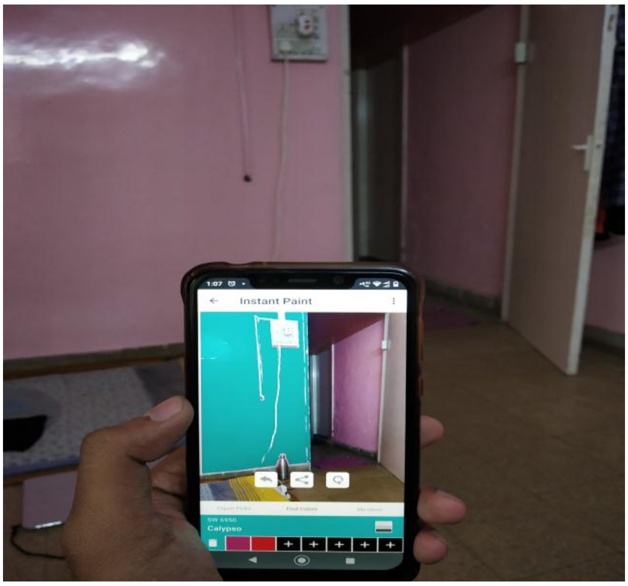

Figure 2. Wall painting feature

V CHALLENGES

Limiting scenarios of the application consists of difficulties caused due to smart phone hardware quality and configurations. Operation system versions being an addition to that . Hardware support like minimal required RAM and processor are a must to obtain augmented reality support .Difference is operating systems like Android and iOS is also a major concern. Other limiting factor is the availability of 3D assets on a large scale along with good quality and small size of the assests. Performance constraints due to the lack of quality of assets can lead to the failure of the whole application. Scaling the application to variety of customers and their adaption to the new ways of visualization can cause some other limitations.

Rendering an AR experience utilizes too much power. We've have achieved a lot when it comes to miniaturized processors and graphics cards, but we're still not able to match the level we require to make high-end everyday AR and reality Computing generates a lot of heat. Basically, the more power used, the more heat that gets generated, and the smaller the device, the slower it gets rid of that heat. Rendering an AR scene is complex process. That heat generated in turn can slow down processors. In AR, all things exist primarily in three dimensions , but its an misconception to sat that Augmented reality has to 3D but the majority of data assets , application and experiences will need at least a little 3D designing knowledge somewhere in our project. Currently the base of people with graphic development skill is still limited. Currently, Computer Vision is a fast growing but limited technology. Giving computers the ability to recognize the full catalog of earthly objects at any time of day and segment them into useful groups, just isn't something we've completely pulled off yet. Successful registration needs an accurate tracking system which mainly depends on various sensors integrated in AR system [6]. Camera is the most functional and flexible one. Combining with vision-based computational technologies, such as graphic recognition algorithms, camera- based registration is potentially competent for any AR application. Another difficulty is that marker-less registration has a high requirement of computational resources which mainly indicates hardware performance

VI CONCLUSION

With advancements in computer vision technology and cheaper hardware, AR can only flourish. AR has its unique advantages and is very good at tackling especially visualization problems. In an AR environment, visualizing 3d objects could be convenient and easy while saving costs by completely lowering the risk of product returns. In this study, we examined how a marker-less AR could be used for interior design. We proposed a mobile application that enables users to visualize home decoration objects in reality. Registration as the key technique highly affects how successful a virtual object is superimposed on the real world To improve the robustness of marker-less registration, advanced camera self-calibration approaches need to be integrated in registration process.

REFERENCES

[1] ARCore. https://developers.google.com/ar

[2] Jianing Wei, Genzhi Ye, Tyler Mullen, Matthias Grundmann, Adel Ahmadyan, Tingbo Hou, "Instant Motion Tracking and Its Applications to Augmented Reality", Google Research 2018.

[3] Eric Marchand, Hideaki Uchiyama and Fabien Spindler, "Pose estima- tion for augmented reality: a hands-on survey",2016.

[4] Gerhard Reitmayr, Tobias Langlotz, Daniel Wagner, Alessandro Mulloni, Gerhard Schall, Dieter Schmalstieg and Qi Pan Graz University of Technology, ”Simultaneous Localization and Mapping for Augmented Reality”,2010.

[5] Julien valentin, adarsh kowdle, jonathan t. barron, neal wadhwa, max dzitsiuk, michael schoenberg, vivek verma, ambrus csaszar, eric turner, ivan dryanovski, joao afonso, jose pascoal, konstantine tsotsos, mira le-ung, mirko schmidt, onur guleryuz, sameh khamis, vladimir tankovitch, sean fanello,shahramizadi, and christoph rhemann, ”Depth from Motion for Smartphone AR”,2018.

[6] Yan ”Registration Issues in Augmented Reality”,2017.