

FilmSpottAR - An Augmented Reality Application for Enhancing the Cinema Experience

Dragoş-Ion Neag

Technical University of Cluj-
Napoca

Str. G. Bariţiu 28, 400027,
Cluj-Napoca, România
Dragosionneag7@gmail.com

Adrian Sabou

Technical University of Cluj-
Napoca

Str. G. Bariţiu 28, 400027,
Cluj-Napoca, România e-mail
address

ABSTRACT

FilmSpottAR is a marker-based AR Android mobile application used for image and object recognition, that makes use of image and model targets. It allows the user to point the camera towards any of the posters displayed in a theatre and receive information, such as trailers, plot, and ratings, which can ultimately lead to the user wanting to see that movie, but also schedule information, allowing the user to book a ticket right away. This system can help cinemas and local theatres promote and invite more people to watch their movies, by simplifying the process of acquiring a ticket. Through these actions, the project brings into the spotlight the advantages Augmented Reality can bring into people's lives. This paper outlines the goals of the project, alongside the stages of development. Use cases and essential features are described, and finally, the user testing results are analyzed.

Author Keywords

Augmented Reality; Location Based Services; Marker-based detection; OpenGL; Mobile Application;

ACM Classification Keywords

H.5.m. Information interfaces and presentation: Human-Computer Interaction; Interaction Techniques;

General Terms

Augmented Reality; Location-Based Services; Image Tracking; Information Gathering;

DOI: 10.37789/rochi.2023.1.1.17

INTRODUCTION

Living in a world where time is extremely valued, people can easily lose their focus unless there is a factor to stimulate their interest. Besides time, money also is an aspect that dictates a lot of decisions an average user makes. Today it seems cheaper and more convenient to subscribe to any of the available streaming platforms, rather than going to the cinema. It advertises the fact that it can be more comfortable and easier to watch movies at home. During this campaign, the theatres remained unchanged without any aggressive promotion and lost the interest of younger people in general.

The main objective of the project is to build a system

capable of enhancing the experience of a user in a movie theatre by allowing him to obtain either a complete program of that specific cinema or details about a movie based solely on its poster.

The term Augmented Reality was first brought up by researcher Tom Caudell, at Boeing in 1990, while attempting to improve marking devices used to guide workers on the factory floor. His idea consisted of plywood floors, containing individually designed wiring instructions, and head-mounted devices which display a plane's specific schematics through advanced eyewear [7]. Overall, AR refers to interactive versions of the real-world environment through virtual elements [13].

Currently, more than 12 distinct classes of augmented reality application domains exist, including well-known domains such as medicine, military, entertainment, and robotics, and also original domains such as education, navigation, or tourism. One field affected in a positive way by the potential of augmented reality is Manufacturing, as AR can enhance the way people understand the product assembly tasks which must be carried out [7].

Technological advancements represented a strong pillar to stand under the evolution of cinema. Having a wider set of tools to work and experiment with, directors were able to have more freedom in the creative process, without having to add boundaries to their imagination.

Being able to insert elements in a scene captured from reality makes room for endless capabilities and opportunities. This project makes use of this innovation and improves the overall experience of a user inside a movie theatre, by allowing him to detect movie posters, usually hung outside cinemas, using his phone and gain information that may lead him into buying a ticket to see that film.

RELATED WORKS

The AR applications allow for a new way of interacting with products in general, offering the public a more engaging shopping experience [6]. As one of this project's goals is to bring exposure back to traditional movie theatres, there are several examples of large-scale companies which experienced marketing benefits following the introduction of such applications. IBM stated that these

kinds of projects can reach 92\% market opportunity [2], as more than 50\% of the consumers prefer receiving information this way while in-store [2]. Although this application is in the initial stage of promoting mostly local theatres, it possesses future potential, as this kind of technology tries to resolve the product uncertainty problem that customers usually face [10].

The entertainment sector is one in particular that made use of AR in the last couple of years. Perhaps the most famous example of augmented reality is represented by the mobile game Pokémon Go. Released back in 2016, it was a huge success, reaching about 232 million users worldwide. The game places fictional creatures, from the famous Japanese series, in the actual world [7]. The initial concept of the application was to distribute the Pokémons throughout various places, such that the user is able to use GPS to try and find them. Because of this, one of its most important features is location-based tracking, as it allows the users to walk around using a real-world map in search of characters.

Another particular application that was developed as an attempt to enhance the marketing potential of movies, was a project built for the Vancouver Turkish Film Festival in 2017, called the VTFF AR app. The idea involved a specific poster, which utilizes Augmented Reality [4], that, when scanned with the application, would display elements from the festival's brochure.

Few systems attempted to develop a behavior similar to the proposed project. The Cinema Station Application took the same approach, attempting to allow users to search for movies and cinema operators, and also purchase tickets to available shows, using AR to display objects that include trailers, ratings, or movie cast [3].

The idea of fetching information based on a recognized target can be implemented in multiple domains, such as art. The application created by the people at the Multimedia University from Cyberjaya aims to enhance the experience of users in museums through introducing AR experience, such that the system can identify objects of interest and obtain information about them in in real-time using feeds from a smartphone camera. This is accomplished via vision-based AR and 3D object tracking, which eliminates the need for markers that can become inaccurate due to damage or obfuscation [9].

Another system that tackles the museum environment is NSMuseum, which can be described as a cross-platform AR solution for art museums [5]. The project seeks to aid the learning process inside a museum by allowing the users to gain information on museum elements. It introduces general information along with audio pieces that can prove helpful for any visitor.

CONCEPTS

Augmented Reality - incorporates items into the real scene in the nearest surroundings, superimposing digital

information over the physical world. Techniques such as depth tracking and the capacity to perceive the environment using computer vision are just a few of the methods used in augmented reality.

Marker-based recognition - markers are represented by distinctive, recognizable patterns, which can be quickly recognized by the AR platform (see Figure 1). They exist and function anywhere, from a QR code, to, in this case, movie posters.

Location-based services - provide the developers with the necessary access to a device's exact position while keeping track of location updates in real time. This type of precision and effectiveness is achieved by the services through a fused location provider, which integrates information from different sources, such as GPS, Wi-Fi, or cellular networks.



Figure 1: Extracting markers in a movie poster

THEORETICAL FOUNDATION

Augmented Reality

AR is the technology that can improve the physical world by superimposing elements such as 3D objects, images, or videos.

The technology resides in three main principles:

- Blending real and virtual environments
- Real-Time Interactivity
- 3D Registration

Image Targets

By scanning pictures and then attaching digital materials to them, images can be successfully transformed into interactive platforms for augmented reality experiences. Displaying 3D objects or other digital content onto an image target happens when the image target is detected by the engine. Hypothetically speaking, a link between the target and the digital content is created, allowing the 3D objects to adapt their position depending on the user's movement around the scene. The engine allows the developer to decide what kind of objects he wants to display when detecting an image target. This content can

take several forms, going from simple 2D images to even more complex, animated 3D objects.

Google Location Services

This application heavily depends on location recognition. This is because of the necessity of acknowledging the current position relative to a cinema. The goal is to recognize which cinema is close to the scanned poster (if any) and verify if the resulting movie is available to watch that day. This can be a complex matter, especially if using tools not sufficiently maintained and updated in relation to the movie theatres' location. Perhaps the most reliable location system is Google Maps. It's only logical to make use of their offered services, hence they can offer not only the highest precision but also the most accurate data. These features are offered by the Google Play services location APIs and can be used for integrating location-based functionality in any Android application.

3D object rendering

Once a target is detected by the engine, the system becomes responsible for rendering whatever information is required on top of the target, as it has the means and the references it needs to do so. Creating the objects, or the digital contents, is a more complex matter, which brings us to this topic - the OpenGL library.

For rendering an object, firstly the geometry must be defined. This means specifying the vertices and topological information of the element. Usually, this vertex data gets stored in the Vertex Buffer Objects (VBOs). These objects are specific buffers designed to hold an array of vertex attributes. Objects are drawn in general using triangles, each one being described by three vertices. Following the initial setup, two shaders must be created to handle the object - one vertex shader and one fragment shader. The vertex shader handles the translation from 3D into 2D screen space. Following that, the fragment one will determine, the correct color of each pixel. After the two shaders were created, the projection and the view transform must be configured. Projection transform shall determine the field of view and the aspect ratio of the projected 3D environment, while the view transform will handle the position and orientation of the camera in the 3D space.

PROPOSED SOLUTION

Objectives

Conventional movie theatres are at a point where advertising isn't as rewarding as they would hope and are slowly driving onto the path of the movie rental shops in the early 2000s, making them more of a nostalgic place rather than a necessity. By adding a fresh and innovative way of finding out what movies a cinema projects, it can encourage younger audiences to attend movie theatres more often.

The way the system is looking to do these things is by:

- Recognizing movie posters and theatres based on specific markers.
- Providing information in the form of objects attached to the interest element.
- Allowing the user to be redirected to the scanned theatre's website in order to purchase tickets to any movie.
- Creating an event in the user's calendar when a ticket is bought.

Use cases

In this case, the use case diagram served as an aid in understanding the flow of the application, along with its main functionalities (see Figure 2). Thus, four main use cases were identified:

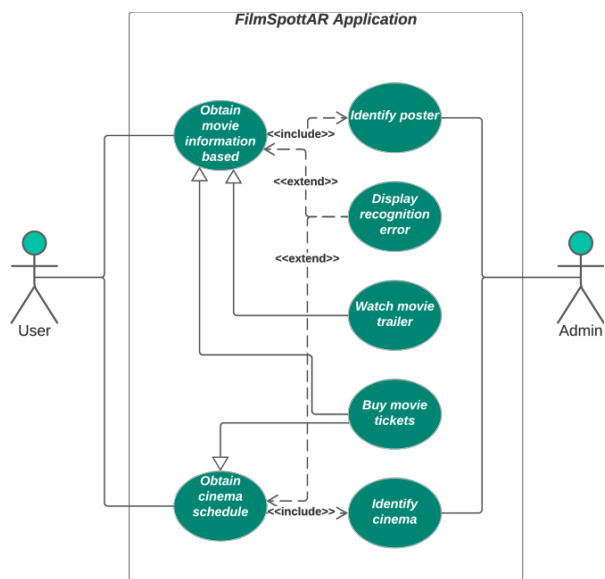


Figure 2: Use case diagram for the system

- Obtaining information about a movie based on its poster.
- Obtaining schedule information of a cinema based on its building and localization.
- Watching a movie's trailer based on its poster.
- Buying tickets to a movie based either on its poster or on the cinema it is played in.

DEVELOPMENT TOOLS

Vuforia

As presented before, Vuforia offers a wide variety of features helpful for developing a seamless AR application. In this case, the goal of the used engine was for it to have advanced image recognition and device tracking. The engine provided by the product performs image recognition very well due to its image target functionalities, based on markers. The device tracking and environmental positioning are given also due to the usage of ARCore functionalities.

OpenGL

The core idea behind OpenGL is a pipeline of operations, where data enters from one and is then processed, manipulated, and rasterized to produce pixels onto the screen. This pipeline was at first a fixed function pipeline, which offered a limited range of predefined functionalities, without giving the developer the ability to modify the algorithms used to transform 3D objects into 2D images. Later, this concept was dropped and the programmable pipeline was introduced, giving programmers the freedom that was missing. This enabled the possibility for developers to determine the color of objects, change the viewpoint, manage textures, and draw polygons in a variety of ways.

Places API

While fetching the current location is important, it must be exploited and try to gain information about the surroundings. For this, the *Places API* comes into play, as it offers access to a sizable database of locations, including shops, landmarks, areas of interest, and so on, allowing us to acquire comprehensive information about these locations. The operation that stands out, in this case, is the feature that allows us to search for locations based on a particular criterion.

Volley HTTP

Being dependent on retrieving information from certain APIs requires high attention to the mechanisms used to fetch that information. Such a mechanism is the one developed by Google, Volley, a library available for Android which offers a simple, yet effective way for sending network requests. It is built to make tasks such as handling answers or collecting data in Android easier. Complex matters like controlling network connections, threading, or response processing are abstracted by the high-level API of Volley.

DESIGN AND IMPLEMENTATION

System Architecture

The general architecture of the system consists of a few principal components, those being the Vuforia Engine, the Android Project, the necessary information providers, and the image target database (see Figure 3). As explained in the previous chapters, Vuforia is the brain of the application, allowing us to make use of Augmented Reality functionalities, such as Image Target recognition, and 3D object rendering in the scene.

The Android project is acting as a controller for the system. It is responsible for calling the methods available in the engine at the proper time, depending on the moment of the lifecycle.

The database holds the image target references that the engine will use to detect a poster and process the information that must be rendered on the screen.

Lastly, the information providers are represented by APIs and movie theatre websites, resources that are necessary for

receiving the required data from the user.

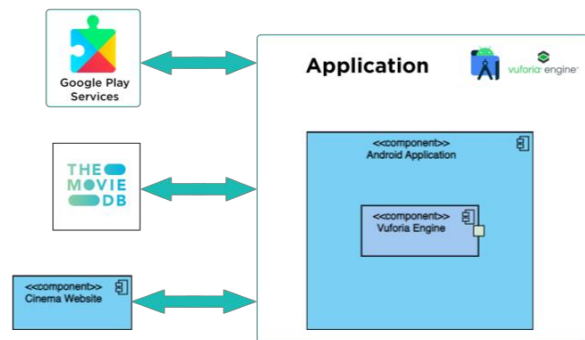


Figure 3: Conceptual system architecture

Vuforia Engine

The core of the application is represented by the Vuforia Engine. As previously described, the engine is the component that allows the use of augmented reality features. It permits us to recognize posters and buildings and display digital content on top of them. Although Vuforia targeted their product at the Unity platform, they created the Vuforia Engine for native platforms, which allows for the development of AR applications using the engine no matter the environment.

The project consists of two main parts: *platform specific code* and *cross-platform code wrapping*.

Information providers

As the final objective of the application is to provide users with useful information about movies, obtaining quality and accurate information is mandatory. The probability of the person wanting to see a film is already existent if that user decides to scan a poster, so more information should only increase that probability. Though, what details he sees is crucial, so they must be picked in order to not be useless or too much.

Movie information API

Finding a resource that can provide all the information you need is pretty hard. Usually, it is split into a number of details, each one being found in another source. This happens in this case too, as there are many movie-related APIs, each one presenting its advantages and disadvantages.

The **Movie Database API (TMDB API)** is an application programming interface built by The Movie Database team, an online platform, that allows users to search for a film, learn details about it, rate it, or save it into a watchlist.

The API follows the principles of REST, allowing the developer to use HTTP methods like GET or POST. The result of the calls is JSON objects, which allow for easy handling of the information. This represents an advantage, as it can be used on a variety of platforms, including Android. As it is a standalone platform, all the movies

presented are labeled with a unique ID, which shall be used for labeling the movie posters in the database.

Web scraping cinema information

The application's focus resides on allowing the user to scan a poster and gain information about the movie. While this feature represents the priority, the user can scan cinema buildings and receive information about the schedule of that day. The details received are not in a specific format, as it depends on the cinema in question how much information it provides on their movies. The reason that they depend on the provided details is because the system is designed in such a way that it uses the cinema's website in order to provide the live information regarding future projected movies.

The way it does this is by making use of **web scraping**, also known as *web harvesting*. This method makes use of software that is able to access a website's source page and extract data from it. Unlike other methods such as *screen scraping*, which can extract pixels displayed on screen, this process can extract underlying HTML code, and data stored in databases along with it [12].

This project makes use of the **jsoup** library, which was designed for Java and is capable of extracting and manipulating data, using methods such as DOM, or CSS. Using this tool, cinema's websites can be accessed and retrieve their daily schedule. As the library not only allows us to receive the source code, but it also possesses the capabilities to parse the information, such that only the things that present interest are shown.

Main functions

Recognizing an Image Target

The process of recognizing represents one of the most important, if not the most important process in this project. The ability of the user to recognize movie posters or cinema buildings just by pointing the camera at them is given by the marker-based technology used in Vuforia.

In order for the engine to recognize images, it must have some references that it can compare the surroundings with. This is accomplished thanks to the existing observers, that were created by extracting information from the respective databases.

This process begins inside the VuforiaActivity, which initializes the engine and configures the rendering. If the configuration has finished successfully, the rendering of the video background and of the augmented elements is launched.

Obtaining nearby cinemas

The flow of the application is designed to assume that the application will most likely be used while being around a movie theatre, as most movie posters are found there. Hence, when a user intends to scan a poster, the application

will immediately check for his/hers current location, and attempt to find a cinema on the map. As previously explained, the Google Play services will be used, more specifically the Fused Location Provider API.

- **Obtaining the current location**

When the application is started, a `FusedLocationProviderClient` will be created, which will be responsible for retrieving the last known location using the `getFusedLocationProviderClient()` method. In order for that to be possible, the device should dispose of two location permissions:

- **ACCESS_COARSE_LOCATION**
- **ACCESS_FINE_LOCATION.**

After making sure these are allowed, the `lastLocation` method can be called, which returns an object describing the last known location, including information such as geographical coordinates, which will be used in determining nearby places.

- **Searching for nearby cinemas**

After obtaining the device's current coordinates, API call can be made to check if there are any movie theatres around the current location. The *Places API* requires a few parameters when receiving a *GET* request. The *type* of such places is set to *movie_theater*, the *keyword* is set to *cinema*, the *radius* is set to 100 meters, and finally, the *location* is represented by the current location's coordinates: `currentLocation.latitude`, `currentLocation.longitude`. The API will return a list of results that match the current search. The response will be deserialized using *GSON*, the only necessary information will be acquired, that being the cinema's **name**.

Fetching necessary information

Considering that the system successfully identified an image target, it may proceed to retrieve the necessary information that should appear on the digital content displayed on top of the target. The reason it needs to start with the fetching of movie or cinema details is that the way the text is displayed on 3D objects is through creating textures. These textures must be created before the object rendering is done so that they can be passed correctly to the rendering function.

Once a target is recognized, the system continuously informs us that it found it. A *boolean* variable is created that is meant to run a code only once after an observation was found.

- **Calling an API**

Unlike C++, Kotlin disposes of a variety of mapping and formatting methods, that allow for an easy JSON to data class conversion.

After a request was sent, and the response was

received, it will be stored inside data classes that respect its JSON format. The way the JSON data is deserialized into classes was by using GSON.

- **Scrapping web pages**

As was described previously, the cinema recognition will return the information provided by every theatre's respective web page.

Using an appropriate web scraping tool does make the coding more efficient. The difference is made by the headers or tags which carry the required information. Because of this, by using **jsoup** the algorithms can be kept quite similar. The fetched HTML code should be parsed and only the headers that contain the interest details selected. A data model was created in order to generalize the information received from the cinemas in question called *CinemaMovie*.

Rendering digital content

Having recognized an image target, and fetched the necessary information that should be displayed over it, the system can proceed to the rendering part. In the *VuforiaWrapper* file, the algorithm gathered the required details, and it must now render the objects. The way in which text is printed onto objects is by creating and applying **textures**.

- **Generating bitmaps**

In order to draw on the bitmap, a structure must be used that is able to handle commands like that. *Canvas* structure was chosen, which allowed us to change the color of the bitmap through the method *drawColor*. For drawing text, however, a *Paint* object must be created first, in order to store details such as text size, or text color. Finally, to draw the text onto the bitmap the *drawText* method is used, which will draw any text at the computed coordinates, according to the specified details. These coordinates are also computed taking into account where the center of the bitmap is.

- **Converting bitmaps into textures**

Every movie poster a user might scan will contain a certain number of details. Displaying these details in an appropriate way is crucial, as it must be done in an attractive and visible manner. The conclusion is that it would require more than one object. This obviously implies the usage of multiple textures.

Since there are multiple textures required, the method generating the bitmaps has to provide as output a list of such structures.

After successfully gathering the necessary information, by using the *glGenTextures* function a new texture object can be generated with a specific ID. This texture must be bound to the current context and set certain specific parameters, that can provide a smooth appearance to it when being scaled.



Figure 4: Applying textures on 3D objects [1]

Creating the 3D objects

There are a few prerequisites necessary before rendering the objects themselves.

- **Projection matrix** is used for representing the 3D objects on 2D screens.
- **Model View Matrix** is used for geometric transformations, such as translations or rotations, that can help us maneuver the object in the scene.
- **Scaled Model View Matrix** refers to a similar matrix as the one above but with extra scaling applied to the object.
- **List of textures IDs** represents an array of IDs, which will be used for applying the adequate textures on each object.

Having these elements prepared, *renderImageTarget* function is called. This function's responsibility is to draw the 3D objects, after applying the necessary textures onto them. The way this is done is by setting up different model view matrices, in order to display multiple objects in their respective positions according to the given reference.

Handling touch in the 3D objects

After various tries, it has been decided that the most efficient and effective solution is to bring the object into screen space. More specifically, to bring two of its coordinates into screen space: the bottom-right corner, and the top-left corner. As the object is a rectangle, its boundaries must be known such that the touch point can be placed inside, or outside its area.

Thus, for each button that a listener must be added to, its translation matrix must be known, used to bring it into the current pose. Knowing this information, the initial rectangle coordinates can be converted into screen space and make a comparison between the touch point coordinates and the rectangle ones (see Figure 5).

REQUIRED RESOURCES

The application was developed to be supported exclusively on Android devices, as it uses certain native functionalities. In order for a user to be able to use the application, he must possess a device that satisfies the following criteria:

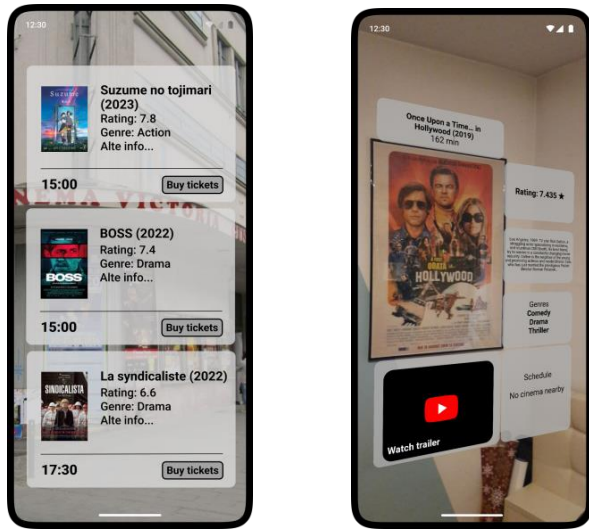


Figure 5: Screenshots of the system

- The device is an Android device, whose OS version is 8.0 (OREO) or later.
- The device has a main camera in working condition.
- The device permits and has access to internet services.

USER TESTING

In order to check the system's capabilities in real-life situations, a group of 6 users was chosen to try the application's final version (see Figure 6). The overall purpose was to see if the project was constructed in such a manner that a person without advanced technical knowledge can use the application in an effective way.

For this phase, a number of six people participated, each of them with different backgrounds and different technological understanding. Two out of the six people were medicine students, two automobile constructions degree students, one was also a student enrolled in a business master's degree, and the last one was a junior programmer.

In the beginning, each person was informed about the context of the application, its functionalities, along with the expected behavior and goals.

RESULTS

Each person involved was somehow skeptical of the capabilities of the project but was pleasantly surprised by what he or she discovered. Everybody was intrigued by the manner the information was displayed and found it extremely immersive. The interaction with the posters, and also with the cinemas, was a fun and enjoyable one.

In the end, the testers agreed on the impact the project would have on conventional movie theatres in terms of user engagement. Each student pointed out that it would definitely encourage them to visit cinemas more often, as they were unaware of what kind of movies they could see in the local theatres.



Figure 6: User testing the application capabilities

Improvements

Apart from giving their opinion on the potential of the project, each test user was asked about what improvements or additional features they would bring into the system. Everybody was extremely creative and gave some interesting responses and suggestions. Two users pointed out that they would improve the visual aspect of the 3D elements, by changing the font and by displaying the information in a different manner. Other than that, everybody was satisfied with the current version of the application.

Ratings

Finally, every participant was asked a series of questions that required evaluation on a scale from 1 to 10.

Usability assessment (How easy is the app to use?)

The participants were asked to rate the system based on the ease of understanding the application. The majority of five out of six respondents gave a rating of 1, which indicates an incredibly high level of ease ($M = 9.83$, $SD = 0.15$). One person gave it a score of 9, which indicates a somewhat lower but still positive impression of usability.

Information Utility (How useful are the details displayed on the AR elements?)

The second question sought thoughts on the information's perceived usefulness. Ratings of 8 and 9 were predominant in responses ($M = 8.83$, $SD = 0.49$), suggesting that participants acknowledged the value of the material but also noted opportunities for improvement.

Design Assessment (How would you rate the overall design of the system?)

Various opinions were expressed regarding the third question. Some participants gave the design their full support and expressed satisfaction with it, without bringing any improvement ideas. Others, on the other hand, thought the design was a good start but recommended changes to the typeface and presentation methods.

Focus of the application (How useful do you find the AR/location tracking capabilities in the context of the application?)

The main functions of the application were universally acknowledged by the participants, who highlighted its emphasis on Augmented Reality (AR) and location-tracking features.

Engagement Assessment (How enjoyable was the usage of the application?)

In the last question, users evaluated how engaged they were with the application. Remarkably, every participant had a high degree of engagement, scoring 9 or 10 ($M = 9.67$, $SD = 0.44$).

SUMMARY AND CONCLUSIONS

Augmented Reality brought innovative functionalities that allow for the integration of virtual objects into the real world, which can represent a great change to the day-to-day life of the average user. This applies to the entertainment industry as well. This thesis aims to propose a useful solution to enhance the movie-watching experience. The application fills the gap between conventional movie posters and digital material through the introduction of AR capabilities. Not only this, but it renews the interaction people have with movie theatres.

The achievements of the project in the Augmented Reality area open the door to possible improvements that can be brought in the entertainment sector by making use of this technology. Previous projects attempted to create a more personalized environment, in which customers can immerse themselves, but without adding a true award for doing so. This project is changing the marketing perspective of movie theatres and many more.

The system was created as an aid to movie enthusiasts and movie theatres. It attempts to enrich the movie-going experience and engage the viewers in a different, new way. The results of the project reflect this intention, as an AR-based application capable of detecting movie posters and theatres and bringing up related information was successfully created. Because of this project, viewers are able to book a movie ticket by simply scanning a poster, which will automatically know at what theatre the client is.

ACKNOWLEDGEMENTS

Would like to thank the available papers that stood as a great reference for this project, along with the testers that made themselves available for giving feedback on the application.

REFERENCES

- [1] 3d texturing in animation production; a short overview of the workflow. [Online]. Available: <https://dreamfarmstudios.com/blog/getting-to-know-3d-texturing-in-animation-production/>. Accessed April 16, 2023.
- [2] Adhani, N.I. and Rambli, D.R.A., 2012. A survey of mobile augmented reality applications. In 1st International conference on future trends in computing and communication technologies (pp. 89-96).
- [3] Alkhuraiji, S.M., 2020, June. Cinema Station Application (CSA): An augmented reality application for accessing film information and online cinema booking. In 2020 8th International Symposium on Digital Forensics and Security (ISDFS) (pp. 1-6). IEEE.
- [4] (2017) Augmented reality brings film festival posters brochures to life. [Online]. Available: <https://www.stambol.com/2017/05/29/augmentedreality-brings-film-festival-posters-brochures-life/>. Accessed April 21, 2023.
- [5] Dragota, M.I. and Sabou, A., 2021. NSMuseum-A Case Study for Developing Cross-platform Mobile Applications in Augmented Reality. In RoCHI (pp. 74-81).
- [6] McLean, G. and Wilson, A., 2019. Shopping in the digital world: Examining customer engagement through augmented reality mobile applications. *Computers in Human Behavior*, 101, pp.210-224.
- [7] Mekni, M. and Lemieux, A., 2014. Augmented reality: Applications, challenges and future trends. *Applied computational science*, 20, pp.205-214.
- [8] Selvam, A., Yap, T.T.V., Ng, H., Tong, H.L. and Ho, C.C., 2016. Augmented reality for information retrieval aimed at museum exhibitions using smartphones. *Journal of Engineering and Applied Sciences*, 100(3), pp.635-639.
- [9] Serino, M., Cordrey, K., McLaughlin, L. and Milanaik, R.L., 2016. Pokémon Go and augmented virtual reality games: a cautionary commentary for parents and pediatricians. *Current opinion in pediatrics*, 28(5), pp.673-677.
- [10] Tan, Y.C., Chandukala, S.R. and Reddy, S.K., 2022. Augmented reality in retail and its impact on sales. *Journal of Marketing*, 86(1), pp.48-66.
- [11] Web scraping attack. [Online]. Available: <https://www.imperva.com/learn/application-security/web-scrapingattack/:text=Web%20scraping%20is%20the%20process,replicate%20entire%20website%20content%20elsewhere>. Accessed April 19, 2023.
- [12] What is augmented reality (ar)? [Online]. Available: <https://dynamics.microsoft.com/enus/mixed-reality/guides/what-is-augmented-realityar/:text=Augmented%20reality%20is%20an%20enhanced,sensory%20stimuli%20via%20holographic%20technology>. Accessed May 15, 2023.