<TITLE OF THE PROJECT>

Taibah University
College of Computer Science Engineering
Computer Engineering Department

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

The world isn't fair for the visually impaired, plenty of normal, everyday routes become hazardous without any way to see them, stairs, crowds, elevators, stores and plenty of stuff they're oblivious about existing unless someone guides one to them; even regular protrusions are tripping hazards and this is prevalent in most buildings where safety standards aren't enforced which accounts for most residential blocks in saudi arabia. There's also a rising concern of people with no family acquaintances or vacant assistants which gets more prevalent year by year thanks to the rise of the internet, which forms another big hurdle to the blind of today.

That's not to say that people didn't try to offer solutions to this; although it's apathetic compliance to authorities, there had been some effort to put braille under hanged labels and on the ground (Figure 0.0 demonstrates an example of braille)

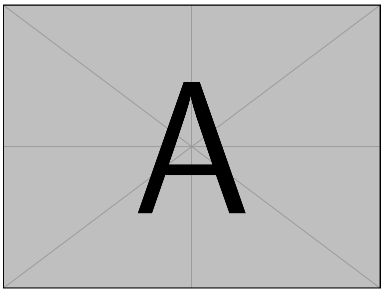

Figure 2.1: Braille

A conundrum arises though, how could a blind person tell where that hanged label or ground protrusion is, much less know how to read it?

You never know when you go blind, there's a good chance you would get it unexpectedly so you might not even know where to look to learn it if the previous paragraph also applies to said individual, in response to this, a much more respectable effort came in the form of tactile pavements, the blind shouldn't have to know anything about how to distinguish each tile, they just have to know it's presence, while they might still be able to miss it in certain cases, the authority of said place or building can just line them up in such a way that it becomes hard to miss since they also have the benefit of being generic tiles that only signal danger or caution, meaning one wouldn't have to make different tiles for each place, they just have to stick with one pattern, reducing cost of manufacturing and maintaining them significantly, below is a figure for an example tactile pavement.

Paragraphs above need citations to confirm their validity

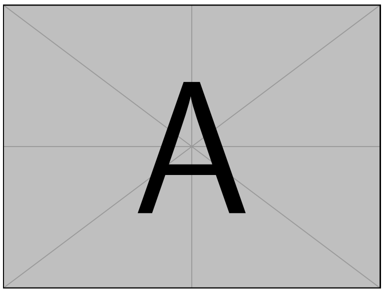

Figure 2.2: Example of a tactile pavement

A problem is still standing however, while they might signal potential danger or sudden elevations of ground or the edges of a track, they don't guide you anywhere so the blind is still hopelessly lost in the midst of a crowd and doesn't know which is where.
We've embarked on a project to change that, with the galactic rise of smartphones, odds are almost everyone is equipped with smartphones.
We aim to guide them with an inexpensive, easy to set up and maintain, and with cellular, you don't even have to have access to the internet to make it, which means higher odds that remote mosques and other such buildings to use it without any fear of cost or the presence of a maintainer.

2.2 Landmarks(Natural & Artificial)

Landmarks can be categorized as either natural or artificial. Natural landmarks are formed by nature, such as mountains, rocks, trees, or any other natural formations. While on the other hand, artificial landmarks are human-made structures such as traffic signs, statues, and QR codes. both categorizes can be used in various important fields in computer science. For example, computer vision algorithms leverage landmarks by detecting, identifying, and tracking them to determine the precise location of an object, create a 3D map of the surrounding environment, or to monitor the state of an object.

While both categorizes can be very useful, natural landmarks are much harder to recognize, very diversified and do not have uniform shapes, difficult to customize, and they do not encode data. All of these characteristics can be opposite in the artificial landmarks, which make them better in terms of control, precision, reliability, and adaptability. Also there are a lot of libraries that support encoding, detecting, decoding, and tracking artificial landmarks. There are also a wide variety of artificial markers such as QR, Arcuo, Topotag. More info on the other kinds of markers are present on reference [2].

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 2.3. This structure is made out of the following parts:

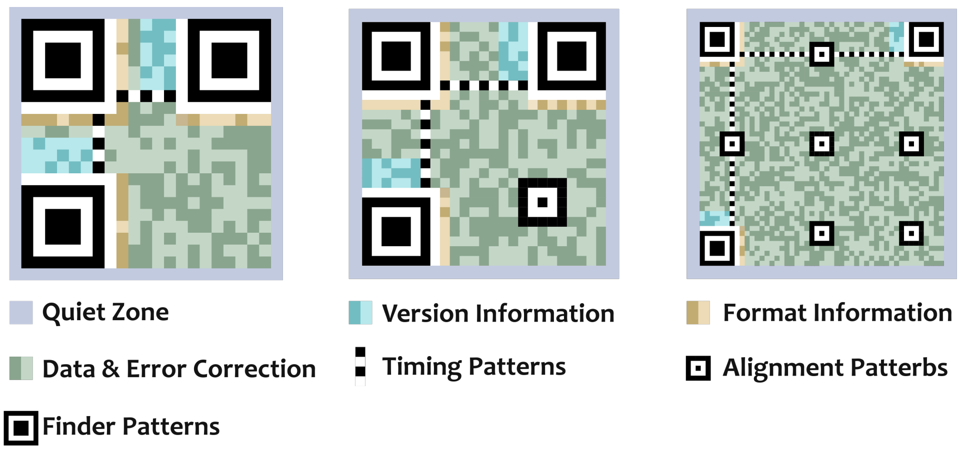

Figure 2.3: These are three QR Codes that store texts. The texts' lengths get larger going from the left to the right. Notice that the alignment pattern only appears at the bigger QR Codes, and there are multiple ones at the biggest QR Code.

- **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 [7], 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 [7].

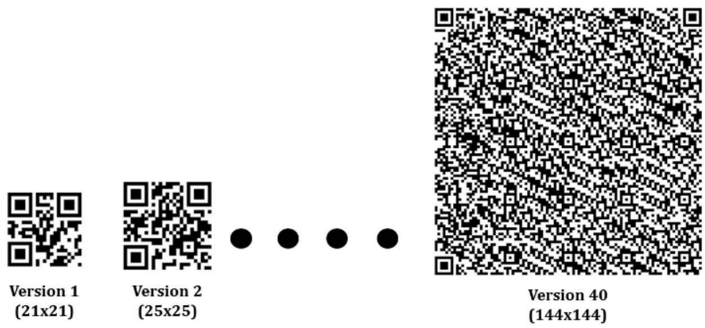

Figure 2.4: QR codes Versions (adapted from [7]).

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.

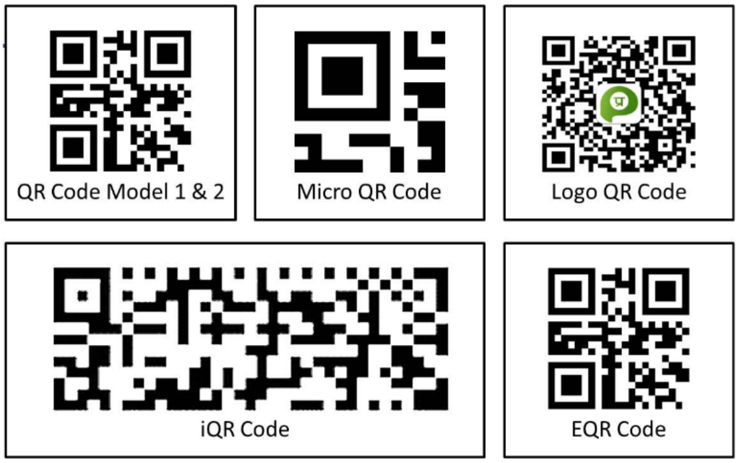

Figure 2.5: QR code types (adapted from [7]).

- **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 [7].
- **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 [7].
- **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 [7].
- **Encrypted QR Code**: Uses encryption techniques to secure the information encoded within, making it suitable for applications requiring data confidentiality [7].

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 [7].

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 [8][9].

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 [10].
- **Java - ZXing (Zebra Crossing)**: An open-source library widely used for encoding QR codes in Java and Android applications [11].
- **iOS - Core Image**: iOS provides native QR code generation capabilities via the 'CIQRCodeGenerator' filter in the Core Image framework [12].

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) [7].

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 [11, 15].

For decoding programmatically, useful libraries include:

- **Python - Segno**: A versatile Python library for generating and reading both QR and Micro QR codes [13].
- **Java - ZXing**: Provides decoding functionality along with encoding, and is widely used for mobile and web apps [11].
- **iOS - AVFoundation**: iOS provides native support for scanning QR codes using the 'AVCaptureMetadataOutput' class in the AVFoundation framework [14].

2.4 Indoor Localization

Indoor localization refers to the process of determining the exact position and orientation of an object or person within indoor environments, where traditional GPS signals are often unreliable. This technology is essential for a wide range of applications, from navigation within large buildings to asset tracking in warehouses.

Various technologies are employed for indoor localization, each with its own strengths and limitations. Wi-Fi, Bluetooth, RFID, and Ultra-Wideband (UWB) are some of the most common methods. While some systems rely on expensive and complex hardware like sensors and Inertial Measurement Units (IMUs), others use more cost-effective solutions such as Bluetooth beacons. The choice of technology often depends on the specific needs of the application, balancing factors such as accuracy, cost, and ease of implementation. For more information, see reference [6].

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

2.4.1 Indoor Localization with QR Codes

Indoor localization using QR codes is a method to determine a user's pose. The system works by strategically placing QR codes around the space—on floors, walls, ceilings, or even hanging panels. Each QR code encodes specific positional information, allowing users to understand their location relative to these codes once they are detected.

Grid Pattern QR codes A simple QR-based localization method involves dividing a $n \times m$ meter room into r squares, each with a QR code indicating its exact position. A device, such as a hat with a camera, detects the QR codes as the user moves, determining their location by the square they're in. While this approach provides discrete positional data, it's computationally efficient and useful in contexts like robotics. Although effective for coarse localization, it lacks the precision required for continuous positioning. This approach is used in the solution proposed by [1].

Pose Estimation with QR codes Another approach of indoor localization involves calculating the relative position and orientation (pose) of the QR code in relation to the camera. After detecting a QR code, the camera determines

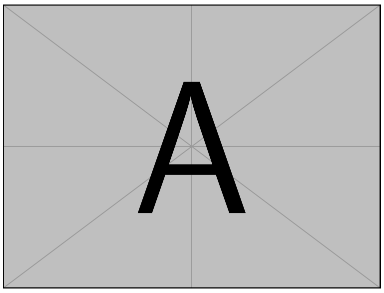

Figure 2.6: Grid Patter Illustration

its position and orientation relative to itself, and by combining this information with the known global position of the QR code, the system can estimate the user's precise, continuous position within the space. Although this method offers significantly higher positional accuracy, it requires more computational resources, as it involves additional steps such as camera calibration to determine intrinsic parameters. This added complexity makes it more resource-intensive compared to grid-based localization, but it delivers continuous localization with greater precision. Cite the papar that use this approach

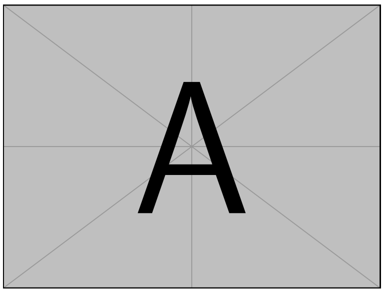

Figure 2.7: Grid Patter Illustration

In a well-designed setup, the system's effectiveness remains high, irrespective of the QR codes' locations. By tailoring the placement and setup of QR codes to suit the environment, the system can deliver robust indoor localization, whether it prioritizes simplicity or precision.

2.5 Related Work

In the area of localization research, artificial landmarks have been used widely due to their difficulty detecting and building a system upon natural landmarks. However, artificial landmarks, specifically QR codes, have gained notable attention due to their ease of making and implementation, low cost, and the amount of information that can be encoded on them.

In the solution proposed by [1], an approach for indoor mobile robot localization using QR codes arranged in a grid pattern on the ceiling is presented.

The robot is equipped with an industrial camera mounted on top, facing upward to detect these QR codes. The camera captures images of the QR codes, and a recognition algorithm processes the position and orientation of the QR codes within the image. By utilizing the coordinates of each QR code and camera calibration data, the robot can accurately determine its position. This setup allows for real-time, precise localization of the robot, which is crucial for effective navigation in structured indoor environments.

The paper [2] introduces a vision-based indoor positioning system that makes use of QR codes and a smartphone camera to accurately determine a user's location indoors. The system positions QR codes at specific locations, which are then identified by the camera. The two-dimensional coordinates of the QR codes are converted into three-dimensional spatial coordinates using camera calibration. By forming a quadrangle from reference symbols on the QR codes, the system calculates the center of gravity to determine the user's position and direction. Experiments have shown an average error of less than 1 meter.

With high accuracy, the paper [3] proposes a cost-effective localization method for indoor mobile robots using QR codes as artificial landmarks. The 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 QR code positions are pre-stored in a database, allowing the robot to calculate its real-world coordinates by processing the image data from the smartphone camera. The accuracy of the system has been confirmed, indicating that the method can produce localization errors ranging from 3.2 cm to 6.55 cm.

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] 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.

[2] 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.

[3] 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 Intelligent Mechatronics (AIM), Busan, Korea (South), 2015, pp. 965-970, doi: 10.1109/AIM.2015.7222664.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

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

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

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

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

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

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

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

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

Appendix A

A.1 Sample Data Tables

Here we provide some example data tables that might be relevant for the study.

Table A.1: Example Data Table

Item	**Quantity**	**Price**
Item 1	10	$100
Item 2	20	$200
Item 3	30	$300

A.2 Sample Figures

Below is an example figure for demonstration purposes.

Appendix B

B.1 Additional Information

This appendix contains additional information relevant to the study, including supplementary materials and further explanations.

B.2 Supplementary Materials

Here you might include additional data or documents that support your research.

B.3 Further Explanations

This section provides further explanations on specific topics covered in the main document. For example, you might elaborate on methodologies or provide additional insights into the analysis.