Enhancing Urban Mobility via AR-based Parking Navigation System

Kimia Afshari[1]

School of Computer Science, University of California, Santa Barbara

ABSTRACT

The rapid advancements in mobile technologies have generated significant interest in Location-Based Services (LBS). As the reliance on outdoor navigation systems within mobile applications continues to grow, conventional map navigation can be upgraded through the strategic integration of Augmented Reality (AR). Meanwhile, by leveraging landmarks and pathways, AR-based navigation systems hold the potential to enhance user experience to unprecedented levels. This paper presents a novel approach combining an automatic camera-based parking space detection method with an AR routing tool, resulting in an interactive driving navigation system with more features than traditional route guidance methods. Furthermore, this research delves into the increasing challenges associated with parking in metropolitan areas and meticulously investigates the potential benefits of integrating AR-based outdoor navigation to enhance urban mobility. The proposed solution combines Machine Learning methods for parking space detection with advanced navigation systems, offering benefits beyond time-saving for drivers. It could potentially play a role in reducing vehicles' energy consumption and thus relieving traffic congestion, which might contribute to creating more sustainable and livable cities.

The source code can be found on https://github.com/kimiaf1998/AR-based-Parking-Navigation.

Keywords: Augmented Reality, Location-Based Service, Car Navigation System, Artificial Intelligence, Computer Vision, Smart Parking System.

1. INTRODUCTION

Finding available parking spots in metropolitan areas might be challenging, time-consuming, and tedious, often leading to increased energy consumption and traffic congestion. To address this issue, various navigation services have been developed, aiming to mitigate the need for prolonged searches and to provide drivers with up-to-date information about available parking lots. These systems affect transportation by streamlining the parking process and helping drivers park more efficiently. Concurrently, recent technological advancements have significantly propelled outdoor routing and made it user-friendly and more accessible. On the other hand, the widespread adoption of the Global Positioning System (GPS) has revolutionized outdoor navigation and empowered individuals to utilize their mobile devices for efficient and reliable guidance. Furthermore, with the increasing demand for Location-Based Services, a corresponding need for additional enhancements in navigation systems has emerged.

Augmented Reality (AR) is a promising solution for enhancing conventional navigation services. By integrating real-time visual information through AR to have a turn-by-turn instruction set, the abstraction of traditional maps is reduced, and users will be able to orient and follow pathways using landmarks. Although using GPS has been instrumental in providing accurate outdoor positioning, pushing this technology further and making it more engaging has always been necessary. Moreover, relying solely on GPS may not always be an efficient solution in densely populated areas or environments with limited satellite visibility, and it may lack a reliable environmental perception. The combination of GPS technology with the Visual Positioning System (VPS) offered by ARCore Geospatial [1] allows for a more precise and interactive outdoor wayfinding experience. It benefits from the broad coverage of GPS, global-scale recognition, and localization capabilities of VPS. This helps attach digital content into physical real world to provide an immersive user experience. By integrating this visual content into conventional guiding maps, users will be able to find their way visually and in a more convenient way.

In this paper, we create a driving navigation system that utilizes markerless AR, visual cues, and camera-based parking space detection technique to actively guide drivers toward the closest available parking spot. Leveraging Machine Learning (ML), the Parking Space Detector effectively identifies vacant parking spaces using a single camera sensor in parking lots. Furthermore, the integration of VPS and GPS technologies offers valuable information about the driver's location and position in the physical world, enhancing their understanding of the surrounding environment. The system offers an immersive and engaging navigation solution that seamlessly guide drivers toward available parking spaces in real time. This solution provides drivers with information that can positively affect their psychological character by releasing their stress, especially in environments with limited vision due to lighting or weather conditions, including darkness, rain, and fog. In the next section, we will discuss related works in context-aware applications and parking space finding methods and highlight the advancements in AR-based navigation technologies. Besides, we will detail the proposed methodology and do some evaluation on it, followed by a conclusion and a compilation of suggestions to encompass a broader range of aspects in the future.

2. RELATED WORKS

The development of context-aware applications and advancements in positioning methods, such as GPS and VPS, has paved the way for innovative solutions in interactive AR-based driving navigation field. On the other hand, leveraging the fast-growing field of AI to automate the process of locating available parking spaces has become increasingly significant. This section explores relevant research and existing systems that have investigated similar approaches, aiming to provide valuable insights into the topic.

2.1. Context-Aware Parking Applications

Context-aware applications have gained significant attention in recent years due to their ability to provide personalized and location-based services. These applications leverage various sensors and data sources to understand the user's context, enabling them to deliver tailored information and assistance. Several studies have focused on context-aware parking systems to alleviate the

[1] kimia_afshari@ucsb.edu

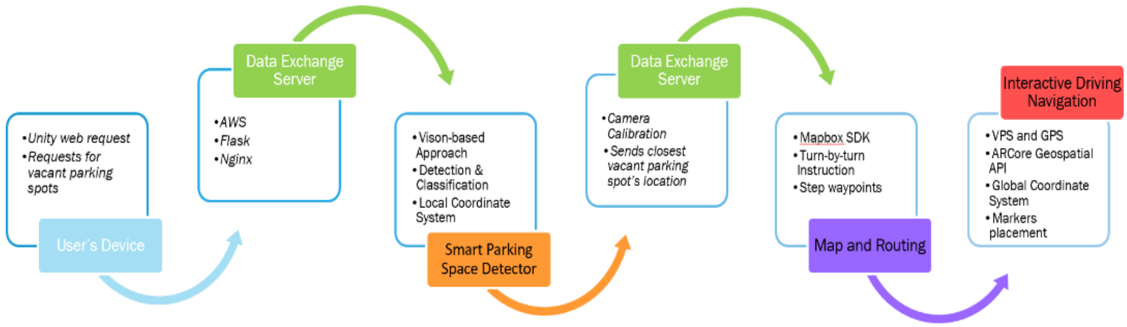

Figure 1: System pipeline

challenges associated with finding available parking spots in metropolitan areas. Hafiz et al. [2] proposed a context-aware parking application to assist drivers in finding parking slots dynamically while moving and/or arriving at the destination. This system allows registered users to reserve parking slots for the estimated time of arrival in the specified location in advance.

2.2. AR-based Navigation Technologies

Augmented Reality (AR) technology provides users with an intuitive and engaging experience for enhancing navigation systems so that they can interact with their surroundings. By integrating virtual content into the real-world environment, AR enhances the navigation experience in several ways, offering unique benefits for users.

Muhannad et al. [3] suggested an AR-based outdoor smart parking system that provides parking space information to drivers and interactively navigates them toward the vacant space by analyzing the scene captured by smartphone's cameras. The system operates by transmitting live video streams from mobile applications to a central server, which analyzes the scenes in real time and provides drivers with up-to-date guidance and information. This only works for a particular parking lot with prior scene knowledge and some surrounding route information used for navigation near the parking lot.

Tashko et al. [4] reviewed the studies on using AR technology in Head-Up Displays (HUD) for driving navigation assistance. The paper discusses various visual cues and information that can be displayed on the windshield, such as speed limits, navigation arrows, and traffic signs, to provide real-time guidance to the driver. It also analyzes different HUD-AR displays and their functions for driver assistance, such as lane departure, lane keeping, and detection of critical events.

Google Maps [5] has introduced AR technology in its walking mode to offer users a more interactive and convenient navigation experience. By leveraging a smartphone's camera and positioning sensors, Google Maps overlays virtual cues onto the user's view, providing enjoyable and easy-to-follow real-time guidance. The application relies on GPS and VPS technologies for accurate localization and recognition of the surroundings for object placement.

2.3. ML Parking space detection

Paulo et al. [6] presented a systematic review of vision-based approaches for parking lot management. The study highlights the need for robust video-based datasets, diverse environmental conditions, and labeling of parking spaces. It reveals that most works focus on individual parking space classification, with a lack of consideration for camera and parking lot changes. The usage of deep learning methods is prevalent, but there is a need for standardized evaluation protocols and the availability of public datasets for reproducibility and advancement in the field.

There are several commercial apps available that provide outdoor parking spot-finding services. Two popular options in this category are ParkWhiz [7] and SpotAngles [8]. These apps offer a wide range of parking spots, both on streets and in parking lots. However, when it comes to parking lots, these apps generally do not provide information about specific parking spots. Instead, they typically indicate the availability of parking spaces in terms of numbers. ParkWhiz allows drivers to reserve parking spots in advance, providing information on parking availability and operating primarily in the United States and Canada. SpotAngles also offers a range of paid and unpaid parking spaces and includes the flexibility to filter spots by price. Additionally, SpotAngles offers a car locator feature, assisting drivers in easily finding their vehicles in crowded parking lots.

3. Methodology

The proposed system is comprised of five major components. A Smart Parking Space Detector, a Coordinate System Mapping, a Map and Routing component, an AR-based Interactive Navigation component, and a Server. The system pipeline is demonstrated in Figure 1.

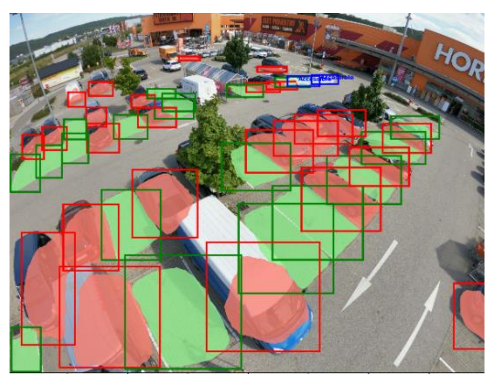

Figure 2: Predictions visualization of parking space detector model. Parking spots are identified by bounding boxes in conjunction with segmentation masks.

3.1. Smart Parking Space Detector

Through the Smart Parking Space Detector, we will employ computer vision techniques and real-time camera-based detection to identify vacant parking spots within parking lots. The approach to detecting vacant parking spaces is to have a classification model that distinguishes the vacant and occupied spots and a detector model to find the location corresponding to each spot. We utilize the ML-based detector model proposed in our previous work [9]. Given an image of a parking lot, the model outputs the location and emptiness status of each detected parking spot in a parking lot in its local coordinate systems (Figure 2). Images are captured from an overhead CCTV camera positioned in a parking lot. To navigate to each parking spot, each spot's location is then mapped to a geographic real-world coordinate, acting as a destination and making the navigation process more convenient.

3.2. Coordinate System Mapping

The Parking Space Detector component operates on its own local coordinate system, which means that the location of each detected parking spot is not directly interpretable by the user's screen coordinate system. To establish communication between these coordinate systems, a mapping process is necessary. This involves using camera calibration techniques with known local coordinates of the camera and their corresponding geographic coordinates in the real world for selected parking spots. By establishing these correspondences, a transformation matrix can be calculated to determine the relationship between a parking spot's location and its corresponding geographic coordinates. The geographic information, including latitude and longitude, for each parking spot is obtained from Google Earth [10], a comprehensive mapping tool offering various features such as street view, 3D buildings, terrain information, and measurement tools. Once the correspondences for the parking spots are collected, the transformation matrix is used to determine the corresponding geographic coordinates for each parking spot.

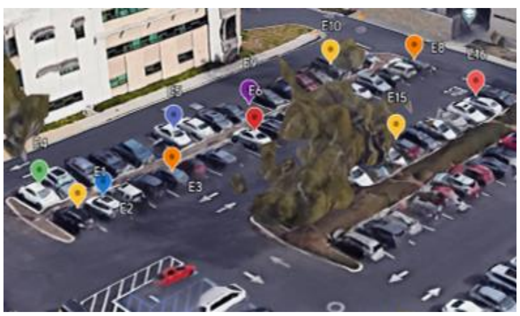

Figure 3: Parking lot view on Google Earth. Markers show the location of the parking spots used in camera calibration.

3.3. Map and Routing

The map component determines the optimal route to the closest available parking spot. To achieve this, we rely on the Mapbox Navigation SDK [11], which offers a range of features including route calculation and turn-by-turn instructions. Each route consists of multiple legs, which represent segments between consecutive waypoints. Since our scenario does not involve any intermediate points, there will be only one leg in the route. Each leg is further divided into steps, which provide detailed instructions and directions to guide the user to their destination. By leveraging the waypoints associated with each step, we can incorporate visual cues and create an interactive real-time route guidance experience (Figure 4: Waypoints of a route provided by Mapbox SDK (Figure 4).

3.4. AR-based Interactive Navigation

With this interactive navigation module, drivers will receive real-time directions and guidance through an immersive AR experience. Drivers will be able to follow directional arrows, virtual signage, and dynamic path indicators projected onto their smartphone screens. The component leverages the routing data result of the map component and integrates it with an AR interface.

Heading arrows are placed along the way at every waypoint in the current step, which is close to the user's location (Figure 5). If the distance of two adjacent waypoints exceeds a proximity threshold, some new waypoints placed between them are calculated to provide a seamless guidance. As users pass the waypoints, it continues placing the next waypoints in the current step. The waypoint proximity to the user's location is defined by a specific threshold. When the driver reaches the destination, a parking visual marker pointing to the parking spot is augmented (Figure 6). This helps the driver interpret and find the spot in a more convenient way.

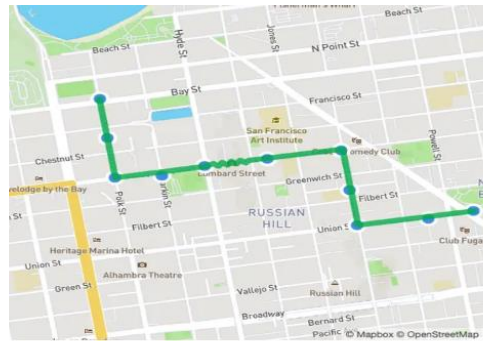

Figure 4: Waypoints of a route provided by Mapbox SDK

Figure 5: 3D heading arrows are placed along the way using waypoints

The object placement in the user's surrounding environment is enabled using ARCore Geospatial API. The ARCore Geospatial API provides comprehensive knowledge about the user's physical environment and creates a single coordinate system to work within. The API can be used in any area covered by Google Street View, helping to offer AR experiences on a global scale. It uses device sensors and GPS data to detect the device's environment, then matches the recognizable parts of that environment to a localization model provided by Google's VPS to determine the precise location of the user.

The physical location of each object is determined by its latitude, longitude, and altitude coordinates. Therefore, to be able to attach a visual content persistently to a desired area, the geographic location needs to be obtained using the Coordinate System Mapping module. The VPS is then employed to map the global location of these contents onto the user's screen, allowing them to be displayed accurately in the real world.

For optimal user experience, it is important to ensure that the heading arrows and destination marker align with the user's altitude. This alignment helps in providing convenience and avoiding confusion during navigation. ARCore's Geospatial API offers anchors as a means to attach objects to the scene. There are three types of anchors available: Geospatial, Terrain, and Rooftop. In our case, Terrain anchors are particularly useful as they enable markers to be placed at an elevation relative to the ground. This eliminates the need to determine the user's altitude separately, simplifying the overall process of attaching markers to the environment.

Figure 6: A 3D P marker pointing to the parking spot

3.5. Server

The server component plays a crucial role in facilitating data transfer between the parking space detection module and the user's device. It is responsible for processing and mapping prediction results obtained from the detection component, sending the global location of the closest available parking spot to the user's device for downstream tasks. To ensure smooth communication between different modules, this component leverages the Flask framework [12].

To effectively run the server, a web server is required, constantly listening for incoming requests. In our specific scenario, these requests correspond to queries for obtaining the nearest parking spot within a parking lot. Deploying the module on Amazon Web Services (AWS) [13] is a recommended solution, as it possesses the necessary capabilities to handle these requests and provide appropriate responses.

In addition, the server component utilizes Nginx [14] and Unicorn [15] for deployment. Nginx serves as a powerful web server and reverse proxy, efficiently handling incoming HTTP requests. Unicorn, on the other hand, acts as the application server, managing multiple worker processes to ensure optimal performance and concurrency.

4. Experiments

4.1. Platform and Packages

The project is specifically deployed on an Android mobile device, allowing us to conduct thorough testing. We have utilized Unity version 2021.3.22 to develop this Android application primarily because it offers a convenient and expedient deployment method, eliminating the need for unnecessary additional development. However, it is worth noting that certain Android packages within Unity may have limited features and functionality. For instance, Mapbox provides an appealing navigation user interface (UI) for Android, but it is typically accessed outside the Unity environment, such as through Android Studio. Given our Android platform, we have incorporated the ARCore XR package version 4.2.7 to create an AR experience while also utilizing Mapbox SDK for Unity version 2.1.1.

4.2. Evaluation

In order to evaluate the effectiveness of the proposed system, we have defined several metrics.

The first metric focuses on the system's performance in accurately placing interactive visual cues during navigation. By selecting different destinations and following the system's guidance, we assess the location offset of the visual markers to determine how closely it aligns with reality. The destination is taken from the user in the main page (Figure 7). Additionally, we verify the accuracy of heading arrows to ensure they point in the correct direction for the next step. The evaluation reveals that the system can generally attach contents in the correct locations with an acceptable offset (Figure 8 and Figure 9). However, there are scenarios where the offset exceeds the desired range.

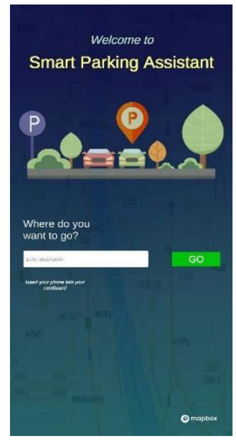

Figure 7: Application home page

The second metric assesses the reliability and stability of the system in driving guidance. We measure the frequency of failures in localizing and recognizing the environment, which directly impacts the placement of visual markers. These failures result in navigation loss, causing driver anxiety and confusion. The experiment demonstrates that the system lacks reliability and sometimes fails to place objects on the screen due to localization issues with ARCore Geospatial API (Figure 10). Without complementary text and audio turn-by-turn guidance, this navigation system cannot be considered reliable for everyday driving scenarios.

The third metric focuses on the overall user experience compared to traditional navigation systems like Google Maps. Our experiment indicates that when the system performs well and provides an interactive and robust AR navigation experience, it can potentially serve as an alternative to traditional methods. This can also be helpful in situations where it is difficult to determine the correct direction due to the similarity of multiple paths ahead.

The last metric involves testing the system across various speeds and driving environments, including roads, highways, and streets. However, as the speed exceeds 40mph, the device fails to localize accurately, resulting in the disappearance of visual markers. This can be potentially due to the increasingly number of frames per second to process and find the recognizable parts. This experiment highlights the unreliability of VPS technology for driving navigation purposes.

5. Conclusion

This paper contributes to the growing field of Location-Based Services and showcases the potential of Augmented Reality in transforming driving navigation and parking assistance. By leveraging machine learning, advanced navigation systems, and AR technology, the proposed system offers a promising solution for enhancing urban mobility and creating more sustainable and enjoyable driving experiences. In addition, it addresses the challenges associated with finding available parking spots in metropolitan areas and assists drivers in saving time and energy by eliminating the need for block circling.

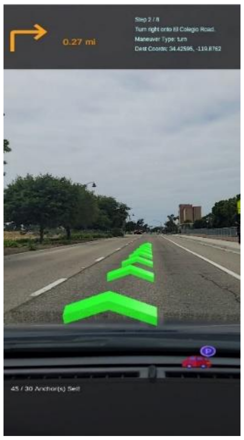

Figure 8: Interactive route navigation using AR markers – straight path

Figure 9: Interactive route navigation using AR markers – curved path

The experiments conducted to evaluate the proposed system have provided valuable insights. While the system generally performs well in accurately placing visual cues and providing interactive driving guidance, there are challenges related to reliability and stability, particularly in localization and object placement. The limitations of VPS technology in highways and roads as well as high-speed driving scenarios, also highlight the need for further improvement.

Despite these challenges, the system shows great potential as an alternative to traditional navigation approaches. When it performs effectively, it offers an immersive and engaging AR navigation experience, which can positively impact user experience and reduce frustration in parking and driving situations.

Future research should focus on addressing the reliability and stability issues by either improving the performance of VPS technology or adding complementary localization methods, particularly for high-speed driving scenarios. Moreover, incorporating a lane detector that accurately identifies the boundaries of the driver's current lane can significantly improve the interpretability of navigation. This ensures that visual cues are appropriately positioned in alignment with the lane.

Figure 10: Mispositioning of heading arrows due to localization loss

References

[1] Google. (2021). ARCore Geospatial API (Version 1.37.0) [Software]. Available at: https://developers.google.com/ar/develop/geospatial

[2] Ul Haque, Hafiz Mahfooz, Haidar Zulfiqar, Sajid Ullah Khan, and Muneeb Ul Haque. "Context-aware parking systems in urban areas: a survey and early experiments." In Context-Aware Systems and Applications, and Nature of Computation and Communication: 7th EAI International Conference, ICCASA 2018, and 4th EAI International Conference, ICTCC 2018, Viet Tri City, Vietnam, November 22–23, 2018, Proceedings 7, pp. 25-35. Springer International Publishing, 2019.

[3] Al-Jabi, Muhannad, and Haya Sammaneh. "Toward mobile AR-based interactive smart parking system." In 2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS), pp. 1243-1247. IEEE, 2018.

[4] Rizov, Tashko, Milan Kjosevski, and Risto Tashevski. "Driver assistance systems in vehicles using augmented reality–benefits and challenges." Trans Motauto World 2, no. 5 (2017): 201-206.

[5] Google LLC. Google Maps [Software]. Retrieved from https://www.google.com/maps

[6] Lisboa de Almeida, Paulo Ricardo, Jeovane Honório Alves, Rafael Stubs Parpinelli, and Jean Paul Barddal. "A Systematic Review on Computer Vision-Based Parking Lot Management Applied on Public Datasets." arXiv e-prints (2022): arXiv-2203.

[7] ParkWhiz. ParkWhiz [Mobile application software]. Retrieved from https://play.google.com/store/apps/details?id=com.parkwhiz.driverApp

[8] SpotAngels. SpotAngels [Mobile application software]. Retrieved from https://play.google.com/store/apps/details?id=com.spotangels.android

[9] kimia A. (2022). Parking-Space-Detection [Source code]. GitHub. Retrieved from https://github.com/kimiaf1998/Parking-Space-Detection

[10] Google LLC. Google Earth [Computer software]. Retrieved from Google Earth website: https://www.google.com/earth/

[11] Mapbox. Mapbox Unity SDK (Version 2.1.1) [Software development kit]. Retrieved from Mapbox website: https://www.mapbox.com/unity-sdk/

[12] Pallets Projects. (n.d.). Flask: A Python Microframework. Retrieved from https://flask.palletsprojects.com/

[13] Amazon Web Services. (n.d.). Amazon Web Services (AWS). Retrieved from https://aws.amazon.com/

[14] Nginx (Version 1.18.0). NGINX. Retrieved from https://nginx.org/

[15] Unicorn. (Version 6.1.0). Unicorn Web Server. Retrieved from https://github.com/defunkt/unicorn