Taibah University
College of Computer Science Engineering
Computer Engineering Department

<TITLE OF THE PROJECT>

Indoor environments navigation assistant, guidance, and localization system for visually impaired and blind individuals

A Project Submitted in partial fulfilment of the requirements for the Bachelor Degree in Computer Engineering

Submitted By

<Name Lastname><Student ID Number>
<Name Lastname><Student ID Number>
<Name Lastname><Student ID Number>

Project Advisor
<Title><Name Lastname>

<Year>-<Term>

ABSTRACT

Visually impaired and blind individuals might face difficult challenges when navigating through indoor environments, especially if they where not familiar with them. Indoor environments can be very unpredictable and filled with obstacles, and they are rich of important visual information which only designed for eyes consumption. These difficulties rise the need of these individuals to assistants and guiders whether by asking strangers, or by taking companions. But these solutions will lead to decreasing the independence of these individuals. We are trying to offer a solution in this project, a wearable and easy to use system that guides and assists the users after determining their positions and scanning their surrounding environments using only a camera and smart phone.

The proposed system is an integration of three systems: Objects Detection, Localization, and Customizable Guidance System. These components rely on analyzing the images captured by a camera and extracting the important visual information from them. The Objects Detection System can detects the surrounding objects and describe them to the users if they asked to, and informusers about obstacles, and users can ask it to alarm them when they face an object with certain characteristics, such as a man wearing red T-Shirt. The localization system will use the camera to detect artificial landmarks to determine the user's location. And finally, the Customizable Guidance System reads to user custom information and instructions after entering specific areas.

- the result obtained

- the significance of the result or finding

KEYWORDS: Localization; Obstacles Avoidance; Objects Detection, Customization; Guidance; Pose; Calibration

ACKNOWLEDGEMENTS

The content of this single page is left to the preference of the student. It is suggested however that the page makes reference to guidance received by the student from his or her supervisor and project Committee members. Reference should also be made to any financial assistance received to carry out the project. Any extraordinary assistance received by the student for example in word processing, data collection, data analysis, and so on, should be properly acknowledged. The acknowledgements should not exceed 250 words.

Contents

List of Tables

List of Figures

Glossary and List of Abbreviations

Glossary

Term 1 Definition of term 1.

Term 2 Definition of term 2.

List of Abbreviations

Abbr. Full form of the abbreviation.

Chapter 1

Introduction

1.1 Overview

Navigation in indoor areas presents considerable difficulties for those with visual impairments. Conventional GPS systems, while proficient in outside navigation, are inadequate for indoor environments due to diminished satellite signals and the intricate designs of spaces such as retail malls, hospitals, and educational institutions. Researchers have investigated alternate ways, including the use of artificial landmarks, to facilitate indoor navigation.

Artificial landmarks, including QR codes, offer a cost-effective and readily implementable alternative for localization and navigation in interior settings. These landmarks are identifiable by cameras or sensors, enabling visually challenged individuals to obtain audio or tactile feedback for navigational assistance. This project proposes a QR code-based navigation system to ensure precise localization and dependable obstacle detection for those with visual impairments. This project aim to develop a low-cost indoor navigation system that will combine accurate indoor localization, reliable obstacle detection, and customizable, for indoor environment using (. . . Tools and Software and QR code) to help with visual impairments.

1.2 Motivation

This project is motivated by the need to enhance the quality of life and autonomy for visually impaired individuals, especially in interior settings where conventional navigation aids such as GPS are inadequate. Visually impaired individuals encounter significant obstacles when navigating intricate indoor environments, which can restrict their mobility and independence. Existing options, such as guide dogs and human aid, although beneficial, may prove inadequate or unworkable in all situations. The motivation also is to develop a prototype solution utilising commonly accessible technology (such as QR codes and mobile devices) enhances the system's accessibility to a broader audience (Cost and Accessibility). This differs from expensive or complex systems that may require specialized hardware.

1.3 Problem Statement

Navigating indoor environments is challenging for visually impaired individuals, directly affecting their independence and daily activities. While some existing technologies have developed separate algorithms for indoor localization, they are often not integrated into a user-friendly device that individuals can easily use.

Moreover, many solutions rely on expensive sensors, increasing the overall cost of the device. Additionally, the majority of current aids do not provide a cohesive approach that combines accurate localization with obstacle detection, and they often lack context-specific, non-visual instructions tailored to the unique needs of visually impaired users. This highlights the urgent need for an affordable, integrated indoor navigation system that enhances mobility and independence in everyday settings.

1.4 Project Objectives

1.5 Project Outline

The remainder of this report is organised as follows: Chapter 2 provides the Background and the related works. Chapter 3 provides the project system design, and Chapter 4 covers the implementation. Chapter 5 provides the results obtained by the system, and Chapter 6 concludes the report.[1]

[1] Next Term. But we can mention it from now.

Chapter 2

Background

2.1 Introduction

This chapter is organized as follows: Section 2 discusses the role of landmarks in navigation, emphasizing artificial landmarks such as QR codes. Section 3 explores the structure, versions, and encoding of QR codes, while Section 4 focuses on QR code-based localization systems. Section 5 covers object detection, followed by Section 6, which addresses customizable guidance systems. Finally, Section 7 reviews related work in the field.

2.2 Landmarks

Landmarks are essential in navigation, tracking, and mapping, helping to determine positions within an environment. They can be categorized into two types: natural and artificial.

2.2.1 Natural Landmarks

Natural landmarks are features formed by nature or naturally present in an environment, such as mountains, rocks, trees, stairs, or doors. In computer science, they can be used in computer vision algorithms to help determine object locations, create 3D maps, or monitor object states. However, natural landmarks are more difficult to recognize due to their diversity, lack of uniform shapes, and inability to encode data.

2.2.2 Artificial Landmarks

Artificial landmarks, on the other hand, are human-made structures like traffic signs, statues, and QR codes. These landmarks offer several advantages over natural ones in computer vision applications due to their uniformity, ease of recognition, ability to encode data, and customization options. Artificial landmarks are more reliable, precise, and adaptable, making them better suited for controlled environments. Furthermore, many libraries support the encoding, detection, decoding, and tracking of artificial landmarks. Examples include various markers such as QR codes, Aruco markers, and Topotags, which are widely used in different computer science fields. More details on different types of markers can be found in reference [1].

2.3 QR code

Quick Response (QR) codes are a type of matrix barcode designed for efficient data encoding and rapid scanning. QR Codes first developed in 1994 by Denso Wave, to track automotive parts in manufacturing. but have since become widely used across various applications due to their versatility, simplicity, and capacity to store significant amounts of information compared to traditional barcodes.

2.3.1 QR Structure

QR Codes can be detected and decoded at various angles and distortion levels due to their unique structure. The code consists of black modules (squares) arranged on a white background in a grid pattern, allowing for rapid and error-resistant scanning. All QR Codes have standard structure as shown in Figure 2.1. This structure is made out of the following parts:

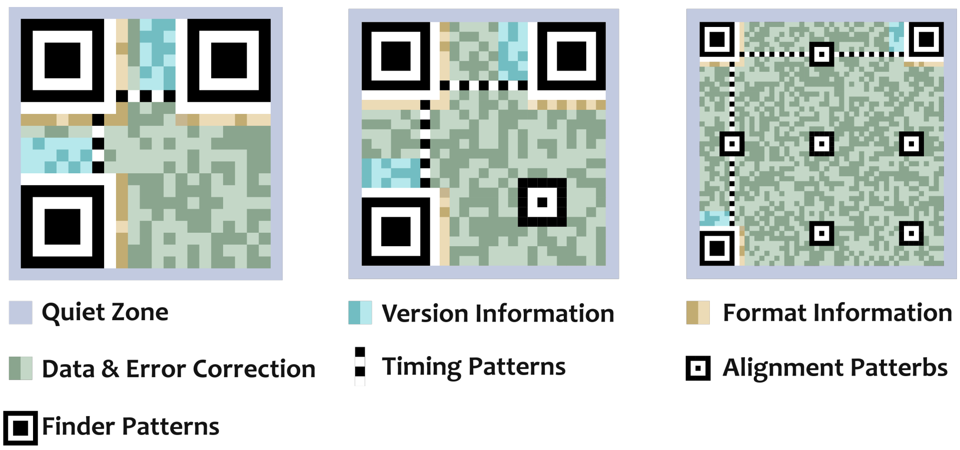

Figure 2.1: These are three QR Codes that store texts. The texts' lengths get larger going from the left to the right. Notice that the alignment pattern only appears at the bigger QR Codes, and there are multiple ones at the biggest QR Code.

- **Quiet Zone:** This is a white empty area surrounds the QR Code that helps distinguishing it from its surroundings.
- **Version Information:** QR Codes have different versions/sizes which specified by these two areas.
- **Format Information:** This part provides details about the error correction level and data mask pattern used.
- **Data & Error Correction:** This is where both encoded data & error correction are. data and error correction are stored together enabling the

QR code to recover and reconstruct the stored data, even if up to 30% of the code is damaged or obscured. Data have different types, such as text, URLs, or other.

- **Timing Patterns:** This part is essential for defining the grid's structure and assists the scanner in establishing the size and coordinate system of the code.
- **Alignment Patterns:** The Alignment Patterns help correct distortion and skewing of the QR code when viewed from different angles. It is especially crucial for larger QR codes that may be prone to bending or misalignment.
- **Finder Patterns:** These patterns assist the scanning device in rapidly locating and orienting the QR code, regardless of its rotation or angle.

See [8], for more information.

2.3.2 QR Code Versions and Types

QR codes come in 40 versions, each representing a different size and data capacity. Version 1 contains 21×21 modules, while Version 40 has 177×177 modules. As the version number increases, so does the data capacity and the complexity of the code, making it capable of storing more information or supporting higher levels of error correction [8].

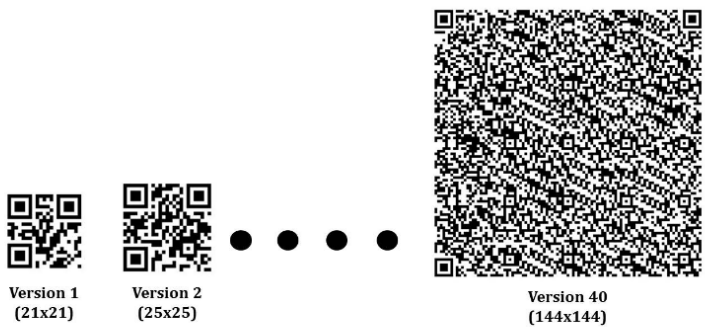

Figure 2.2: QR codes Versions (adapted from [8]).

There are also specialized QR code types designed for specific applications:

- **QR Code Model 1 and 2**: Model 1 is the original version of the QR code, developed in 1994 by Denso Wave. While model 2 is an improved version of model 1, and it is the one most commonly used today. These models are used for daily basis.

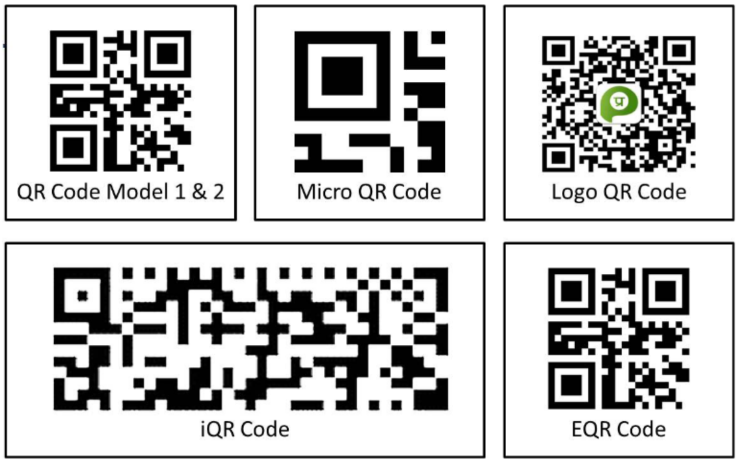

Figure 2.3: QR code types (adapted from [8]).

- **Micro QR Code**: Designed to be smaller and simpler, it uses only one Finder Pattern, making it more compact than traditional QR codes. It is often used when space is limited [8].
- **Logo QR Code**: Allows logos or images to be embedded within the QR code, enhancing the code's visual appeal for marketing or branding purposes [8].
- **iQR Code**: A flexible matrix-type QR code capable of being printed in various sizes and configurations, from small, high-capacity codes to large codes. It can store more data than standard QR codes and can be inverted or turned into dot patterns for direct part marking [8].
- **Encrypted QR Code**: Uses encryption techniques to secure the information encoded within, making it suitable for applications requiring data confidentiality [8].

2.3.3 QR Code Encoding

The encoding process for a QR code involves converting data into a matrix of black and white modules. First, the input data is analyzed to determine the appropriate encoding mode, such as numeric, alphanumeric, byte, or Kanji. The data is then transformed into binary code, and error correction is added using Reed-Solomon algorithms, ensuring that the code can still be scanned if partially damaged [8].

Popular online tools for generating QR codes include:

- **Online Tools**: These free online tools offer simple and fast solutions for both generating and decoding QR codes. QRickit allows users to decode

QR codes from uploaded images, while GOQR.me provides various output formats, such as PNG, SVG, and EPS, making it versatile for different use cases [10][11].

For generating QR codes programmatically, popular libraries include:

- **Python - PyQRCode**: A Python library that simplifies QR code creation, allowing output in SVG, PNG, and other formats [12].
- **Java - ZXing (Zebra Crossing)**: An open-source library widely used for encoding QR codes in Java and Android applications [13].
- **iOS - Core Image**: iOS provides native QR code generation capabilities via the ‘CIQRCodeGenerator‘ filter in the Core Image framework [14].

2.3.4 QR Code Decoding

Decoding a QR code starts by scanning the Finder Patterns, which allow the scanner to properly align the code. Next, the Format Information is read to apply the correct error correction and mask pattern. Once decoded, the binary data is translated back into the original format (e.g., text, URL) [8].

Common online tools for decoding QR codes include:

- **Online Tools**: These tools offer versatile QR code decoding solutions across platforms. ZXing is an open-source decoder integrated into Android and web applications, QRickit provides online QR code decoding from uploaded images, and Google Lens allows users to scan and decode QR codes directly via smartphone cameras [13, 18].

For decoding programmatically, useful libraries include:

- **Python - Segno**: A versatile Python library for generating and reading both QR and Micro QR codes [15].
- **Java - ZXing**: Provides decoding functionality along with encoding, and is widely used for mobile and web apps [13].
- **iOS - AVFoundation**: iOS provides native support for scanning QR codes using the ‘AVCaptureMetadataOutput‘ class in the AVFoundation framework [16].

2.4 Indoor Localization

Indoor localization refers to the process of determining the precise position and orientation of an object or individual within indoor environments, where traditional Global Positioning System (GPS) signals are often unreliable. This technology is critical for a wide range of applications, including navigation within large buildings and asset tracking in warehouses.

Various technologies are employed for indoor localization, each with distinct advantages and limitations. Wi-Fi, Bluetooth, Radio-Frequency Identification (RFID), and Ultra-Wideband (UWB) are among the most widely used methods. Some systems rely on complex and expensive hardware, such as sensors and Inertial Measurement Units (IMUs), while others utilize more cost-effective solutions, including Bluetooth beacons and pose estimation using Quick Response (QR) codes. The selection of the appropriate technology depends on the specific requirements of the application, taking into account factors such as accuracy, cost, and ease of implementation [9].

As the field of indoor localization continues to evolve, new techniques and innovative approaches are emerging, such as the use of QR codes for precise positioning and tracking.

2.4.1 Indoor Localization with QR Codes

Indoor localization using QR codes enables position determination by detecting QR codes strategically placed throughout an indoor space. These codes can be affixed to floors, walls, ceilings, or suspended on hanging panels, each containing encoded positional information.

2.4.1.1 QR Code Placement

The placement of QR codes plays a critical role in ensuring effective indoor localization. Codes can be positioned in various locations:

- **Ceilings**: QR codes on ceilings are often out of the way and can provide an unobstructed view for overhead cameras, such as those mounted on hats or handheld devices. This placement is ideal for applications where the user's line of sight remains upward.

- **Walls**: QR codes can also be positioned on walls at different heights to accommodate various camera angles. This setup is particularly useful when the user or device is at eye level with the code.

- **Floors**: Placing QR codes on floors can be beneficial in environments where overhead cameras or downward-facing sensors (such as those on a robot or mobility aid) are used. However, codes on floors might be more prone to wear and tear and may require periodic maintenance.

- **Hanging Panels**: In environments where flexibility is needed, QR codes can be placed on hanging panels suspended from the ceiling. This allows for better visibility while keeping the codes elevated from foot traffic or other obstacles.

2.4.1.2 Data Encoding in QR Codes

Each QR code encodes essential information for localization purposes, including:

- **Coordinates**: The most important data that QR codes can encode are the precise coordinates of their position within the environment. These coordinates are predefined and allow the system to accurately calculate the user's or object's location when the QR code is detected. The coordinates might be in terms of x, y, z positions or based on a grid system specific to the environment.

- **Unique Identifier (ID)**: In addition to coordinates, each QR code will have a unique ID that differentiates it from others in the system. The ID can be referenced in the system's database to retrieve additional information, such as the room name or floor level.

- **Orientation Data**: QR codes can also encode information about their orientation, which helps in determining the user's orientation in relation to the environment (e.g., the angle of the code in reference to a global axis).

- **Additional Metadata**: If needed, QR codes can encode further metadata, such as room names, nearby landmarks, or points of interest. This is especially useful in environments where additional contextual information enhances the user's navigation experience.

2.4.1.3 Approaches to Localization with QR Codes

Several methods exist for implementing indoor localization using QR codes, each with its own benefits and trade-offs. Two common approaches include the divided environment method, which offers simplicity and low computational requirements, and the pose estimation method, which provides higher precision but at the cost of increased computational complexity. These approaches are discussed below.

The Divided Environment Method In the divided environment method, the space is partitioned into distinct pieces(e.g., squares, triangles, or hexagons), with each piece containing a QR code encoding its position. QR codes can be placed on the floor, ceiling, walls, or hanging panels depending on the system's configuration. A similar approach has been demonstrated by Zhang et al. [3].

To illustrate this method, consider a room measuring 4 meters in both width and height, divided into 16 equal squares, each with an area of $1m^2$. If we customized a hat for example, that embeds a camera in its top, if we put the QR codes at the ceiling, then the user's position will get determined while navigating in the room wearing the hat.

Although this method is computationally efficient, it has limitations. The position values are discrete, which may not provide the continuous and precise localization required in certain applications. Nonetheless, this method can be highly effective for specific use cases, as discussed in Appendix A.

Pose Estimation with QR Codes Another method for indoor localization involves calculating the relative position and orientation (pose) of the QR code with respect to the camera. Upon detecting a QR code, the system estimates its position and orientation relative to the camera and, by combining this data with the known global position of the QR code, determines the user's precise and continuous position within the environment.

While this approach offers significantly higher accuracy, it requires greater computational resources due to additional processes, such as camera calibration to determine intrinsic parameters. This increased complexity makes the method more resource-intensive compared to the divided environment approach. However, it is suitable for applications where continuous and precise localization is essential, as demonstrated by Lucag et al. [4].

In a well-designed system, the effectiveness of localization remains high, regardless of the specific locations of the QR codes. By optimizing QR code placement and system design to match the environment, the indoor localization system can provide robust performance, whether prioritizing simplicity or precision.

2.5 Objects Detection

2.6 Customizable Guidance

2.7 Related Work

In localization research, artificial landmarks have been widely utilized due to the challenges associated with detecting and building systems based on natural landmarks. Among artificial landmarks, QR codes have garnered significant attention because of their ease of creation, low implementation costs, and the substantial amount of information they can encode.

The work of Zhang et al. [3] presents an approach for indoor mobile robot localization using QR codes arranged in a grid pattern on the ceiling. The system uses an industrial camera mounted on top of the robot, facing upward to detect these QR codes. The camera captures images of the QR codes, and a recognition algorithm processes the codes' position and orientation within the image. By leveraging the coordinates of each QR code and applying camera calibration data, the robot can accurately determine its position. This setup enables real-time, precise localization, which is crucial for efficient navigation in structured indoor environments.

Kim et al. [5] introduce a vision-based indoor positioning system that employs QR codes and a smartphone camera to accurately determine a user's indoor location. In this system, QR codes are placed at predefined locations and detected by the smartphone camera. The two-dimensional coordinates from the QR codes are converted into three-dimensional spatial coordinates using camera calibration techniques. By forming a quadrilateral shape from reference symbols on the QR codes, the system calculates the center of gravity to determine the user's position and orientation. Experiments demonstrated an average localization error of less than one meter, highlighting the system's high accuracy.

Lee et al. [6] propose a cost-effective indoor localization method for mobile robots, using QR codes as artificial landmarks. QR codes are strategically placed on the ceiling, and a smartphone mounted on the robot detects these codes to determine the robot's position and heading direction. The positions of the QR codes are pre-stored in a database, enabling the robot to compute its real-world coordinates by processing image data captured by the smartphone's camera. The system has been experimentally validated, showing localization errors ranging from 3.2 cm to 6.55 cm, confirming its accuracy.

We should mention as a conclusion of this section how our solution will differ from them whether by accuracy, multiple tech integration, targeted audience, real world validation method, etc.

Chapter 3

System Design

3.1 System Design

The system design chapter presents the overall architecture and key components.

Chapter 4

Results and Discussion

4.1 Results

This chapter presents the results obtained from the system.

Scenario	**Expected Result**	**Actual Result**
Scenario 1	Success	Success
Scenario 2	Failure	Success

Table 4.1: Results of Various Scenarios

Table 4.1 shows the results obtained during testing in various scenarios.

Chapter 5

Conclusion and Future Work

5.1 Conclusion and Future Work

The project concludes by summarizing the findings and suggesting future work.

References

[1] Michail K., Brennan C., Sabrina C., Anand A., Camden W., & Nikolaos V. (2021). Fiducial Markers for Pose Estimation. Journal of Intelligent & Robotic Systems (2021) 101:71. https://link.springer.com/article/10.1007/s10846-020-01307-9

[2] Tiwari, V., "QR Code Technology and Its Applications," International Journal of Advanced Research in Computer Science and Software Engineering, vol. 6, no. 4, pp. 123-127, 2016.

[3] H. Zhang, C. Zhang, W. Yang and C. -Y. Chen, "Localization and navigation using QR code for mobile robot in indoor environment," 2015 IEEE International Conference on Robotics and Biomimetics (ROBIO), Zhuhai, China, 2015, pp. 2501-2506, doi: 10.1109/ROBIO.2015.7419715.

[4] Luca C., Gionata C., Francesco F., Alessandro F., Gianluca I., Andrea M. Federica V. (2017). A QR-code Localization System for Mobile Robots: Application to Smart Wheelchairs. https://ieeexplore.ieee.org/document/8098667

[5] J.-I. Kim, H.-S. Gang, J.-Y. Pyun, and G.-R. Kwon, "Implementation of QR Code Recognition Technology Using Smartphone Camera for Indoor Positioning," Energies, vol. 14, no. 10, p. 2759, 2021, doi: 10.3390/EN14102759.

[6] S. -J. Lee, G. Tewolde, J. Lim and J. Kwon, "QR-code based Localization for Indoor Mobile Robot with validation using a 3D optical tracking instrument," 2015 IEEE International Conference on Advanced Intel-

ligent Mechatronics (AIM), Busan, Korea (South), 2015, pp. 965-970, doi: 10.1109/AIM.2015.7222664.

[7] M. Kalaitzakis, B. Cain, S. Carroll, A. Ambrosi, C. Whitehead, and N. Vitzilaios, "Fiducial Markers for Pose Estimation: Overview, Applications, and Experimental Comparison of the ARTag, AprilTag, ArUco, and STag Markers," Journal of Intelligent & Robotic Systems, vol. 101, no. 71, pp. 1-26, 2021, doi: 10.1007/s10846-020-01307-9.

[8] S. Tiwari, "An Introduction to QR Code Technology," 2016 International Conference on Information Technology (ICIT), Bhubaneswar, India, pp. 39-44, 2016, doi: 10.1109/ICIT.2016.021.

[9] S.G. Leitch, Q.Z. Ahmed, W.B. Abbas, M. Hafeez, P.I. Laziridis, P. Sureephong, and T. Alade, "On Indoor Localization Using WiFi, BLE, UWB, and IMU Technologies," Sensors, vol. 23, no. 20, pp. 8598, 2023, doi: 10.3390/s23208598.

[10] "QRCode Monkey - Free QR Code Generator," QRCode Monkey, Available: `https://www.qrcode-monkey.com`.

[11] "QR TIGER - Free QR Code Generator," QR Tiger, Available: `https://www.qrcode-tiger.com`.

[12] "PyQRCode Documentation," PyQRCode, Available: `https://pyqrcode.readthedocs.io`.

[13] "ZXing Decoder," ZXing Project, Available: `https://github.com/zxing/zxing`.

[14] "Core Image Framework," Apple Developer Documentation, Available: `https://developer.apple.com/documentation/coreimage`.

[15] "Segno - Python QR Code Generator," Segno Documentation, Available: `https://segno.readthedocs.io`.

[16] "AVFoundation Framework," Apple Developer Documentation, Available: `https://developer.apple.com/documentation/avfoundation`.

[17] https://www.youtube.com/watch?v=-L-WgKMFuhE&list= PLFtAvWsXl0cq5Umv3pMC9SPnKjfp9eGW

[18] "QRickit QR Code Decoder," QRickit, Available: `https://qrickit.com`.

Appendix A

A.1 The divided environment method

The idea of dividing an environment into smaller pieces is widely used in some industries in different ways and purposes. One of these industries is the video game industry. A huge amount of video games divide the game world into a chunk of squares, triangles, hexagons, and other shapes depending on the game's need and performance. But the difference here is that they do not use this technique for localization since the positions for all the objects in the games are known and stored at the RAMs already. They use the technique to categorize the pieces such as walkable ground, water, lava, rocks, and so on. Then these pieces are used along with their categories to find a proper path between two points. Different path finding algorithms can be used, but the most popular and simple one is A* algorithm. See this [17] incredible youtube tutorial made by Sebastian Lague, that explain an implementation of the A* algorithm and how it can be optimized.