

BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HỒ CHÍ MINH  
VIỆN KỸ THUẬT HUTECH



**BÁO CÁO**  
**ĐỒ ÁN IOT**

**Hệ thống giám sát môi trường lớp học kết hợp  
điểm danh ứng dụng YOLO và công nghệ LoRa**

**Sinh viên thực hiện:**

- |                       |                  |
|-----------------------|------------------|
| 1. Nguyễn Thanh Nhân  | MSSV: 2286301149 |
| 2. Hồ Đức An          | MSSV: 2286300050 |
| 3. Nguyễn Vũ Huy Khôi | MSSV: 2286300027 |

**Ngành:** Robot & Trí tuệ nhân tạo

**Lớp:** 22DRTA1

**Giảng viên hướng dẫn:** ThS. Ngô An Thuyên

TP. Hồ Chí Minh, 2025

# Mục lục

<b>1</b>	<b>Giới thiệu</b>	<b>4</b>
<b>2</b>	<b>Cơ sở lý thuyết</b>	<b>4</b>
2.1	Động lực nghiên cứu . . . . .	4
2.2	Đặt vấn đề . . . . .	4
2.3	Đóng góp chính . . . . .	4
2.4	Các nghiên cứu liên quan . . . . .	5
<b>3</b>	<b>Thiết kế hệ thống</b>	<b>5</b>
3.1	Hệ thống tổng thể . . . . .	5
3.1.1	Phân hệ Edge node (Thu thập và xử lý) . . . . .	5
3.1.2	Phân hệ Gateway & Cloud (Trung chuyển & Lưu trữ) . . . . .	6
3.1.3	Phân hệ Quản lý (Application) . . . . .	6
3.2	Thiết kế phần cứng . . . . .	7
3.2.1	Cấu hình Edge Node . . . . .	7
3.2.2	Thiết kế LoRa Gateway . . . . .	8
3.3	Thiết kế phần mềm . . . . .	8
3.3.1	Giải thuật xử lý AI tại biên (Edge AI – Jetson Nano) . . . . .	8
3.3.2	Giải thuật điều khiển và truyền thông tại Node (Raspberry Pi Pico) . . . . .	9
<b>4</b>	<b>Kết quả thực nghiệm</b>	<b>11</b>
4.0.1	Hiệu năng thực tế trên Jetson Nano . . . . .	11
4.1	Kết quả đánh giá truyền thông LoRa . . . . .	11
<b>5</b>	<b>Kết luận</b>	<b>13</b>

# LỜI CẢM ƠN

Trong quá trình thực hiện đồ án này, nhóm chúng em đã nhận được sự hướng dẫn, hỗ trợ và giúp đỡ tận tình từ nhiều cá nhân và tập thể.

Trước hết, chúng em xin bày tỏ lòng biết ơn sâu sắc đến giảng viên hướng dẫn **ThS. Ngô An Thuyên**, người đã trực tiếp hướng dẫn, định hướng và đóng góp nhiều ý kiến quý báu giúp nhóm hoàn thành đồ án một cách hiệu quả. Những góp ý và sự tận tâm của thầy là nền tảng quan trọng giúp chúng em củng cố kiến thức chuyên môn và nâng cao kỹ năng nghiên cứu, thực hành.

Chúng em cũng xin gửi lời cảm ơn đến quý thầy cô trong Khoa/Viện đã truyền đạt những kiến thức nền tảng và chuyên môn trong suốt quá trình học tập, tạo điều kiện thuận lợi để chúng em thực hiện đồ án này.

Cuối cùng, nhóm xin cảm ơn gia đình, bạn bè đã luôn động viên, hỗ trợ tinh thần và tạo điều kiện để chúng em có thể tập trung hoàn thành đồ án đúng tiến độ.

Mặc dù đã rất cố gắng, nhưng do thời gian và kiến thức còn hạn chế, đồ án không tránh khỏi những thiếu sót. Nhóm chúng em rất mong nhận được sự đóng góp và chỉ dẫn thêm từ quý thầy cô để đồ án được hoàn thiện hơn.

Chúng em xin chân thành cảm ơn.

# 1 Giới thiệu

Nhu cầu quản lý sinh viên tự động trong Smart Campus ngày càng tăng; các giải pháp dựa trên WiFi hoặc điểm danh thủ công gặp phải các vấn đề như tắc nghẽn mạng và tính riêng tư.

Báo cáo đề xuất hệ thống điểm danh kết hợp Edge-AI (NVIDIA Jetson Nano chạy YOLOv8 cho phát hiện khuôn mặt kết hợp FaceNet cho trích xuất đặc trưng) và truyền thông tầm xa năng lượng thấp bằng LoRa. Kiến trúc không đồng nhất (heterogeneous computing) được áp dụng: Jetson xử lý ảnh, Raspberry Pico thu/gửi dữ liệu cảm biến và chuyển tiếp qua LoRa.

Thực nghiệm cho thấy độ chính xác nhận diện đạt khoảng 9x% (tùy dataset), tần suất truyền tin ổn định qua 4 tầng bê tông với PDR > 90% dưới cấu hình TX power hợp lý.

Hệ thống hoạt động độc lập, giảm tải băng thông cho hạ tầng trung tâm và có khả năng triển khai diện rộng trong khuôn viên trường học.

## 2 Cơ sở lý thuyết

### 2.1 Động lực nghiên cứu

Trong kỷ nguyên IoT, mô hình Tòa nhà thông minh (Smart Building) đang chuyển dịch từ tự động hóa đơn thuần sang khả năng thích ứng (adaptability) dựa trên dữ liệu cư dân thực tế. Việc nắm bắt chính xác số lượng và danh tính sinh viên trong thời gian thực là chìa khóa để tối ưu hóa năng lượng và quản lý cơ sở vật chất hiệu quả.[1]

### 2.2 Đặt vấn đề

Tuy nhiên, việc thu thập dữ liệu trong các tòa nhà cao tầng gặp thách thức lớn về kết nối. Mạng Wifi thường tiêu tốn năng lượng cao và phạm vi phủ sóng hạn chế (<40m trong nhà). Công nghệ LoRa nổi lên như một giải pháp thay thế nhờ khả năng truyền xa, nhưng nghiên cứu của Xu et al. chỉ ra rằng tín hiệu LoRa trong môi trường bê tông cốt thép nhiều tầng (multi-floor) chịu sự suy hao lớn và biến động kênh truyền phức tạp, đòi hỏi thiết kế hệ thống phải được tối ưu hóa kỹ lưỡng.[2] Hơn nữa, băng thông thấp của LoRa không cho phép truyền tải dữ liệu hình ảnh/video. Các kiến trúc truyền thống dựa trên Cloud computing gặp trở ngại về độ trễ và tắc nghẽn băng thông khi quy mô cảm biến tăng lên. Theo Fraga-Lamas et al., kiến trúc tính toán sương mù/tại biên (Fog/Edge Computing) là giải pháp thiết yếu cho các khuôn viên thông minh (Smart Campus) để đảm bảo độ trễ thấp và giảm tải cho hạ tầng mạng.[3]

### 2.3 Đóng góp chính

Để giải quyết các thách thức trên, bài báo đề xuất một hệ thống điểm danh lai ghép (hybrid approach). Chúng tôi sử dụng Jetson Nano để xử lý AI tại biên, trích xuất thông tin điểm danh và chỉ gửi dữ liệu văn bản gói gọn qua mạng LoRa riêng biệt. Đặc biệt, nghiên cứu thực hiện so sánh thực nghiệm giữa mô hình YOLOv8 thuần và kiến trúc đường ống YOLOv8 kết hợp FaceNet. Kết quả nhằm chứng minh rằng kiến trúc hai giai đoạn (two-stage pipeline) mang lại khả năng mở rộng (scalability) và tính

thích ứng tốt hơn cho môi trường giáo dục thay đổi liên tục, khắc phục nhược điểm phải huấn luyện lại (re-training) của các mô hình truyền thống.

## 2.4 Các nghiên cứu liên quan

Bất chấp những tiến bộ đáng kể trong các hệ thống điểm danh thông minh đã thảo luận ở trên, vẫn tồn tại một khoảng trống quan trọng trong việc kết nối hiệu quả giữa nhận diện sinh trắc học độ chính xác cao với các giao thức truyền thông băng thông thấp, tầm xa trong môi trường tòa nhà nhiều tầng.

Các giải pháp dựa trên thị giác máy tính hiện có, chẳng hạn như nghiên cứu của Giau et al. (2024) , đã chứng minh độ chính xác cao khi triển khai các mô hình học sâu như YOLOv8 và FaceNet trên thiết bị biên (Jetson Nano). Tuy nhiên, các hệ thống này thường phụ thuộc vào hạ tầng mạng băng thông cao (ví dụ: Wi-Fi hoặc Ethernet) để truyền dữ liệu video hoặc đồng bộ hóa cơ sở dữ liệu lớn với đám mây, khiến chúng không phù hợp với những khu vực có kết nối internet không ổn định hoặc bị giới hạn băng thông nghiêm ngặt.[4]

Ngược lại, các nghiên cứu tập trung vào công nghệ LoRa trong tòa nhà, như Khurshid et al. (2025) và Liang et al. (2020) , đã chứng minh sự vượt trội về vùng phủ sóng và khả năng xuyên bê tông cốt thép so với Wi-Fi. Tuy nhiên, các triển khai này thường sử dụng các phương pháp định danh đơn giản (ví dụ: thẻ RFID) hoặc chỉ truyền dữ liệu môi trường dạng vô hướng. Chúng thiếu khả năng thực hiện xác thực sinh trắc học phức tạp, khiến hệ thống dễ bị điểm danh hộ ("buddy punching") và không đáp ứng được yêu cầu về "khả năng thích ứng"(adaptability) của các tòa nhà thông minh hiện đại như định nghĩa của Buckman et al. (2014) .[5] [6] [1]

Do đó, hiện nay vẫn thiếu các nghiên cứu về kiến trúc lai (hybrid architectures) tận dụng sức mạnh tính toán của điện toán biên không đồng nhất (Jetson Nano kết hợp với vi điều khiển) để xử lý dữ liệu hình ảnh phức tạp ngay tại chỗ và chỉ truyền các gói tin văn bản nhẹ, đã được tối ưu hóa qua LoRa. Nghiên cứu này nhằm mục đích lấp đầy khoảng trống đó bằng cách đề xuất một hệ thống Edge-AI dựa trên LoRa, đảm bảo cả khả năng nhận diện khuôn mặt chính xác cao và khả năng truyền thông tin cậy, năng lượng thấp trong các tòa nhà giáo dục nhiều tầng.

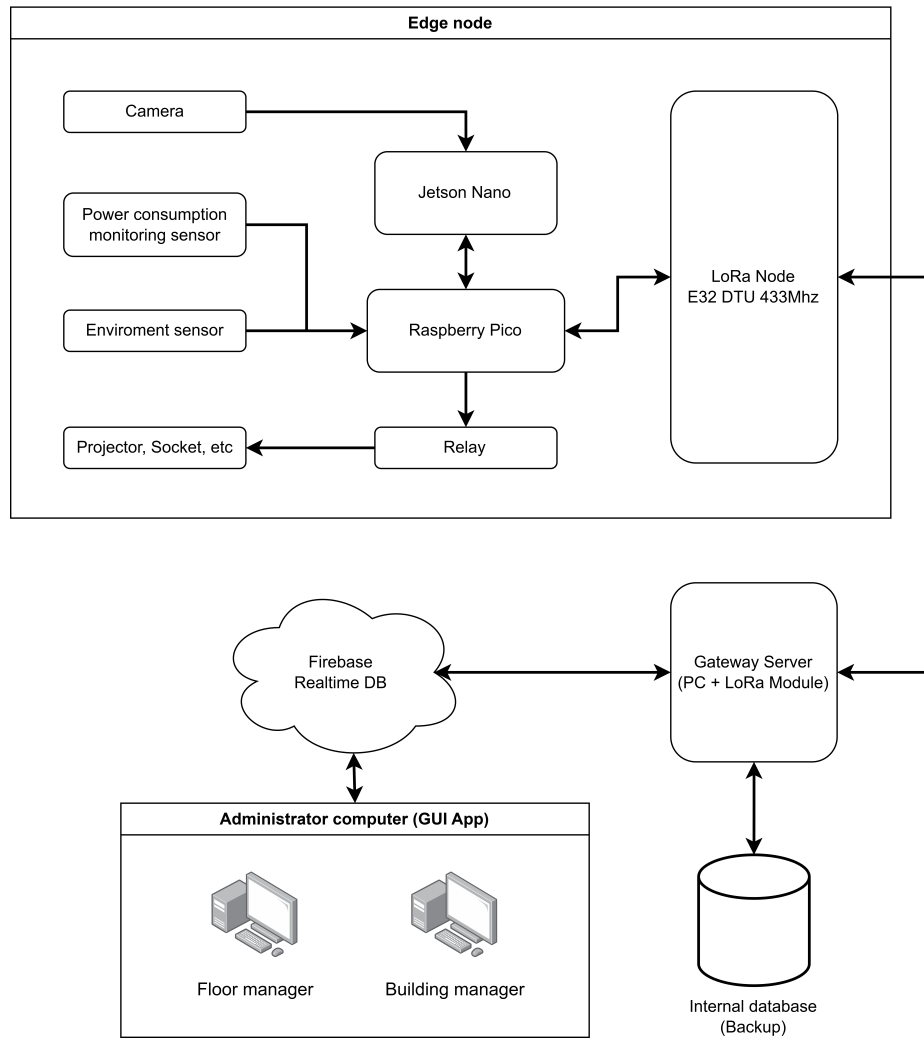
## 3 Thiết kế hệ thống

### 3.1 Hệ thống tổng thể

Ứng dụng trên máy tính quản trị (Administrator GUI) sẽ kết nối trực tiếp tới Firebase để lấy dữ liệu hiển thị cho quản lý tòa nhà (Building Manager) và quản lý tầng (Floor Manager). Giải pháp này cho phép giám sát hệ thống từ xa qua Internet mà không phụ thuộc vào vị trí vật lý của thiết bị Gateway.

#### 3.1.1 Phân hệ Edge node (Thu thập và xử lý)

Sử dụng Jetson Nano tích hợp AI (YOLOv8) để nhận diện khuôn mặt, FaceNet để xác minh danh tính và Raspberry Pi Pico để thu thập thông số môi trường/điện năng. Dữ liệu sau xử lý được đóng gói và truyền đi qua giao thức LoRa để đảm bảo khoảng cách xa và tiết kiệm năng lượng.



Hình 1: Sơ đồ khối hệ thống đề xuất

### 3.1.2 Phân hệ Gateway & Cloud (Trung chuyển & Lưu trữ)

- **LoRa Gateway:** Đóng vai trò trạm thu nhận tín hiệu từ các Edge Node. Tại đây, dữ liệu được lưu trữ cục bộ vào Internal Database (dự phòng sự cố mạng), đồng thời được đồng bộ hóa thời gian thực (Real-time) lên **Firestore Cloud**.
- **Firestore Realtime DB:** Là cầu nối trung gian, giúp lưu trữ dữ liệu tập trung và đảm bảo tính toàn vẹn dữ liệu.

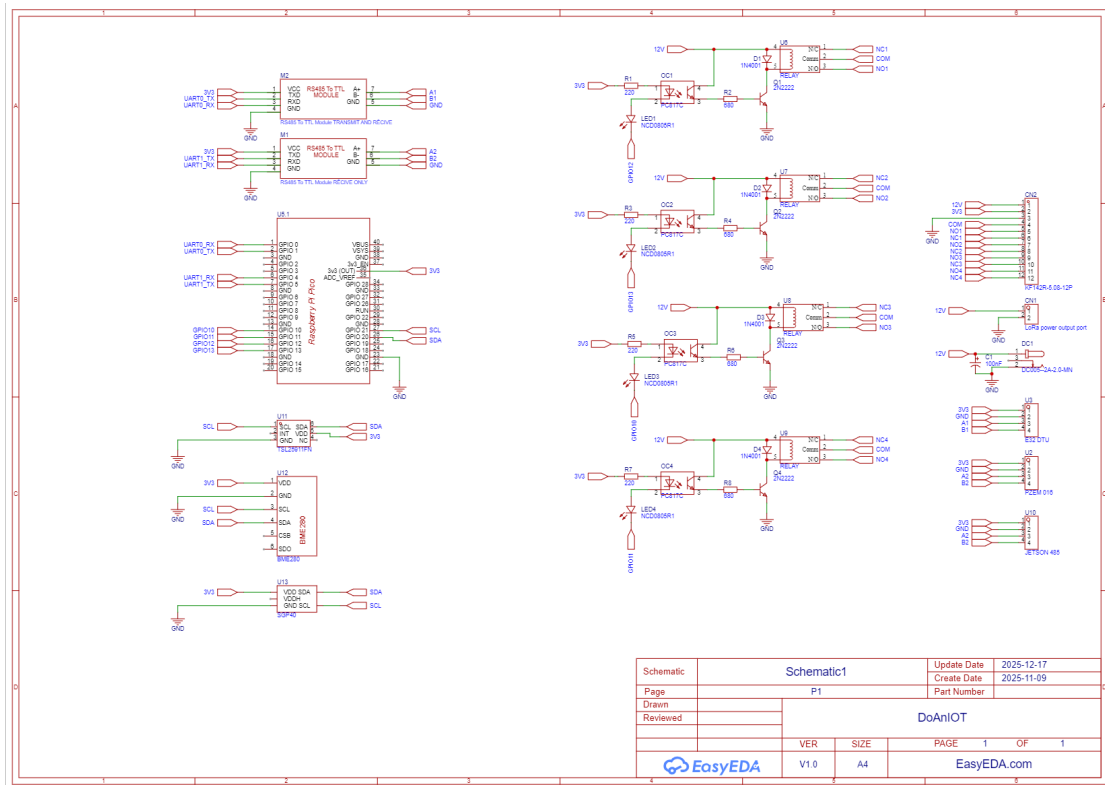
### 3.1.3 Phân hệ Quản lý (Application)

Ứng dụng trên máy tính quản trị (Administrator GUI) sẽ kết nối trực tiếp tới Firestore để lấy dữ liệu hiển thị cho quản lý tòa nhà (Building Manager) và quản lý tầng (Floor Manager). Giải pháp này cho phép giám sát hệ thống từ xa qua Internet mà không phụ thuộc vào vị trí vật lý của thiết bị Gateway.

## 3.2 Thiết kế phần cứng

### 3.2.1 Cấu hình Edge Node

Mỗi Edge Node được thiết kế như một tổ hợp phần cứng nhằm cân bằng giữa khả năng xử lý AI tại biên và mức tiêu thụ năng lượng thấp. Kiến trúc phần cứng của Edge Node bao gồm bốn khối chức năng chính: khối xử lý AI, khối điều khiển và cảm biến, khối điều khiển chấp hành và khối giao tiếp truyền thông. Sơ đồ khối tổng thể của Edge Node được trình bày trong Hình 2.



Hình 2: Sơ đồ khối phần cứng của Edge Node.

**Khối xử lý AI (AI Processing Unit)** Khối xử lý AI sử dụng nền tảng NVIDIA Jetson Nano làm bộ xử lý trung tâm cho các tác vụ thị giác máy tính. Jetson Nano được trang bị GPU Maxwell 128 nhân và CPU ARM Cortex-A57 bốn nhân, cho phép thu nhận hình ảnh từ camera, phát hiện khuôn mặt bằng mô hình YOLOv8 và trích xuất vector đặc trưng bằng FaceNet. Việc xử lý hình ảnh trực tiếp tại biên giúp giảm tải băng thông truyền thông và tăng cường bảo mật dữ liệu, do hình ảnh thô không cần truyền về hệ thống trung tâm.

**Khối điều khiển và cảm biến (Control & Sensor Unit)** Khối điều khiển và cảm biến sử dụng vi điều khiển Raspberry Pico (chip RP2040) làm trung tâm điều phối, chịu trách nhiệm thu thập dữ liệu môi trường, giám sát tiêu thụ năng lượng và điều khiển các thiết bị ngoại vi. Pico được lựa chọn nhờ khả năng hỗ trợ đa dạng các chuẩn giao tiếp như I2C, UART và SPI, đồng thời có mức tiêu thụ năng lượng thấp.

Các cảm biến môi trường được tích hợp thông qua module mở rộng Waveshare Pico Environment Sensor và giao tiếp với Pico thông qua bus I2C dùng chung. Các cảm biến

chính bao gồm BME280 (đo nhiệt độ, độ ẩm và áp suất), TSL2591 (đo cường độ ánh sáng), LTR390-UV-1 và SGP40 (đo tia cực tím và nồng độ hợp chất hữu cơ dễ bay hơi). Do sử dụng Sensor HAT thương mại, các mạch điều hòa tín hiệu và điện trở kéo lên bus I2C đã được tích hợp sẵn.

Để giám sát tiêu thụ điện năng của phòng học, hệ thống sử dụng module PZEM-016 giao tiếp với Pico thông qua giao thức Modbus-RTU trên nền tảng RS485. Do Pico hoạt động ở mức logic UART TTL, một mạch chuyển đổi UART TTL sang RS485 được sử dụng nhằm đảm bảo tương thích điện áp và chuẩn truyền thông.

**Khối điều khiển chấp hành (Actuator Control)** Khối điều khiển chấp hành cho phép Raspberry Pico điều khiển các thiết bị điện trong phòng học thông qua các module relay công suất. Tín hiệu điều khiển từ Pico được cách ly quang bằng optocoupler và khuếch đại bằng transistor, giúp đảm bảo an toàn cho khối điều khiển và giảm nhiễu từ tải điện xoay chiều.

**Giao tiếp nội bộ và truyền thông** Jetson Nano và Raspberry Pico trao đổi dữ liệu nội bộ thông qua giao tiếp UART/Serial, trong đó Jetson gửi kết quả nhận diện (ID sinh viên) về Pico. Pico kết nối với module LoRa E32 DTU (433 MHz, chip SX1278) thông qua UART để truyền dữ liệu cảm biến và dữ liệu nhận dạng về Gateway.

### 3.2.2 Thiết kế LoRa Gateway

Gateway đóng vai trò là cầu nối giữa mạng LoRa và hệ thống quản lý trung tâm. Gateway sử dụng module LoRa E32 DTU để thu nhận các gói tin từ các Edge Node và truyền dữ liệu qua giao tiếp USB/Serial tới máy tính quản trị, nơi dữ liệu được giải mã, lưu trữ và hiển thị trên giao diện Dashboard.

## 3.3 Thiết kế phần mềm

### 3.3.1 Giải thuật xử lý AI tại biên (Edge AI – Jetson Nano)

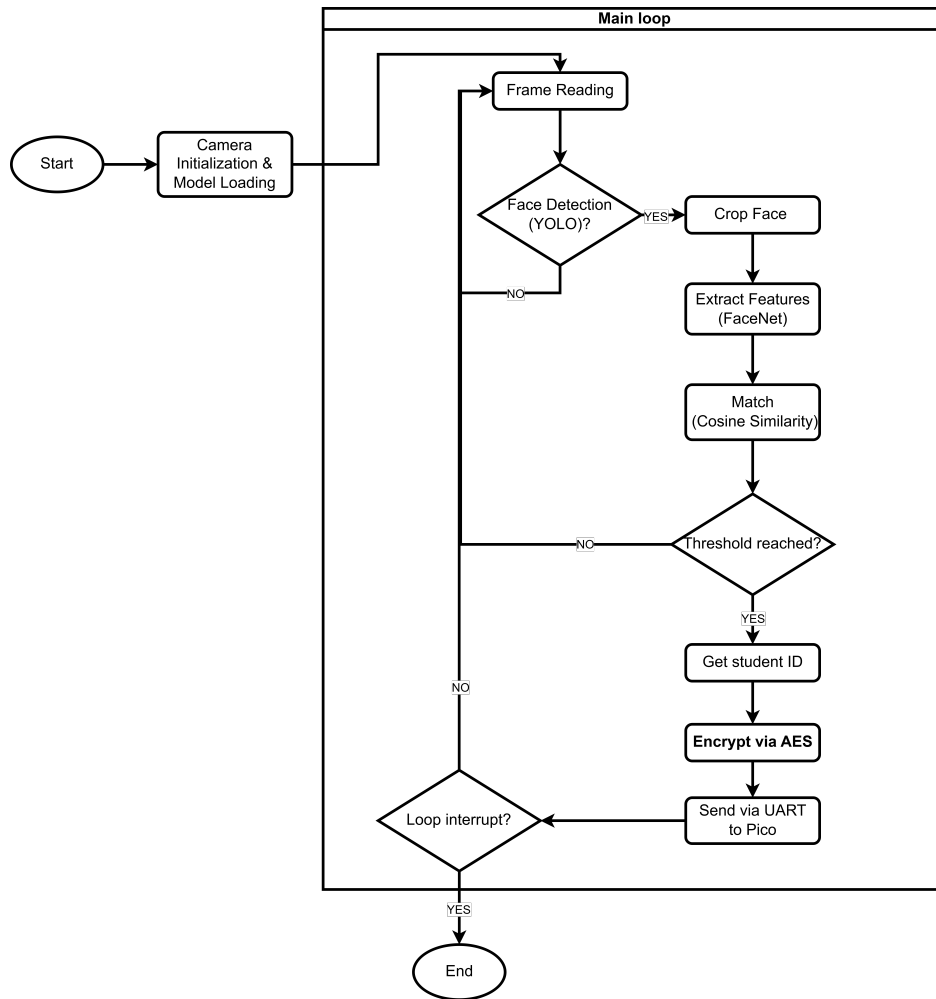
Giải thuật xử lý AI tại biên được triển khai trực tiếp trên nền tảng NVIDIA Jetson Nano nhằm thực hiện nhận diện khuôn mặt và xác định danh tính sinh viên ngay tại Edge Node, qua đó giảm thiểu phụ thuộc vào hạ tầng mạng trung tâm. Luồng xử lý dữ liệu hình ảnh được thiết kế theo chuỗi tuần tự, như minh họa trong Hình 3.

Đầu tiên, Jetson Nano thu nhận luồng hình ảnh thời gian thực từ camera thông qua thư viện OpenCV. Các khung hình sau khi được tiền xử lý cơ bản sẽ được đưa vào mô hình YOLOv8 để thực hiện phát hiện khuôn mặt. Mô hình YOLOv8 cho phép xác định nhanh vị trí khuôn mặt trong ảnh với độ chính xác cao và độ trễ thấp, phù hợp cho các ứng dụng thời gian thực tại biên.

Sau khi khuôn mặt được phát hiện, vùng ảnh tương ứng sẽ được cắt (crop) từ khung hình gốc và đưa vào mô hình FaceNet để trích xuất vector đặc trưng. FaceNet ánh xạ mỗi khuôn mặt thành một vector đặc trưng có kích thước cố định trong không gian đặc trưng, trong đó khoảng cách giữa các vector phản ánh mức độ tương đồng giữa các khuôn mặt. Vector đặc trưng thu được sau đó được so khớp với cơ sở dữ liệu cục bộ lưu trữ trên Jetson Nano bằng các phép đo khoảng cách (ví dụ: Euclidean distance hoặc cosine similarity) để xác định danh tính sinh viên tương ứng (MSSV).

Một điểm nhấn quan trọng của hệ thống là cơ chế bảo mật dữ liệu ngay tại biên. Sau khi quá trình nhận diện hoàn tất và chỉ còn lại kết quả định danh (MSSV), toàn





Hình 3: Luồng xử lý AI tại biên trên Jetson Nano.

bộ dữ liệu hình ảnh và vector đặc trưng không được truyền ra khỏi Jetson Nano. Trước khi gửi MSSV sang Raspberry Pico thông qua giao tiếp UART, dữ liệu này được mã hóa ngay lập tức bằng thuật toán AES với độ dài khóa 128 hoặc 256 bit. Cơ chế mã hóa này đảm bảo rằng thông tin sinh viên không thể bị khai thác ngay cả trong trường hợp gói tin bị bắt trên đường truyền LoRa.

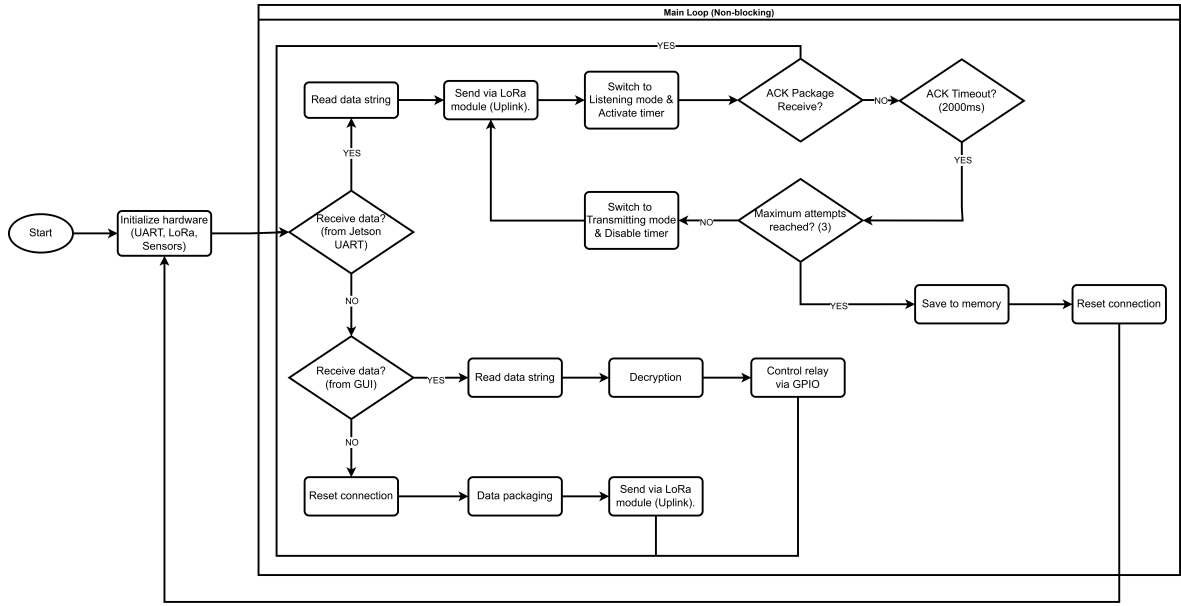
Việc kết hợp xử lý AI tại biên và mã hóa dữ liệu trước truyền thông giúp hệ thống vừa đảm bảo độ trễ thấp, vừa nâng cao mức độ riêng tư và an toàn thông tin. Kiến trúc này đặc biệt phù hợp với các ứng dụng Smart Campus, nơi yêu cầu cao về bảo mật dữ liệu cá nhân và khả năng mở rộng hệ thống.

### 3.3.2 Giải thuật điều khiển và truyền thông tại Node (Raspberry Pi Pico)

Để đảm bảo khả năng xử lý đa nhiệm trên vi điều khiển đơn nhân, phần mềm trên Raspberry Pi Pico được thiết kế theo kiến trúc vòng lặp không chặn (Non-blocking I/O) kết hợp với máy trạng thái (State Machine). Lưu đồ giải thuật chi tiết được trình bày trong Hình 4.

Quy trình hoạt động của hệ thống được chia thành ba luồng xử lý chính trong vòng lặp vô hạn (Main Loop):

- **Luồng xử lý dữ liệu AI (Uplink Priority):** Đây là tác vụ có độ ưu tiên cao



Hình 4: Lưu đồ giải thuật xử lý đa nhiệm và truyền tin cậy tại Raspberry Pi Pico.

nhất. Pico liên tục kiểm tra bộ đệm UART kết nối với Jetson Nano. Khi nhận được chuỗi dữ liệu (MSSV đã mã hóa), hệ thống thực hiện quy trình *Gửi tin cậy (Reliable Transmission)*:

1. Gửi gói tin qua module LoRa (Transmitting Mode).
  2. Chuyển ngay sang chế độ *Listening Mode* và kích hoạt bộ định thời (Timer).
  3. Chờ tín hiệu xác nhận (ACK) từ Gateway trong khoảng thời gian Timeout (2000ms).
  4. **Cơ chế thử lại (Retry Logic):** Nếu hết thời gian Timeout mà không nhận được ACK, hệ thống sẽ tự động gửi lại gói tin. Quá trình này lặp lại tối đa 3 lần ( $Maximum\ attempts = 3$ ).
  5. **Chế độ ngoại tuyến (Offline Mode):** Nếu sau 3 lần gửi thất bại, dữ liệu sẽ được lưu vào bộ nhớ (Flash/EEPROM) để gửi lại sau, đảm bảo không mất mát dữ liệu điểm danh.
- **Luồng điều khiển thiết bị (Downlink):** Nếu không có dữ liệu từ Jetson, Pico kiểm tra tín hiệu đến từ Gateway (GUI). Gói tin nhận được sẽ được giải mã (Decryption). Nếu xác thực đúng lệnh điều khiển, Pico sẽ thay đổi trạng thái GPIO để đóng/ngắt Relay điều khiển thiết bị ngoại vi.
  - **Luồng giám sát môi trường (Periodic Task):** Trong trường hợp không có sự kiện ngắt nào xảy ra, hệ thống thực hiện đóng gói dữ liệu từ các cảm biến (Nhiệt độ, độ ẩm, điện năng) và gửi định kỳ về Gateway để cập nhật Dashboard.

Việc áp dụng cơ chế xác thực ACK và lưu trữ cục bộ giúp hệ thống duy trì độ tin cậy cao ngay cả trong điều kiện môi trường truyền dẫn vô tuyến không ổn định.

## 4 Kết quả thực nghiệm

### 4.0.1 Hiệu năng thực tế trên Jetson Nano

Để tìm ra cấu hình tối ưu cho thiết bị biên, nhóm nghiên cứu đã tiến hành đo đặc tốc độ xử lý (FPS) tại ba mức độ phân giải đầu vào khác nhau. Đặc biệt, thuật toán FaceNet được tối ưu hóa bằng cách chỉ kích hoạt trích xuất đặc trưng một lần mỗi 30 khung hình (hoặc khi đối tượng đứng yên), giúp giảm tải tính toán.

Kết quả thực nghiệm được tổng hợp trong Bảng 1:

Bảng 1: So sánh hiệu năng xử lý tại các độ phân giải khác nhau trên Jetson Nano

Độ phân giải (px)	FPS trung bình	Đánh giá khả năng đáp ứng
640 × 480	< 6.0	<b>Thấp.</b> Video bị giật lag, độ trễ cao, gây khó khăn cho việc nhận diện khi đối tượng di chuyển nhanh.
480 × 320	8.4	<b>Trung bình.</b> Hiệu năng có cải thiện nhưng chưa đảm bảo độ mượt mà khi có nhiều khuôn mặt xuất hiện cùng lúc.
320 × 240	<b>12.2</b>	<b>Tốt.</b> Video mượt mà, độ trễ thấp. Với sự hỗ trợ của cơ chế FaceNet ngắt quãng (1 lần/30 frames), độ chính xác vẫn được đảm bảo trong khi tài nguyên CPU được giải phóng cho các tác vụ nền.

**Kết luận:** Dựa trên Bảng 1, nhóm quyết định lựa chọn độ phân giải 320 × 240 làm cấu hình triển khai chính thức cho hệ thống.

### 4.1 Kết quả đánh giá truyền thông LoRa

Để đánh giá hiệu năng thực tế của hệ thống truyền thông LoRa, nhóm nghiên cứu đã tiến hành hai đợt thử nghiệm độc lập và thu thập dữ liệu về thời gian khứ hồi (Round Trip Time - RTT). Các tham số cấu hình LoRa được thiết lập gồm: Spreading Factor (SF) 12, băng thông (Bandwidth) 125 kHz, Coding Rate 4/5 và kích thước tải trọng (payload) dao động quanh mức 50 byte.

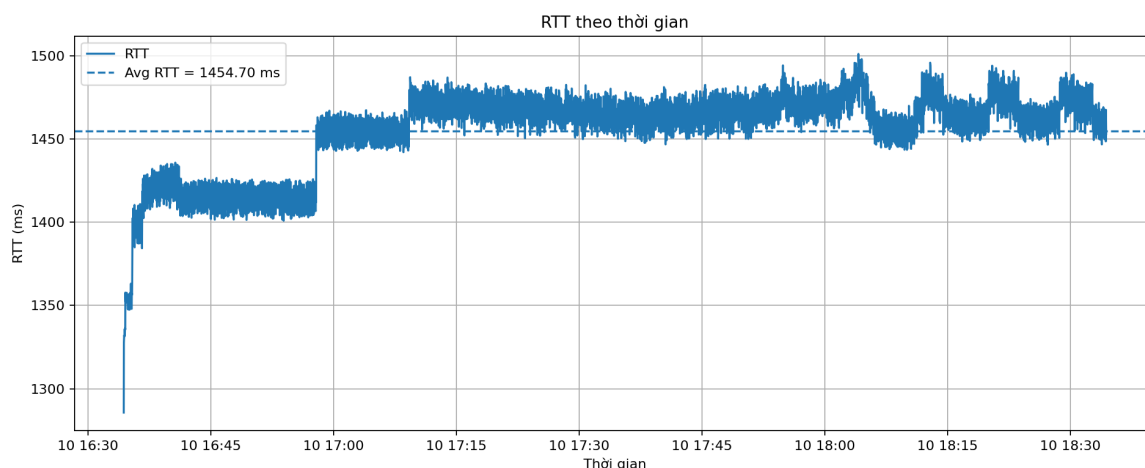
Biểu đồ biến thiên của RTT theo thời gian trong hai đợt thử nghiệm được thể hiện lần lượt trong Hình 5 và Hình 6.

Kết quả đo đạc cho thấy RTT trung bình luôn duy trì trong khoảng 1,45 đến 1,46 giây. Trong toàn bộ quá trình đo, hệ thống không ghi nhận bất kỳ trường hợp mất gói tin nào, dẫn đến tỷ lệ chuyển gói thành công (Packet Delivery Ratio - PDR) đạt mức tuyệt đối 100%. Payload trung bình xấp xỉ 49 byte, phản ánh đúng kích thước gói tin ứng dụng được định dạng trong hệ thống (bao gồm tiêu đề, dữ liệu cảm biến và mã hóa).

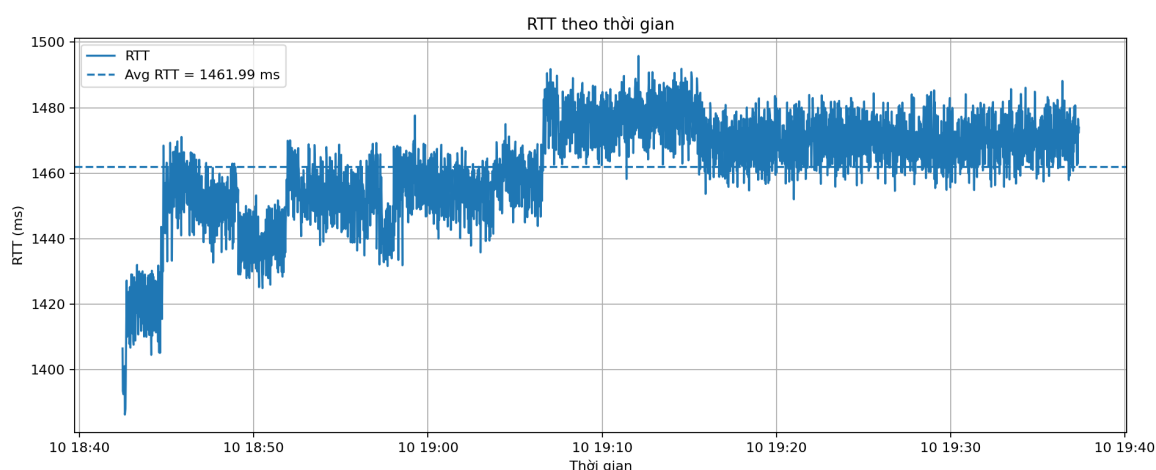
Bảng 2 dưới đây tóm tắt các chỉ số hiệu năng chính thu được từ hai lượt thử nghiệm:

#### Đánh giá và Thảo luận:

- **Độ ổn định:** RTT của hệ thống có độ ổn định cao, dao động chủ yếu trong khoảng  $\pm 200$  ms quanh giá trị trung bình. Mức dao động này là chấp nhận được đối với giao thức truyền thông vô tuyến tầm xa.



Hình 5: Biểu đồ RTT theo thời gian - Lượt thử nghiệm 1 (Thời gian đo: 2 giờ).



Hình 6: Biểu đồ RTT theo thời gian - Lượt thử nghiệm 2 (Thời gian đo: 55 phút).

Bảng 2: Tổng hợp kết quả thử nghiệm hiệu năng mạng LoRa

Lượt test	RTT TB (ms)	PDR (%)	Payload TB (byte)	Thời lượng (giờ)	Số mẫu
1	1454,70	100	49,3	7200 (2h)	4945
2	1461,99	100	49,5	3294 (55m)	2252

- **Độ tin cậy:** Với PDR đạt 100% trong cả hai kịch bản (với tổng số mẫu hơn 7000 gói tin), hệ thống chứng minh được mức độ tin cậy rất tốt, đảm bảo không thất thoát dữ liệu điểm danh quan trọng.
- **Độ trễ (Latency):** Việc sử dụng SF12 (để tối ưu khoảng cách truyền và khả năng xuyên tường) kết hợp với payload lớn ( 50 byte) làm tăng thời gian truyền sóng (Time-on-Air), dẫn đến RTT ở mức cao ( 1,45s). Tuy nhiên, kết quả này hoàn toàn đáp ứng được yêu cầu của bài toán thu thập dữ liệu môi trường và điểm danh sinh viên, vốn không đòi hỏi độ trễ cực thấp (Ultra-low latency) như

các ứng dụng điều khiển thời gian thực.

## 5 Kết luận

### Tài liệu

- [1] A. H. Buckman, M. Mayfield, and S. BM Beck, “What is a smart building?” *Smart and sustainable built environment*, vol. 3, no. 2, pp. 92–109, 2014.
- [2] W. Xu, J. Y. Kim, W. Huang, S. S. Kanhere, S. K. Jha, and W. Hu, “Measurement, characterization, and modeling of lora technology in multifloor buildings,” *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 298–310, 2019.
- [3] P. Fraga-Lamas, M. Celaya-Echarri, P. Lopez-Iturri, L. Castedo, L. Azpilicueta, E. Aguirre, M. Suárez-Albela, F. Falcone, and T. M. Fernández-Caramés, “Design and experimental validation of a lorawan fog computing based architecture for iot enabled smart campus applications,” *Sensors*, vol. 19, no. 15, p. 3287, 2019.
- [4] N. G. Pham, N. L. Bui, Q. H. Tran, L. T. H. Tong, and H. N. Tran, “An effective approach to face recognition with artificial intelligence and the internet of things using nvidia jetson nano,” *Dalat University Journal of Science*, pp. 39–64, 2024.
- [5] I. Khurshid, T. Abbas, S. A. Rammy, M. A. Baloch *et al.*, “Fusion of lora rfid and ai base smart security recognition systems for rural areas,” *STATISTICS, COMPUTING AND INTERDISCIPLINARY RESEARCH*, vol. 7, no. 2, pp. 139–156, 2025.
- [6] R. Liang, L. Zhao, and P. Wang, “Performance evaluations of lora wireless communication in building environments,” *Sensors*, vol. 20, no. 14, p. 3828, 2020.