Reification of furnishings for interior design using Augmented Reality

Prof. Dr. Ahmed El-Abbasay
Higher Institute of Computers and Information Technology, Computer Depart. , El. Shorouk Academy, Cairo, Egypt
Email: dr.ahmed.abbasi@sha.edu.eg
Aml Ibrahim Kamal
Higher Institute of Computers and Information Technology, Computer Depart. , El. Shorouk Academy, Cairo, Egypt
Email:amal.ibrahim@sha.edu.eg

Abstract- ***Augmented reality (AR) technology and 3D interior modeling is crucial in designing. AR technology has been useful in numerous fields like architectural design and construction of buildings Augmented Reality furniture arrangement systems are useful for viewing room or building layouts without purchasing or moving actual furniture. Such systems, however, are typically constrained to 2D tablet or phone interfaces and frequently need users to physically and regularly alter their viewpoint of the real world, requiring human scene manipulation. This paper will present a system can add virtual object to the real environment (Augmented Reality) using camera and using real-time tracking marker less with Simultaneous Localization and Mapping (SLAM)algorithm for corner detection and then place objects. This paper assists in the improvement and evaluation of the design project; a complex decoration modeling structure is created to ensure that the user can connect with all points of the project design in order to achieve perfect coordination between structure and function.***

Index Terms— Augment reality , SLAM, Marker-less Augmented Reality

I. Introduction

The release of innovative hardware such as Google Glass has re-ignited interest in augmented reality([1]), a topic that has been a hot topic in software development circles for several years. The camera on your device is used in conjunction with computer vision-based recognition algorithms to superimpose digital information onto a live feed of the real world. It's a great tool for visualizing data and presenting it in an engaging way, making virtual objects feel like they belong in the physical world. Displays that use augmented reality superimpose data onto the user's field of view, creating a surreal environment where the real and virtual worlds merge. It works on both desktop and mobile platforms.

As more and more people shop for and buy furniture online, it's important to be able to accurately gauge the size of the pieces before you commit to buying. While there are currently AR-based applications out there, most of them are either not optimized for real-time processing or are limited to a single image plane. To get around that, he can use this app to see, through the medium of augmented reality images, if the furniture is, in fact, adjustable, so that it can be placed in the customer's actual living space, be it a home or an office. As a first step in this direction, our app uses augmented reality to show users a 3D rendered model, which is a digital representation of the real furniture without obstructing the markers. In this research, we propose a novel approach to applying Augmented Reality technology to furniture, wherein a user can view virtual furniture and communicate with 3D virtual furniture data via a dynamic and flexible user interface. The paper is organized as follows. In section II-Background, In section III-Related technology, In section IV-System implementation.

II. Background

Two distinct methods were considered during the planning and development of our project. It can be broken down into two categories: 1) Marker-based Augmented Reality, and 2) markers-less Augmented Reality.

Existing augmented reality [11] applications frequently make use of visual markers. Marker-based augmented reality systems frequently take input from real-world video. This means that the tracking system employed for marker detection and pose estimation has a significant impact on the effectiveness of a marker-based augmented reality system. Marker-based AR is widely regarded as a mature [12] method. Mark-free tracking is another, albeit less common, method for developing augmented reality visualizations. Model-based and Structure-from-Motion (SfM)-based techniques dominate the field of Marker-less Augmented Reality (MAR) [13]. Model-based techniques estimate the camera's pose by using a 3D model pre-populated with data collected before tracking begins. Without requiring any pre-existing scene knowledge, SfM-based approaches estimate camera motion across frames. The primary objective here is to acquire a means of estimating a camera's position, orientation, and three-dimensional movement from images captured with just one calibrated camera and without the use of any markers. Using AR with and without markers is depicted in Figures 1 and 2, respectively.

Figure 1 AR with markers

Figure 2 AR marker-less

This section summarizes the state of the art of two markerless approaches, the available technology and the main future challenges AR will have to face.

A. Location-based and SLAM

Location-based augmented reality (AR) takes advantage of smartphones' built-in GPS, digital compass, accelerometer, and camera to pinpoint the user's precise location and then match that with nearby points of interest (POIs) to serve up contextually relevant content. When a user's device data matches a specific location, the corresponding information and virtual objects are displayed.

Marker-less augmented reality (AR) is more flexible than marker-based AR because it relies on positional information gleaned from a device's camera, GPS, digital compass, and accelerometer rather than an image or object cue to deploy.

In the realm of autonomous mobile robotics, method of Simultaneous Localization and Mapping (SLAM) [14] is widely used as a map-building approach in Figure 3.

SLAM is still being actively studied because users want more efficient and faster algorithms. While this can simplify the algorithms required for sensor positioning and augmentation, it also limits the scope of what can be operated on and how users can interact with the environment, which is why most visual augmented reality work relies on predefined markers or models. The goal of many augmented reality (AR) applications is to superimpose digital information onto the real world in ways that were not possible before. That, in turn, necessitates systems that are accurate, robust, and interactive, with the ability to jointly position sensor and scene with minimal input. The use of SLAM [15] techniques is a big help in dealing with or getting rid of this problem.

Despite extensive research into the computational complexity, map representation, and data association of SLAM's core problems [16], many difficulties persist. The creation of algorithms for ever-growing and less-structured environments is the most pressing [17] open question. Unstructured environments cause problems with sensing and linearization, both of which raise concerns.

In order to better categorise the current state-of-the-art in vision-based SLAM methods, Muhammad, Fofi, and Ainouz [18] introduced a detailed classification based on I imaging systems used for performing SLAM, such as single cameras, stereo pairs, multiple camera rigs, and catadioptric sensors, (ii) features extracted from the environment in order to perform SLAM, such as point features and line/edge features, and (iii) It is possible to use either delayed or immediate landmark initialization, SLAM techniques [19] like Extended Kalman Filtering and Particle Filtering, biologically inspired methods like RatSLAM, and other methods like Local Bundle Adjustment, and (v) wheel odometry data. Durrant and Bailey [20, 21] provide a comprehensive introduction to the SLAM method and the related literature. They explained what SLAM is, how it's traditionally been done, and what recent developments in computational methods and new formulations of the SLAM problem for large scale and complex environments have brought.

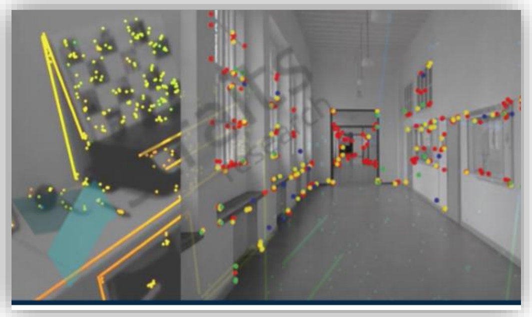

Figure 3 SLAM

III. Related technology

A. AR Technology

Augmented reality (AR) uses a smartphone's camera to blend digital elements into the real world, making it seem as though holographic information is actually there [8] [4]. Not like in VR [5], where you're completely submerged in a fabricated setting. Augmented reality alters the surrounding environment by incorporating 3D objects, animations, sounds, video, and graphics. that draws the user's interest to the technology.

B. AR Toolkit

It is a free and opensource software library for creating augmented reality (AR) applications that superimpose digital information onto physical space. The connection between the real world and the virtual one is established by using this toolkit library [3]. Several computer vision algorithms are incorporated into this toolkit. This facilitates the rapid creation of an enormous variety of AR-based applications.

- With a single camera and some black geometric squares, we can trace the code and figure out where something is in space.
- It can work with designs that are square or rectangular.
- A simple coding standard for cameras. A sufficient speed for use in real-time settings.
- The full source code is available for download.

C. Vuforia Augmented Reality SDK

Vuforia AR SDK is an augmented reality software development kit (SDK) for mobile devices. It employs computer vision techniques to detect flat surface tracking and superimpose 3D models onto the real world. [3]. Vuforia's API, or Application Programming Interface, is available in a number of languages. [7] Vuforia is a cross-platform solution for developing augmented reality (AR) and mixed reality (MR) mobile applications, boasting robust tracking and high performance across a wide range of hardware (including mixed reality HMD (Head Mounted Displays) and mobile devices). [2]. There is a wide variety of third-party hardware, including AR/VR/MR glasses, that is compatible with Vuforia. This facilitates the development of apps and games for both the Android and iOS platforms. The drag-and-drop function makes it simple to create new content. In order to showcase the platform's primary capabilities, Unity offers a Vuforia package through its Asset Store. The Vuforia SDK for Unity allows us to quickly and easily create augmented reality (AR), virtual reality (VR), and mixed reality (MR) applications.

D. UNITY

The Unity 3D engine is robust, and the Unity 3D environment is helpful for designers. Its ease of use makes it ideal for students, and its power makes it useful for experts. Unity's script is compiled with its own built-in C# compiler. Everything related to Visual Studio, including the making of new files and the upkeep of existing ones, is handled automatically. The Unity 3D engine allows us to easily add, remove, and transfer files. Your solution can include our Visual Studio files as well. When a new project is being made in Unity, the user has the option of importing new files and folders with ease. We can also make them a brand new file with the option to track images. [8]

E. Image Tracking

During image tracking, the computer calculates the camera's coordinates [6]. Based on the user's input, the system then displays only the content the user has indicated they are interested in. In conditions not yet known, On the up and up; Automatically recognizing camera-captured images or videos and adding 3D models to them is known as "image tracking," which we discuss in [9]. The detection and superimposition of an enhanced image in image tracking are both processes that rely solely on cutting-edge computational technology. [10]
AR tracking can be divided into 4 types:

1. Token-Based Fiduciary Tracking: it is a way to overcome obstacles to set easily detectable parameters Enter the real world and employ computer techniques. Using an image-tracking method, the system can track videos. It utilized pattern recognition and computer observation techniques. AR utilizes black and white squares as markers [4]. [Fig.2]. The use of black and white markers is due to the background being highly contrasted. A glaring error in the investigation. While zooming, a maker's technique that must be visible cannot be easily obscured by another object. [9]

2. Hybrid Tracking: Hybrid tracking is the combination of two or more data sources, such as a compass, GPS, and camera position, to determine the actual location and direction of an object. The information that can be used and what must be supplemented in the field of view of the actual image must be calculated, but the actual image is used to hold the developed layer.

3. Simulated Tracking: The modular approach incorporates the initial data of three-dimensional objects in the actual environment. Using the geometric representation of three-dimensional objects can alter their position and orientation. The model approach is based on the concept of edge detection to generate new three-dimensional models. This method requires additional processing power to track objects.

4. Natural Tracking: Natural tracking is a type of tracking in which the real world is used as a marker after its characteristics and behaviors have been identified. Natural landmarks are heavily utilized by marker-free currency trackers (Often referred to as point of interest or major point) the user's environment

IV. SYSTEM IMPLEMENTATION

A.Modules

First, as we declared in the architecture of our project Figure 4, we'll be using Flutter to design and activate the interface of the application to make it usable on both Android and IOS systems using Dart programming language. We'll also integrate between Flutter and Unity to work on the interface of the AR orders too and communicate them to the rest of the application.

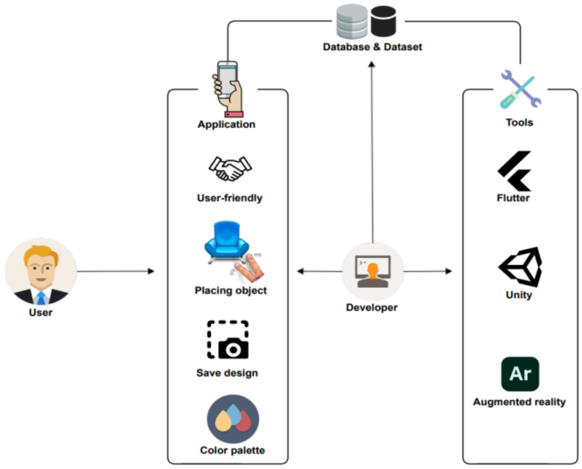

Figure 4 project Architecture

The application implementation consists of eight modules.

- Creating Augmented Reality Objects.
- Designing Views for the User Interface.
- View and choose a category
- Object Selection
- Placing Furniture Objects
- Rendering Objects
- Sharing the final Design
- Placing multi-objects

1. Creating Augmented Reality Object:

Firstly, We have to create a virtual model using 3Dmax to create a three dimensional models for interior design, models mainly use polygons method of modeling. After the model is implemented in autodesk 3dmax, you can save your .max files directly into your **Project** or export them into Unity using the **Autodesk .FBX** or other generic formats. Unity imports meshes from 3ds Max present the screen via Unity 3D [22]

2. Designing Views for the User Interface.

We use Flutter to make the app's views. The primary interface of the application is the rolling view of the key. Models of furniture, navigation buttons to go on to the next model, the size of the user's living room, specifics about each model (such as the model's dimensions when scanned) and the option to purchase the model from an online shop make up the bulk of the primary application. Huh.
All the furniture's keys can be kept in one place in a furniture column, and the column can be used to display and move individual pieces of furniture. To accomplish this, we show the view in context with the Android screen and add navigation buttons to the view. After that, we bring in a 3D model made in 3d max, move a chair using C# script for that object, and then use the function to switch views.

3. View and choose a category

In this module, we created unity scenes to represent:
One scene for all the categories Then, another multiple scenes for each category of furniture objects
We have eight categories which are: Beds, Tables ,Sofas, Chairs, Stools, Wardrobes, Decoration ,Full Scenes
Each category in our scene is accessed by an individual button that takes us to the objects of this category, then the user can choose whatever object they want to place in the real world. Furthermore, we used photoshop besides unity to create and edit the icons and images we used for each button, and also to create a pleasant design for the application to fulfill our user-friendly feature.

4. Object Selection

In this module , we started to build unity project that makes our 3D furniture objects we created before to be shown without other any image target(marker-less) and to be shown only on the ground plane or any vertical plane.
So, in this module, its idea is to let the object be free and be detected anywhere in any time, the user has just to open the camera phone from the application and select any object they want then they can find the 3D object appear in the place they choose and after that they could change the object properties as needed like scaling, dragging and rotating.

5. Placing Furniture Objects

After choosing an object in this module, the AR camera will automatically open.
Then the user can place the object they chose on a flat surface. Using AR camera and plane finder, the environment is detected and scanned to choose a suitable location for object placement as shown in Figure 5.

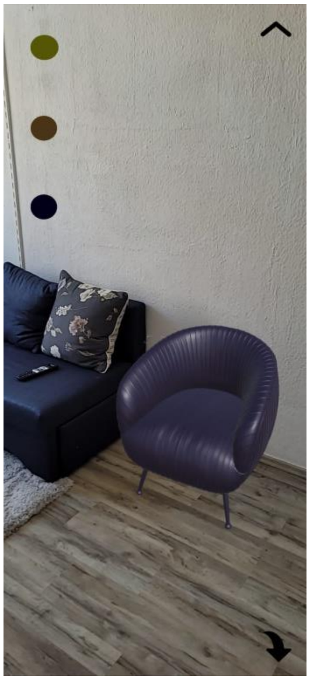

Figure 5 Object(chair) placement

6. Rendering Objects

In this module, User’s ability to scale, rotate and change the position of the objects.
Also, there are options to choose among multiple colors to change the object’s color to whatever color they want. The scenes of the objects have different options to apply to the object once they’re placed by the user in the real world, they can resize, rotate, drag the object, and even change its color.

7. Sharing the final Design

In this module, The application helps user natively to share files (images, videos, documents, etc.) and/or plain text on Android.
Firstly, get it from the assets store the import it inside the unity project. Secondly, create C# script where we write the functions of Native share. Finally, we drag the C# script inside the button function to make it work.

8. Placing multi-objects

In this module, we create a scene with more than one object. It’s difficult to combine more than one object in one scene and place all these objects in real and control each one separately to change its properties: rotate, locate, size as shown in figure 6.
The main idea to create a full scene in our project is to allow the user to feel the design.
Firstly, we add the objects on the ground plane then arrange the objects on suitable places

Secondly, we add to each object the box collider and to each object the “LeanSelectableByFinger”code and Create PressToSelect code.

Finally , we add to “PressToSelect”, “LeanFingerDown” & “LeanSelectByFinger” code and connect “LeanFingerDown.OnFinger” code with “LeanSelectByFinger.SelectScreenPosition” code.

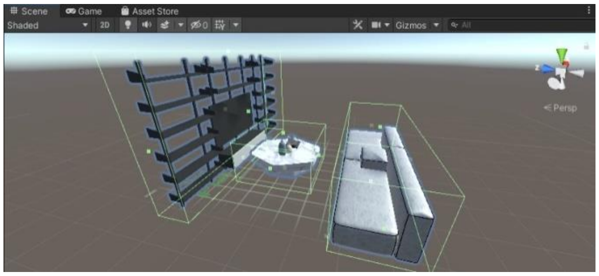

Figure 6 Multi-object placement

V. CONCLUSION

Augmented reality is becoming common. Game engine Unity 3D and AR development platforms allow us to integrate augmented reality to more apps. This study proposes a real-time app for viewing 3D furniture models. This application cuts costs and enhance real-time simulations using multimedia. Also, it helps the customer comprehend the project's concept and accomplish specific requirements and superior design.

People want to see virtual interior design concepts in real time, and AR lets them. It's easy to view furniture on our app by positioning the object model from your smartphone to your home or business with augmented reality. We all know that people won't buy furniture without seeing it. With augmented reality, customer fear is eased. Using augmented reality, consumers can visualize furniture in their home spaces. Those who want to buy furniture have more color and fabric options. Augmented reality lets users develop furniture alternatives based on their ideal world, experience in living environments, and acquire more comfortable furniture.

VI. FUTURE WORK

Thus, with the goal of having our app serve as the de facto standard for all Augmented Interior Design software, we've gone ahead and included the following new features to ensure that our app can do it all, from A to Z:

Color-changing walls- Using the ARCore and Unity Config.PlaneFindingMode module, we can select between three different detection modes:

A horizontal detector would have to look for a horizontal surface like a floor or ceiling, which is not what we were going for.

While the automatic detector of both horizontal and vertical surfaces at once is still in its infancy, Google's development kit now includes a new feature that allows us to detect vertical objects like walls using public static final Config.PlaneFindingMode VERTICAL.

VII. REFERENCES

[1] Mami Mori, Jason Orlosky, Kiyoshi Kiyokawa, Haruo Takemura. (2016, Sep.). A Transitional AR Furniture Arrangement System with Automatic View Recommendation. IEEE Adjunct. [Online]. 21(3).pp.21-24. ISBN: 978-1-5090-3740-7. Available: https://ieeexplore.ieee.org/document/7836488

[2] Etienne Mouragnon, Maxime Lhuillier, Michel Dhome, Fabien Dekeyser, and Patrick Sayd. Real time localization and 3d reconstruction.In 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’06), volume 1, pages 363–370. IEEE, 2006.

[3] Kalyani Pampattiwar, Akshay Adiyodi, Manasvini Agrahara, and Pankaj Gamnani. Interior design using augmented reality environment. Interior Design, 5(10), 2016.

[4] Viet Toan Phan and Seung Yeon Choo. Interior design in augmented reality environment. International Journal of Computer Applications, 5(5):16–21.

[5] Bernhard E Riecke, Daniel Feuereissen, John J Rieser, and Timothy P McNamara. Self-motion illusions (vection) in vr—are they good for anything? In 2012 IEEE Virtual Reality Workshops (VRW), pages 35–38. IEEE, 2012.

[6] Shigeru Saito, Atsushi Hiyama, Tomohiro Tanikawa, and Michitaka Hirose. Indoor marker-based localization using coded seamless pattern for interior decoration. In 2007 IEEE Virtual Reality Conference, pages 67–74. IEEE, 2007.

[7] Christian Sandor, Andrew Cunningham, Arindam Dey, and Ville-Veikko Mattila. An augmented reality x-ray system based on visual saliency. In 2010 IEEE International Symposium on Mixed and Augmented Reality, pages 27–36. IEEE, 2010.

[8] Konstantinos Smparounis, Dimitris Mavrikios, Menelaos Pappas, Vagelis Xanthakis, Giovanni Paolo Vigan`o, and Katharina Pentenrieder. A virtual and augmented reality approach to collaborative product design and demonstration. In 2008 IEEE International Technology Management Conference (ICE), pages 1–8. IEEE, 2008.

[9] Waraporn Viyanon, Thanadon Songsuittipong, Phattarika Piyapaisarn, and Suwanun Sudchid. Ar furniture: Integrating augmented reality technology to enhance interior design using marker and markerless tracking. In IIP’17, 2017.

[10] Liyan Chen, Xiaoyuan Peng, Junfeng Yao, Hong Qiguan, Chen Chen, and Yihan Ma. Research on the augmented reality system without identification markers for home exhibition. In 2016 11th International Conference on Computer Science & Education (ICCSE), pages 524–528. IEEE, 2016.

[11]. Bimber, Oliver, and Ramesh Raskar. (2005). Spatial augmented reality: Merging real and virtual worlds. Scientist. Vol. 6. AK Peters Ltd. doi:10.1260/147807708784640126.

[12]. Schmalstieg, Dieter, and Daniel Wagner. (2007). Experiences with Handheld Augmented Reality. 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality 07, pp: 1-13, no. 16. doi:10.1109/ISMAR.2007.4538819.

[13]. Zhang, Liang, Xu-Jiong Meng, and Yao-Wu Chen. (2009).Convergence and consistency analysis for FastSLAM. In 2009 IEEE Intelligent Vehicles Symposium, 447-452. IEEE. doi:10.1109/IVS.2009.5164319.

[14]. Chekhlov, Denis, Andrew P Gee, Andrew Calway, and Walterio Mayol-Cuevas. (2007). Ninja on a Plane: Automatic Discovery of Physical Planes for Augmented Reality Using Visual SLAM. 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality: 1-4. doi:10.1109/ISMAR.2007.4538840.

[15]. Veronica Teichrieb et al., “A Survey of Online Monocular Markerless Augmented Reality”, International Journal of Modeling and Simulation for the Petroleum Industry, Bol. 1, nº 1, pp. 1-7, August 2007.

[16]. Nuetzi, Gabriel, Stephan Weiss, Davide Scaramuzza, and Roland Siegwart. (2010). Fusion of IMU and Vision for Absolute Scale Estimation in Monocular SLAM. In Journal of Intelligent Robotic Systems, 61:287-299. doi:10.1007/s10846-010-9490-z.

[17]. Andreasson, H, T Duckett, and A Lilienthal. (2007). Mini-SLAM: Minimalistic Visual SLAM in Large-Scale Environments Based on a New Interpretation of Image Similarity. In Robotics and Automation 2007 IEEE International Conference on, no. April: 4096-4101.doi:10.1109/ROBOT.2007.364108.

[18]. Muhammad, Naveed, David Fofi, and Samia Ainouz. (2009). Current state of the art of vision based SLAM. Proceedings of SPIE 7251: 72510F-72510F-12. doi:10.1117/12.805839.

[19]. Yan, Jiang, Liu Guorong, Luo Shenghua, and Zhou Lian. (2009). A Review on Localization and Mapping Algorithm Based on Extended Kalman Filtering. 2009 International Forum on Information Technology and Applications 2: 435-440. doi:10.1109/IFITA.2009.284.

[20]. Durrant-Whyte, H, and T Bailey. (2006) Simultaneous localization and mapping: part I. IEEE Robotics Automation Magazine 13, no. 2: 99-110. doi:10.1109/MRA.2006.1638022.

[21]. Bailey, T, and H. Durrant-Whyte. (2006). Simultaneous localization and mapping (SLAM): part II. Ed. Bruno Siciliano and Oussama Khatib. IEEE Robotics Automation Magazine 13, no. 3:108-117. doi:10.1109/MRA.2006.1678144.

[22] (Unity Documentation Manual) https://docs.unity3d.com/2017.4/Documentation/Manual/HOWTO-ImportObjectMax.html