

ĐẠI HỌC BÁCH KHOA HÀ NỘI

TRƯỜNG ĐIỆN – ĐIỆN TỬ



ĐỒ ÁN
TỐT NGHIỆP KỸ SƯ

ĐỀ TÀI:

NGHIÊN CỨU, THIẾT KẾ NỀN TẢNG TRÒ CHƠI
THỰC TẠI TĂNG CƯỜNG DÀNH CHO TRẺ EM
ỨNG DỤNG TRONG GIÁO DỤC

Sinh viên thực hiện:

Trần Thị Thìn

20182800 Điện tử 01 – K63

Ngành

Kỹ thuật Điện tử Viễn thông

Giảng viên hướng dẫn: TS. Nguyễn Đình Văn

Hà Nội, tháng 8 năm 2023

LỜI NÓI ĐẦU

Ngày nay, khi xã hội hiện đại, hội nhập quốc tế đang phát triển, giao lưu về khoa học công nghệ được nâng cao, mạng internet và các thiết bị điện tử thông minh trở nên phổ biến với mỗi hộ gia đình và là phương tiện không thể thiếu trong việc cập nhật thông tin, học tập hay giải trí. Sự phát triển mạnh mẽ của các trò chơi điện tử là minh chứng cho sự phát triển và tiến bộ vượt bậc của mạng internet trong thời đại này. Theo báo cáo của Newzoo, công ty hàng đầu về trò chơi điện tử và dữ liệu người chơi, ngành công nghiệp game toàn cầu đạt doanh thu 197 tỷ USD trong năm 2022, tăng 2,1% so với 2021. Cùng với đó là sự phát triển không ngừng của các thị trường mới nổi ở khu vực châu Á - Thái Bình Dương, trong đó có Việt Nam.

Việc sử dụng thiết bị điện tử và các trò chơi trên nền tảng công nghệ đã trở thành một phần không thể thiếu trong cuộc sống hàng ngày của con người đặc biệt là trẻ em. Ngoài những trò chơi chỉ mang tính giải trí, đã nhiều trò chơi điện tử mang tính giáo dục được ra đời, nhằm giúp trẻ em vừa được chơi vừa học tập. Tuy nhiên việc tiếp xúc quá mức với các trò chơi trên các thiết bị thông minh có thể gây nghiện và ảnh hưởng đến sức khỏe và khả năng tương tác xã hội của trẻ. Hơn nữa, việc ngồi lâu trước màn hình điện tử không chỉ ảnh hưởng đến thể chất của trẻ, mà còn có thể gây ra các vấn đề sức khỏe như cận thị, chứng mỏi mắt và thiếu vận động. Vì vậy, việc tìm kiếm các giải pháp giáo dục kết hợp hoạt động vật lý, tương tác bạn bè và học tập là cực kỳ quan trọng để giúp trẻ phát triển toàn diện mà vẫn duy trì sức khỏe tốt.

Đề tài "Nghiên cứu, thiết kế nền tảng trò chơi thực tại tăng cường dành cho trẻ em, ứng dụng trong giáo dục" được đưa ra nhằm tạo ra một giải pháp giáo dục mới mẻ, kết hợp hoạt động vật lý, tương tác bạn bè và học tập trong một môi trường an toàn và hấp dẫn. Mô hình trò chơi này sẽ sử dụng mã ArUco để nhận diện vị trí và đối tượng của người chơi ở môi trường thực, tạo ra một trải nghiệm tương tác độc đáo và thú vị.

Qua đây, em xin gửi lời cảm ơn chân thành đến TS. Nguyễn Đình Văn – khoa Kỹ thuật truyền thông, trường Điện – Điện tử, Đại học Bách Khoa Hà Nội, người đã tận tình hướng dẫn, chỉ bảo em trong suốt quá trình hoàn thành đề tài. Em cũng xin cảm ơn đến các thành viên của lab **PSI** đã giúp đỡ em trong suốt quá trình nghiên cứu và hoàn thành đồ án.

Trong quá trình thực hiện đồ án, tuy đã rất cố gắng song do những hạn chế về thời gian cũng như kiến thức có hạn nên không thể tránh khỏi những thiếu sót. Em rất mong nhận được ý kiến đóng góp của thầy cô để hoàn thiện Đồ án một cách tốt nhất.

Em xin chân thành cảm ơn!

Hà Nội, ngày 28 tháng 07 năm 2023

Sinh viên thực hiện

A handwritten signature in blue ink, appearing to read 'Thìn', with a long horizontal stroke extending to the right.

Trần Thị Thìn

LỜI CAM ĐOAN

Tôi là **Trần Thị Thìn**, MSSV **20182800**, sinh viên lớp **Điện tử 01 – K63**. Tôi thực hiện đồ án dưới sự hướng dẫn của thầy TS. Nguyễn Đình Văn. Tôi xin cam đoan toàn bộ nội dung được trình bày trong đồ án "*Nghiên cứu, thiết kế nền tảng trò chơi thực tại tăng cường dành cho trẻ em, ứng dụng trong giáo dục*" là kết quả quá trình tìm hiểu và nghiên cứu của tôi. Các dữ liệu được nêu trong đồ án là hoàn toàn trung thực, phản ánh đúng kết quả thực tế. Mọi thông tin trích dẫn đều tuân thủ các quy định về sở hữu trí tuệ, các tài liệu tham khảo được liệt kê rõ ràng. Tôi xin chịu hoàn toàn trách nhiệm với những nội dung được viết trong đồ án này.

Người cam đoan

A handwritten signature in blue ink, appearing to read 'Thìn', with a long horizontal stroke extending to the right.

Trần Thị Thìn

MỤC LỤC

DANH MỤC TỪ VIẾT TẮT	vii
DANH MỤC HÌNH ẢNH	viii
DANH MỤC BẢNG BIỂU	viii
TÓM TẮT ĐỒ ÁN.....	xi
ABSTRACT.....	xii
CHƯƠNG 1. TỔNG QUAN ĐỒ ÁN.....	1
1.1 Đặt vấn đề.....	1
1.2 Ý nghĩa khoa học và ý nghĩa thực tiễn.....	2
1.3 Mục tiêu và phạm vi của đề tài.....	3
1.4 Phương pháp nghiên cứu	3
1.5 Nội dung thực hiện và kế hoạch dự kiến	4
1.6 Hạn chế đề tài.....	5
CHƯƠNG 2. CƠ SỞ LÝ THUYẾT	7
2.1 Thực tại ảo và thực tại tăng cường là gì?.....	7
2.2 ArUco Marker.....	9
2.2.1 Giới thiệu về mã đánh dấu Aruco	9
2.2.2 Ứng dụng thẻ ArUco trong thực tại tăng cường	11
2.2.3 Ưu điểm ArUco Marker.....	14
2.2.4 Nguyên lý hoạt động ArUco Marker	15
2.3 Giao thức truyền phát MQTT.....	23
2.3.1 Internet of Things (IoT)	24
2.3.2 Kiến trúc MQTT	26
2.3.3 Nguyên lý làm việc.....	27
2.4 Các phần mềm hỗ trợ.....	30
2.4.1 PyCharm Community Edition	30
2.4.2 Mosquitto Broker.....	31
2.4.3 Gdevelop5	31

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG	33
3.1 Phân tích hệ thống.....	33
3.1.1 Mục đích hệ thống.....	33
3.1.2 Đối tượng của hệ thống	33
3.1.3 Chức năng của hệ thống	33
3.2 Kiến trúc tổng quan	33
3.2.1 Mô hình trò chơi.....	33
3.2.2 Sơ đồ nguyên lý.....	35
3.3 Nguyên lý hoạt động hệ thống	36
3.3.1 Khởi điều khiển	36
3.3.2 Khởi Giao tiếp	39
3.3.3 Khởi Hiển thị	40
CHƯƠNG 4. KẾT QUẢ MÔ PHỎNG VÀ KẾT LUẬN	43
4.1 Kịch bản trò chơi	43
4.1.1 Yêu cầu về thiết bị.....	43
4.1.2 Kịch bản trò chơi chi tiết.....	43
4.2 Kết quả mô phỏng	44
4.2.1 Kết quả nhận diện thẻ tag ArUco	44
4.2.2 Chạy hoàn thiện kịch bản trò chơi.....	45
4.3 Kết luận	53
4.3.1 Kết quả đạt được	53
4.3.2 Đánh giá kết quả.....	54
4.4 Hướng phát triển	55
TÀI LIỆU THAM KHẢO	57

DANH MỤC TỪ VIẾT TẮT

STT	Chữ viết tắt	Thuật ngữ đầy đủ	Ý nghĩa
1	AR	Augmented Reality	Thực tại tăng cường
2	VR	Virtual Reality	Thực tại ảo
3	ArUco	Augmented Reality University of Cordoba	Thẻ tag ArUco, được nghiên cứu từ một đại học ở Cordoba
4	MQTT	Message Queuing Telemetry Transport	Giao thức truyền tải tin nhắn hàng đợi
5	HTTP	HyperText Transfer Protocol	Giao thức truyền tải siêu văn bản
5	IoT	Internet of Things	Internet vạn vật
6	QoS	Quality of service	Chất lượng dịch vụ
7	OpenCV	Open Source Computer Vision	Thư viện mã nguồn mở chuyên về xử lý ảnh và thị giác máy tính.
8	TCP/IP	Transmission Control Protocol (TCP) và Internet Protocol (IP)	Giao thức truyền tải điều khiển truyền nhận/liên kết mạng
9	SSL/TLS	Secure Sockets Layer (SSL) và Transport Layer Security (TLS)	Kỹ thuật mã hóa truyền tin trên internet.

DANH MỤC HÌNH ẢNH

Hình 2.1 Triển lãm thực tại ảo đầu tiên của Việt Nam (2020)	7
Hình 2.2 Công nghệ AR ứng dụng trong đời sống.....	8
Hình 2.3 Pokemon Go (2016) – AR tạo tiếng vang lớn.....	8
Hình 2.4. ArUco Marker	10
Hình 2.5. Robot dọn dẹp sử dụng ArUco theo dõi vị trí	12
Hình 2.6 Vẽ bản đồ sử dụng ArUco	12
Hình 2.7 Dự án AR Piano – sản phẩm thử nghiệm tại Aeon Mall Tân Phú, TP.HCM .	13
Hình 2.8 ArUco ứng dụng thực tại tăng cường trong Unity	13
Hình 2.9 Cấu trúc thẻ ArUco 5X5.....	15
Hình 2.10 Lưu đồ nhận diện thẻ tag.....	17
Hình 2.11 Quá trình phát hiện ID thẻ tag ArUco.	17
Hình 2.12 Ảnh đầu vào	19
Hình 2.13 Ảnh đã phân đoạn.....	20
Hình 2.14 Các mã được chấp nhận (a) và bị loại bỏ (b)	21
Hình 2.15 Mã bị lỗi do điều kiện lọc	21
Hình 2.16 Thay đổi phối cảnh.....	22
Hình 2.17 Lưới ô đánh dấu	22
Hình 2.18 Lê ô đánh dấu	23
Hình 2.19. Internet of Things	24
Hình 2.20 Độ phổ biến của các giao thức truyền dữ liệu IoT năm 2018	25
Hình 2.21 Quá trình Subscribe và publish với broker	26
Hình 2.22 Ví dụ cụ thể Facebook sử dụng MQTT.....	28
Hình 2.23. Nền tảng Gdevelop5.....	32
Hình 3.1 Mô hình trò chơi.....	35
Hình 3.2 Sơ đồ khối nguyên lý hoạt động hệ thống.....	35
Hình 3.3 Mosquitto publish.....	37
Hình 3.4 Lưu đồ thuật toán khối Điều khiển	38

Hình 3.5 MQTT broker	39
Hình 3.6 Gdevelop5 subscribe để lắng nghe từ topic MQTT	40
Hình 3.7 Thiết kế trò chơi trên Gdevelop5	41
Hình 4.1 Mô hình thu dữ liệu kiểm thử	44
Hình 4.2 Bộ thẻ được sử dụng trong trò chơi	46
Hình 4.3 Giao diện chọn trò chơi khi chưa kết nối với Broker	46
Hình 4.4 Giao diện trò chơi khi đã kết nối với broker	47
Hình 4.5 Gdevelop connected MQTT	47
Hình 4.6 Trò chơi “Tìm kiếm kim cương”	48
Hình 4.7 Hình ảnh người chơi xuất hiện trên màn hình (3 ngôi sao)	48
Hình 4.8 Hình ảnh từ khung hình camera	49
Hình 4.9 Vị trí người chơi được gửi liên tục	49
Hình 4.10 Người chơi di chuyển và chạm với bom	50
Hình 4.11 Bảng điểm người chơi	50
Hình 4.12 Giao diện trò chơi “Trò chơi toán học”	51
Hình 4.13 Hiện thị câu hỏi và các đáp án	51
Hình 4.14 Đáp án biến mất sau 10 giây	52
Hình 4.15 Tích vào ô chứa đáp án đúng	52
Hình 4.16 Bảng điểm trò chơi toán	53

DANH MỤC BẢNG BIỂU

Bảng 1.1 Kế hoạch dự kiến	4
Bảng 2.1 So sánh một số đặc trưng 2 giao thức HTTP và MQTT	25
Bảng 4.1 Kết quả đánh giá mô hình nhận diện ID thẻ tag	45
Bảng 4.2 Đánh giá mức độ hoàn thành dự kiến	54

TÓM TẮT ĐỒ ÁN

Đồ án tốt nghiệp "Nghiên cứu, thiết kế nền tảng trò chơi thực tại tăng cường dành cho trẻ em, ứng dụng trong giáo dục" tập trung tìm hiểu về lý thuyết, đưa ra giải pháp và xây dựng một mô hình trò chơi thực tại tăng cường sử dụng mã ArUco dành cho một nhóm trẻ em hay một lớp học. Cụ thể Đồ án gồm các nội dung sau:

Chương 1. Tổng quan đồ án

Trình bày về lý do chọn đề tài, các vấn đề liên quan, ý nghĩa khoa học và ý nghĩa thực tiễn của đề tài, mục đích nghiên cứu.

Chương 2. Cơ sở lý thuyết

Chương này nghiên cứu các lý thuyết cơ bản của giải pháp như là: Thực tại tăng cường, công nghệ mã code ArUco, Giao thức truyền tải MQTT, các ứng dụng liên quan và các công nghệ được sử dụng

Chương 3. Phân tích và thiết kế hệ thống

Phân tích đưa ra thiết kế hệ thống, sơ đồ nguyên lý, mô hình trò chơi và các luồng xử lý dữ liệu

Chương 4. Kết quả mô phỏng và kết luận

Sau khi lên kế hoạch, phân tích và thiết kế hệ thống, ở chương này sẽ trình bày kết quả mô phỏng hệ thống game như là kịch bản, giao diện và các game state. Đưa ra kết luận và hướng phát triển.

ABSTRACT

The graduation thesis "Research and design the platform of augmented reality games for children, application in education" focuses on exploring theoretical foundations, proposing solutions, and building an augmented reality game model using ArUco markers for a group of children or a classroom setting. The thesis comprises the following sections:

Chapter 1: Overview of topic

This section presents the reasons for choosing the topic, related issues, scientific and practical significance of the topic, and research purposes.

Chapter 2: Theoretical Background

This chapter investigates fundamental theories relevant to the proposed solutions, including Augmented Reality, ArUco marker technology, MQTT data transmission protocol, related applications, and utilized technologies.

Chapter 3: System Analysis and Design

The analysis in this chapter leads to the system design, block diagrams, game model, and data processing flow.

Chapter 4: Simulation Results and Conclusion

After planning, analyzing, and designing the system, this chapter presents the simulation results of the augmented reality game system, such as scenarios, interfaces, and game states. It also provides conclusions drawn from the results and outlines potential future directions for development.

CHƯƠNG 1. TỔNG QUAN ĐỒ ÁN

1.1 Đặt vấn đề

Trong thời đại công nghệ hiện đại, việc sử dụng thiết bị điện tử và các trò chơi trên nền tảng công nghệ đã trở thành một phần không thể thiếu trong cuộc sống hàng ngày của trẻ em. Sự phát triển của công nghệ đã đem đến cho trẻ nhỏ rất nhiều trò chơi hấp dẫn để giải trí cũng như giáo dục trẻ. Các trò chơi như "Minecraft" và "Angry Birds" giúp trẻ phát triển tư duy sáng tạo và khả năng lập kế hoạch, trong khi "ABCmouse" và "Duolingo" hỗ trợ trẻ em học hỏi và rèn luyện kỹ năng ngôn ngữ. Các trò chơi này được sử dụng hoàn toàn trên các thiết bị thông minh như điện thoại hay máy tính. Sử dụng trò chơi điện tử một cách thông minh giúp các bạn trẻ rèn luyện được các kỹ năng giải quyết tình huống cũng như lên kế hoạch, quản lý thời gian, nâng cao tư duy logic, sáng tạo hay phát triển tí tưởng tượng cho trẻ...Tuy nhiên, nếu không có sự tỉnh táo trong việc lựa chọn trò chơi, không biết cân bằng thời gian chơi, không làm chủ được bản thân trước những cám dỗ, quá chìm đắm trong thế giới ảo trên màn hình sẽ gây ra những hậu quả không lường về vật chất và tinh thần, tổn hại về sức khỏe, trí tuệ và nhận thức cho trẻ.

Vì lý do trên, cần có những giải pháp sáng tạo để kết hợp giữa hoạt động vật lý, tương tác xã hội và học tập trong trò chơi giáo dục cho trẻ em. Một trong những phương pháp tiềm năng là xây dựng mô hình trò chơi thực tại tăng cường (Augmented Reality - AR). Công nghệ Thực tế tăng cường AR là sự tương tác đồng thời giữa thế thực và thế giới ảo thông qua các thiết bị điện tử và nội dung 2D/3D đi kèm. Theo các chuyên gia, thực tế tăng cường hỗ trợ người học dễ dàng tiếp thu, xử lý và ghi nhớ thông tin, giúp cho việc học trở nên hấp dẫn và thú vị. Tuy nhiên trong việc sử dụng thực tại tăng cường cho trẻ em việc nhìn vào màn hình điện thoại hay các thiết bị AR trong thời gian dài có thể gây căng thẳng và mỏi mắt cho trẻ em, đặc biệt là đối với nhóm tuổi phát triển. Một vấn đề khác là để triển khai xây dựng một hệ thống trò chơi AR đòi hỏi phải đối mặt với nhiều khó khăn về kỹ thuật và thiết kế cùng với đó việc đầu tư về công nghệ và phần cứng là không thể tránh khỏi. Do đó, chi phí cao có thể trở thành rào cản đối với việc tiếp cận AR cho các trường học và gia đình có nguồn lực hạn chế. Vì vậy, một giải pháp về trò chơi thực tại tăng cường giúp trẻ phát toàn diện, tránh những môi nguy

hại về sức khỏe của trẻ như được nêu ở trên cùng với đó là giảm thiểu chi phí để trò chơi trở nên phổ biến là hết sức cần thiết.

Trong báo cáo đề tài "Nghiên cứu, thiết kế nền tảng trò chơi thực tại tăng cường dành cho trẻ em, ứng dụng trong giáo dục", em đặt mục tiêu xây dựng một nền tảng trò chơi giáo dục thực tại tăng cường thú vị, có chi phí thấp và giảm thiểu tác hại mà các trò chơi điện tử khác mang lại, để trẻ em một trải nghiệm giáo dục mới, kết hợp giữa hoạt động vật lý, tương tác bạn bè và học tập, từ đó tạo điều kiện cho sự phát triển toàn diện của trẻ.

Đề tài này đề xuất giải pháp sử dụng mã đánh dấu ArUco, để nhận diện đối tượng người chơi và vị trí người chơi trong không gian thực, liên kết với giao diện trò chơi được thiết kế trên ứng dụng game engine Gdevelop 5 qua giao thức mạng MQTT (Message Queuing Telemetry Transport). Em hi vọng rằng nghiên cứu này sẽ đóng góp vào lĩnh vực giáo dục sáng tạo cho trẻ em, tạo ra những trải nghiệm mới mà trẻ em vừa học tập, vừa tương tác bạn bè và vừa được hoạt động vật lý, từ đó giúp trẻ em phát triển toàn diện và duy trì sức khỏe tốt.

1.2 Ý nghĩa khoa học và ý nghĩa thực tiễn

Ý nghĩa khoa học:

- Nghiên cứu thiết kế trò chơi thực tại tăng cường trong giáo dục dành cho trẻ nhỏ
- Xây dựng và thử nghiệm nền tảng trò chơi sử dụng thực tại ảo ứng dụng các công nghệ: mã tag ArUco, giao thức truyền tin MQTT và lập trình trò chơi với nền tảng Gdevelop 5

Ý nghĩa thực tiễn

- Phát triển trò chơi sáng tạo mang tính giáo dục: Mục đích của đề tài là xây dựng một mô hình trò chơi mới, kết hợp hoạt động vật lý, tương tác xã hội và học tập. Mô hình trò chơi này có thể giúp trẻ em vừa hoạt động, vừa học tập và tương tác bạn bè một cách tích cực. Điều này đáng chú trọng trong bối cảnh hiện nay khi trẻ em dễ dàng tiếp cận các trò chơi điện tử gây nghiện.
- Ứng dụng thực tế trong giáo dục và giải trí: Mô hình trò chơi này không chỉ tạo ra một môi trường giáo dục tích cực, mà còn khuyến khích sự phát triển toàn diện của trẻ em. Mô hình có thể được áp dụng trong các cơ sở giáo dục và giải trí cho trẻ em. Các trường học, trung tâm giáo dục, và các tổ chức giáo dục có thể sử

dụng mô hình này để tạo ra môi trường học tập mới mẻ và thú vị. Ngoài ra, nó cũng có thể được ứng dụng trong các sự kiện và hoạt động giải trí dành cho trẻ em, giúp tạo ra những trải nghiệm đáng nhớ và bổ ích.

1.3 Mục tiêu và phạm vi của đề tài

Mục tiêu của đề tài "Nghiên cứu, thiết kế nền tảng trò chơi thực tại tăng cường dành cho trẻ em, ứng dụng trong giáo dục" là tạo ra một nền tảng trò chơi giáo dục độc đáo và hấp dẫn, giúp trẻ em vừa hoạt động vật lý, vừa tương tác với bạn bè và vừa học tập, nhằm tránh xa các trò chơi trên các thiết bị điện tử gây nghiện và có ảnh hưởng đến sức khỏe.

Trò chơi được nghiên cứu áp dụng cho nhóm học sinh hoặc một lớp học độ tuổi từ 6 – 10 tuổi.

Cụ thể, các mục tiêu và phạm vi chính của đề tài bao gồm:

- Nghiên cứu và áp dụng mã ArUco trong việc nhận diện và theo dõi vị trí của đối tượng trong không gian. Điều này bao gồm việc tìm hiểu về cấu trúc và cách hoạt động của mã ArUco, cùng với việc xây dựng và in các thẻ ArUco để sử dụng trong mô hình trò chơi.
- Tìm hiểu và ứng dụng giao thức MQTT (Message Queuing Telemetry Transport) nhằm truyền tải và kết nối Module nhận dạng tag ArUco và giao diện trò chơi.
- Thiết kế và phát triển các trò chơi giáo dục sáng tạo trong mô hình trò chơi. Các trò chơi sẽ được thiết kế nhằm kích thích hoạt động, tương tác xã hội và học tập cho trẻ em. Các hoạt động như tìm kiếm kim cương, tính toán và các trò chơi học tập khác sẽ được tích hợp trong mô hình trò chơi để đem lại trải nghiệm học tập tích cực cho trẻ em.

1.4 Phương pháp nghiên cứu

Để thực hiện đề tài em áp dụng một phương pháp nghiên cứu kết hợp giữa nghiên cứu lý thuyết và thực nghiệm. Phương pháp này nhằm đảm bảo sự hiểu rõ về các khái niệm lý thuyết liên quan và kiểm chứng tính khả thi và hiệu quả của mô hình trò chơi.

Nghiên cứu lý thuyết: Tiến hành nghiên cứu về công nghệ thực tại tăng cường, mã ArUco, và giao thức MQTT. Tập trung tìm hiểu về nguyên lý hoạt động, các công nghệ và công cụ hỗ trợ, cũng như những ứng dụng và lợi ích của chúng trong lĩnh vực giáo

dục và giải trí cho trẻ em. Nghiên cứu lý thuyết này giúp em xác định cách thức áp dụng và tối ưu hóa các công nghệ này trong mô hình trò chơi.

Xây dựng mô hình trò chơi: Đề tài sử dụng Python và GDevelop 5 để xây dựng mô hình trò chơi. Mô hình trò chơi sẽ sử dụng camera để nhận diện thẻ ArUco và xác định vị trí và đối tượng của từng người chơi và hiển thị trên Gdevelop. Trong tương lai, các trò chơi giáo dục có thể được thiết kế và xây dựng thêm nhằm mở rộng đối tượng và mục tiêu phục vụ của sản phẩm trong đề án.

Thực nghiệm và đánh giá: Đề tài được thực hiện và thực nghiệm để đánh giá tính khả thi và hiệu quả của mô hình trò chơi thông qua việc tham gia và thử nghiệm trò chơi với đối tượng trẻ em. Sau đó tiến hành phân tích và đánh giá dữ liệu thu thập được để xác định mức độ đáp ứng và cải thiện mô hình trò chơi.

1.5 Nội dung thực hiện và kế hoạch dự kiến

Bảng 1.1 Kế hoạch dự kiến

Thứ tự	Phần việc	Nội dung công việc	Thời gian dự kiến
1	Tìm hiểu lý thuyết	Tìm hiểu ArUco Marker và thư viện Paho Python	03/04 – 17/04
		Giao thức mạng MQTT (Pub/Sub, mosquitto,..)	
2	Kết nối MQTT broker	Connect Broker MQTT sử dụng Paho Python	18/04 – 26/04
		Pub/Sub Topic lên server, xử lý message gửi và nhận trên MQTT	
3	Xử lý ảnh	Nhận diện ArUco Marker bằng thư viện OpenCV	24/04 – 06/05
4	Thiết kế kịch bản và chạy thử kịch bản trên python	Thiết kế kịch bản trò chơi (Nội dung trò chơi, các bước chạy chương trình)	10/05 – 10/06
		Viết chương trình theo luồng, gameState bằng ngôn ngữ Python	

5	Thiết kế giao diện trò chơi	Thiết kế giao diện trên Gdevelop5 xử lý các sự kiện game	10/6 – 30/6
6		Xây dựng hệ thống game demo, giao tiếp giữa các khối, ứng dụng cho 1 tập thẻ ArUco	15/6 – 10/7
7	Hoàn thiện	Thiết kế app	Hoàn thành trước 30/7
		Thử nghiệm với người chơi thật trên diện rộng.	

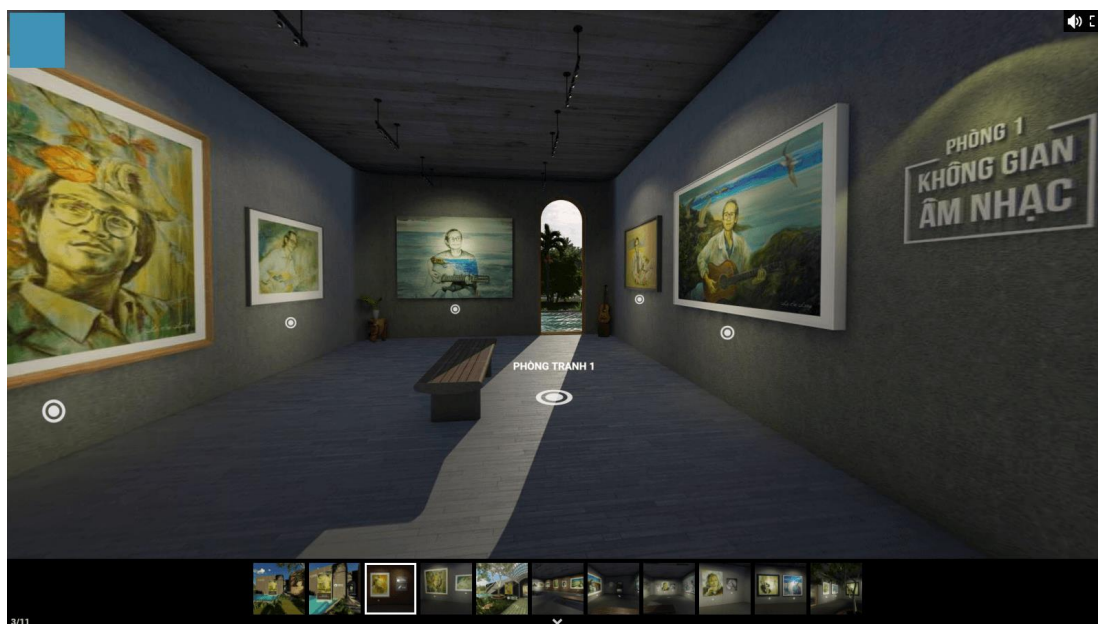
1.6 Hạn chế đề tài

Đồ án đã hoàn thiện mô hình trò chơi thực tế tăng cường và đáp ứng được mục tiêu đặt ra. Tuy nhiên, do giới hạn thời gian và phạm vi của đồ án tốt nghiệp, việc thực hiện khảo sát và đánh giá thực nghiệm với người dùng là không thực hiện được. Do đó, chưa có việc tiến hành khảo sát thực nghiệm và thu thập trải nghiệm của người chơi để đưa ra đánh giá về độ đáp ứng của trò chơi.

CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

2.1 Thực tại ảo và thực tại tăng cường là gì?

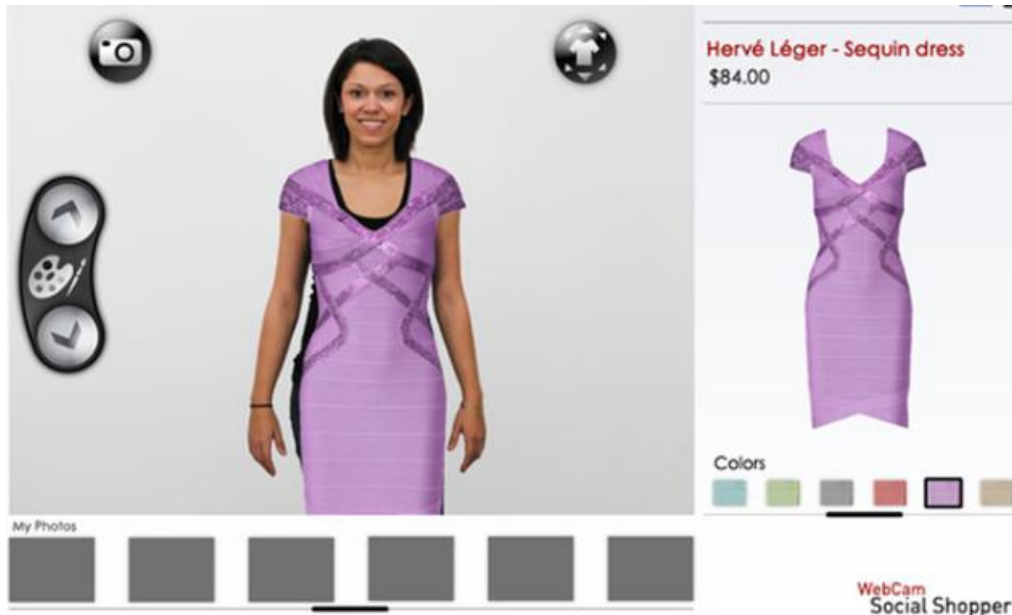
Thực tại tăng cường (Argument Reality) là một xu hướng công nghệ mới được phát triển trên nền tảng công nghệ thực tại ảo (Virtual Reality). Thực tại ảo đề cập đến việc mô phỏng bằng máy tính tạo ra các hình ảnh của một thế giới mà dường như tất cả các giác quan của ta sẽ có và cảm nhận được tương tự như cách chúng ta nhận thức thế giới thực hoặc thực thể vật lý như ngoài đời. VR đã được ứng dụng trong mọi lĩnh vực trên thế giới: Y học, khoa học kỹ thuật, quân sự, kiến trúc, du lịch, địa ốc,... à đáp ứng mọi nhu cầu: Nghiên cứu – Giáo dục – Thương mại – dịch vụ. VR có tiềm năng ứng dụng vô cùng lớn, có thể nói mọi lĩnh vực có thật trong cuộc sống đều có thể ứng dụng thực tại ảo để mô phỏng, nghiên cứu và phát triển hoàn thiện hơn. Đa phần các môi trường tạo ra bởi công nghệ thực tại ảo chủ yếu là hình ảnh hiển thị trên máy tính thông qua góc nhìn ba chiều (3D), tuy nhiên một vài mô phỏng cũng liên quan đến các loại giác quan khác như thính giác hay khứu giác.



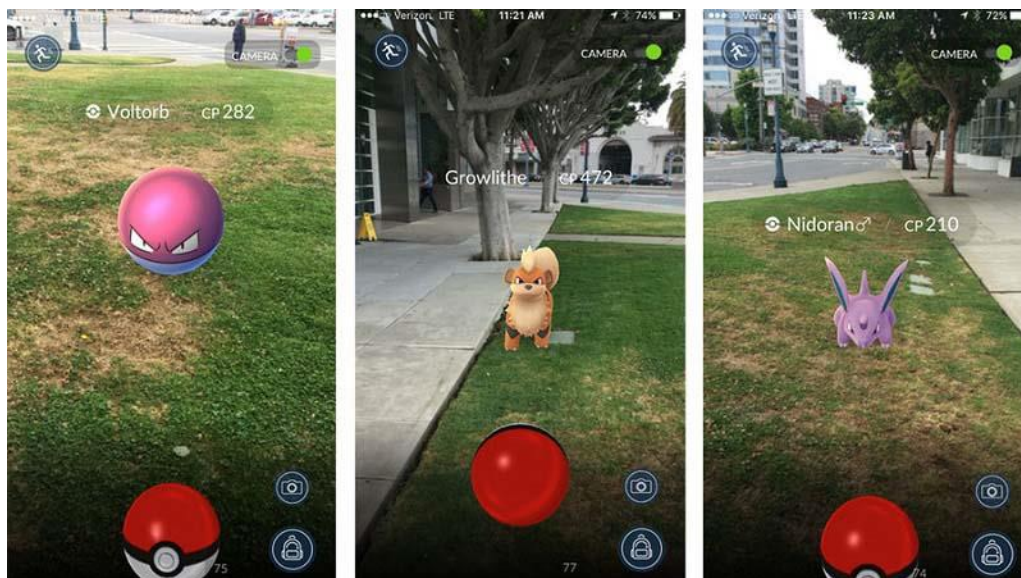
Hình 2.1 Triển lãm thực tại ảo đầu tiên của Việt Nam (2020)

Khác với VR, AR được hiểu là công nghệ cho phép con người quan sát những vật thể trong thế giới thực thông qua một thiết bị điện tử thông minh nào đó. AR là sự tích hợp thông tin kỹ thuật số với môi trường của người dùng trong thời gian thực. Không giống như thực tại ảo, công nghệ tạo ra một môi trường hoàn toàn nhân tạo, người trải nghiệm thực tại tăng cường trải nghiệm môi trường kỹ thuật số trong thế giới thực bằng

cách phủ lên quang cảnh thực tại một số vật thể ảo do máy tính tạo ra hoặc tương tác với thế giới thực thông qua các môi trường kỹ thuật số.



Hình 2.2 Công nghệ AR ứng dụng trong đời sống



Hình 2.3 Pokemon Go (2016) – AR tạo tiếng vang lớn

Thực tại ảo tăng cường hỗ trợ thêm thành phần kỹ thuật số tương tác. Vì vậy, người trải nghiệm AR có thể kích hoạt camera trên các thiết bị thông minh, xem thế giới thực xung quanh trên màn hình và tương tác với vật thể thông qua lớp phủ kỹ thuật số hoặc tương tác thế giới thực. Thực tại ảo tăng cường sử dụng các kỹ thuật xử lý ảnh để phát hiện và định vị vị trí các đối tượng trong thế giới thực. Sau đó, các đối tượng ảo được thêm vào bằng cách sử dụng các kỹ thuật đồ họa 3D hoặc 2D để tạo ra một hình ảnh số động và thêm vào trong thế giới thực.

Trong công nghệ thực tại tăng cường, xử lý ảnh đóng vai trò quan trọng. Nó liên quan đến việc xử lý và phân tích thông tin từ hình ảnh để nhận diện và theo dõi đối tượng trong không gian thực. Các thuật toán xử lý ảnh trong AR nhằm xác định vị trí, hướng, và đặc điểm của đối tượng trong thế giới thực, từ đó tạo ra các hiệu ứng và đối tượng ảo tương tác. Xử lý ảnh được sử dụng để nhận diện và theo dõi vị trí của các đối tượng trong không gian thực. Các kỹ thuật như nhận dạng đặc trưng (feature recognition), phân đoạn (segmentation), và phối hợp (registration) được sử dụng để xác định và theo dõi vị trí, hình dạng và các thuộc tính của đối tượng.

Với sự phát triển của công nghệ xử lý ảnh, AR đã tạo ra những tiềm năng ứng dụng rộng rãi trong nhiều lĩnh vực. AR mang đến những trải nghiệm tương tác độc đáo và thú vị, cho phép người dùng tương tác trực tiếp với môi trường xung quanh và tạo ra những trải nghiệm mới mẻ và tương tác. Tương lai của thực tế tăng cường và thực tế ảo hứa hẹn rất sáng rõ và đầy tiềm năng. Với sự phát triển không ngừng của công nghệ và sự gia tăng của việc áp dụng AR và VR trong nhiều lĩnh vực, chúng ta đang chứng kiến một cuộc cách mạng trong trải nghiệm người dùng và cách chúng ta tương tác với thế giới xung quanh.

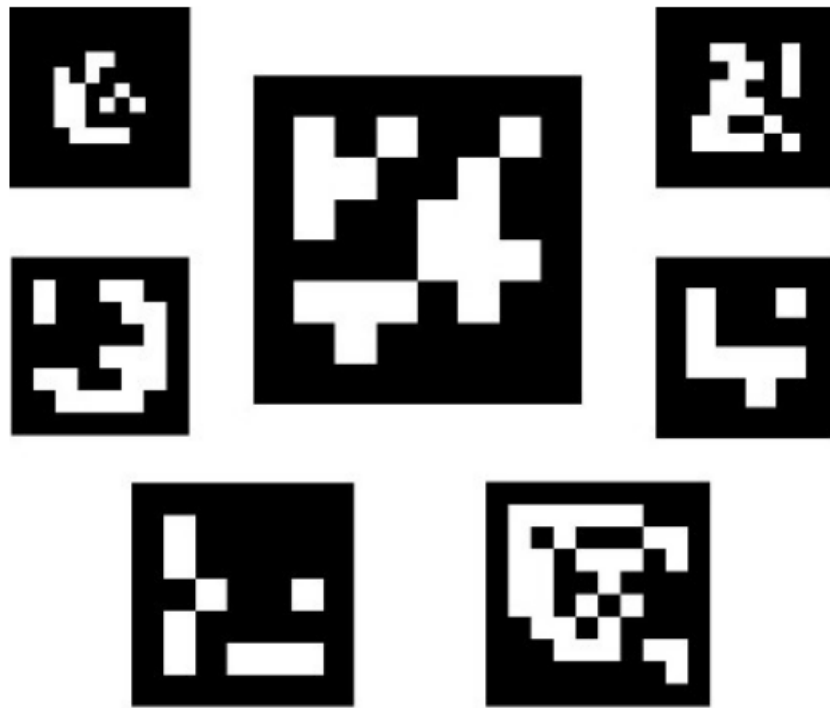
2.2 ArUco Marker

2.2.1 Giới thiệu về mã đánh dấu Aruco

ArUco là một công nghệ xử lý ảnh được thường được sử dụng trong thực tại tăng cường, ArUco là viết tắt của "Augmented Reality University of Cordoba" - một đại học tại Tây Ban Nha. Công nghệ Aruco được phát triển bởi Rafael Muñoz Salinas và Sergio Garrido-Jurado vào năm 2014. Ý tưởng ban đầu của họ là sử dụng các marker đơn giản và dễ nhận diện để theo dõi vị trí của các vật thể trong thực tế ảo tăng cường. Để phát triển công nghệ này, các nhà nghiên cứu đã sử dụng các thuật toán xử lý ảnh để nhận diện và theo dõi các marker trong không gian 3D. Các marker được thiết kế với các đường phân cách đen trắng đơn giản và độc đáo, giúp cho việc nhận diện và theo dõi chúng trở nên dễ dàng hơn. Sau khi hoàn thành việc phát triển công nghệ, các nhà nghiên cứu đã công bố mã nguồn của công nghệ Aruco và cung cấp tài liệu hướng dẫn cho người dùng.

Mã ArUco là một loạt các thẻ đánh dấu bao gồm một mạng lưới ô vuông được chia thành các ô đen và trắng. Cấu trúc của mỗi mã ArUco được xác định bằng một ID duy

nhất. ID này thường được mã hóa dưới dạng một chuỗi nhị phân hoặc thập phân. Các mã ArUco có thể có độ phân giải khác nhau, từ những mã nhỏ gồm chỉ vài ô cho đến những mã lớn gồm hàng trăm ô.



Hình 2.4. ArUco Marker

Quá trình nhận dạng mã ArUco thường được thực hiện bằng cách sử dụng camera và các thuật toán xử lý hình ảnh. Khi camera quan sát mã ArUco, hình ảnh được chụp bởi camera được truyền vào thuật toán xử lý. Thuật toán sẽ phân tích hình ảnh để tìm ra các đặc điểm đặc trưng của mã, chẳng hạn như các góc và đường biên. Dựa trên các đặc điểm này, mã ArUco được nhận dạng và vị trí của nó trong không gian 3D được xác định. Thông qua việc nhận dạng mã ArUco và xác định vị trí của chúng, hệ thống thực tế tăng cường có thể tạo ra các trải nghiệm tăng cường ảo, bổ sung thông tin hoặc tương tác với thế giới thực. Các nhà nghiên cứu đã công bố mã nguồn của công nghệ Aruco dưới dạng phần mềm mã nguồn mở và cung cấp tài liệu hướng dẫn cho người dùng thông qua các trang web và tài liệu trực tuyến. Cụ thể, mã nguồn của công nghệ Aruco đã được công bố trên Github, một nền tảng lưu trữ mã nguồn mở trực tuyến, và có sẵn cho mọi người tải xuống và sử dụng miễn phí. Bên cạnh đó, các tài liệu hướng dẫn cũng đã được công bố trên trang web chính thức của OpenCV, trong đó có cách cài đặt và sử

dụng thư viện Aruco, các ví dụ về sử dụng thư viện và các tài liệu tham khảo khác liên quan đến công nghệ này.

Các ứng dụng của công nghệ Aruco rất đa dạng và được sử dụng trong nhiều lĩnh vực khác nhau, từ giáo dục đến sản xuất công nghiệp và giải trí. Công nghệ này đã được cộng đồng phát triển và sử dụng rộng rãi, đặc biệt là trong các ứng dụng thực tế ảo tăng cường.

2.2.2 Ứng dụng thẻ ArUco trong thực tại tăng cường

Lịch sử phát triển của ArUco đi kèm với việc nghiên cứu và phát triển trong lĩnh vực thực tế tăng cường và thị giác máy tính. Công cụ này đã được sử dụng rộng rãi trong cộng đồng nghiên cứu và các ứng dụng thực tế. Từ khi được giới thiệu, ArUco đã trở thành một công cụ phổ biến cho việc nhận dạng và theo dõi vị trí các đối tượng trong thực tế tăng cường.

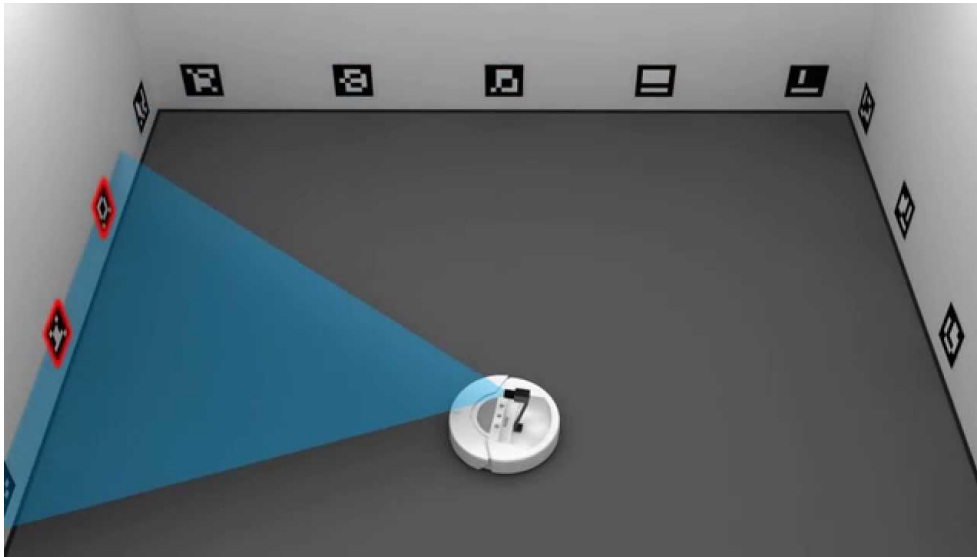
Sự phát triển tiếp tục của mã nguồn ArUco phụ thuộc vào sự đóng góp từ cộng đồng và nhóm phát triển OpenCV. Được hỗ trợ bởi một cộng đồng lớn và nhiều nghiên cứu trong lĩnh vực thực tế tăng cường, ArUco tiếp tục được cải tiến và cập nhật để đáp ứng các yêu cầu và xu hướng mới trong lĩnh vực này.

Một trong những ứng dụng quan trọng của thẻ ArUco là trong đánh dấu và gắn kết thông tin. Khi camera của thiết bị thực tế tăng cường nhìn thấy thẻ ArUco, hệ thống có thể nhận dạng và xác định vị trí của nó. Điều này cho phép hệ thống gắn kết thông tin bổ sung như hình ảnh, video, văn bản hoặc đối tượng 2D hay 3D với đối tượng đánh dấu. Việc gắn kết thông tin ảo vào đối tượng thực tế tạo ra trải nghiệm tăng cường độc đáo và tạo ra tiềm năng ứng dụng rộng lớn trong các lĩnh vực như học tập, kiến trúc, marketing và nhiều lĩnh vực khác. Cụ thể:

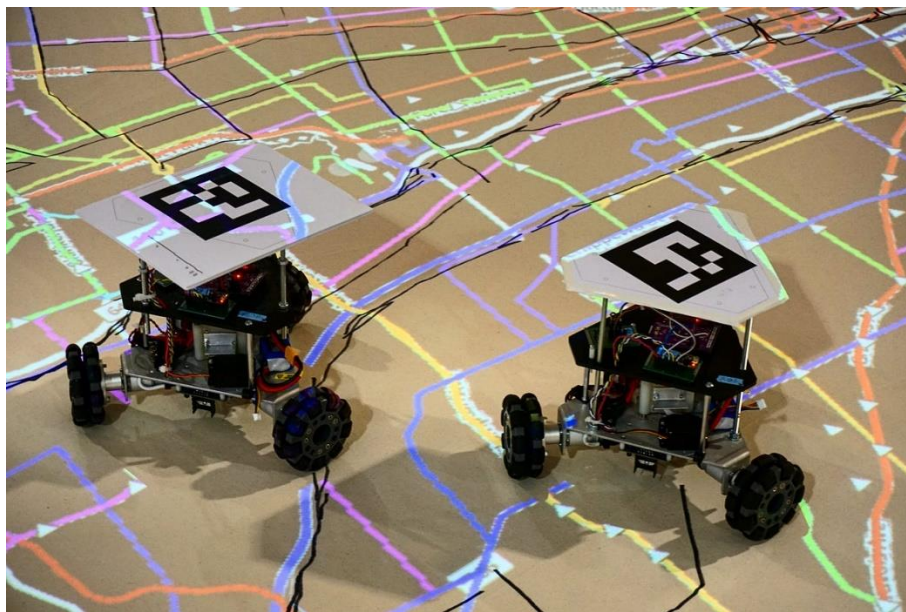
Thẻ ArUco cũng được sử dụng để theo dõi vị trí và hướng của đối tượng trong không gian 3D. Bằng cách nhận dạng và xác định vị trí của thẻ ArUco trong cảnh quan sát, hệ thống thực tế tăng cường có thể xác định được tọa độ và hướng của đối tượng liên quan đến camera. Điều này cho phép tạo ra các trải nghiệm tăng cường chính xác và nhất quán, bằng cách định vị và hiển thị các đối tượng ảo một cách chính xác trong không gian thực tế.

Thẻ ArUco được sử dụng rộng rãi trong nghiên cứu robot để nhận dạng và theo dõi vị trí của robot trong không gian 3D. Thẻ ArUco có thể được gắn kết trên các đối tượng hoặc môi trường để định vị và xác định vị trí của robot. Robot có thể sử dụng

camera để nhận dạng mã ArUco trong hình ảnh và dựa trên đó, xác định vị trí và hướng di chuyển của nó hoặc được sử dụng để phân biệt các đối tượng khác nhau.



Hình 2.5. Robot dọn dẹp sử dụng ArUco theo dõi vị trí



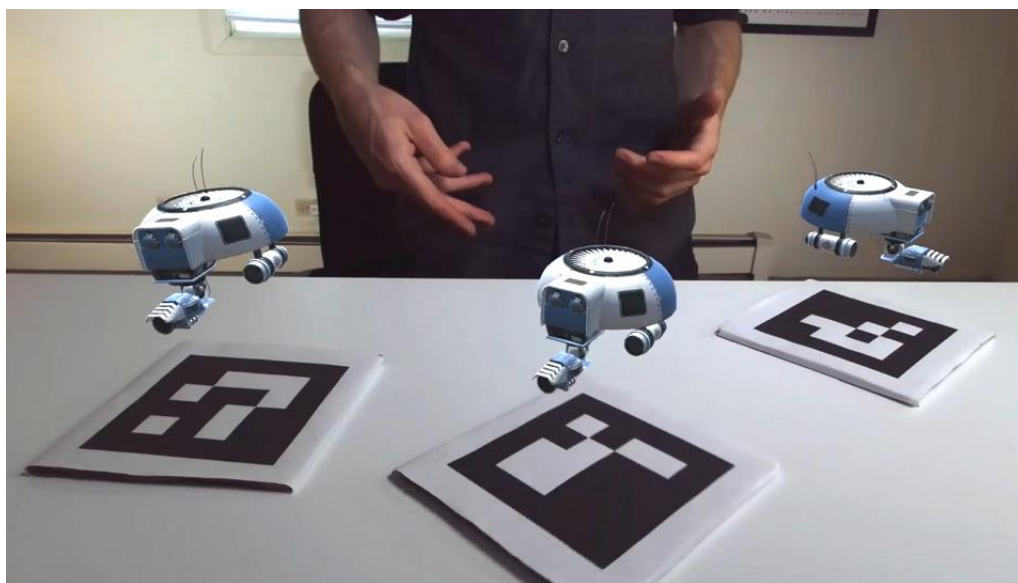
Hình 2.6 Vẽ bản đồ sử dụng ArUco

ArUco marker cũng cho phép tương tác và điều khiển trong các ứng dụng thực tế tăng cường. Bằng cách di chuyển, xoay hoặc tương tác với thẻ ArUco, người dùng có thể tương tác với đối tượng ảo trên màn hình và thực hiện các chức năng hoặc hành động cần thiết. Việc tạo ra các giao diện tương tác thông qua thẻ ArUco mở ra nhiều cơ hội cho các ứng dụng thực tế tăng cường trong lĩnh vực giải trí, giáo dục, y tế và nhiều ngành công nghiệp khác.



Hình 2.7 Dự án AR Piano – sản phẩm thử nghiệm tại Aeon Mall Tân Phú, TP.HCM

ArUco cũng được ứng dụng trong thực tế ảo (virtual reality) và trò chơi. Sử dụng thẻ ArUco trong thực tế ảo hoặc trò chơi cho phép người chơi tương tác với đối tượng ảo trên màn hình thông qua việc di chuyển hoặc xoay thẻ. Điều này tạo ra một trải nghiệm chơi game hoặc tham gia vào thế giới ảo tương tác và thú vị.



Hình 2.8 ArUco ứng dụng thực tại tăng cường trong Unity

Aruco marker cũng được sử dụng trong giáo dục và đào tạo. Các giáo viên và học sinh có thể sử dụng Aruco marker để tạo ra các tài liệu giảng dạy tăng cường thực tế, giúp học sinh hình dung và hiểu bài học một cách trực quan và sinh động hơn.

Quảng cáo và trưng bày sản phẩm: Aruco marker cũng được sử dụng trong quảng cáo và trưng bày sản phẩm. Các nhà sản xuất và nhà quảng cáo có thể sử dụng Aruco marker để tạo ra các quảng cáo tăng cường thực tế để thu hút sự chú ý của khách hàng..

2.2.3 Ưu điểm *ArUco Marker*

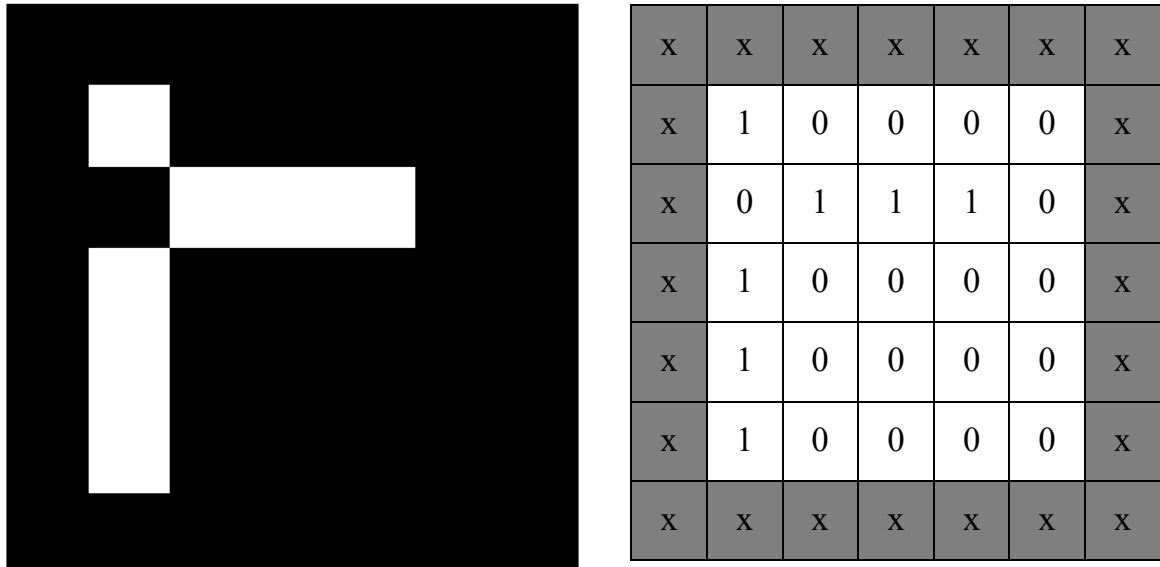
Việc sử dụng ArUco là một loại mã chi phí thấp mang lại nhiều lợi ích cho người lập trình vì mã nguồn mở dễ dàng ứng dụng và cho mô hình vì độ chính xác cao:

- Aruco marker được thiết kế để có thể dễ dàng in và đặt trên đối tượng cần theo dõi, điều này giúp cho việc sử dụng Aruco marker trở nên đơn giản và tiện lợi.
- Độ chính xác cao: Với các thuật toán xử lý ảnh hiện đại, Aruco marker có độ chính xác cao trong việc xác định vị trí và hướng của đối tượng trong không gian 3D.
- Tính ổn định: Aruco marker có tính ổn định cao, không bị ảnh hưởng bởi ánh sáng môi trường hay góc nhìn của camera.
- Đa dạng: Aruco marker cung cấp nhiều loại marker với độ phân giải khác nhau, từ đó có thể được sử dụng cho các mục đích khác nhau.
- Ứng dụng rộng rãi: Aruco marker được sử dụng trong nhiều lĩnh vực, từ theo dõi vị trí và hướng của đối tượng trong không gian 3D đến điều khiển robot và máy bay không người lái, giáo dục và đào tạo, quảng cáo và trưng bày sản phẩm, ...
- Mã nguồn mở: Thư viện Aruco trong OpenCV được phát triển dưới dạng mã nguồn mở, giúp cho người dùng có thể tùy chỉnh và sử dụng theo nhu cầu của mình

2.2.4 Nguyên lý hoạt động ArUco Marker

2.2.4.1 Cấu trúc và từ điển mã ArUco (Marker Dictionary)

Cấu trúc của một mã ArUco bao gồm một ma trận vuông được chia thành các ô nhỏ, trong đó mỗi ô có thể có giá trị đen (0) hoặc trắng (1) viên mã được quy ước là màu đen (không tính vào ma trận giá trị ID). Mã ArUco thường có cấu trúc đặc biệt để đảm bảo tính duy nhất và khả năng phát hiện tin cậy.



Hình 2.9 Cấu trúc thẻ ArUco 5X5

Cụ thể về cấu trúc mã ArUco:

- Kích thước của mã ArUco: Mã ArUco có kích thước được xác định trước, chẳng hạn như 4x4 tức 4 hàng 4 cột, hoặc 5x5. Điều này chỉ ra rằng mã ArUco sẽ được chia thành một ma trận vuông với số lượng ô tương ứng.
- Ô trong mã ArUco: Mỗi ô trong mã ArUco có thể là đen (0) hoặc trắng (1), biểu thị màu sắc của ô đó. Trạng thái màu sắc của mỗi ô được đặt dựa trên các yếu tố như độ tương phản và khả năng phát hiện.
- Cấu trúc mẫu của mã ArUco: Mã ArUco được xây dựng từ một mẫu cụ thể, trong đó các ô đen và trắng được sắp xếp theo một cấu trúc đặc biệt. Mỗi thẻ khác nhau cho phép xác định ID duy nhất của mã và đảm bảo tính duy nhất của nó trong từ điển mã ArUco.
- Bit kiểm tra sửa lỗi: Mã ArUco có thể bao gồm các bit kiểm tra sửa lỗi để tăng tính tin cậy và khả năng phát hiện lỗi trong quá trình nhận dạng mã. Các bit

kiểm tra này giúp xác định xem một mã ArUco có hợp lệ hay không và có thể sửa chữa các lỗi nhỏ trong quá trình nhận dạng.

- Sự kết hợp của các mã: Trong ứng dụng thực tế, ArUco Marker có thể được sử dụng cùng nhau để xác định vị trí và hướng của các đối tượng trong không gian 3D. Sự kết hợp của các mã ArUco cho phép xác định vị trí và hướng của đối tượng dựa trên quan hệ tương đối giữa các mã ArUco trong khung hình.

Marker Dictionary (Từ điển mã ArUco) là tập hợp các mã ArUco duy nhất và được đánh số có cấu trúc đặc biệt. Từ điển này chứa các mã ArUco khác nhau và được sử dụng để nhận dạng và theo dõi vị trí các đối tượng trong không gian 3D. Mỗi từ điển trong mã ArUco được đặc trưng bởi các thông số cụ thể như số lượng mã có sẵn, kích thước của mã, cấu trúc của mã và các thuộc tính khác.

Ví dụ, trong thư viện OpenCV, có sẵn một số từ điển mã ArUco phổ biến, bao gồm:

DICT_4X4_50: Từ điển gồm 50 mã ArUco, mỗi mã là một ma trận 4x4.

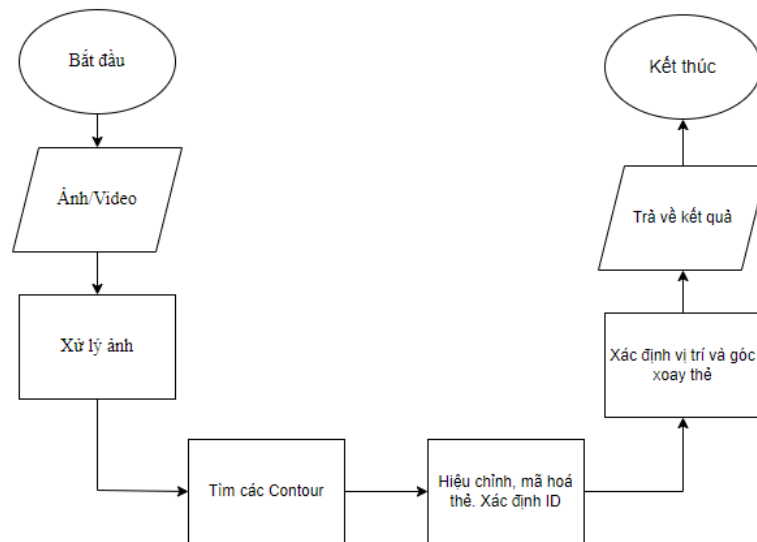
DICT_5X5_100: Từ điển gồm 100 mã ArUco, mỗi mã là một ma trận 5x5.

DICT_5X5_250: Từ điển gồm 250 mã ArUco, mỗi mã là một ma trận 5x5.

Mỗi mã trong từ điển có một ID duy nhất, và các mã được thiết kế sao cho có tính duy nhất và khả năng phát hiện tin cậy.

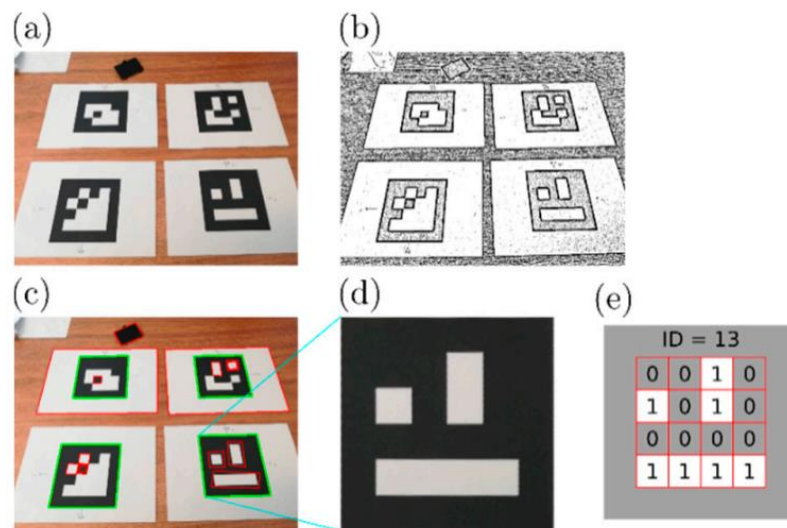
2.2.4.2 Nguyên lý nhận diện mã

Để nhận dạng một thẻ mã Aruco đại diện cho ID nào, chúng ta cần phân tách được mã đúng từ ảnh đầu vào, xử lý ảnh và số hóa thành các bit dữ liệu. Từ đó, so sánh với thư viện có sẵn của Aruco để xác định giá trị của mã đó. Quy trình thực hiện nhận dạng ID thẻ tag được mô tả như Hình 2.10:



Hình 2.10 Lưu đồ nhận diện thẻ tag

Chúng ta có thể nhìn trực quan hơn về quá trình nhận diện thẻ tag trong Hình 2.11



Hình 2.11 Quá trình phát hiện ID thẻ tag ArUco. (a) Khung gốc. (b) Kết quả phân ngưỡng. (c) Phát hiện đường viền; màu xanh lục, các ô vuông sẽ được xác định là điểm đánh dấu được hiển thị, thay vào đó, màu đỏ là tất cả các ô vuông bị từ chối. (d) Loại bỏ triển vọng. (e) Nhận dạng từ nhị phân.

Sau khi hoàn thành quá trình hiệu chuẩn camera, quá trình xác định vị trí mã Aruco bắt đầu. Quá trình này được mô tả một cách trực quan trong Hình 2.11 và thường được sử dụng trong các ứng dụng thị giác máy tính, với những biến thể nhỏ tùy thuộc vào các thuật toán cụ thể được sử dụng.

Đầu tiên, xác định hình ảnh đầu vào (a), có thể là một ảnh tĩnh hoặc một khung hình từ video, sau đó được chuyển đổi sang ảnh xám để tăng cường kết quả của bước tiếp theo, quá trình ngưỡng hóa thích nghi (b). Bước này giúp tạo ra một ảnh nhị phân với các đường viền liên quan đến sự thay đổi màu sắc đáng kể. Tiếp theo, các đường viền, đặc biệt là các đường viền liên quan đến sự thay đổi màu sắc đáng kể, được lọc bỏ bằng thuật toán phát hiện đường viền để loại bỏ các đường viền không được đóng và không xấp xỉ bởi bốn cạnh (c). Các đường viền còn lại có thể tạo thành các đánh dấu hình vuông. Đối với mỗi đường viền, quá trình loại bỏ góc nhìn (d) và mã hóa nhị phân (e) được thực hiện để kiểm tra xem đường viền có phải là một mã Aruco hay không. Mã hóa nhị phân chuyển đổi mẫu duy nhất của các ô vuông đen và trắng thành một ma trận nhị phân, mã hóa ID duy nhất của mã Aruco. Nếu một mã Aruco được xác định, ID của nó cùng với các góc tương ứng trong mặt phẳng được lưu trữ. Quá trình này được lặp lại cho tất cả các đường viền có khả năng là mã Aruco.

Khi đã kiểm tra tất cả các đối tượng có thể, quá trình xác định vị trí mã Aruco hoàn tất. Quá trình này tổng hợp các bước xử lý ảnh như chuyển đổi sang ảnh xám, phát hiện đường viền và mã hóa nhị phân để xác định và lưu trữ thông tin về vị trí của các mã Aruco được xác định trong hình ảnh hoặc khung hình.

Thuật toán nhận dạng các mã Aruco markers được tóm tắt theo đoạn mã giả sau đây:

while *Thu nhận frame từ camera do*

begin Phát hiện Đánh dấu Tự động

Lấy khung hình từ video.

Chuyển đổi khung hình sang ảnh xám.

Áp dụng ngưỡng thích nghi (adaptive thresholding) cho khung hình để tạo ra ảnh nhị phân.

Tìm các đường viền (contours) trong ảnh nhị phân.

Loại bỏ các đường viền có số điểm thấp.

Xấp xỉ các đường viền thành các hình chữ nhật.

Loại bỏ các hình chữ nhật quá gần nhau.

Loại bỏ các hình chữ nhật tiềm năng để có góc nhìn mặt trước.

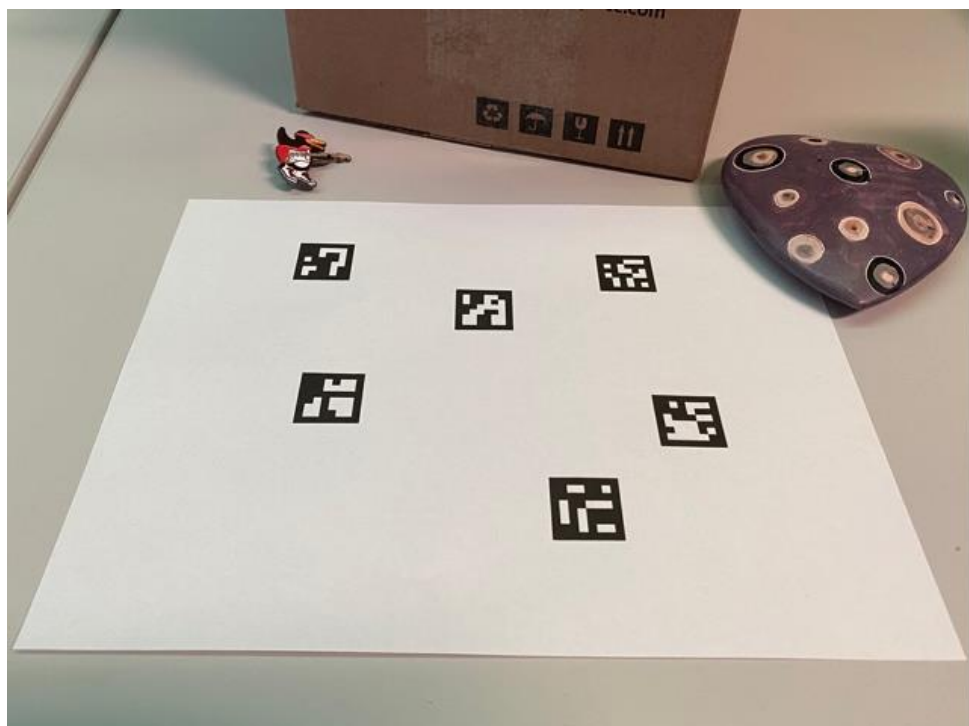
Áp dụng thuật toán ngưỡng Otsu để nhận dạng đánh dấu từ mã nhị phân.


```

        Nếu,
        if nhận dạng được đánh dấu then
            Tinh chỉnh các góc.
        end
        Đưa ra danh sách các góc và ID của đánh dấu.
    end
    begin Ước lượng Vị trí Đánh dấu:
        For mỗi ID do
            Áp dụng phép biến đổi từ 2D sang 3D cho các góc.
            Tạo danh sách các vị trí đối với máy ảnh (poses).
        end
    end
end

```

Các bước nhận diện mã Aruco được thể hiện cụ thể hơn dưới đây dựa theo hướng dẫn chi tiết trong Detection of ArUco Markers - OpenCV Documentation.

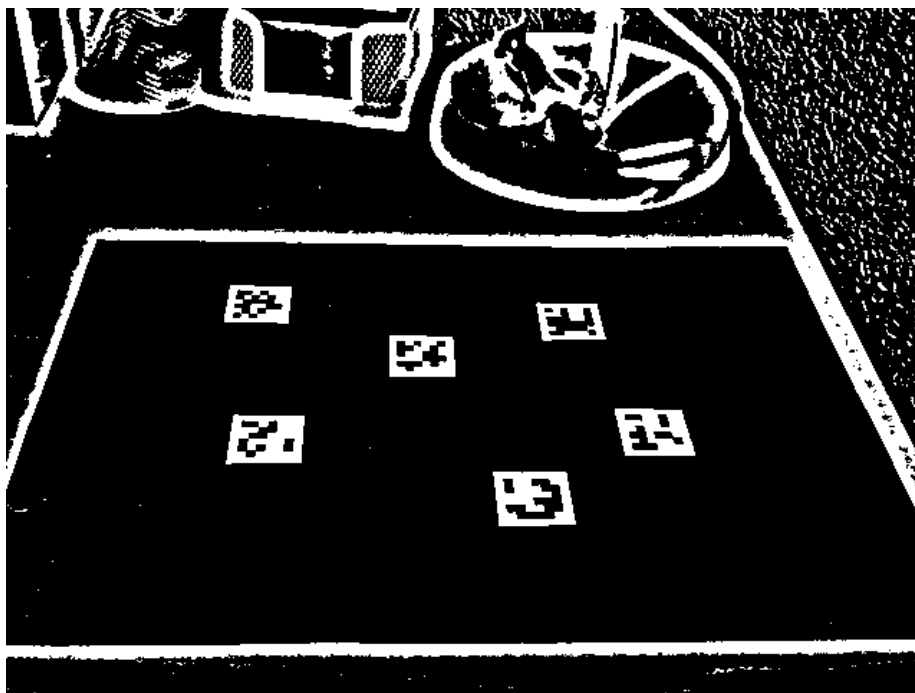


Hình 2.12 Ảnh đầu vào

Bước 1: Phân đoạn ảnh

Một trong những bước đầu tiên của quá trình nhận dạng mã là phân đoạn thích nghi hình ảnh đầu vào để phân đoạn ảnh thành hai lớp: lớp nền và lớp vật thể. Phương thức này tính giá trị trung bình của các điểm xung quanh mỗi pixel rồi trừ giá trị trung bình đó cho mỗi pixel. Quá trình này tạo ra một ảnh mới trong đó giá trị của mỗi pixel được dịch chuyển và căn chỉnh để cải thiện độ tương phản và phân biệt các đối tượng trong ảnh.

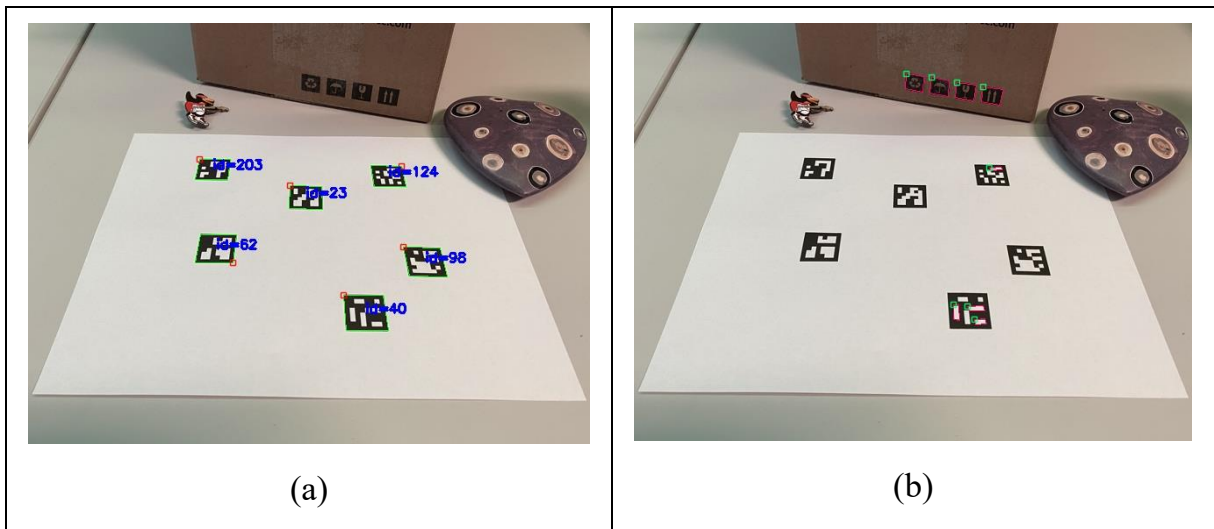
Ví dụ về phân đoạn cho hình ảnh mẫu được sử dụng ở trên là:



Hình 2.13 Ảnh đã phân đoạn

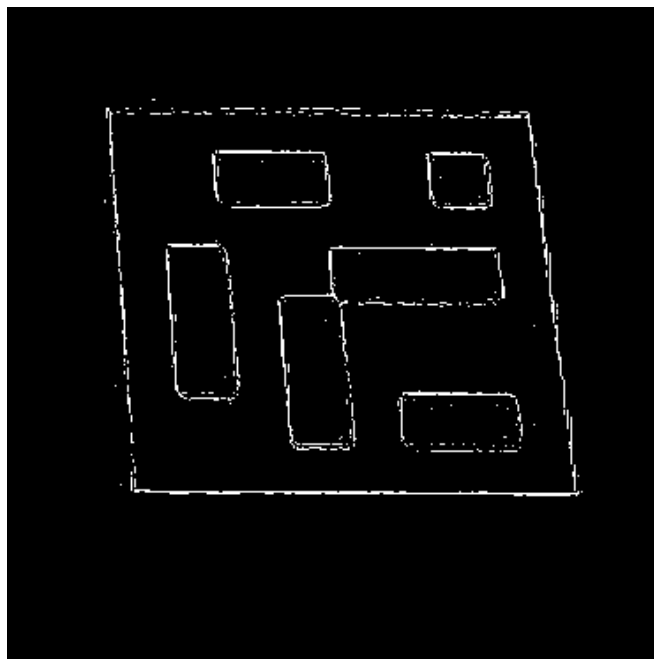
Bước 2: Lọc đường viền

Sau khi phân đoạn ảnh, đường viền sẽ được nhận diện, việc lọc và loại bỏ các đối tượng sai là một bước quan trọng để đảm bảo độ chính xác và hiệu suất của quá trình nhận diện mã. Tuy nhiên, không phải tất cả các đường viền được coi là các đối tượng mã. Chúng được lọc ra theo các bước khác nhau để các đường viền không chắc sẽ bị đánh dấu bị loại bỏ.



Hình 2.14 Các mã được chấp nhận (a) và bị loại bỏ (b)

Nếu tất cả các đường viền được xem xét và sẽ được xử lý trong các giai đoạn sau sẽ gây mất thời gian, kéo theo chi phí tính toán cao hơn. Vì vậy, việc loại bỏ các đối tượng sai trong giai đoạn này là thích hợp hơn. Mặt khác, nếu các điều kiện lọc quá chặt chẽ, các đường viền thực của mã có thể bị loại bỏ, và do đó không được phát hiện.



Hình 2.15 Mã bị lỗi do điều kiện lọc

Bước 3. Trích xuất bits

Trước khi phân tích mã nhị phân chính, chúng ta cần thực hiện quá trình số hóa (binarization) để trích xuất các bit của mỗi đối tượng. Chuyển đổi sang ảnh nhị phân làm cho việc phân tích mã trở nên dễ dàng hơn và giúp loại bỏ các yếu tố không mong muốn trong quá trình phân tích. Sau khi nhận dạng đối tượng, các bit của mỗi đối tượng

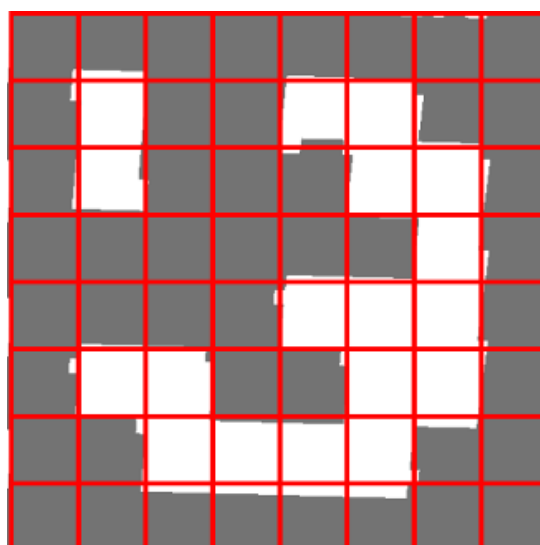
sẽ được phân tích để xác định xem chúng có phải là mã thực sự hay không. Trước khi phân tích mã nhị phân chính nó, ta cần trích xuất các bit (quá trình số hóa). Để làm như vậy, sự méo mó bị loại bỏ và hình ảnh kết quả được phân đoạn sử dụng thuật toán Otsu để tách riêng các điểm ảnh màu đen và trắng.

Đây là một ví dụ về hình ảnh thu được sau khi loại bỏ sự méo mó:



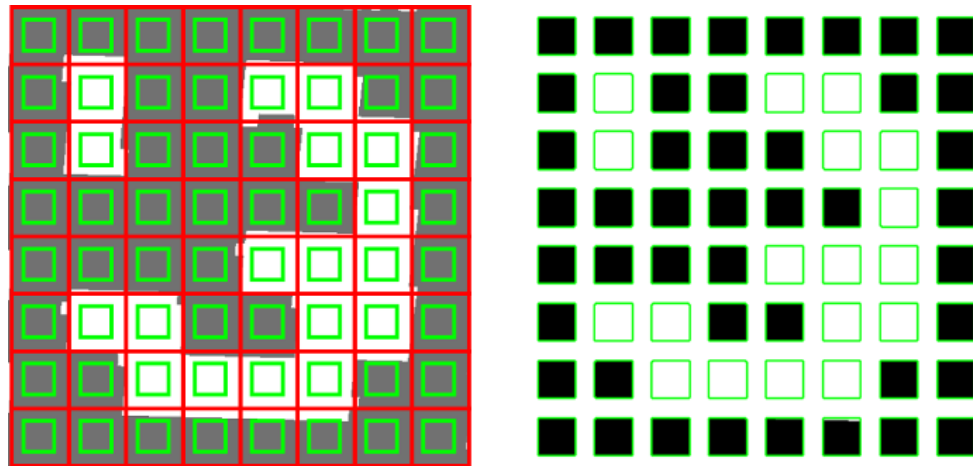
Hình 2.16 Thay đổi phối cảnh

Sau khi số hóa hình ảnh và chia thành lưới, quá trình giải nén bit của mỗi phần được thực hiện bằng cách tính số lượng điểm ảnh màu đen và trắng trong mỗi ô vuông của lưới. Trên mỗi ô, số lượng điểm ảnh màu đen và trắng được tính để quyết định bit được gán cho ô nào (từ giá trị lớn):



Hình 2.17 Lưới ô đánh dấu

Khi giải nén bit của mỗi phần, số lượng điểm ảnh màu đen và trắng được tính. Nói chung, không nên xem xét tất cả các điểm ảnh di động. Thay vào đó, tốt hơn là bỏ qua một số điểm ảnh ở mép của các phần. Vì sau khi loại bỏ sự méo mó, màu của các phần nói chung không tách rời hoàn toàn và các phần trắng có thể lấn chiếm một số pixel của các phần đen (và ngược lại). Vì vậy, tốt hơn là bỏ qua một số điểm ảnh để tránh tính các điểm ảnh sai. Chỉ các pixel bên trong các ô vuông màu xanh lá cây được dùng để phân loại.



Hình 2.18 Lê ô đánh dấu

Bước 4: Nhận diện mã

Sau khi các bit đã được trích xuất, bước tiếp theo kiểm tra nếu chiết xuất mã thuộc về từ điển mã ArUco có sẵn trong thư viện, và nếu cần thiết, sửa lỗi có thể được thực hiện. Các thẻ Aruco có nhiều loại và kích thước mã ID khác nhau (ví dụ: 4x4, 5x5, 6x6, 7x7). Các kích thước mã ID nhỏ hơn thường cho phép nhận diện chính xác hơn, nhưng có giới hạn về số lượng mã.

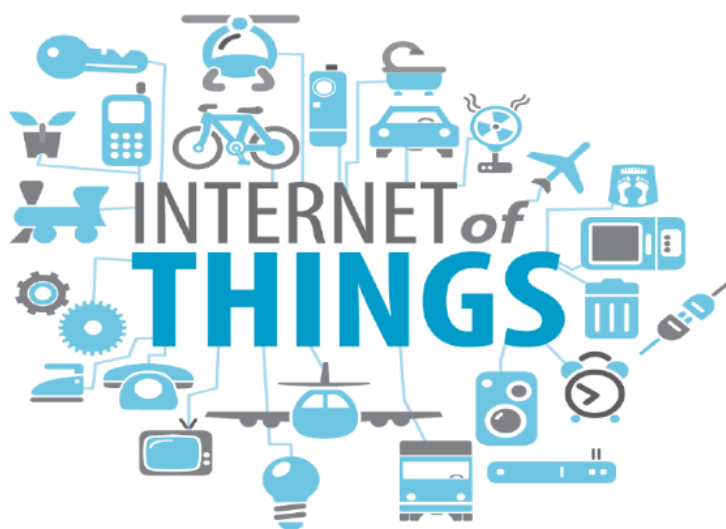
2.3 Giao thức truyền phát MQTT

MQTT (Message Queue Telemetry Transport) là một giao thức gửi tin nhắn nhẹ nhàng và hiệu quả trong mạng IoT (Internet of Things). Nó được thiết kế để hoạt động trong môi trường có băng thông hạn chế, đặc biệt là trên các thiết bị nhúng có tài nguyên hạn chế. MQTT được phát triển bởi Andy Stanford - Clark và Arlen Nipper vào cuối năm 1999. Nhiệm vụ của họ là tạo ra một giao thức sao cho sự hao phí băng thông và năng lượng là thấp nhất. Từ đó kết nối đến đường ống dẫn dầu thông qua sự kết nối giữa các vệ tinh.

Đến năm 2011, Eurotech và IBM đã trao lại MQTT cho Paho - một dự án của Eclipse. Năm 2013, MQTT được trình lên Organization for the Advancement of Structured Information Standards (OASIS). MQTT được chuẩn hoá và dần trở thành một trong những giao thức được sử dụng phổ biến nhất được sử dụng trong IoT hiện nay.

2.3.1 Internet of Things (IoT)

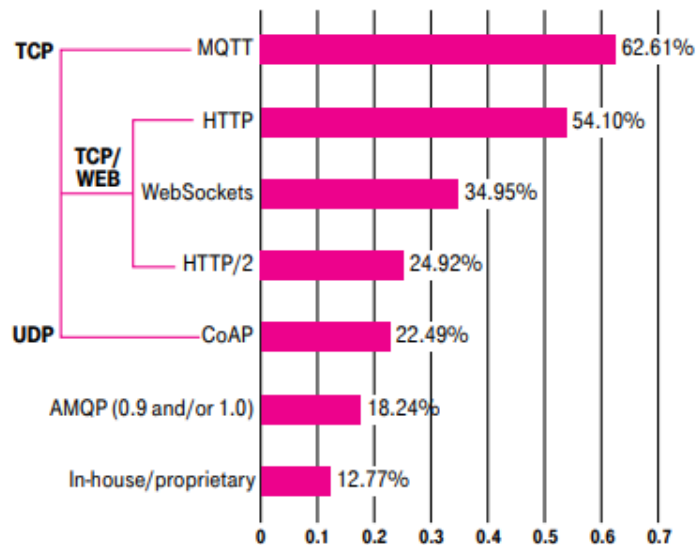
Internet of Things (IoT) được hiểu một cách đơn giản là mô tả sự kết nối của rất rất nhiều các thiết bị hay nói chung là vật thể (things) xung quanh chúng ta trong một mạng lưới như mạng Internet mà hàng ngày chúng ta vẫn sử dụng. Và tất cả các thiết bị này đều được quản lý, kiểm soát thông qua kết nối không dây, giao thức kết nối mạng. Cụm từ này được đưa ra bởi Kevin Ashton vào năm 1999 Ông là một nhà khoa học đã sáng lập ra Trung tâm Auto-ID ở đại học MIT (Massachusetts Institute of Technology hay Viện công nghệ Massachusetts).



Hình 2.19. Internet of Things

IoT nhằm đưa thế giới tiến tới những kỹ thuật tiên tiến nhất, thông minh nhất, một thế giới mà các thiết bị có thể đo lường và đưa ra các nhận xét chính xác nhất. Để thực hiện được một thế giới thông minh như vậy thì việc tìm hiểu về cách thức hoạt động và sự giao tiếp giữa các thiết bị IoT là cực kỳ quan trọng.

Có nhiều giao thức khác nhau cho các lớp khác nhau trong kiến trúc IoT. Dưới đây là bảng so sánh độ phổ biến của các giao thức truyền dữ liệu IoT trong năm 2018 (Theo nguồn *t-mobile.com*).



Hình 2.20 Độ phổ biến của các giao thức truyền dữ liệu IoT năm 2018

HTTP là giao thức phổ biến và được sử dụng rộng rãi, tuy nhiên trong những năm qua MQTT nhanh chóng đạt được các tín nhiệm từ thị trường IoT. Các nhà phát triển IoT hiện nay đang phải lựa chọn giữa hai giao thức HTTP và MQTT. Dưới đây là bảng so sánh giữa hai giao thức:

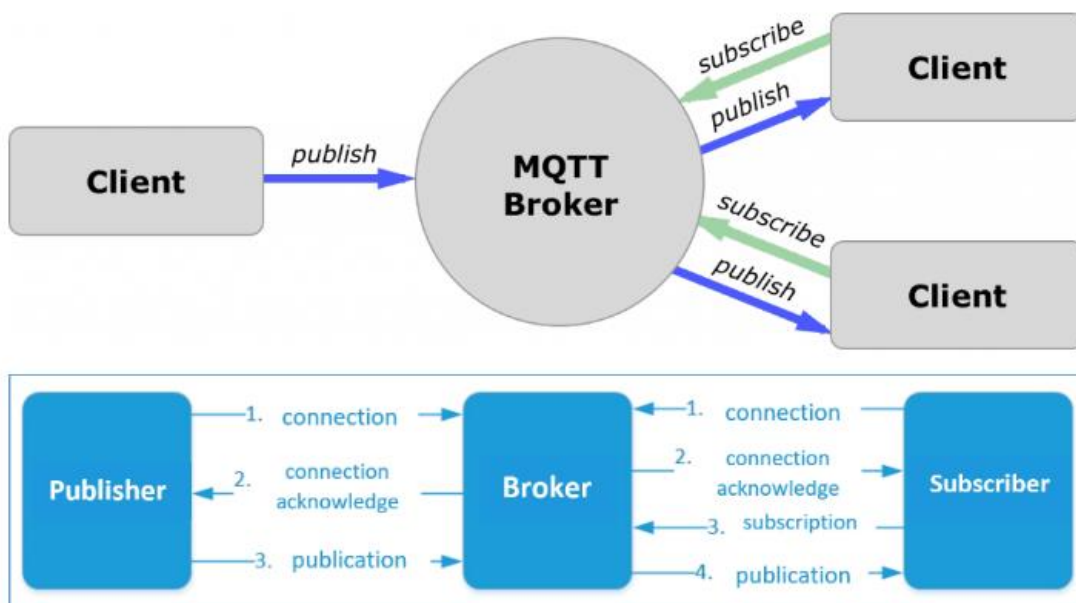
Bảng 2.1 So sánh một số đặc trưng 2 giao thức HTTP và MQTT

Đặc điểm	MQTT	HTTP
Hình thức	Tin nhắn xếp hàng từ xa	Giao thức truyền siêu văn bản
Kiến trúc	Publisher/subscriber: các thiết bị có thể publish hoặc subscribe chủ đề bất kỳ	Kiến trúc Yêu cầu/ đáp ứng
Độ phức tạp	Đơn giản	Phức tạp hơn
Bảo mật dữ liệu	Có	Không
Giao thức lớp trên	TCP	UDP
Kích thước tin nhắn	Nhỏ - nhị phân	Lớn – dạng ASCII
Thư viện	30KB (C), 100KB (Java)	Lớn
Phân phối	1-N, N-1, 1-1	1-1

MQTT thích hợp cho các ứng dụng IoT có yêu cầu thời gian thực, tài nguyên hạn chế và yêu cầu tiết kiệm năng lượng. MQTT sử dụng mô hình Publish/Subscribe, trong đó các thiết bị gửi tin nhắn (publish) tới các địa chỉ (topics) và các thiết bị khác nhận và đăng ký (subscribe) vào các địa chỉ đó để nhận tin nhắn. Giao thức này tiết kiệm năng lượng và có khả năng xử lý lượng tin nhắn lớn.

2.3.2 Kiến trúc MQTT

MQTT có mô hình client/server, nơi mà mỗi cảm biến là một khách hàng (client) và kết nối đến một máy chủ, có thể hiểu như một nhà môi giới (broker) thông qua giao thức TCP. MQTT là giao thức định hướng bản tin (message), mỗi bản tin là một tín hiệu rời rạc và broker không thể biết trước. Mỗi message được publish lên một địa chỉ, có thể hiểu là một kênh. Client đăng kí vào một vài kênh để nhận và gửi dữ liệu, gọi là subscribe. Client có thể subscribe vào nhiều kênh, và nhận được dữ liệu khi bất kì trạm nào khác gửi dữ liệu lên kênh đã đăng kí. Khi một client gửi một message đến một kênh nào đó, gọi là publish. Các mô hình publish/subscribe cho phép các client MQTT có thể giao tiếp 1-1, 1-N và N-1. Dưới đây là hình ảnh mô tả kết nối giao thức MQTT:



Hình 2.21 Quá trình Subscribe và publish với broker

MQTT hỗ trợ ba mức độ chất lượng dịch vụ (QoS – Quality of service) để đảm bảo giao tiếp tin cậy trong mạng. Các mức QoS bao gồm:

- QoS 0 (At most once): Tin nhắn được gửi một lần mà không có xác nhận hoặc đảm bảo giao đến đích. Mức độ này đảm bảo tin nhắn được gửi nhanh nhưng có thể bị mất hoặc trùng lặp.
- QoS 1 (At least once): Tin nhắn được gửi ít nhất một lần và được xác nhận bởi MQTT Broker. Nếu tin nhắn không được gửi thành công, nó sẽ được gửi lại cho đến khi được giao đến đích.
- QoS 2 (Exactly once): Tin nhắn được gửi đúng một lần và đảm bảo giao đến đích một lần duy nhất. Mức độ này đảm bảo tính chính xác và đáng tin cậy nhưng yêu cầu nhiều tài nguyên và độ trễ cao hơn.

Topic và Payload: Trong MQTT, tin nhắn được gửi qua các topics, là các địa chỉ để xác định mục tiêu nhận tin nhắn. Mỗi topic có một cây phân cấp, cho phép phân loại và quản lý dữ liệu trong mạng. Tin nhắn MQTT cũng chứa payload, là nội dung chính của tin nhắn, có thể là dữ liệu cảm biến, trạng thái thiết bị, hoặc thông điệp điều khiển. Topic được phân cấp theo hình thức cây thư mục, sử dụng dấu “/” phân cách giữa các cấp. Tin nhắn có thể là bất kỳ dữ liệu nào bao gồm kí tự văn bản, hình ảnh. Âm thanh,..

Security: MQTT hỗ trợ các phương pháp bảo mật như xác thực và mã hóa dữ liệu để bảo vệ thông tin trong mạng. Xác thực có thể được thực hiện bằng cách sử dụng tên người dùng và mật khẩu hoặc chứng chỉ số. Mã hóa dữ liệu có thể sử dụng giao thức TLS/SSL để bảo vệ quá trình truyền tải tin nhắn.

2.3.3 Nguyên lý làm việc

Một quá trình làm việc của MQTT được chia làm bốn giai đoạn: kết nối, xác thực, giao tiếp và kết thúc. Một client bắt đầu bằng cách tạo ra một kết nối TCP/IP tới broker bằng cách sử dụng một cổng tiêu chuẩn hoặc một cổng tùy chỉnh được xác định bởi các phương thức của broker. Các cổng tiêu chuẩn là 1883 không được mã hoá giao tiếp và 8883 được mã hoá giao tiếp sử dụng giao thức SSL/TLS. Các client cũng có thể cung cấp một chứng chỉ client tới broker để trong sự kết hợp này broker có thể sử dụng để xác thực cho client.

Mỗi gói MQTT bao gồm các header cố định (2byte), header thay đổi được và payload có giới hạn là 256MB chứa thông tin và chất lượng dịch vụ (QoS). QoS càng cao càng đáng tin cậy, tuy nhiên độ trễ sẽ lớn hơn và yêu cầu băng thông lớn hơn, khách hàng có thể đăng kí xác nhận mức độ QoS mà họ muốn nhận.

Trong quá trình truyền, một client có thể thực hiện xuất bản, đăng ký, hủy đăng ký và hoạt động ping. Hoạt động xuất bản gửi một khối dữ liệu nhị phân tới một chủ đề được xác định bởi các thiết bị xuất bản. MQTT hỗ trợ tin nhắn BLOBS lên đến 256MB. Các định dạng của nội dung là các ứng dụng cụ thể. Các chủ đề đăng ký sử dụng một cặp gói tin Subscribe/Suback, hủy đăng ký sử dụng cặp gói tin Unsubscribe/Unsuback.



Hình 2.22 Ví dụ cụ thể Facebook sử dụng MQTT

Facebook sử dụng MQTT để chat trên messenger mỗi "nhóm chat" hay mỗi "Facebook group" là một topic được tạo ra và tất cả thành viên trong cuộc trò chuyện đó đăng ký (subscribe) để lắng nghe và gửi tin (publish) đến topic đó. Khi một thành viên publish message bất kì lên "nhóm chat" hay group trên Facebook, tất cả các thành viên subscribe topic đó sẽ nhận được thông báo và xem được tin nhắn. Cơ sở hạ tầng máy chủ MQTT của ứng dụng được tạo thành từ "nhiều" broker con khác nhau, và sử dụng topic để định hướng các gói tin MQTT.

Trong quá trình truyền thông, một client có thể thực hiện xuất bản, đăng ký, hủy đăng ký và hoạt động ping. Hoạt động xuất bản gửi một khối dữ liệu nhị phân tới một chủ đề được xác định bởi các thiết bị xuất bản. MQTT hỗ trợ tin nhắn BLOBS lên đến 256MB. Các định dạng của nội dung là các ứng dụng cụ thể. Các chủ đề đăng ký sử dụng một cặp gói tin Subscribe/Suback, hủy đăng ký sử dụng cặp gói tin Unsubscribe/Unsuback. Hoạt động thứ tư mà client thực hiện trong quá trình liên lạc là ping tới broker sử dụng gói Pingreq Pingresp. Hoạt động này không có chức năng nào khác ngoài việc duy trì kết nối trực tiếp và đảm bảo kết nối TCP/IP không bị đóng lại bởi một gateway hoặc router.

Khi một thiết bị muốn chấm dứt một quá trình MQTT, nó sẽ gửi một tin nhắn Disconnect cho broker và sau đó đóng kết nối. Điều này được gọi là tắt máy bởi vì nó cung cấp cho client với khả năng dễ dàng kết nối lại bằng cách cung cấp danh tính khách hàng và khôi phục lại nơi nó bị tắt. Các ngắt kết nối xảy ra đột ngột sẽ không có thời gian cho các thiết bị gửi tin nhắn Disconnect, các broker có thể gửi các tin nhắn đăng ký từ các thiết bị mà broker đã lưu trữ trước đó. Các tin nhắn đó cung cấp cho các thiết bị phải làm gì nếu nhà xuất bản bất ngờ ngắt kết nối.

Tóm lại, ta có thể tổng kết ưu và nhược điểm của hệ giao thức truyền tải MQTT như sau:

Ưu điểm:

- Nhẹ và hiệu quả: MQTT là một giao thức nhẹ và hiệu quả, được thiết kế để tiết kiệm năng lượng và băng thông. Điều này làm cho nó phù hợp cho các thiết bị có tài nguyên hạn chế và mạng có băng thông thấp.
- Linh hoạt và dễ triển khai: MQTT hỗ trợ nhiều loại thiết bị và nền tảng, và có thể triển khai dễ dàng trên các môi trường khác nhau.
- Khả năng mở rộng: MQTT hỗ trợ cơ chế publish-subscribe, cho phép các thiết bị gửi và nhận thông tin một cách linh hoạt. Điều này giúp giải quyết vấn đề mở rộng hệ thống IoT khi số lượng thiết bị tăng lên.
- Hỗ trợ nhiều ngôn ngữ và nền tảng: MQTT được hỗ trợ bởi nhiều ngôn ngữ lập trình và nền tảng, giúp cho việc phát triển ứng dụng trở nên đơn giản và linh hoạt.

Nhược điểm: Bảo mật yếu: MQTT không cung cấp bảo mật mặc định và phụ thuộc vào các biện pháp bảo mật bổ sung như SSL/TLS để đảm bảo tính bảo mật trong việc truyền tải dữ liệu. Nếu không được cấu hình chính xác, giao thức có thể dễ bị tấn công và thông tin nhạy cảm có thể bị lộ.

Tuy có nhược điểm nhất định, nhưng MQTT vẫn là một giao thức mạnh mẽ và phổ biến, đáp ứng nhu cầu truyền tải dữ liệu đơn giản, hiệu quả và đáng tin cậy.

2.4 Các phần mềm hỗ trợ

2.4.1 PyCharm Community Edition

Chương trình game được lập trình bằng ngôn ngữ Python 3.9 trên phần mềm PyCharm Community Edition phiên bản 2022. Để chạy trích xuất được hình ảnh thực tế game cần một webcam có độ phân giải tốt để lấy hình ảnh một cách chính xác nhất. Python là ngôn ngữ lập trình có mục đích chung do Guido van Rossum bắt đầu, ngôn ngữ này đã trở nên rất phổ biến rất nhanh chóng, chủ yếu là do tính đơn giản và dễ đọc mã của nó. Nó cho phép lập trình viên diễn đạt ý tưởng bằng ít dòng mã hơn mà không làm giảm khả năng đọc.

So với các ngôn ngữ như C/C++, Python chậm hơn. Điều đó nói rằng, Python có thể dễ dàng được mở rộng bằng C/C++, cho phép chúng ta viết mã chuyên sâu về tính toán trong C/C++ và tạo các trình bao bọc Python có thể được sử dụng làm mô-đun Python. Điều này mang lại cho chúng tôi hai lợi thế: thứ nhất, mã này nhanh như mã C/C++ gốc (vì đây là mã C++ thực tế hoạt động ở chế độ nền) và thứ hai, mã Python dễ viết hơn C/C++. OpenCV-Python là một trình bao bọc Python để triển khai OpenCV C++ ban đầu được thiết kế để giải quyết các vấn đề về thị giác máy tính.

Thư viện OpenCV đi kèm với hỗ trợ ArUco tích hợp sẵn, cả để tạo các điểm đánh dấu ArUco và để phát hiện chúng. Phát hiện các điểm đánh dấu ArUco bằng OpenCV là một quy trình gồm ba bước có thể thực hiện được thông qua mô hình con: cv2.aruco. Thư viện Aruco trong OpenCV được cài đặt dưới dạng một module riêng biệt, có thể được tải xuống và cài đặt thông qua OpenCV.

Các chức năng chính của thư viện Aruco trong OpenCV bao gồm:

- Tạo ra các marker Aruco với kích thước và mức độ chi tiết khác nhau.
- Phát hiện các marker Aruco trong hình ảnh và tính toán vị trí và góc xoay của chúng trong không gian 3D.
- Cung cấp các công cụ để hiệu chỉnh camera, giúp cải thiện độ chính xác của việc tính toán vị trí và góc xoay của marker.
- Tính toán các thông số của camera như ma trận camera và vector nghiêng, giúp cải thiện độ chính xác của việc tính toán vị trí và góc xoay của marker.
- Cung cấp các ví dụ về sử dụng thư viện để giúp bạn bắt đầu sử dụng thư viện Aruco trong ứng dụng của mình.

2.4.2 Mosquitto Broker

Mosquitto MQTT là một phần mềm broker MQTT mã nguồn mở, được phát triển bởi Eclipse Foundation. Nó là một phần mềm nhẹ, đơn giản và hiệu suất cao, được sử dụng để triển khai hệ thống giao tiếp MQTT trong các ứng dụng IoT và máy tính.

Mosquitto cung cấp một môi trường đáng tin cậy và linh hoạt cho việc gửi và nhận các thông điệp MQTT qua mạng. Nó tuân theo các tiêu chuẩn và giao thức MQTT, cho phép các thiết bị và ứng dụng khác nhau kết nối và giao tiếp với nhau một cách dễ dàng. Một trong những điểm mạnh của Mosquitto là tính nhẹ nhàng và hiệu suất cao. Nó được thiết kế để tiêu thụ ít tài nguyên hệ thống và hoạt động ổn định ngay cả trên các thiết bị có tài nguyên hạn chế. Điều này làm cho Mosquitto MQTT phù hợp với các hệ thống IoT có số lượng thiết bị lớn và đòi hỏi khả năng xử lý nhanh và hiệu quả. Mosquitto MQTT cung cấp các tính năng quan trọng như chủ đề (topic), đăng ký (subscribe) và xuất bản (publish) thông điệp. Người dùng có thể tạo và quản lý các chủ đề, cho phép các thiết bị và ứng dụng đăng ký nhận thông điệp từ chủ đề cụ thể và gửi thông điệp đến chủ đề tương ứng.

Ngoài ra, Mosquitto MQTT hỗ trợ bảo mật thông qua sử dụng giao thức TLS/SSL để bảo vệ kết nối và truyền thông. Điều này đảm bảo rằng dữ liệu gửi đi và nhận về qua mạng được mã hóa và bảo mật, ngăn chặn việc truy cập trái phép và đảm bảo tính riêng tư.

2.4.3 Gdevelop5

GDevelop 5 được phát triển bởi một nhóm các nhà phát triển tại Flowlab, một công ty chuyên về phát triển công cụ và nền tảng cho trò chơi. Nền tảng này được viết bằng ngôn ngữ lập trình C++ và sử dụng một số thư viện và công nghệ mã nguồn mở như PixiJS và Box2D. Đây là một nền tảng phát triển trò chơi đa nền tảng được thiết kế để giúp người dùng tạo ra các trò chơi một cách dễ dàng và hiệu quả. Nền tảng này có một loạt các tính năng và công cụ giúp người dùng xây dựng trò chơi 2D và 3D mà không cần kiến thức lập trình sâu.



Hình 2.23. Nền tảng Gdevelop5

Một trong những đặc điểm quan trọng của GDevelop5 là giao diện đồ họa trực quan. Với giao diện này, người dùng có thể kéo và thả các thành phần của trò chơi để xây dựng các cảnh, tạo các sự kiện và thiết lập các tương tác giữa các đối tượng. Việc sử dụng giao diện đồ họa giúp giảm thiểu sự phụ thuộc vào việc viết mã lập trình truyền thống và cho phép người dùng tập trung vào việc tạo nội dung và trải nghiệm trò chơi.

GDevelop 5 cung cấp một loạt các công cụ và trình chỉnh sửa tài nguyên giúp người dùng tạo ra các đồ họa, âm thanh và video cho trò chơi của mình. Người dùng có thể nhập khẩu tài nguyên tồn tại hoặc tạo ra tài nguyên mới trong chính nền tảng. Điều này giúp tạo ra các trò chơi có nội dung đa dạng và phong phú. Nền tảng GDevelop 5 cho phép người dùng triển khai trò chơi trên nhiều nền tảng khác nhau như Windows, macOS, Linux, Android, iOS và HTML5. Điều này giúp người dùng tiếp cận một lượng lớn người chơi trên các thiết bị khác nhau và tận dụng tối đa thị trường trò chơi đa dạng.

Hơn nữa, GDevelop 5 hỗ trợ tích hợp các công nghệ và dịch vụ bên thứ ba, cho phép người dùng tạo ra trò chơi có tính tương tác cao và tích hợp các chức năng mạng xã hội. Người dùng có thể sử dụng các công nghệ như bảng xếp hạng, chia sẻ kết quả và tích hợp mạng xã hội để tạo ra trò chơi phong phú và kết nối với cộng đồng người chơi. Công nghệ này hỗ trợ giao thức MQTT để kết nối và giao tiếp với các thiết bị và máy chủ thông qua mạng. Việc sử dụng MQTT trong GDevelop 5 cho phép người dùng tạo ra các ứng dụng và trò chơi liên quan đến giao tiếp máy tính và Internet of Things (IoT).

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

3.1 Phân tích hệ thống

3.1.1 Mục đích hệ thống

Mục đích của hệ thống là tạo ra một nền tảng trò chơi chi phí thấp cho trẻ em, ứng dụng trong môi trường giáo dục như trường học, lớp học hay các trung tâm giáo dục. Trẻ em sử dụng trò chơi dưới sự hướng dẫn của người lớn. Các trò chơi được thiết kế mang tính giáo dục và giải trí để trẻ em chơi trong một không gian được quy định, người chơi sẽ cùng nhìn lên một màn hình cỡ lớn, tương tác trực tiếp với trò chơi bằng việc bằng di chuyển trong không gian đó và tương tác lẫn nhau. Từ đó giúp trẻ em phát triển toàn diện và duy trì sức khỏe tốt, tránh được những tác hại từ các trò chơi điện tử gây ra.

3.1.2 Đối tượng của hệ thống

Trò chơi ứng dụng trong môi trường giáo dục, áp dụng cho nhóm trẻ em từ 6-10 tuổi, số lượng người chơi mỗi lượt tối đa là 10 người.

3.1.3 Chức năng của hệ thống

Từ những yêu cầu đó, hệ thống có các chức năng chính là:

- Nhận diện thẻ ArUco: nhận diện người chơi và vị trí của người chơi và hiển thị lên màn hình
- Cho phép người chơi tương tác thông qua các thẻ tag ArUco. Các tương tác bao gồm:
 - Di chuyển theo hướng và tốc độ mong muốn
 - Chạm vào vật cản trên màn hình
- Sử dụng tương tác được định nghĩa trên để tham gia các trò chơi được thiết kế sẵn.

3.2 Kiến trúc tổng quan

3.2.1 Mô hình trò chơi

Mô hình trò chơi được thiết kế bao gồm các thành phần sau:

Camera Top-down:

Trò chơi sử dụng vị trí thực của người chơi trong không gian thực nên cần sử dụng một camera top-down để theo dõi toàn bộ khu vực trò chơi. Camera này được đặt ở trên cao và có góc nhìn rộng để bao phủ toàn bộ không gian trò chơi. Nhiệm vụ của camera là chụp hình ảnh và ghi lại các di chuyển và tương tác của người chơi trong trò chơi.

Máy chiếu và Màn chiếu:

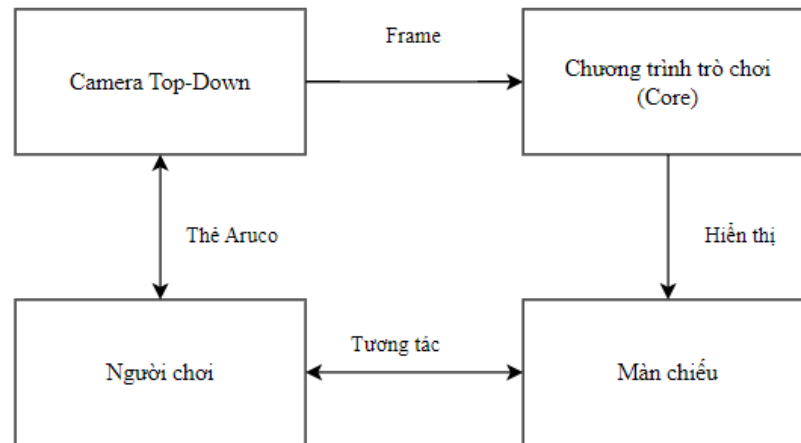
Máy chiếu được đặt tại một vị trí phù hợp trong không gian trò chơi để hiển thị hình ảnh đồ hoạ được thiết kế - giao diện chính của trò chơi. Người chơi sẽ trực tiếp quan sát máy chiếu và tìm kiếm vị trí của mình để tương tác trò chơi.

Màn chiếu có chức năng hiển thị frame từ camera top – down, giúp nhận diện người chơi đứng trong khung hình nhằm trích xuất vị trí để người chơi xuất hiện trên máy chiếu. Màn chiếu có thể được quản lý bởi người hướng dẫn trò chơi.

Thẻ tag ArUco:

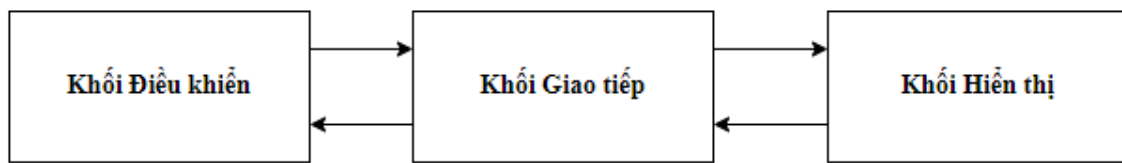
Người chơi đeo thẻ tag ArUco, để thích hợp với việc di chuyển trong không gian thực thẻ tag ArUco có thể được gắn trên mũ. Thẻ tag ArUco có mã nhận dạng riêng biệt và duy nhất, cho phép hệ thống nhận biết và định danh người chơi. Khi người chơi di chuyển hoặc tương tác với thẻ tag ArUco trong không gian thực, các tín hiệu từ thẻ được gửi đến trò chơi để xử lý và tạo ra các tương tác tương ứng trong trò chơi.

Ta có thể mô hình hoá trò chơi bằng sơ đồ sau:



Hình 3.1 Mô hình trò chơi

3.2.2 Sơ đồ nguyên lý



Hình 3.2 Sơ đồ khối nguyên lý hoạt động hệ thống

Hệ thống trò chơi bao gồm ba khối chính: Khối Điều khiển, Khối Giao tiếp và Khối Hiển thị. Mỗi khối thực hiện một nhiệm vụ cụ thể trong hệ thống. Dưới đây là mô tả chi tiết về sơ đồ nguyên lý của hệ thống:

Khối Điều khiển:

Khối Điều khiển được triển khai bằng mã Aruco và Python. Nhiệm vụ của Khối Điều khiển là thu thập thông tin từ các nguồn dữ liệu, xử lý và quyết định tạo ra các lệnh điều khiển dựa trên các dữ liệu này. Sau khi xử lý, Khối Điều khiển gửi lệnh điều khiển tới Khối Giao tiếp thông qua giao thức MQTT. Lệnh này có thể là các yêu cầu di chuyển, dừng, đổi hướng, hoặc bất kỳ hành động điều khiển nào cần thiết.

Khối Giao tiếp:

Khối Giao tiếp chịu trách nhiệm nhận lệnh điều khiển từ Khối Điều khiển thông qua MQTT. Sau khi nhận được message lệnh, Khối Giao tiếp thực hiện gửi các lệnh

được yêu cầu sang Khối Hiển thị. Thông tin được vận chuyển có thể bao gồm trạng thái vận chuyển hiện tại, tọa độ, hướng di chuyển và bất kỳ thông tin quan trọng nào liên quan đến trò chơi.

Khối Hiển thị:

Khối Hiển thị là giao diện người dùng được triển khai bằng GDevelop. Khối Hiển thị chịu trách nhiệm hiển thị thông tin được gửi từ Khối Giao tiếp qua giao thức MQTT. Khối hiển thị có thể cung cấp các chức năng tương tác cho người dùng, cho phép họ tương tác với các vật thể được hiển thị trên màn hình. Khối hiển thị cũng có thể gửi thông tin cần thiết cho Khối điều khiển thông qua Khối Giao tiếp trong những trường hợp cần xử lý.

3.3 Nguyên lý hoạt động hệ thống

3.3.1 Khối điều khiển

Khối Điều khiển là một phần quan trọng nhất trong hệ thống, chịu trách nhiệm nhận diện mã ArUco từ hình ảnh và video của camera, sau đó xử lý, trích xuất thông tin cần thiết của người chơi hay tạo các lệnh điều khiển dựa trên thông tin nhận diện. Dưới đây là mô tả chi tiết về nhiệm vụ của khối điều khiển:

Tiếp nhận khung hình từ camera để thực hiện quá trình nhận diện:

Khối Điều khiển hiệu chỉnh và nhận và xử lý dữ liệu từ Camera. Sử dụng các thuật toán xử lý hình ảnh và công nghệ nhận dạng, Khối Điều khiển tìm kiếm các mã ArUco trong ảnh và nhận diện các mã đã được định danh trước đó. Sử dụng các thư viện và công cụ phân tích hình ảnh, khối Điều khiển sẽ tiến hành nhận diện mã ArUco từ hình ảnh hoặc video. Mã ArUco là một loại mã nhận dạng được in trên các thẻ có tính độc lập và có thể dễ dàng nhận diện từ hình ảnh. Chi tiết quá trình nhận diện đã được mô tả ở chương trước.

Trích xuất thông tin:

Sau khi nhận diện được mã ArUco, khối Điều khiển trích xuất thông tin từ mã đó, bao gồm ID (định danh) và vị trí của mã ArUco trong không gian trò chơi. Thông tin này bao gồm các thông số như tọa độ x, y của mã ArUco trong hệ tọa độ của Camera.

Tạo lệnh điều khiển:

Dựa trên thông tin nhận diện từ mã ArUco, khối Điều khiển sẽ thực hiện các xử lý và quyết định để tạo ra các lệnh điều khiển cần thiết. Lệnh điều khiển có thể là các yêu cầu di chuyển, dừng, đổi hướng, hoặc các hành động khác liên quan đến quá trình điều khiển trò chơi. Tùy vào cách thức lập trình, mỗi ID sẽ được quy định một lệnh riêng để được sử dụng làm thẻ lệnh, điều này người chơi không cần thao tác trên màn hình các thiết bị điện tử nhằm phục vụ các mục đích trước đó.

Gửi lệnh điều khiển lên MQTT:

Sau khi đã tạo các lệnh điều khiển, thông tin của người chơi khối Điều khiển sẽ gửi (publish) chúng lên Broker MQTT vào các topic tương ứng để các thành phần khác trong hệ thống có thể nhận và thực hiện theo lệnh. Các lệnh điều khiển này sẽ được gửi tới Khối Giao tiếp và các thành phần khác có thể tương tác với hệ thống.

Ví dụ dưới đây là thông báo của mosquitto Mqtt về việc publish một message có độ dài 8bytes lên topic “game66/box/sound”, IP: 127.0.0.1

```
1690233371: Client auto-D51538E2-BFF5-8E3D-D278-A21F420FA726 disconnected.
1690233376: New client connected from 127.0.0.1:63910 as auto-AB31FC90-E056-6516-635F-19EEFCCB1957 (p2, c1, k60).
1690233376: No will message specified.
1690233376: Sending CONNACK to auto-AB31FC90-E056-6516-635F-19EEFCCB1957 (0, 0)
1690233376: Received PUBLISH from auto-AB31FC90-E056-6516-635F-19EEFCCB1957 (d0, q0, r0, m0, 'game66/box/sound', ... (8 bytes))
1690233376: Received DISCONNECT from auto-AB31FC90-E056-6516-635F-19EEFCCB1957
1690233376: Client auto-AB31FC90-E056-6516-635F-19EEFCCB1957 disconnected.
```

Hình 3.3 Mosquitto publish

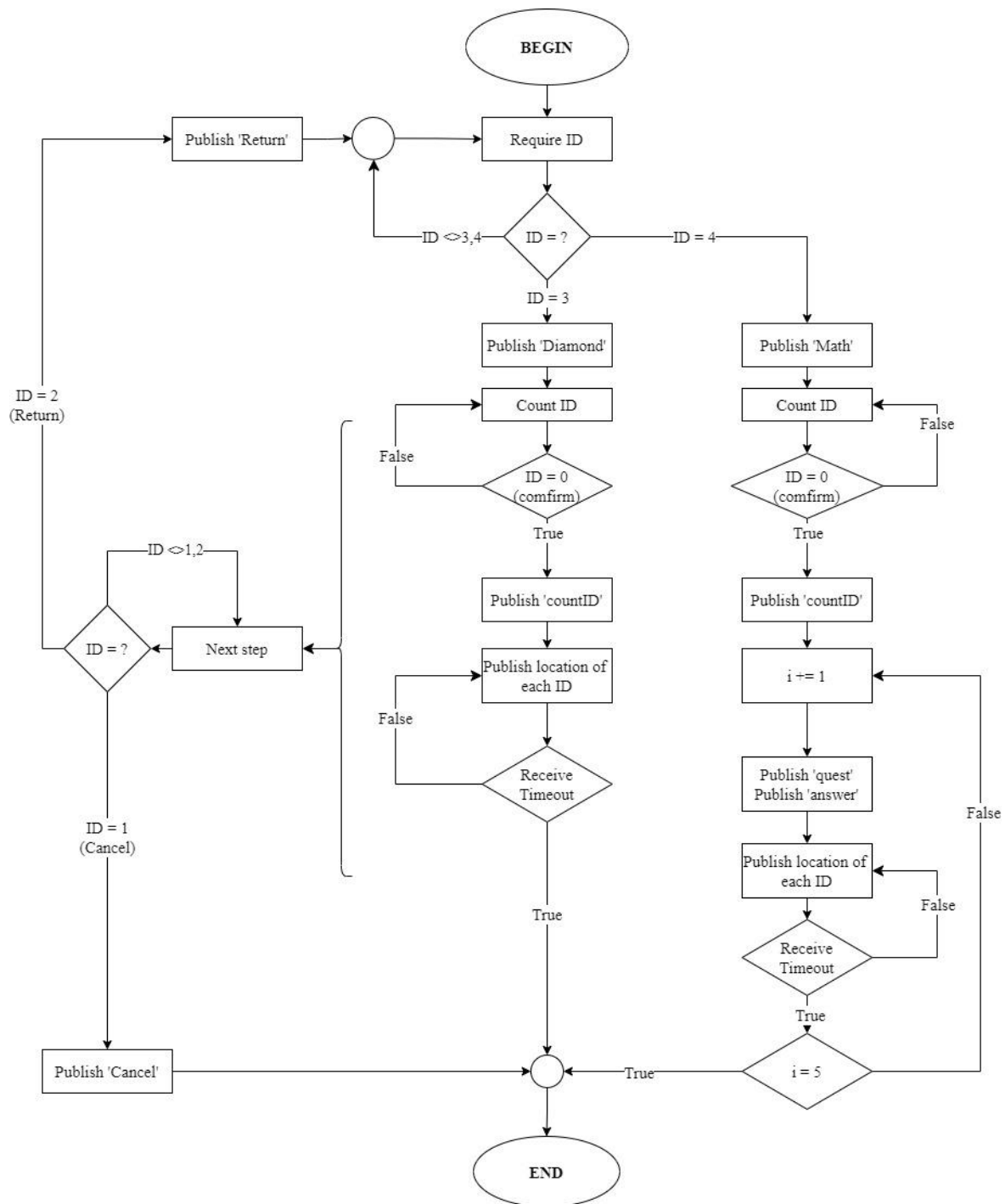
Khối Điều khiển gửi message, khối Hiển thị sẽ lắng nghe và thực hiện theo yêu cầu để trò chơi trở nên mượt mà hơn.

Nhận thông tin từ khối khác qua khối Giao tiếp

Khi khối Điều khiển subscribe vào một topic, nó sẽ lắng nghe và chờ đợi các thông điệp (messages) được gửi tới từ khối khác trên cùng topic đó. Những thông điệp này có thể chứa thông tin cần thiết để điều khiển hoạt động của khối Điều khiển hoặc cung cấp các dữ liệu hữu ích cho việc xử lý của nó

Tóm lại, khối Điều khiển chịu trách nhiệm nhận diện và trích xuất thông tin từ mã ArUco, sau đó gửi thông tin này lên MQTT. Dựa trên thông tin nhận diện, Khối Điều khiển tạo ra các lệnh điều khiển và gửi chúng lên MQTT để điều khiển và tương tác với các thành phần khác trong hệ thống. Quá trình này đảm bảo việc điều khiển và hiển thị trong trò chơi diễn ra một cách chính xác và linh hoạt.

Mô tả chi tiết về hoạt động của khối điều khiển được minh họa bằng lưu đồ thuật toán dưới đây:



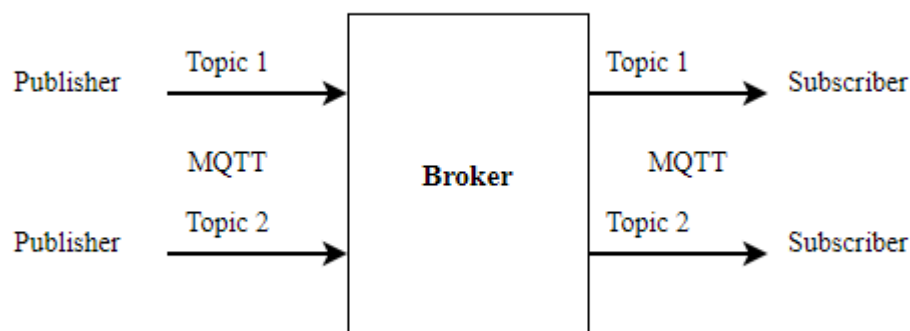
Hình 3.4 Lưu đồ thuật toán khối Điều khiển

Luồng xử lý và kết quả của quá trình xử lý sẽ được mô tả chi tiết ở Chương 4.

3.3.2 Khối Giao tiếp

Khối Giao tiếp là một phần quan trọng trong hệ thống trò chơi, có nhiệm vụ tạo ra và quản lý các giao tiếp giữa các khối khác nhau. Dưới đây là mô tả chi tiết về Khối Giao tiếp:

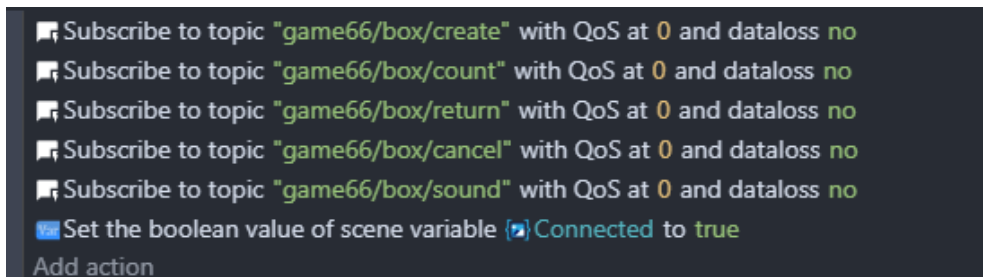
Khối Giao tiếp trong hệ thống trò chơi sử dụng giao thức MQTT để thiết lập kết nối và tạo các topic (chủ đề) cho các khối khác như Khối Điều khiển và Khối Hiển thị. Điều này cho phép các khối khác có thể subscribe (đăng ký) vào các topic này và nhận thông tin từ Khối Giao tiếp.



Hình 3.5 MQTT broker

Khi Khối Giao tiếp nhận được các message lệnh từ Khối Điều khiển thông qua MQTT, nó sẽ tiến hành xử lý để hiểu nội dung của lệnh và thực hiện các hành động tương ứng. Sau đó, Khối Giao tiếp gửi các lệnh đã được yêu cầu tới Khối Hiển thị qua giao thức MQTT. Các lệnh này có thể liên quan đến việc thay đổi trạng thái trò chơi, tọa độ hiển thị các vật thể, thay đổi trạng thái người chơi, hoặc cung cấp thông tin về vị trí và hành động của vật thể để Khối Hiển thị có thể hiển thị chúng trên màn hình. Để nhận được thông tin từ khối Điều khiển, khối Hiển thị phải subscribe vào topic tương ứng để nhận đúng thông tin.

Để đảm bảo việc nhận đúng thông tin từ Khối Điều khiển, Khối Hiển thị phải subscribe vào các topic tương ứng với thông tin mà nó cần nhận.



Hình 3.6 Gdevelop5 subscribe để lắng nghe từ topic MQTT

MQTT hỗ trợ ba mức độ chất lượng dịch vụ QoS (Quality of Service) để đảm bảo giao tiếp tin cậy trong mạng như đã trình bày ở chương trước. Ở hệ thống trò chơi này, QoS = 0 được lựa chọn. Một trong những lợi ích chính của việc sử dụng QoS = 0 là giảm tải cho hệ thống. Khi Khối Giao tiếp gửi các lệnh với QoS = 0, không cần phải đợi và xác nhận từ Khối Hiển thị để biết rằng lệnh đã được nhận. Điều này giúp giảm thiểu độ trễ và tải cho hệ thống, đồng thời cải thiện hiệu suất và thời gian đáp ứng của trò chơi. Việc sử dụng QoS = 0 cũng đơn giản hóa quá trình giao tiếp giữa Khối Giao tiếp và Khối Hiển thị. Vì không có xác nhận hoặc tái gửi các lệnh, quá trình truyền tải thông điệp trở nên đơn giản và hiệu quả hơn. Điều này đặc biệt hữu ích trong các trò chơi yêu cầu tương tác nhanh và thời gian thực. Nhược điểm của việc sử dụng QoS = 0 không đảm bảo tính tin cậy trong việc truyền tải dữ liệu tuy nhiên điều này có thể chấp nhận được. Các trò chơi thường có yêu cầu tương tác nhanh và thời gian đáp ứng ngay lập tức, và việc sử dụng QoS = 0 giúp đảm bảo trải nghiệm chơi game mượt mà và không bị gián đoạn.

3.3.3 Khối Hiển thị

Khối Hiển thị trong hệ thống là giao diện trò chơi được triển khai bằng GDevelop5 - một công cụ phát triển trò chơi mã nguồn mở dựa trên hình ảnh. Giao diện này có nhiệm vụ hiển thị các yếu tố trò chơi và tương tác với người chơi thông qua các hành động và hiệu ứng đồ họa.

Dưới đây là mô tả chi tiết về Khối Hiển thị:

Hiển thị hình ảnh và đồ họa: Khối Hiển thị sử dụng GDevelop5 để hiển thị các yếu tố trò chơi như người chơi, nhân vật, đối tượng trò chơi, môi trường, và các hiệu ứng đồ họa khác. GDevelop5 cho phép thiết kế và vẽ các hình ảnh tùy chỉnh, hoặc sử dụng các hình ảnh có sẵn để tạo nên các cảnh quan và trải nghiệm trực quan cho người chơi.

Tương tác với người chơi: Khối Hiển thị cung cấp các chức năng tương tác cho người chơi. Điều này bao gồm việc xử lý sự kiện khi người chơi nhấn các phím, di chuyển chuột hoặc tương tác với màn hình cảm ứng. GDevelop5 cho phép định nghĩa các hành động tương ứng với từng sự kiện, cho phép người chơi tương tác và tham gia vào trò chơi một cách tự nhiên và dễ dàng.

Hiển thị thông tin từ Khối Giao tiếp: Khối Hiển thị nhận thông tin từ khối Giao tiếp thông qua giao thức MQTT. Thông tin này bao gồm các lệnh điều khiển và thông tin về trạng thái của các yếu tố trong trò chơi như đã được mô tả ở trên. Khối Hiển thị sẽ xử lý và hiển thị các yếu tố trò chơi một cách đồng bộ với thông tin nhận được, giúp tạo ra một trải nghiệm trò chơi mượt mà và liền mạch cho người chơi.

Tạo hiệu ứng và âm thanh: Khối Hiển thị cũng có thể tạo hiệu ứng đồ họa và âm thanh để làm cho trò chơi sống động và hấp dẫn hơn. GDevelop5 hỗ trợ tạo và thêm các hiệu ứng đặc biệt, âm thanh nền và âm thanh tương tác để tăng cường trải nghiệm trò chơi của người chơi.

Ghi điểm và lưu trạng thái trò chơi: Khối Hiển thị cũng có khả năng ghi điểm và lưu trạng thái trò chơi của người chơi. Điều này cho phép người chơi lưu lại kết quả và tiến độ của họ trong trò chơi và chơi tiếp từ nơi họ đã dừng lại trong lần chơi tiếp theo.

Khối Hiển thị là một phần quan trọng trong hệ thống, giúp tạo ra giao diện trực quan và tương tác cho người chơi, đồng thời hiển thị các yếu tố trò chơi và hiệu ứng đồ họa để làm cho trò chơi hấp dẫn và thú vị.



Hình 3.7 Thiết kế trò chơi trên Gdevelop5

Nhiệm vụ của khối Hiển thị trong mô hình trò chơi thực tại tăng cường:

- Hiển thị các yếu tố trò chơi: Khối Hiển thị đảm nhận vai trò hiển thị các yếu tố quan trọng trong trò chơi như nhân vật, đối tượng, môi trường chơi và các yếu tố khác lên màn hình. Nhờ vào khối này, người chơi có thể thấy và tương tác với các thành phần trong trò chơi một cách trực quan và chân thực.
- Kết nối và nhận thông tin từ khối Điều khiển: Khối Hiển thị subscribe vào các topic mà khối Điều khiển gửi dữ liệu qua MQTT. Sau đó, nó xử lý và hiển thị các thông tin nhận được theo yêu cầu của khối Điều khiển. Điều này đảm bảo rằng người chơi nhận được các thông tin chính xác và phản hồi một cách thích hợp.
- Hiệu ứng đồ họa, âm thanh: Khối Hiển thị có khả năng tạo ra các hiệu ứng đồ họa và âm thanh để làm cho trò chơi thêm hấp dẫn và thú vị. Các hiệu ứng này bao gồm chuyển động mượt mà, ánh sáng, âm thanh, và các hiệu ứng đặc biệt khác để nâng cao trải nghiệm của người chơi.
- Tương tác người chơi mượt mà với thời gian thực: Khối Hiển thị đảm bảo thông tin nhận được từ khối Điều khiển được hiển thị một cách chính xác và mượt mà. Nhờ đó, người chơi có trải nghiệm tương tác thực thụ và đáp ứng trong thời gian thực, tạo cảm giác thực tế và hứng thú khi chơi trò chơi.
- Ghi điểm và lưu trạng thái trò chơi: Sau khi người chơi kết thúc trò chơi, khối Hiển thị tính và ghi điểm của người chơi.

CHƯƠNG 4. KẾT QUẢ MÔ PHỎNG VÀ KẾT LUẬN

4.1 Kịch bản trò chơi

4.1.1 Yêu cầu về thiết bị

Trò chơi thực tại tăng cường yêu cầu các thiết bị sau:

- Thẻ tag ArUco 4X4: Số lượng 21 thẻ in kích thước 9x9cm nền trắng bao gồm 6 thẻ điều kiện có ID từ 0 đến 5 và 15 thẻ ID từ 6 đến 20 để định danh người chơi. Các thẻ được gắn trên mũ hoặc bất kì một vật dụng gì để camera top-down có thể nhận diện được mà không bị che khuất.
- Một Camera top-down độ phân giải tối thiểu 720p (1280x720 pixel), tốc độ khung hình 30FPS: Điều này cho phép hiển thị đủ chi tiết về nhân vật, đối tượng và môi trường trong trò chơi mà không làm mờ hay làm mất đi trải nghiệm người chơi.
- Một màn hình chiếu lớn, đủ để người chơi quan sát từ xa.
- Một máy tính (PC/Laptop) để chạy chương trình điều khiển trò chơi.

4.1.2 Kịch bản trò chơi chi tiết

Chuẩn bị: Người quản lý (có thể là giáo viên, bố mẹ, hoặc người thân) sẽ khởi động hệ thống thông qua máy tính. Mỗi người chơi sẽ được phát một chiếc mũ có gắn mã tag ArUco với giá trị ID khác nhau. Người chơi di chuyển vào khu vực chơi đã được quy định sẵn, đảm bảo nằm trong khung hình thu nhận được từ camera.

Các thẻ lệnh điều khiển thực hiện các nhiệm vụ như: chọn trò chơi, bắt đầu, kết thúc, quay lại,... sẽ được người quản lý sử dụng để điều hướng trò chơi.

Hệ thống trò chơi thực tại tăng cường dành cho trẻ em, ứng dụng trong giáo dục sẽ bao gồm 2 trò chơi chính: “Đi tìm kim cương” (Diamonds Game) và “Trò chơi toán học” (Easy Math). Khi bắt đầu, người quản lý sẽ đưa thẻ để chọn một trong hai trò chơi, sau đó xác nhận số người chơi để hiển thị lên màn hình máy chiếu, mỗi người chơi sẽ được hiển thị thành một ngôi sao với các màu khác nhau.

Đối với trò chơi “Đi tìm kim cương”, trên màn hình ngẫu nhiên kim cương và bom. Nhiệm vụ của người chơi là di chuyển để chạm vào nhiều kim cương nhất có thể đồng

thời tránh chạm phải bom. Mỗi khi thu thập được kim cương người chơi sẽ được cộng 5 điểm, khi va phải bom thì bị trừ 2 điểm. Khi trò chơi kết thúc, số điểm sẽ được thông báo trên màn hình. Trò chơi này có thể giúp trẻ em vừa hoạt động vật lý, vừa giải trí sau những giờ học và tương tác bạn bè một cách tích cực. Từ đó tạo hứng thú trong học tập và gắn kết bạn bè.

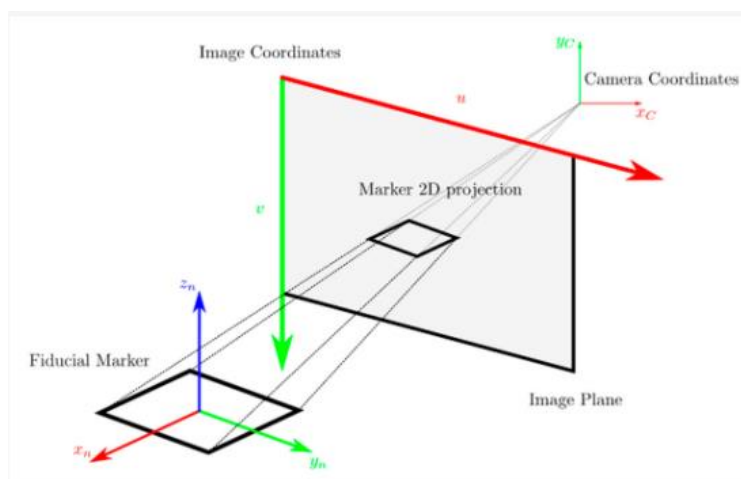
Đối với trò chơi “Trò chơi toán học”, trong mỗi lượt chơi màn hình sẽ được chia thành 9 ô theo ma trận 3x3 chứa 9 đáp án, trong đó có một đáp án đúng của một phép tính cơ bản (cộng, trừ, nhân, chia) ngẫu nhiên. Các con số sẽ biến mất sau 10 giây hiển thị. Người chơi có 30 giây để di chuyển đến ô chứa đáp án đúng. Trả lời đúng sẽ được cộng 1 điểm, sai không bị trừ điểm. Sau 5 lượt chơi, điểm của người chơi sẽ được hiển thị trên giao diện bảng điểm. Trò chơi tạo ra một luồng học tập mới, một mô hình học tập tích cực, giúp các em học tập kết hợp vận động và giải trí. Cụ thể quá trình và kết quả chạy trò chơi sẽ được mô tả ở phần tiếp theo: Kết quả mô phỏng

4.2 Kết quả mô phỏng

4.2.1 Kết quả nhận diện thẻ tag ArUco

Kịch bản kiểm thử: Thiết lập camera có độ phân giải 720p (1280x720 pixel) cách vị trí đặt thẻ tag là 3 mét. Thu dữ liệu đầu vào là 42 video có thời lượng 15 giây tương ứng với 21 mã tag ArUco có ID từ 0 đến 20, với 2 kích thước 9x9cm và 6x6cm.

Mô hình thu dữ liệu:



Hình 4.1 Mô hình thu dữ liệu kiểm thử

Tiêu chí đánh giá cho module này là độ chính xác trong quá trình nhận diện ID thẻ tag ArUco. Kết quả nhận diện đúng được chấp nhận, kết quả nhận diện sai được tính là lỗi, với frame không nhận diện được ID nào sẽ không được tính. Sau đây là kết quả đánh giá mô hình trên bộ dữ liệu trên:

Bảng 4.1 Kết quả đánh giá mô hình nhận diện ID thẻ tag

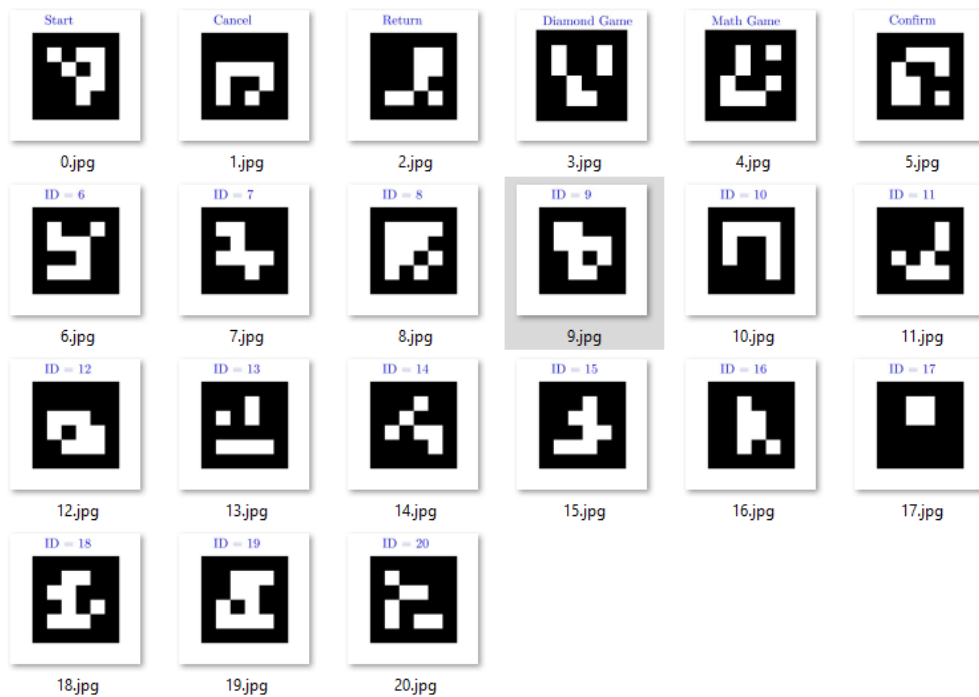
Kích thước thẻ	6x6cm	9x9cm
Số lượng video thử	21	21
Tỉ lệ nhận dạng	60.33%	79.77%
Độ chính xác	100%	99.99%

Theo bảng 4.1, tỉ lệ nhận dạng được tính theo tổng số frame nhận diện được đúng ID trên tổng số frame. Độ chính xác được tính là tổng số frame nhận diện được đúng ID trên tổng số frame phát hiện ra ID (không tính những frame không phát hiện ra ID nào).

Từ bảng kết quả, ta thấy độ chính xác khi nhận diện ID thẻ tag gần như là tuyệt đối, vì vậy việc ứng dụng mã tag ArUco cho mô hình trò chơi là hoàn toàn hợp lý.

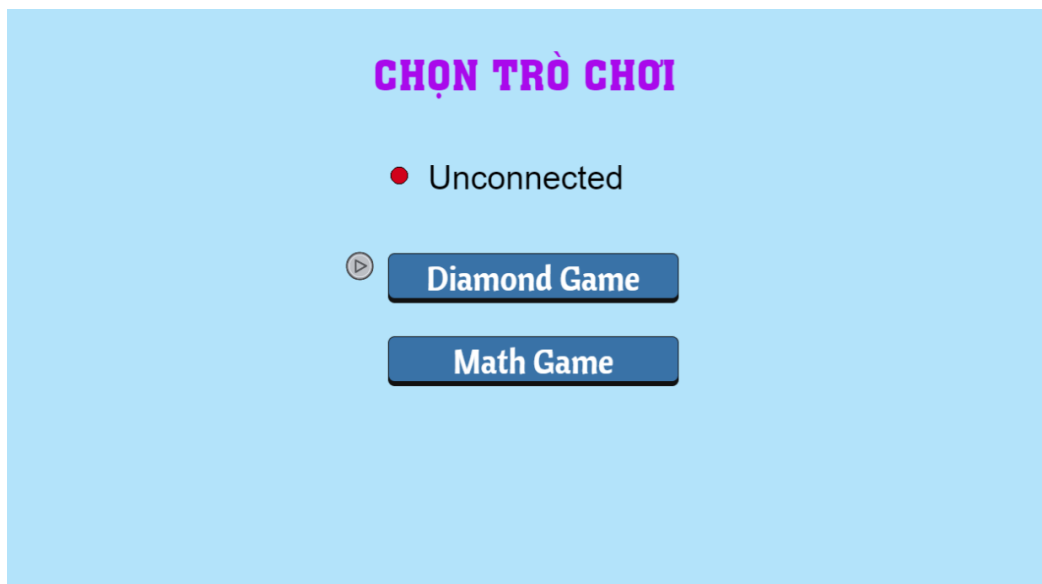
4.2.2 Chạy hoàn thiện kịch bản trò chơi

Bộ thẻ ArUco 4X4 bao gồm 6 thẻ lệnh ID từ 0 đến 5 và 15 thẻ người chơi có ID từ 6 đến 20 được in kích thước mã là 9x9cm như sau:

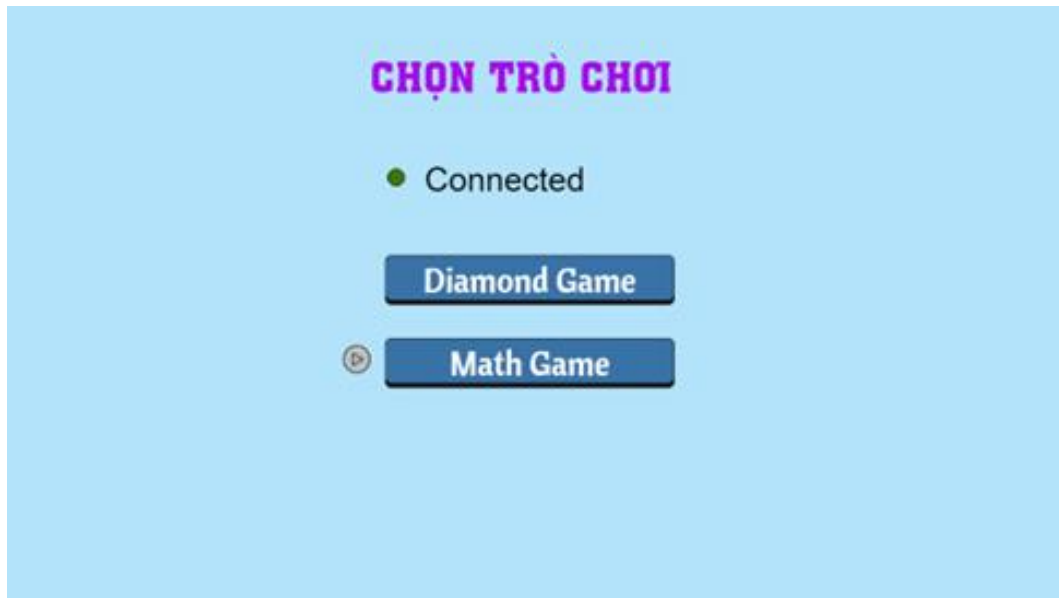


Hình 4.2 Bộ thẻ được sử dụng trong trò chơi

Trò chơi bắt đầu: Gdevelop tiến hành kết nối trò chơi với Broker và subscribe vào các topic đã được thiết kế trước để lắng nghe thông tin điều khiển từ chương trình điều khiển. Khi giao diện Gdevelop 5 kết nối được với Broker MQTT, màn hình sẽ hiển thị text: Connected nếu chưa được kết nối text: Unconnected



Hình 4.3 Giao diện chọn trò chơi khi chưa kết nối với Broker



Hình 4.4 Giao diện trò chơi khi đã kết nối với broker

```
1690662189: New client connected from 127.0.0.1:55923 as mqttjs_518136eb (p2, c1, k60).
1690662189: No will message specified.
1690662189: Sending CONNACK to mqttjs_518136eb (0, 0)
1690662189: Received SUBSCRIBE from mqttjs_518136eb
1690662189:   game66/box/start (QoS 0)
1690662189: mqttjs_518136eb 0 game66/box/start
1690662189: Sending SUBACK to mqttjs_518136eb
1690662189: Received SUBSCRIBE from mqttjs_518136eb
1690662189:   game66/box/scene (QoS 0)
1690662189: mqttjs_518136eb 0 game66/box/scene
1690662189: Sending SUBACK to mqttjs_518136eb
1690662189: Received SUBSCRIBE from mqttjs_518136eb
1690662189:   game66/box/sound (QoS 0)
1690662189: mqttjs_518136eb 0 game66/box/sound
1690662189: Sending SUBACK to mqttjs_518136eb
```

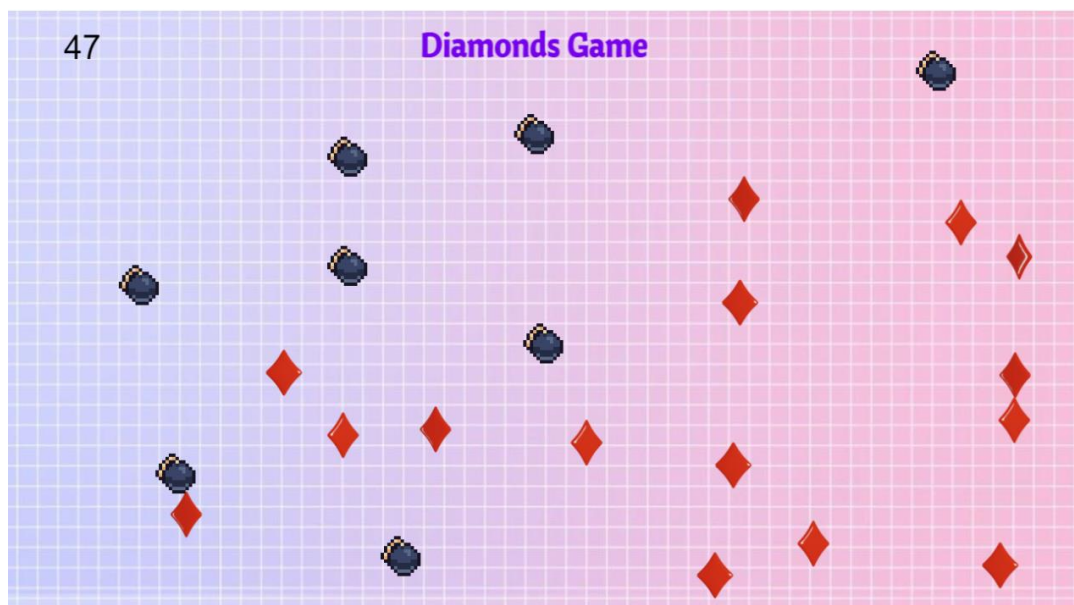
Hình 4.5 Gdevelop connected MQTT

Người quản lý đưa thẻ để chọn trò chơi hoặc bấm các button trên màn hình, trẻ em cũng có thể là người trực tiếp đưa thẻ để chọn trò chơi để vào khung hình của camera.

Khi người chơi đưa thẻ có ID = 3, trò chơi “Tìm kiếm kim cương” được mở. Khi người chơi đưa thẻ có ID = 4, trò chơi “Trò chơi toán học” được mở. Trong quá trình phát trò chơi, bất kì thời điểm nào người quản lý đưa thẻ “Return” (ID = 2) trò chơi sẽ được chuyển về ban đầu.

4.2.2.1 Trò chơi “Tìm kiếm kim cương”

Khi người chơi đưa thẻ có ID = 3, giao diện người chơi chuyển thành:



Hình 4.6 Trò chơi “Tìm kiếm kim cương”

Màn hình hiển ngẫu nhiên số lượng bom và kim cương nhất định. Thời gian đếm ngược được hiển thị ở góc trái màn hình. Người chơi sẽ có 3 phút để di chuyển và thu thập kim cương.

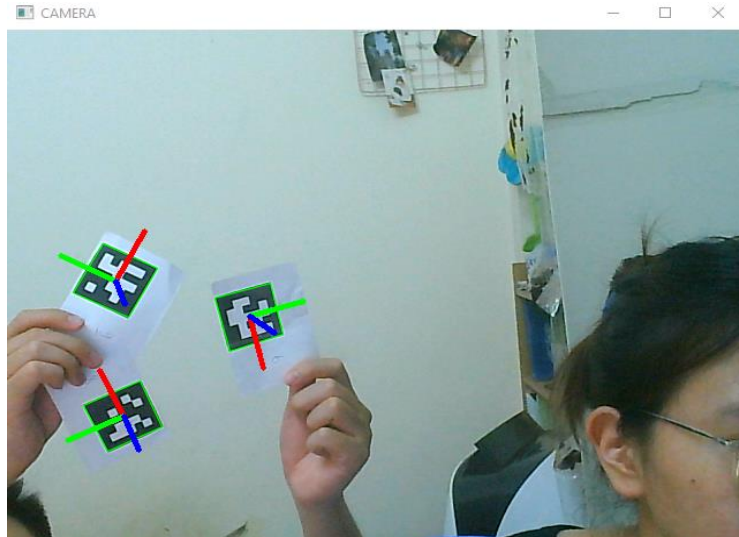
Hệ thống điều khiển trò chơi sẽ thông báo số lượng người chơi được nhận diện từ camera, người quản lý sẽ chịu trách nhiệm xác nhận số lượng người chơi sau đó hiển thị lên màn hình:

Số lượng người chơi là 3: 3 ngôi sao với 3 màu khác nhau



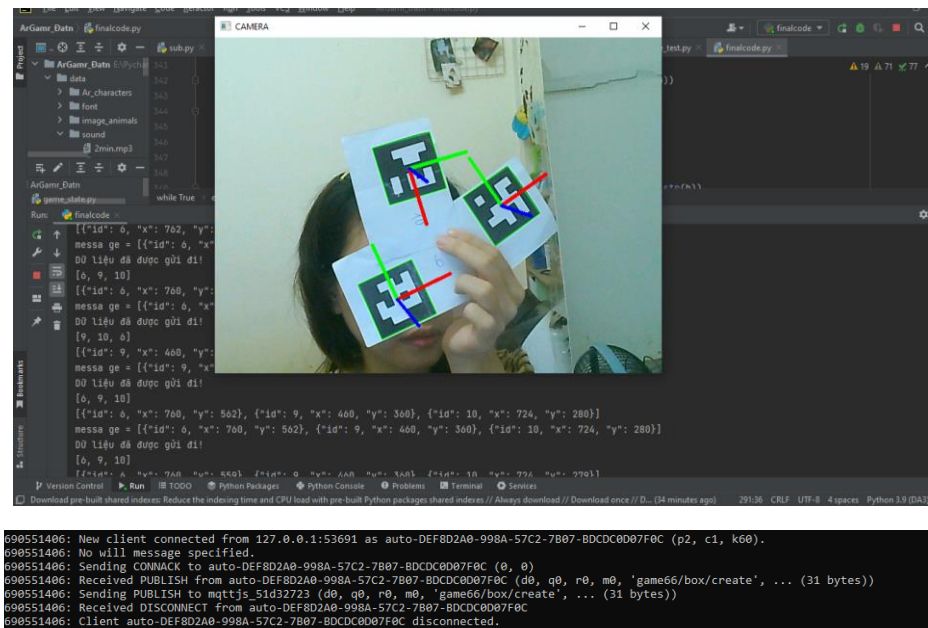
Hình 4.7 Hình ảnh người chơi xuất hiện trên màn hình (3 ngôi sao)

Khi người chơi di chuyển trong khuôn khổ khung hình camera, vị trí của người chơi sẽ được nhận diện và gửi liên tục để phân hiển thị người chơi lên màn hình không vị gián đoạn. Nếu người chơi biến mất khỏi khung hình hoặc thẻ bị che không nhận diện được, trò chơi sẽ lưu lại vị trí cuối cùng được gửi:



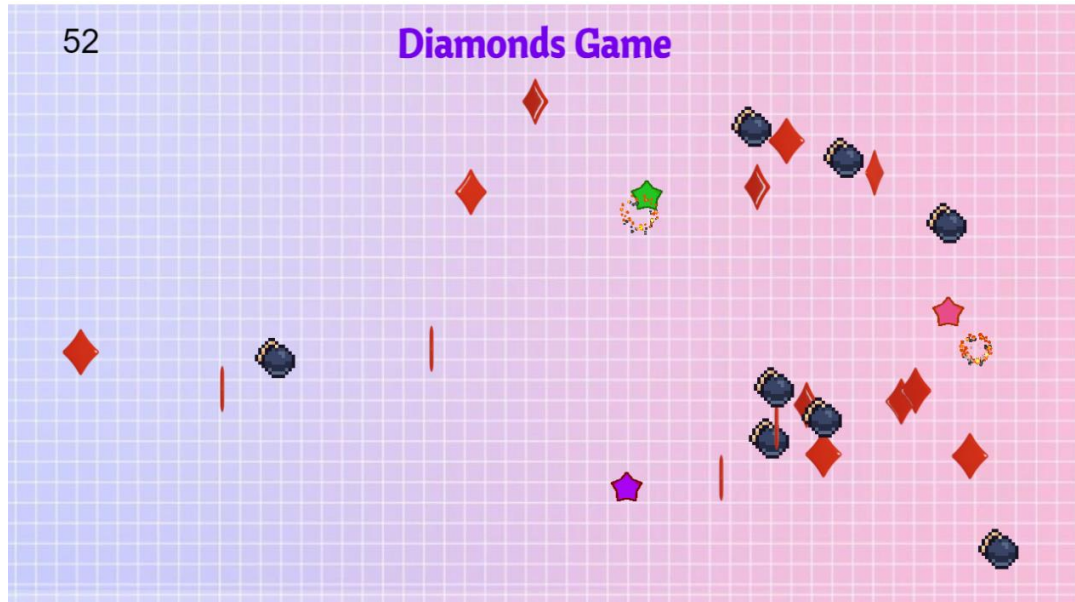
Hình 4.8 Hình ảnh từ khung hình camera

Vị trí của người chơi (các ngôi sao) trên màn hình sẽ được thay đổi liên tục khi người chơi di chuyển để người chơi biết mình đang ở vị trí nào trên màn hình. Từ đó di chuyển đến vị trí có kim cương và tránh bom.



Hình 4.9 Vị trí người chơi được gửi liên tục

Nếu người chơi va chạm với kim cương: Âm thanh va chạm sẽ phát, kim cương đó sẽ biến mất. Nếu người chơi va chạm với bom: Bom sẽ nổ, âm thanh nổ được phát khi đó bom sẽ được di chuyển sang vị trí khác. Số điểm của người chơi được cộng 5 khi người chơi va chạm với kim cương và trừ 1 khi người chơi va chạm với bom.



Hình 4.10 Người chơi di chuyển và chạm với bom

Trò chơi kết thúc và hiển thị bảng điểm sau 3 phút chơi hoặc khi người quản lý đưa thẻ “Cancel” (ID = 1) hoặc số lượng kim cương ở trên màn hình bằng 0.

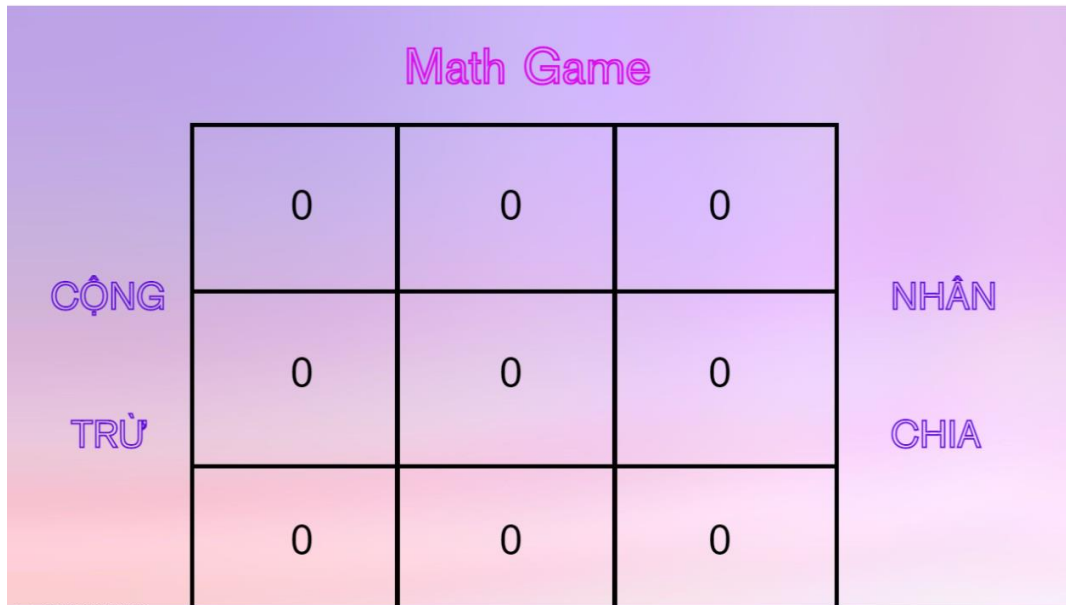
BẢNG ĐIỂM	
ID: 9	SCORE: 81
ID: 6	SCORE: 77
ID: 8	SCORE: 25

Hình 4.11 Bảng điểm người chơi

4.2.2.2 Trò chơi Toán học

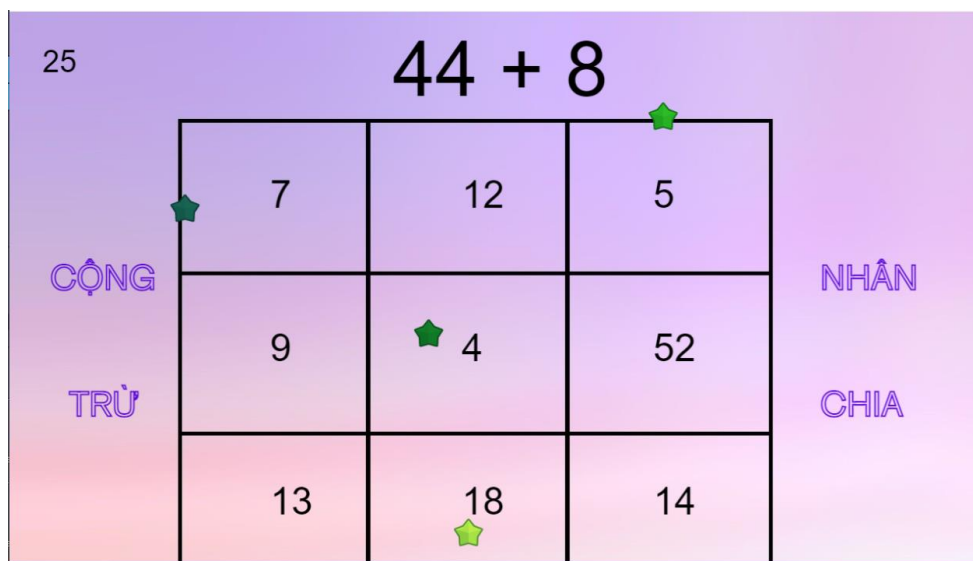
Khi người chơi đưa thẻ có ID = 4, trò chơi “Tìm kiếm kim cương” được mở.

Hệ thống điều khiển trò chơi sẽ thông báo số lượng người chơi được nhận diện từ camera, người quản lý sẽ chịu trách nhiệm xác nhận số lượng người chơi sau đó hiển thị lên màn hình:



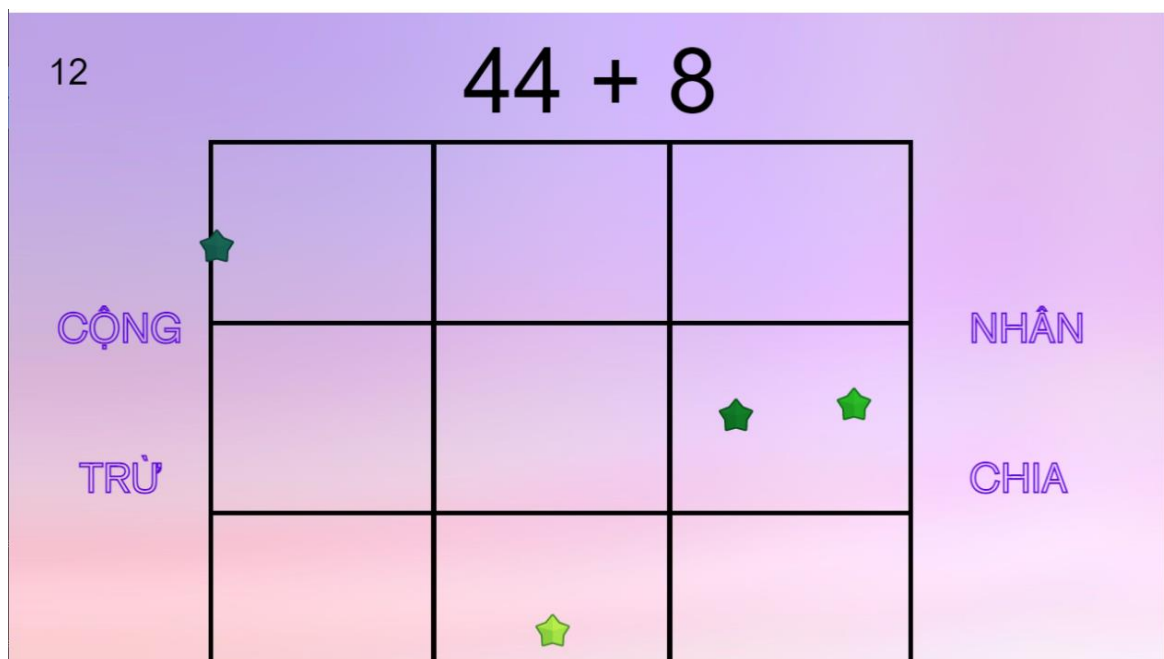
Hình 4.12 Giao diện trò chơi “Trò chơi toán học”

Sau khi số lượng người chơi được xác nhận, trò chơi sẽ nhận được một phép toán cơ bản ngẫu nhiên và một kết quả đúng sẽ được hiển thị ở một ô bất kì trong 9 ô vuông trên màn hình.



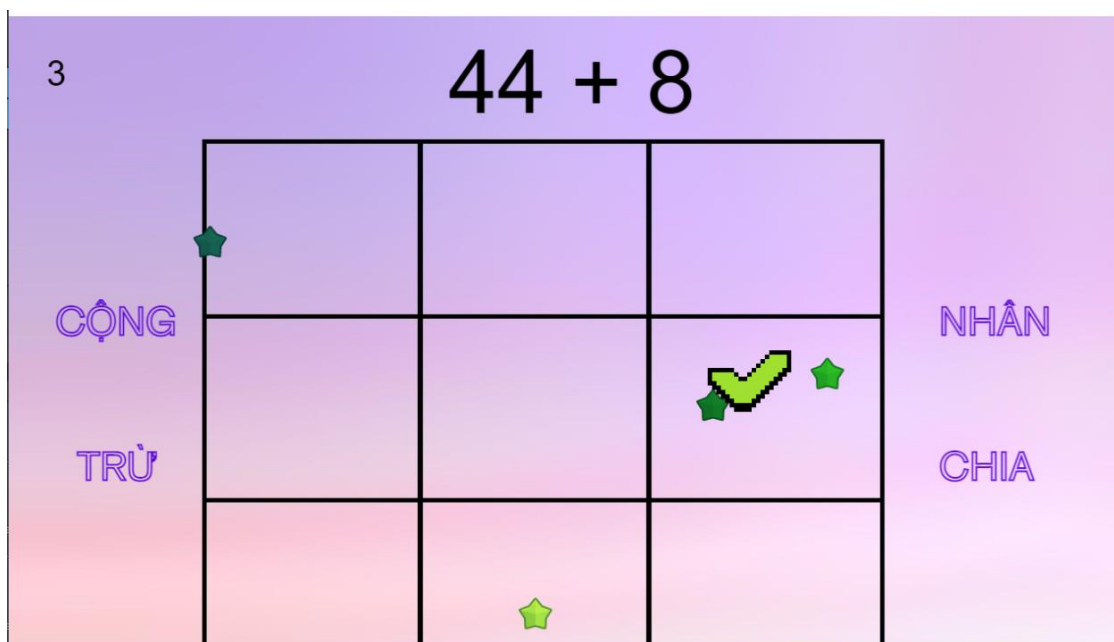
Hình 4.13 Hiển thị câu hỏi và các đáp án

Người chơi có 30 giây để di chuyển vào ô có đáp án mình cho là đúng. Sau 10 giây các con số trên màn hình sẽ biến mất, việc của người chơi là phải nhớ vị trí của đáp án ở trên màn hình.



Hình 4.14 Đáp án biến mất sau 10 giây

Trong 3 giây cuối, vị trí đáp án đúng sẽ hiển thị dấu tích.



Hình 4.15 Tích vào ô chứa đáp án đúng

Sau 5 lượt chơi hoặc người quản lý đưa thẻ Cancel, trò chơi kết thúc và hiện thị bảng điểm tương tự như bảng điểm trò chơi “Tìm kiếm kim cương”

BẢNG ĐIỂM	
ID: 8	SCORE: 10
ID: 9	SCORE: 10
ID: 6	SCORE: 10
ID: 10	SCORE: 10

Hình 4.16 Bảng điểm trò chơi toán

Sau khi trò chơi kết thúc, người quản lý có thể đưa thẻ “Return” (ID = 2) để trở lại trạng thái game đầu tiên.

4.3 Kết luận

4.3.1 Kết quả đạt được

Qua quá trình hoàn thành đề tài “Nghiên cứu, thiết kế nền tảng trò chơi thực tại tăng cường dành cho trẻ em, ứng dụng trong giáo dục”, em đã hoàn thành được những mục tiêu sau:

- Nắm được cách xây dựng một hệ thống game thực tại tăng cường, các kết nối Broker MQTT, thiết kế topic và ứng dụng giao thức đó trong quá trình xây dựng nền tảng game.
- Nắm được quá trình nhận diện mã tag ArUco, đánh giá độ chính xác nhận diện ID, và thực hiện ứng dụng mã ArUco tạo ra các kịch bản trò chơi thú vị.
- Thiết kế được nền tảng trò chơi, giao diện trò chơi trên nền tảng Gdevelop 5.
- Hoàn thành thiết kế các trò chơi giáo dục sáng tạo trong mô hình trò chơi. Các trò chơi được thiết kế nhằm kích thích hoạt động, tương tác xã hội và học tập cho trẻ em. Các hoạt động như tìm kiếm kim cương, tính toán và các trò chơi học tập khác sẽ được tích hợp trong mô hình trò chơi để đem lại trải nghiệm học tập tích cực cho trẻ em.

4.3.2 Đánh giá kết quả

Đánh giá hệ thống trò chơi thực tại tăng cường dành cho trẻ em:

Ưu điểm:

- Mô hình tương đối dễ thực hiện đối với người làm trò chơi.
- Trò chơi được thiết kế hoàn thành được những mục tiêu đề ra: Các hoạt động như tìm kiếm kim cương, tính toán và các trò chơi học tập khác sẽ được tích hợp trong mô hình trò chơi để đem lại trải nghiệm học tập tích cực cho trẻ em.
- Chi phí thấp: Các lớp học, trung tâm giáo dục hiện tại hầu hết đã có sẵn màn chiếu và máy tính, camera. Bộ thẻ ArUco có thể in ra dễ dàng.
- Độ chính xác trong quá trình nhận diện cao: Trò chơi không bị gián đoạn làm mất cảm giác của người chơi

Nhược điểm:

- Hệ thống chưa được đóng gói: tương đối cồng kềnh trong quá trình sử dụng
- Chưa có quá trình đánh giá từ trải nghiệm của trẻ em.

Dưới đây là bảng đánh giá mức độ hoàn thành cho những công việc em đã đặt ra, được nêu ra ở **Chương 1**:

Bảng 4.2 Đánh giá mức độ hoàn thành dự kiến

Thứ tự	Phần việc	Nội dung công việc	Đánh giá
1	Tìm hiểu lý thuyết	Tìm hiểu ArUco Marker và thư viện Paho Python	Hoàn thành
		Giao thức mạng MQTT (Pub/Sub, mosquitto,..)	Hoàn thành
2	Kết nối MQTT broker	Connect Broker MQTT qua API	Hoàn thành
		Pub/Sub Topic lên server, xử lý message gửi và nhận trên MQTT	Hoàn thành
3	Xử lý ảnh	Nhận diện ArUco Marker bằng thư viện OpenCV	Hoàn thành

4	Thiết kế kịch bản và chạy thử kịch bản trên python	Thiết kế kịch bản trò chơi (Nội dung trò chơi, các bước chạy chương trình)	Hoàn thành
		Viết chương trình theo luồng, gameState bằng ngôn ngữ Python	Hoàn thành
5	Thiết kế trò chơi trên GDevelop	Thiết kế giao diện trên Gdevelop5 xử lý các sự kiện game	Hoàn thành
6	Demo	Xây dựng hệ thống game demo, ứng dụng cho 1 tập thẻ ArUco	Hoàn thành
7	Hoàn thiện	Thiết kế app đóng gói sản phẩm	Chưa hoàn thành
		Thử nghiệm với người chơi thật trên diện rộng.	

4.4 Hướng phát triển

Trong thời gian tới, em sẽ tiếp tục phát triển và nâng cấp hệ thống trò chơi thực tại tăng cường cho trẻ em với những mục tiêu đặt ra:

- Hoàn thiện hệ thống game, thiết kế nhiều kịch bản game hấp dẫn hơn để tăng hứng thú cho người chơi.
- Hiệu chỉnh góc cho camera.
- Thử nghiệm trò chơi trên trẻ em, để có đánh giá khách quan về mô hình.
- Đóng gói sản phẩm thành app để người quản lý dễ dàng sử dụng.

TÀI LIỆU THAM KHẢO

- [1] Aufar Zakiev, Ksenia Shabalina, Tatyana Tsoy and Evgeni Magid: “*Pilot Virtual Experiments on ArUco and ArTag Systems Comparison for Fiducial Marker Rotation Resistance*” - January 2020
- [2] OpenCV Document – Modul ArUco: https://docs.opencv.org/4.x/d5/dae/tutorial_aruco_detection.html - truy cập lần cuối ngày 30/7/2023.
- [3] Andrea Botta and Giuseppe Quaglia: “*Performance Analysis of Low-Cost Tracking System for Mobile Robots*”
- [4] Luan C. Klein, João Braun, João Mendes, Vítor H. Pinto, Felipe N. Martins 8, Andre Schneider de Oliveira, Heinrich Wörtche, Paulo Costa and José Lima: “*A Machine Learning Approach to Robot Localization Using Fiducial Markers in RobotAtFactory 4.0 Competition*”
- [5] Zakiev, Aufar; Tsoy, Tatyana; Shabalina, Ksenia; Magid, Evgeni; Saha, Subir Kumar (2020): “*Virtual Experiments on ArUco and AprilTag Systems Comparison for Fiducial Marker Rotation Resistance under Noisy Sensory Data*”
- [6] MQTT: The Standard for IoT Messaging - https://medium.com/@calle_4729/using-mathematica-to-detect-aruco-markers-197410223f62 - truy cập lần cuối ngày 30/7/2023.
- [7] MQTT: The Standard for IoT Messaging - <https://mqtt.org/> - truy cập lần cuối ngày 20/6/2023.
- [8] How to Use The Paho MQTT Python Client for Beginners - <http://www.steves-internet-guide.com/into-mqtt-python-client/> - truy cập lần cuối ngày 30/7/2023.
- [9] GDevelop document: <https://wiki.gdevelop.io/gdevelop5/extensions/> - truy cập lần cuối ngày 2/8/2023.