

# CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI

## 1.1. Lý do chọn đề tài

Internet là một sáng tạo lớn và quan trọng trong lịch sử nhân loại. Được phát triển từ khoảng những năm 1960 cho đến nay và không ngừng phát triển trong tương lai. Internet là một mạng lưới cho phép gửi email, truy cập trình duyệt, tín hiệu truyền hình... Với công nghệ Internet vạn vật - Internet of Things (IoT) là tương lai phát triển của Internet. IoT có khả năng giao tiếp, kết nối các thiết bị, mở rộng các lĩnh vực, cung cấp các dịch vụ tới cho mọi người, mọi lúc và mọi nơi. [1]

Ứng dụng của IoT là vô cùng nhiều, tập trung ở các lĩnh vực: chăm sóc sức khỏe; ngôi nhà, tòa nhà thông minh; thành phố thông minh; giao thông vận tải; năng lượng... [1] Trong ứng dụng cho lĩnh vực sức khỏe có phần mềm quản lý hình ảnh y tế (Radiology Information System - RIS), phần mềm lưu trữ hình ảnh y tế - Picture archiving and communication system - PACS),... Hiện nay, việc sử dụng bức xạ, nguồn phóng xạ trong y tế là phổ biến. Do đó việc giám sát, đảm bảo an toàn cho nhân viên y tế là rất quan trọng. Đã có những đề xuất về ứng dụng IoT lên liều kế, ứng dụng lên ngôi nhà, tòa nhà thông minh để thực hiện việc kiểm soát bức xạ.

Với sự phổ biến việc sử dụng bức xạ vào trong y tế, đặc biệt là y học hạt nhân, việc giám sát, kiểm soát và đảm bảo an toàn là điều quan trọng trong môi trường bệnh viện, môi trường làm việc với phóng xạ. Trong một thống kê từ năm 2017 đến năm 2020 về trang thiết bị, nhân lực và lượng dược chất phóng xạ trong y học hạt nhân, Việt Nam hiện có hơn 45 cơ sở trong đó có 36 cơ sở là có máy SPECT hay SPECT/CT và máy PET/CT. Ngoài ra còn có 1 lò phản ứng hạt nhân ở Đà Lạt, 6 cyclotron ở Hà Nội, Đà Nẵng, thành phố Hồ Chí Minh. Đội ngũ nhân lực 138 bác sĩ, hơn 225 nhân lực kỹ thuật bao gồm: vật lý y khoa và kỹ thuật viên [2].

Với sự lớn mạnh của lĩnh vực y học hạt nhân, thì việc quan trọng là giám sát, quản lý chặt chẽ, đảm bảo an toàn trong quá trình làm việc với bức xạ, nguồn phóng xạ hạt nhân. Trong an toàn bức xạ, để giám sát môi trường ở các phòng thuộc các khoa có sử dụng thiết bị bức xạ hoặc nguồn phóng xạ thì sẽ có những thiết bị có thể đo liều lượng, suất liều hay phát hiện bức xạ. Radiation Alert Ranger®; liều kế cá nhân OSLD hoặc TLD là những ví dụ tiêu biểu.

Năm 2018, Ramacos cùng với các cộng sự của mình thiết ra loại liệu kế cá nhân có module IoT kết nối với mạng không dây. Sử dụng cảm biến bức xạ bỏ túi Geiger type 5, module Xbee series 2 và module Arduino microcontroller để giao tiếp tới máy tính thông qua mạng không dây. Với thiết kế này cho phép người dùng truy cập trực tiếp liệu mà nhân viên y tế nhận được do cảm biến Geiger type 5 ghi nhận [3].

Nhìn chung, sự phát triển của lĩnh vực IoT cũng như sự phổ biến ứng dụng bức xạ trong y tế đang là điều được quan tâm và chú trọng. Với việc sử dụng IoT với các máy đo, thiết bị ghi đo bức xạ, thì việc lưu trữ dữ liệu rất dễ dàng, phục vụ cho công tác kiểm tra và đánh giá mức độ an toàn trong các khoa sử dụng bức xạ hoặc nguồn phóng xạ ở bệnh viện.

Radiation Alert Ranger® (RAR) là máy đo được sử dụng cho nhiều mục đích như: khảo sát ô nhiễm bề mặt; theo dõi, kiểm tra, phát hiện các bức xạ khi làm việc trong môi trường có tồn tại bức xạ, nguồn phóng xạ; phát hiện hạt nhân năng lượng thấp; sàng lọc ô nhiễm môi trường. Tuy nhiên, máy đo RAR chỉ có thể truyền dữ liệu bằng các phương pháp như kết nối bluetooth hoặc cổng USB. Ngoài ra việc lấy dữ liệu đo được phải thông qua một phần mềm độc quyền, phần mềm này phải mua từ nhà sản xuất của máy đo. Mặt khác, phần mềm này không có khả năng tích hợp các máy đo của hãng khác hoặc đo đa điểm. Do đó chúng tôi quyết định thực hiện đề tài “Ứng dụng Module ESP32-CAM và Thuật toán Tesseract Optical Character Recognition (Tesseract OCR) trong trích xuất số đo phóng xạ của thiết bị Radiation Alert Ranger”. Module ESP32-CAM sẽ thu thập ảnh màn hình LCD của máy đo RAR, gửi ảnh này qua kết nối wifi đến phần mềm xử lý trên máy tính. Thuật toán Tesseract OCR được sử dụng để trích xuất số đọc từ ảnh. Số đọc này sẽ được hiển thị trên đồ họa giao diện người dùng theo thời gian thực hoặc lưu trữ vào tập tin.

## **1.2. Mục tiêu đề tài**

Chúng tôi mong muốn thực hiện đề tài này nhằm mục đích phát triển một hệ thống bao gồm phần cứng và phần mềm có tên là RadOCR. Hệ thống có khả năng kết nối mạng và trích xuất từ xa số đọc của máy đo RAR. Đề tài “Ứng dụng Module ESP32-CAM và Thuật toán Tesseract OCR trong trích xuất số đo phóng xạ của thiết bị Radiation Alert Ranger” sẽ bao gồm các mục tiêu:

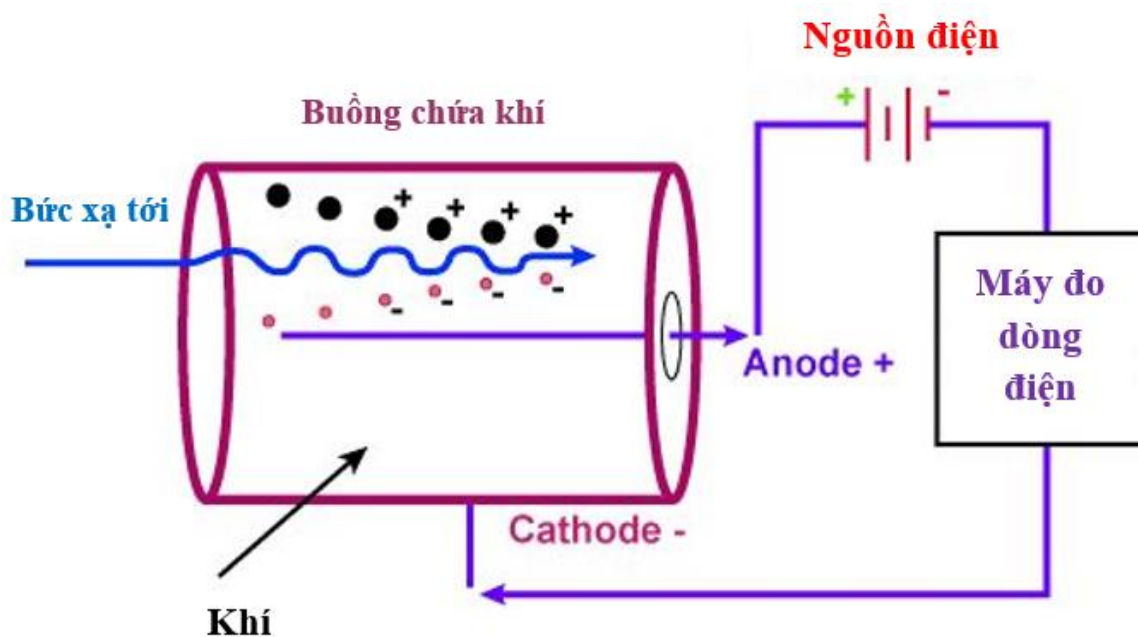
- Nghiên cứu cơ sở lý thuyết liên quan đến đề tài
- Thiết kế phần cứng và phần mềm
- Gia công phần cứng và lập trình phần mềm
- Chạy thử và đánh giá

## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

### 2.1. Các loại đầu dò

#### 2.1.1. Đầu dò chứa khí

Đầu dò chứa khí được cấu tạo từ một buồng chứa khí (Hình 2.1). Vỏ buồng được làm bằng chất dẫn điện, được sử dụng như một điện cực gọi là cathode. Ở trung tâm của buồng có một sợi dây bằng kim loại được đặt cách điện với vỏ, gọi là anode. Để đầu dò hoạt động, ta cần phân cực cho nó bằng nguồn điện một chiều có hiệu điện thế khoảng vài trăm vôn. Anode được nối vào cực dương, cathode được nối vào cực âm của nguồn [4].

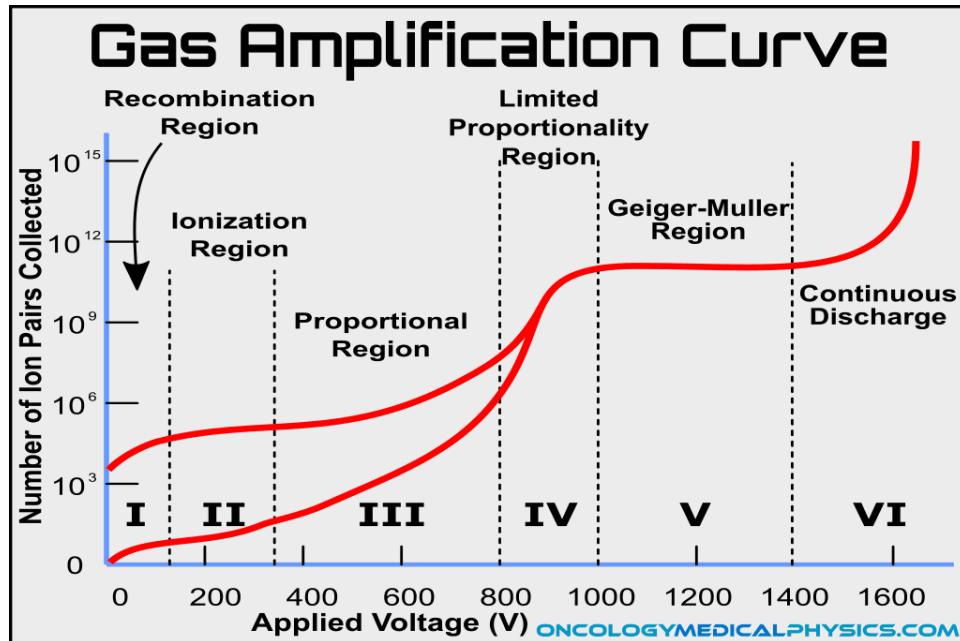


Hình 2.1. Sơ đồ cấu tạo đầu dò chứa khí

Nguyên lý hoạt động của đầu dò chứa khí dựa trên hiệu ứng ion hóa của bức xạ gây ra đối với các phân tử chất khí bên trong buồng. Bức xạ tới tương tác với các nguyên tử của khí trong buồng, làm bật ra các electron của nguyên tử, tạo thành các electron (điện tích âm) tự do. Đồng thời, nguyên tử bị mất electron sẽ bị mất cân bằng điện tích và trở thành các ion dương (điện tích dương) tự do trong thể tích của đầu dò. Dưới tác dụng của điện trường phân cực giữa hai điện cực của đầu dò, các điện tích âm sẽ chuyển động ngược chiều điện trường, điện tích dương sẽ chuyển động thuận chiều điện trường. Sự chuyển động như thế sẽ tạo ra dòng điện và có thể được

ghi nhận ở mạch ngoài. Như vậy, thông qua việc đo dòng điện ở mạch ngoài, ta biết được khi nào có tia bức xạ đến tương tác với đầu dò [4]

Trong hình 2.2, buồng ion hóa phụ thuộc vào điện trường, nên có sáu vùng mô tả đường khuếch đại khí:



Hình 2.2. Đồ thị về đường khuếch đại khí ở các mức điện áp khác nhau

Trong trường hợp áp vào một điện áp rất thấp, điện trường không đủ khả năng tác động tới các điện tích, dẫn đến việc các điện tích lúc này sẽ tái hợp lại thành các nguyên tử, hiện tượng này được gọi là tái tổ hợp [5]. Vùng điện áp thấp này được gọi là vùng tái tổ hợp [6]. Vùng này khiến cho việc đo lường bức xạ tới gặp khó khăn khi không thu thập được các tín hiệu từ các điện tích [4]. Trường hợp khác khi ta tăng điện áp dần, vào khoảng từ 50V – 300V, tín hiệu điện đo được gần như không thay đổi. Ở vùng điện áp này sẽ chỉ có các điện tích sơ cấp mới được tiếp nhận. Các thiết bị buồng ion hóa được sử dụng để hiệu chuẩn liều thường sẽ thực hiện ở vùng bão hòa này [6].

Khi ta tăng điện áp lên, vượt qua vùng bão hòa sẽ là vùng tỷ lệ. Sự kiện ban đầu ở vùng tỷ lệ này sẽ giống với vùng bão hòa, khi bức xạ tới tương tác với electron của nguyên tử và cung cấp năng lượng cho electron làm bật nó ra khỏi nguyên tử, electron bị bật này được gọi là electron sơ cấp. Khi electron sơ cấp này đang đi về anode, nếu

năng lượng của electron được cấp bởi bức xạ tới đủ lớn, electron sơ cấp này sẽ làm bật một electron khác của một nguyên tử khác. Electron vừa bị bật ra được gọi là electron thứ cấp [4]. Ở vùng tỷ lệ, Cả các electron thứ cấp lẫn electron sơ cấp đều gây ra tín hiệu. Số lượng electron thứ cấp tỷ lệ thuận với electron sơ cấp, được gọi là khuếch đại khí [5].

Vùng Geiger Muller là vùng tiếp theo sau vùng tỷ lệ. Trong vùng này, các electron sơ cấp có thể tương tác và tạo ra nhiều electron thứ cấp. Đến lượt mình, các electron thứ cấp lại có khả năng tương tác để tạo thêm nhiều electron khác. Cứ như vậy, quá trình ion hóa xảy ra dây chuyền, ồ ạt dẫn đến gần như toàn bộ phân tử khí trong thể tích đầu dò đều bị ion hóa hết. Hiệu ứng này gọi là “thác lũ” [4]. Tín hiệu tạo ra trong trường hợp này có giá trị bão hòa và không tăng ngay cả khi tăng điện thế phân cực. Trong trường hợp ta tiếp tục tăng điện thế lên, thì ngoài việc hiện tượng ion hóa xảy ra do tương tác của bức xạ với nguyên tử khí, thì còn có sự ion hóa tự phát do điện trường quá lớn gây ra phóng điện giữa hai điện cực. Việc vận hành đầu dò trong vùng điện áp này rất nguy hiểm, hậu quả có thể làm hư hỏng máy đo hoặc gây sai số đếm... Hiện tại vẫn chưa có tiềm năng cho thấy vùng này sẽ áp dụng cho thiết bị ghi đo bức xạ [7].

Bằng cách phân cực đầu dò chứa khí ở các vùng điện áp khác nhau, ta thu được các loại đầu dò chứa khí khác nhau, bao gồm: buồng ion hóa, ống đếm tỷ lệ, và ống đếm Geiger Muller.

Buồng ion hóa là loại đầu dò dùng để đo các hạt alpha, beta, và tia gamm. Buồng ion hóa hoạt động ở điện áp thấp. Dòng điện chạy trong buồng thấp, khoảng  $10^{-12}$  Ampe, nên loại máy đo này luôn phải có module phức tạp để có thêm khả năng khuếch đại dòng điện. Tùy vào mục đích sử dụng, máy đo sẽ có thiết kế khác. Trường hợp máy đo ghi đo nhỏ tính di động cao, thì buồng sẽ chứa đầy khí và vật liệu được dùng chế tạo sẽ có số nguyên tử Z thấp. Trường hợp đo hạt alpha, hạt beta thì thiết kế thành hay vỏ phải mỏng, cửa sổ đo cũng mỏng. Để phân biệt hạt alpha, beta, và tia gamma sẽ cần một lá chắn ngăn hạt alpha hoặc beta đi vào buồng [4].

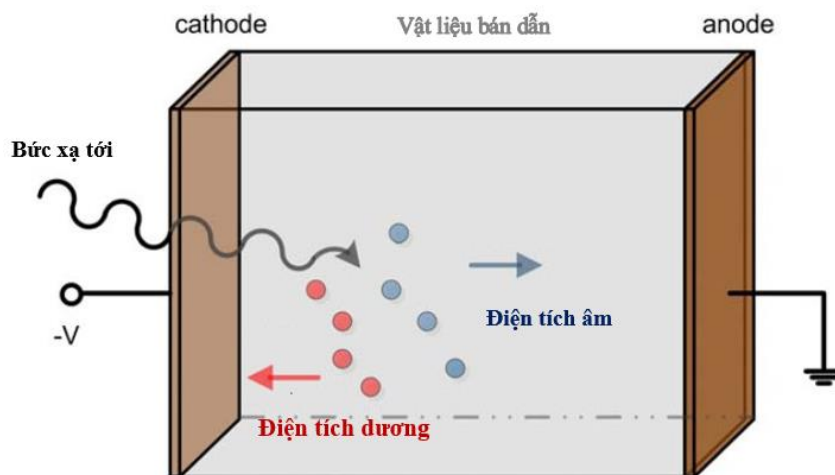
Ống đếm tỷ lệ là loại đầu dò được thiết kế cho phép đếm hạt alpha và hạt beta. Nó hoạt động ở vùng điện áp tỷ lệ. Độ ion hóa của hạt alpha cao hơn so với hạt beta

nên máy đo có thể phân biệt được giữa hai hạt này. Sử dụng khí P-10, là loại khí bao gồm 90% khí Argon và 10% khí Metan và không sử dụng loại khí thông thường để thực hiện việc ion hóa, chỉ sử dụng P-10. [6]

Ống đếm Geiger Muller (GM) là loại đầu dò chứa khí hoạt động ở vùng Geiger Muller. Sử dụng khí P-10, giống với khí trong ống đếm tỷ lệ. Ống đếm này không phụ thuộc hay phân biệt được năng lượng của bức xạ tới, và năng lượng bức xạ truyền vào các ion âm hay electron khi tương tác, đồng nghĩa với việc là nó không phân biệt được hạt alpha và hạt beta. Để phân biệt hạt alpha, beta thì chỉ cần một lá chắn hoặc nắp để che cửa sổ đầu dò. Các đầu dò GM không đo được suất liều hoặc liều lượng. Với loại ống đếm GM này, cần phân cực bằng một điện áp cao nên vùng hoạt động của nó gần với vùng phóng điện. Để giải quyết sự phóng điện có diễn ra trong ống đếm GM với tỷ lệ nhỏ, người ta sử dụng các loại khí hữu cơ hoặc halogen để dập tắt sự phóng điện. So sánh giữa khí hữu cơ và khí halogen, thì halogen được đánh giá tốt về mặt tuổi thọ cho máy đo trong tình huống có tốc độ đếm cao. Việc dập tắt có ý nghĩa là đưa máy đo hoạt động dưới vùng GM nhằm làm các ion đang ở trạng thái kích thích hoặc phóng điện bị khử đi [4].

### 2.1.2. Đầu dò bán dẫn

Các đầu dò bán dẫn dạng chất rắn có cấu tạo tương tự với đầu dò chứa khí. Điểm khác biệt là thay vì chứa khí, thì loại đầu dò bán dẫn này dùng vật liệu bán dẫn trong thể tích của đầu dò như trong hình 2.3 trình bày. Vật liệu bán dẫn là loại vật liệu vừa có thể cách điện và cũng vừa có thể dẫn điện.



Hình 2.3. Sơ đồ cấu tạo đầu dò bán dẫn

Đầu dò bán dẫn bao gồm hai loại vật liệu: loại p và loại n. Hai loại này tiếp giáp nhau tạo thành tiếp giáp p-n. Trong đó, loại n là loại vật liệu mà khi ta đưa thêm tạp chất vào và quá trình pha tạp (doping) cung cấp thêm các electron cho vùng hóa trị, dẫn đến việc chỉ với một năng lượng nhỏ được hấp thụ thì các electron sẽ di chuyển lên vùng dẫn. Còn với loại p, quá trình pha tạp cung cấp thêm lỗ trống ở vùng hóa trị. Xung quanh tiếp giáp p-n là vùng nghèo, vùng này sẽ không có lỗ trống và cũng không có các điện tích, nên lớp nghèo này có khả năng phát hiện sự hiện diện của bức xạ khi đi tới [4].

Bức xạ tới sẽ đi qua lớp nghèo và tương tác với các electron của nguyên tử tinh thể. Khi đủ năng lượng sẽ làm cho electron hấp thụ và bật ra, đồng thời electron mang năng lượng di chuyển lên vùng dẫn. Khi electron nhảy lên vùng dẫn, nó sẽ để lại một lỗ trống ở vùng hóa trị. Lỗ trống này sẽ có một electron từ nguyên tử khác đến lấp đầy và lại để lại một lỗ trống mới [4].

Các loại đầu dò bán dẫn phổ biến bao gồm: diode khuếch tán; đầu dò hàng rào mặt; đầu dò hàng rào mặt; đầu dò Lithium (Li), đầu dò Germanium – độ tinh khiết cao.

Điốt khuếch tán là dạng đầu dò dùng để phát hiện các hạt mang điện, tia gamma cụ thể đo lường gamma cá nhân. Thiết kế của Điốt khuếch tán bao gồm: 1 vật liệu loại p mỏng 1 micromet, và phần còn lại là vật liệu loại n [4].

Đầu dò hàng rào mặt có cấu tạo khác với điốt khuếch tán, khi vật liệu loại p sẽ là một lớp rất mỏng, mỏng hơn ở điốt khuếch tán, và loại p ở đầu dò hàng rào mặt sẽ phủ lên vật liệu loại n. Do cấu tạo như vậy nên bề mặt tinh thể phải luôn được giữ sạch, không để bám chất dạng dầu hoặc các vật thể lạ bám vào. Đầu dò này còn rất nhạy với ánh sáng photon, các ánh sáng photon khi đạt thể tích nhạy sẽ tạo ra các cặp lỗ trống – điện tử. Công dụng của loại đầu dò này được sử dụng để thực hiện lập phổ các hạt mang điện [4].

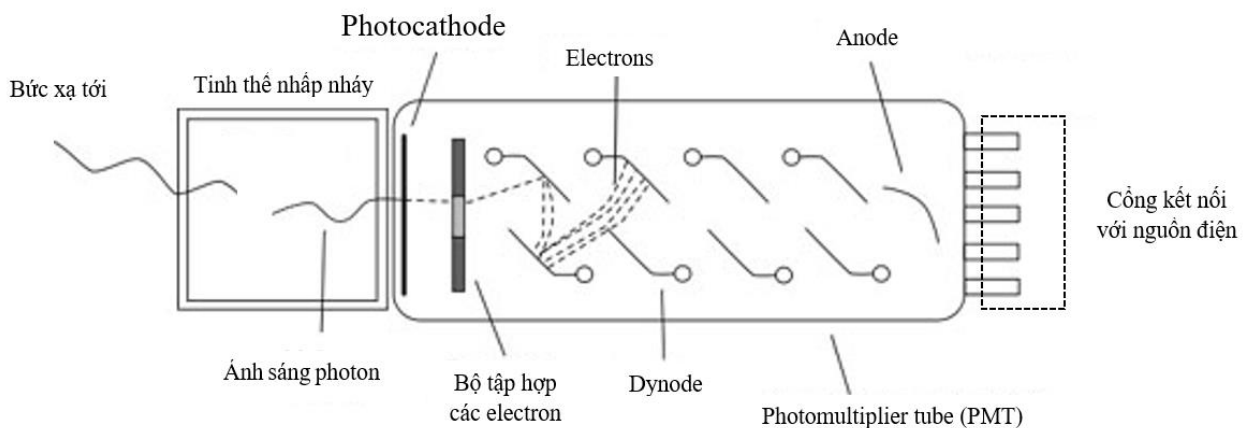
Đầu dò Lithium có cấu tạo pha tạp với Germanium (Ge) hoặc pha tạp với Silicon (Si). Khi pha tạp với Ge thì sẽ có khả năng thực hiện việc đo phổ cho tia gamma với độ phân giải năng lượng tốt, nhưng bất lợi là việc phải duy trì nitrogen lỏng ở nhiệt

độ phù hợp. Khi pha tạp với Si, đầu dò ngoài đo được tia gamma như khi pha tạp Ge với Lithium, còn thêm khả năng đo được phổ beta và phổ tia X. Có thể đo được các tia gamma năng lượng thấp hơn. Tuy nhiên về khả năng đo tia gamma thì kém hơn so với Lithium pha tạp Si [4].

Đầu dò Germanium – độ tinh khiết cao được sử dụng đo và phát hiện bức xạ gamma. Các tạp chất trong tinh thể đầu dò được giữ ở mức thấp nhất. Khả năng đo và phát hiện bức xạ tương tự với đầu dò Lithium – Ge nhưng cải tiến hơn ở phần bảo quản ở nhiệt độ phòng, khi không sử dụng. Bất lợi là phải làm lạnh đến nhiệt độ nitrogen lỏng khi hoạt động [4].

### 2.1.3. Đầu dò nhấp nháy

Cấu tạo của một đầu dò nhấp nháy bao gồm: một tinh thể nhấp nháy và ống nhân quang điện (photo-multiplier tube - PMT) được thể hiện qua sơ đồ cấu tạo hình 2.4. Sau ống PMT là các mạch điện tử: tiền khuếch đại, khuếch đại, bộ phân tích độ cao xung, bộ thu thập tín hiệu xung. [4]



Hình 2.4. Sơ đồ cấu tạo đầu dò nhấp nháy

Tinh thể trong đầu dò bao gồm các loại tiêu biểu như Natri Iodua (NaI) pha tạp Thallium (Tl) thành tinh thể NaI (Tl), Sunfua Kẽm (ZnS) pha tạp thêm bạc (Ag). Các tinh thể có vai trò làm thành phần tương tác với bức xạ tới để tạo ra các nhấp nháy sáng. Các photon ánh sáng sẽ chạm vào bề mặt của một vật liệu nhạy sáng gọi là photocathode.

Photocathode là một vật liệu nhạy sáng có vai trò chuyển đổi các photon ánh sáng thành các tín hiệu electron. Photon ánh sáng sẽ tương tác với các nguyên tử của

vật liệu Photocathode là Phot-pho và làm bật các electron ra khỏi Photocathode. Các electron bị bật ra sẽ đi qua ống nhân quang và được khuếch đại lên. [4]

Ống PMT là bộ phận quan trọng của cả đầu dò nhấp nháy. Nó đóng vai trò là bộ phận khuếch đại các tín hiệu điện electron sau khi các photon ánh sáng đi qua lớp photocathode. Trong ống PM bao gồm các Dynode, Anode, và nguồn cao thế cấp cho ống. Các dynode có chức năng là khuếch đại một electron mà nó thu thập được thành 4 electron. Số lượng dynode trong ống PMT là gần 10 cái và dynode đầu sẽ được cấp điện áp là 200V, các dynode sau sẽ được cấp điện áp gấp đôi dynode trước.

Sau ống PMT là các mạch điện như Bộ tiền khuếch đại, bộ khuếch đại, bộ phân tích xung. Trong đó, bộ tiền khuếch đại có vai trò khuếch đại tín hiệu điện từ ống PMT. Do tín hiệu điện là electron có biên độ nhỏ cần được khuếch đại hay điều chỉnh xung của electron. Vai trò khác của bộ tiền khuếch đại là ghép nối trở kháng giữa detector với bộ khuếch đại.

Bộ khuếch đại có vai trò khuếch đại tín hiệu từ bộ tiền khuếch đại lên khoảng 1000 lần để có được các tín hiệu mạnh cho bộ phía sau làm việc. Độ khuếch đại của xung được tính bằng: tỷ lệ của biên độ của xung phát ra từ bộ khuếch đại với biên độ của xung phát ra từ ống PMT. Ở bộ này còn có vai trò định hình các xung, tránh các tín hiệu xung này chồng chất tín hiệu xung khác.

Bộ phân tích xung dựa vào biên độ điện áp của xung (pulse-high analyser – PHA) có thể suy ra năng lượng bức xạ tương ứng. Có 2 loại PHA: bộ phân tích đơn kênh – (single-channel analyser – SCA) và bộ phân tích đa kênh (multi-channel analyser – MCA). Bộ phân tích đơn kênh có thể phát hiện ra một tín hiệu xung và chỉ xử lý tín hiệu này cho đến khi hoàn thành, nên bộ đơn kênh sẽ bỏ lỡ các tín hiệu xung kế tiếp trong quá trình xử lý tín hiệu trên. Bộ phân loại đa kênh có thể xử lý nhiều xung cùng lúc, rất ít khả năng bị mất tín hiệu, nhưng các tín hiệu xung có thể chồng lên nhau làm cho bộ đa kênh nhận định các tín hiệu bị chồng chập là một tín hiệu.

Chức năng bộ đếm/thống kê/xử lý và hiển thị: Thiết bị không ghi hình có nhiệm vụ đếm số xung (số sự kiện bức xạ) từ bộ phân tích, hiển thị các thông tin này dưới dạng số đếm hoặc số đếm/thời gian,... Thiết bị ghi hình sẽ thực hiện thống kê số

xung (số sự kiện bức xạ) từ bộ phân tích, xử lý các dữ liệu này bằng thuật toán dựng ảnh, lưu trữ, hiển thị,...

Khi có bức xạ tới tương tác với nguyên tử của tinh thể nhấp nháy, Các nguyên tử khi tương tác với bức xạ sẽ bị kích thích tạm thời. Trong quá trình bị kích thích, các electron hấp thụ năng lượng mà bức xạ để lại, sau đó nhảy lên mức năng lượng cao trong nguyên tử. Sau một thời gian ngắn ở mức năng lượng cao, electron bị kích thích đó sẽ nhảy về mức năng lượng ban đầu, đồng thời nó sẽ thải ra năng lượng. Năng lượng bị thải ra đó là photon ánh sáng. Photon ánh sáng này có thể nhìn thấy được. Số lượng photon ánh sáng tỷ lệ thuận với năng lượng của bức xạ tới, nhờ vậy mà đầu dò nhấp nháy có thể dùng để phân tích phổ, ngoài cái chức năng phát hiện bức xạ.

Các loại đầu dò nhấp nháy phổ biến bao gồm: đầu dò Sunfua Kẽm (ZnS) và đầu dò Natri Iodua (NaI (Tl)).

Đầu dò Sunfua Kẽm thường pha tạp thêm các nguyên tử bạc để là chất kích thích (kích hoạt), có tên gọi khác là đầu dò ZnS(Ag). Vật liệu này không cho phép các photon ánh sáng đi qua dễ dàng, nên nó được thiết kế cho mỏng lại. Do bị thiết kế dạng màng mỏng, nên loại vật liệu này rất dễ bị các vật nhọn đâm xuyên, gây thủng màng. Đầu dò Sunfua Kẽm thích hợp cho việc phát hiện các hạt alpha và các hạt ion nặng (ion nặng là hạt nó có điện tích và khối lượng nặng hơn hạt alpha). [4]

Đầu dò Natri Iodua được pha thêm một lượng nhỏ Thallium giúp tăng cường khả năng tạo ra các photon ánh sáng. Tinh thể ở đầu dò này rất dễ suy giảm bởi sự hấp thụ độ ẩm không khí, nên tinh thể được đặt trong một hộp nhôm kín và có một cửa sổ mỏng. Đầu dò thường được dùng để thực hiện đo quang phổ gamma. Trung bình 20 – 30 photon ánh sáng trên 1 keV năng lượng. [4] [6]

Đầu dò này sử dụng tinh thể khác nhau cho các mục đích khác nhau, sự khác nhau là nằm ở độ dày tinh thể. Một tinh thể được thiết kế độ dày 3mm, rất hiệu quả để phát hiện các tia gamma ở mức năng lượng 150 keV. Tinh thể dày hơn có thể đo được các tia gamma năng lượng cao hơn. Khi so sánh với đầu dò bán dẫn, thì nó có hiệu suất đo tốt hơn nhưng độ phân giải năng lượng kém hơn. Nhược điểm của đầu dò là dễ vỡ, ngoài ra việc thay đổi nhiệt độ phòng đột ngột có thể gây nứt tinh thể và làm giảm hiệu suất của đầu dò. [4]

## 2.2. Máy đo phóng xạ Radiation Alert Ranger®

Radiation Alert Ranger ® (RAR) là loại máy đo đo phóng xạ cầm tay, được sản xuất bởi S.E.International. Máy đo được chế tạo dựa trên buồng ion hóa, cụ thể là đầu dò Geiger Muller (GM), cửa sổ mỏng phía sau màn hình LCD, và mặt ngoài cùng là dạng lưới mà ta thấy được ở mặt sau của máy đo. Máy đo có kích thước dài x rộng x dày là 120 x 68 x 33 mm, trọng lượng là 292 g, khả năng chịu nhiệt từ -10 °C tới hơn 50 °C. (Hình 2-5) [8]



Hình 2.5. Máy đo Radiation Alert Ranger®

Máy đo RAR có các bộ phận như màn hình LCD, 1 đèn LED nhỏ có màu đỏ nằm ở đầu máy. Các nút bao gồm: Power (Enter), Count, Audio (Minus), Menu, Backlight (Plus), Mode. Các kết nối như cổng USB, kết nối Bluetooth [8]. Chức năng các nút:

- Power (Enter): Nằm ở trung tâm các nút của của máy. Vai trò bật hoặc tắt máy đo.
- Alarm: Nút cho phép đặt chuông báo hoặc phát ra âm thanh thông báo hoàn tất đo sau một thời gian do ta cài đặt.
- Count: được sử dụng để đếm thời gian, mặc định là 10 phút.

- Audio (Minus): Nút này có chức năng bật hoặc tắt âm thanh khi đang hoạt động. Trong chế độ cài đặt các số, nút này có chức năng giảm số, cụ thể là giảm thời gian đo.
- Menu: Nút này mở ra giao diện menu của máy.
- Backlight (Plus): Cho phép bật đèn nền trên màn hình LCD trong vòng 7 giây theo cài đặt mặc định, nếu nhấn giữ trong 3 giây sẽ bật vĩnh viễn và có thể tắt chế độ đèn vĩnh viễn bằng cách này. Cũng như nút Audio (Minus), nút này cũng có chức năng đổi số cài đặt trong giao diện menu, ở đây là tăng số.
- Mode: Nút này hiển thị và cho phép lựa chọn các đơn vị đo máy có: mR/h, CPS – đếm trên mỗi giây,  $\mu\text{Sv/h}$ , Counts, CPM đếm trên mỗi phút,  $\mu\text{R/h}$ . [8]

Các giới hạn đo tương ứng với các đơn vị thể hiện trong bảng 2.1.

Bảng 2.1. Khoảng đo của từng đơn vị trong máy đo Radiation Alert Ranger®

Chế độ (đơn vị)	mR/h	$\mu\text{Sv/h}$	CPS	CPM	Counts
Giới hạn đo	0.001 - 100	0.01 - 1000	0 - 5000	0 – 350,000	9,990,000

Máy đo RAR có độ nhạy phát hiện đối với tia alpha có năng lượng tối thiểu 2 MeV, tia beta tối thiểu 0.40 keV, tia gamma tối thiểu 10 keV [8]. Máy đo RAR sử dụng nguồn phóng xạ để hiệu chỉnh đầu dò, nhằm tối ưu hóa hiệu suất đo cho đầu dò GM: Sulfur (S-35), Stronium (Sr-90), Phosphorus (P-32), Carbon (C-14), Iodine (I-131), Cobalt (Co-60), Cesium (Cs-137) và hạt Alpha. Do Cs-137 được tối ưu hóa đo đặc cho các mode đơn vị: mR/h,  $\mu\text{Sv/h}$ ,  $\mu\text{R/h}$ , nên sẽ không chính xác hoàn toàn khi sử dụng các nguồn khác ngoài Cs-137 để thực hiện đo đặc phóng xạ. Nếu muốn đạt được độ chính xác cao, thì ta cần phải hiệu chỉnh thủ công bằng nguồn phóng xạ tương ứng nguồn phóng xạ ta muốn đo. [8]

Ứng dụng của Máy đo RAR [8]:

- Phát hiện và đo độ nhiễm bản bề mặt
- Giám sát khả năng bị phơi chiếu với chất phóng xạ khi làm việc với các hạt nhân phóng xạ

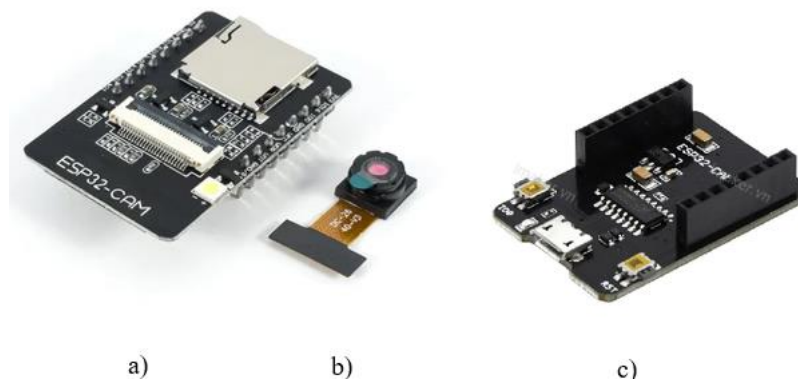
- Sàng lọc ô nhiễm môi trường (phân biệt bức xạ hồng và bức xạ có nguồn, có phóng xạ)
- Phát hiện khí hiếm và các hạt nhân phóng xạ năng lượng thấp khác

### 2.3. Module ESP32-CAM và phần mềm Arduino IDE

ESP32-CAM là sản phẩm của Espressif và có bán sẵn trên thị trường. ESP32-CAM là một bảng mạch được ứng dụng trong lĩnh vực mạng internet kết nối vạn vật (Internet of thing – IoT). Sản phẩm này có tích hợp khả năng kết nối mạng wifi, kết nối bluetooth, kết nối thông qua dây USB để có thể sử dụng được camera – phần chính của cả ESP32-CAM. Sản phẩm này còn có thể bật đèn hay đèn flash nhằm cung cấp thêm ánh sáng cho camera. Camera của module ESP32-CAM là máy ảnh OV2640 (hình 2.6). OV2640 là một máy ảnh thường thấy trong các ứng dụng Internet vạn vật (IoT), có thể mã hóa hình ảnh dạng JPEG với mức tiêu thụ điện năng thấp, phù hợp với module ESP32-CAM, khi module này chỉ sử dụng 5 Volt. [9]

Thông số kỹ thuật [9]:

- Module ESP32-CAM AI-Thinker
- Tốc độ truyền: 115200 bps
- Kết nối dòng điện 5 Volts
- Có khả năng kết nối Bluetooth 4.2 BR/EDR – tiêu chuẩn BLE, USB, Wifi.
- Máy ảnh loại OV2640, định dạng ảnh là JPEG
- Nhiệt độ an toàn từ  $-20^{\circ}\text{C}$  đến hơn  $85^{\circ}\text{C}$ , trọng lượng khoảng 20 g, kích thước  $27 \times 40.5 \times 4.5 \text{ mm}$



Hình 2.6. a) Module ESP32-CAM, b) Camera OV2640, c) bảng mạch

Module ESP32-CAM được lập trình trên Arduino IDE, một môi trường phát triển hay phần mềm lập trình viết code như các chương trình Pycharm, Matlab... Ở Arduino IDE thì sử dụng ngôn ngữ C++. Các tập tin lập trình trên Arduino IDE sẽ có đuôi tập tin là “.ino” không giống với những chương trình khác. Lập trình trên Arduino làm cho module có thể kết nối với mạng internet với bộ code đã có sẵn trên chương trình. Tuy nhiên code sẽ chỉ có thể được nạp vào module bằng kết nối có dây USB, việc này phục vụ cho chuyển đổi code và tải code lên module khi có sự thay đổi trong phần lập trình cho module [10]

## **2.4. Ngôn ngữ Python và lập trình giao diện trong Python**

Python được phát triển vào cuối những năm 1980 bởi Guido van Rossum, tại Viện Nghiên cứu Quốc gia về Toán học và Khoa học Máy tính ở Hà Lan. Python là ngôn ngữ kế thừa từ ABC, Modula 3, small talk và Algol-68. Python là ngôn ngữ lập trình hướng đối tượng. Các bước thực hiện sẽ xoay quanh đối tượng mà chúng ta đang muốn thực hiện. Ngôn ngữ này phổ biến vì câu lệnh đơn giản, dễ tiếp cận và hiện nay rất nhiều bài viết, video hướng dẫn trên mạng. Python chỉ là một ngôn ngữ, không như Matlab là một phần mềm có ngôn ngữ riêng, có sẵn các thư viện làm việc của riêng nó, thì ở Python phải cần có môi trường (hay environment) để có thể sử dụng. Một số môi trường phổ biến như: Arduino IDE, PyCharm, Visual Studio Code, Sublime Text 3, Spyder...

Python được ứng dụng cho việc: Lập trình và phát triển Website, Internet; Phát triển phần mềm – ứng dụng; Phân tích dữ liệu, trực quan dữ liệu...

Python có khá nhiều thư viện phục vụ cho tùy mục đích hay nhu cầu của lập trình viên. Trong đề tài này sẽ sử dụng các thư viện: Tkinter GUI, Numpy, Tesseract Engine – OCR, OpenCV-Python, Threading.

## **2.5. Các bộ thư viện trong python**

### **2.5.1. Bộ thư viện thiết kế giao diện người dùng Tkinter**

Tkinter là thư viện có các thành phần giúp người sử dụng tạo ra giao diện đồ họa theo ý muốn cá nhân. Thư viện này được áp dụng tạo giao diện ứng dụng, phần mềm. Do dễ lập trình và rất phổ biến nên có thể tìm được sự giúp đỡ từ các nhóm cộng đồng lập trình.

Mặc dù dễ lập trình nhưng để thành thực khá khó khăn. Khác với thư viện giao diện khác như PyQT hoặc GUI của Matlab, hai thư viện này có thể tương tác trước bằng cách kéo và thả (drag and drop) các công cụ được cho sẵn, sau đó mới tiến hành lập trình cho từng thành phần. Việc này giúp việc thiết kế giao diện dễ hình dung, dễ chỉnh sửa và có thể tạo ra một giao diện hoàn chỉnh, đẹp mắt nhanh. Dù vậy thì việc lập trình sẽ khó khăn hơn do sự phức tạp của các thành phần và các hoạt động của các chương trình cho sẵn không dễ nắm bắt.

Ở thư viện Tkinter, khó khăn trong việc thiết kế và chỉnh sửa là sau khi phải lập trình thì mới có kết quả. Mỗi khi viết xong một đoạn code cần phải chạy thử để xem giao diện hiện ra như thế nào. Dù vậy, thì cách viết code lại dễ dàng hơn do câu lệnh mà bộ thư viện này mang lại rất dễ hiểu và dễ thực hiện, không quá phức tạp về cách viết và chạy chương trình.

Trong Tkinter còn một nhánh thư viện mở rộng khác là Custom Tkinter. Đây là bộ thư viện có chức năng và cách viết tương tự như thư viện Tkinter gốc. Sự khác biệt là ở các câu lệnh sẽ có thêm từ “ctk” tức là “custom tkinter” nhằm phân biệt với bộ Tkinter gốc khi người sử dụng muốn dùng cả hai bộ Tkinter và Custom Tkinter cùng một lúc [11]. Custom Tkinter được thiết kế lại phần giao diện cho hiện đại hơn, đẹp mắt hơn so với bộ gốc. Cả hai bộ thư viện được sử dụng cùng lúc mà không gây ảnh hưởng lẫn nhau, việc này sẽ dễ cho một số câu lệnh bị thay đổi hoặc không có ở Custom Tkinter có thể dùng câu lệnh ở bộ Tkinter gốc. Tuy nhiên, không thể áp dụng câu lệnh của bộ này lên bộ kia, chỉ có thể dùng song song.

Chức năng của bộ Tkinter cung cấp bao gồm: Tạo giao diện; thêm các nút, các nút tương tác; cho phép thêm và hiển thị hình ảnh, video...

### **2.5.2. Bộ thư viện Numpy**

Numpy là bộ thư viện xử lý dữ liệu chủ yếu là các mảng (array) hoặc ma trận... Bộ thư viện này có chức năng là tính toán các mảng hay các ma trận đa chiều, đa kích thước với các hàm, các câu lệnh được tối ưu sẵn trong bộ thư viện [12].

Trong giới hạn đề tài này, thì bộ thư viện numpy được sử dụng để xử lý hình ảnh. Sau khi hình ảnh được đưa vào Python, nó sẽ được phân tích ra thành các ma

trận kích thước bằng kích thước hình ảnh gốc. Việc xử lý hình ảnh của bộ thư viện này thực chất là xử lý ma trận bằng các phép toán như: cộng; trừ; nhân; chia... Numpy còn có thể tạo hình ảnh từ ma trận.

### **2.5.3. Bộ thư viện Tesseract Engine**

Bộ thư viện Tesseract Engine, Pytesseract hay Tesseract Optical Character Recognition (Tesseract OCR) là những cách gọi khác nhau cùng chỉ một bộ thư viện Tesseract. Bộ thư viện này được tạo ra cho mục đích là công cụ nhận dạng các ký tự hay nhận dạng chữ, số ngoài ra còn có thể nhận vật thể, khuôn mặt. Công dụng nhận dạng vật thể có thể ứng dụng trong việc làm ra camera OV2640 quan sát kết hợp với module ESP32-CAM hoặc các loại camera khác. Trong đề tài này thì chỉ nằm trong việc nhận dạng số đọc của máy đo RAR. Bộ này nhận dạng dạng các ký tự từ hình ảnh, video hoặc thông qua stream bởi ESP32-CAM .

### **2.5.4. Bộ thư viện CSV**

Bộ thư viện CSV là một thư viện cho phép tạo nên một tập tin có đuôi là “.csv” đây là một tập tin có thể mở trên phần mềm Excel của Microsoft. So với tập tin excel hiện có “.xls” thì tập tin “.csv” có phần thua kém hơn. Tập tin này không thể sử dụng các hàm mà một tập tin excel bình thường có thể làm, cũng như bị thiếu đi các chức năng cần thiết, nhưng với sự phát triển của công nghệ cũng như internet thì các tập tin “.csv” này có thể chuyển đổi thành tập tin excel đơn thuần. Nhưng việc sử dụng thư viện CSV trong python để tạo ra một tập tin “.csv” thay vì tập tin excel là do các tập tin “.csv” dễ tạo ra và nhẹ hơn do thua kém tập tin excel về khả năng sử dụng hàm, câu lệnh tính toán, nên tập tin “.csv” được sử dụng cho mục đích ghi dữ liệu [13].

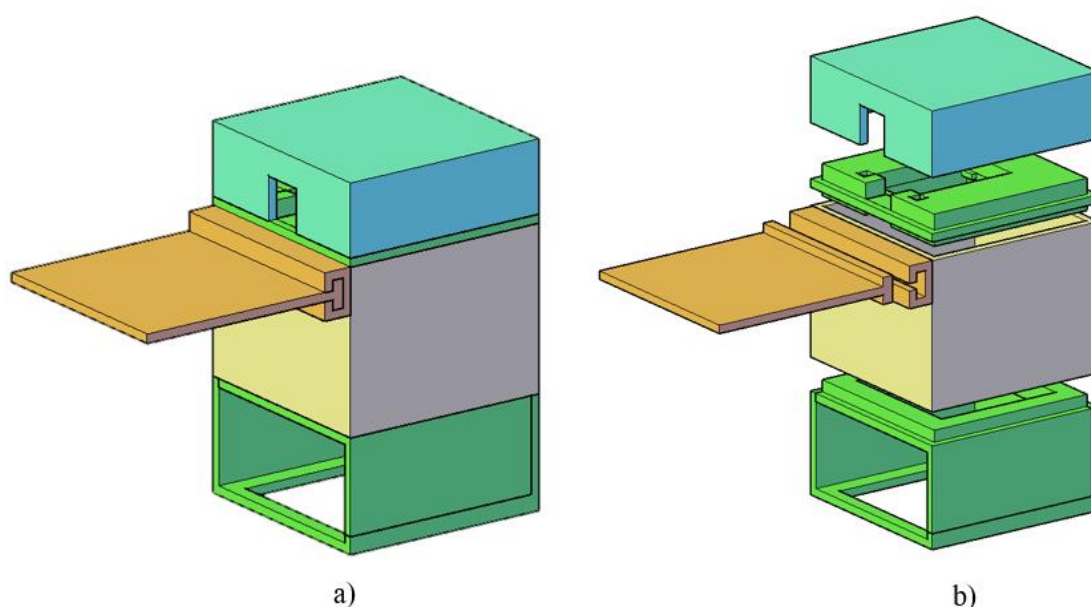
### **2.5.5. Bộ thư viện OpenCV2**

Là bộ thư viện dùng để xử lý hình ảnh. Bộ thư viện này có mặt ở các ngôn ngữ lập trình: Python, MATLAB, C++ và Java. Các kiểu xử lý ảnh như: Chuyển đổi tông màu ảnh RGB (Red-Green-Blue), Làm mờ - Blur, Lọc nhiễu, Đặt ngưỡng Threshold, Cắt cúp ảnh, Thay đổi kích thước ảnh,...

## CHƯƠNG 3. PHƯƠNG PHÁP LUẬN

### 3.1. Thiết kế phần cứng

Phần cứng bao gồm một module ESP32-CAM có tích hợp camera OV2640, nguồn điện 5V, và một bộ giá đỡ để cố định vị trí của camera và máy đo. Module ESP32-CAM và nguồn điện được mua từ các nhà cung cấp địa phương. Giá đỡ được tác giả phát triển như một phần của đề tài này, sau đó được gia công bằng phương pháp in 3D bởi một xưởng in địa phương.



Hình 3.1. a) Bộ giá đỡ khi được lắp ráp, b) Bộ giá đỡ khi tách rời

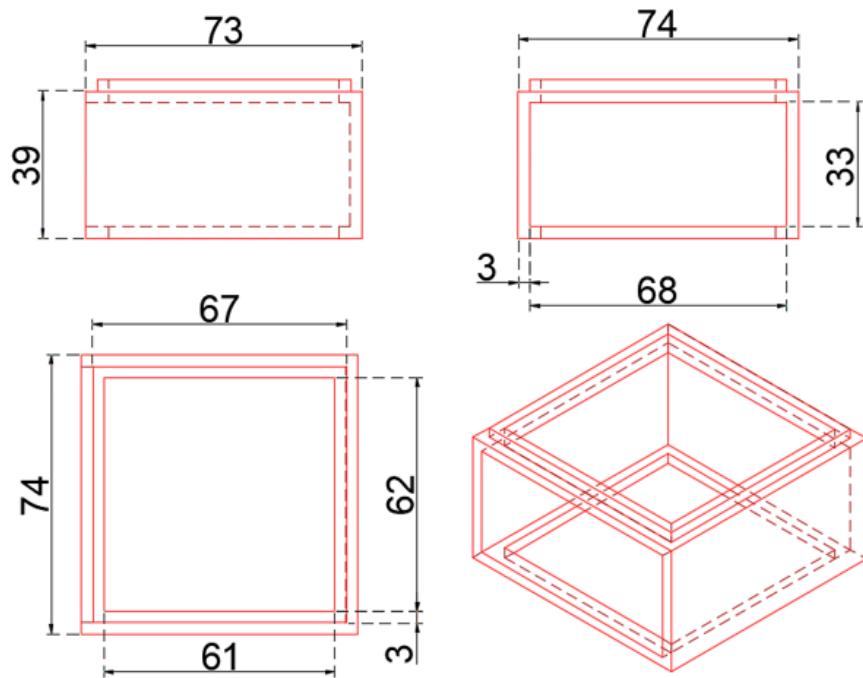
Giá đỡ có tác dụng cố định vị trí giữa ESP32-CAM và máy đo phóng xạ Radiation Alert Ranger ® (RAR). Giá đỡ được thiết kế trên phần mềm AutoCAD sao cho có thể tháo lắp dễ dàng lên máy đo RAR. Hình 3.1 là mô phỏng 3D của giá đỡ này. Nó được thiết kế thành 5 phần riêng biệt, có thể tách rời, và được kết nối với nhau thông qua các khớp nối hình khối chữ nhật nhô lên. Các phần này, theo thứ tự từ dưới lên, bao gồm:

- Tầng 1 là bộ phận gắn lên RAR bằng cách trượt vào khu vực màn hình của máy. Các kích thước của tầng 1 được cho trong hình 3.2.
- Tầng 2 là một bộ phận trung gian kết nối tầng 1 và tầng 3 lại, đồng thời là bộ phận tạo khoảng từ đầu camera tới màn hình LCD của máy đo. Khoảng cách này cần được cố định để hình ảnh thu được rõ nét. Các kích thước được trình

bày trong hình 3.3. Bên ngoài tầng 2 được gắn thêm một chân chống, được gắn vào tầng 2 thông qua một rãnh trượt. Chân chống này giúp cho giá đỡ có thể đứng vững kể cả khi gắn máy đo RAR vào. Hình 3.4 Trình bày kích thước chân chống

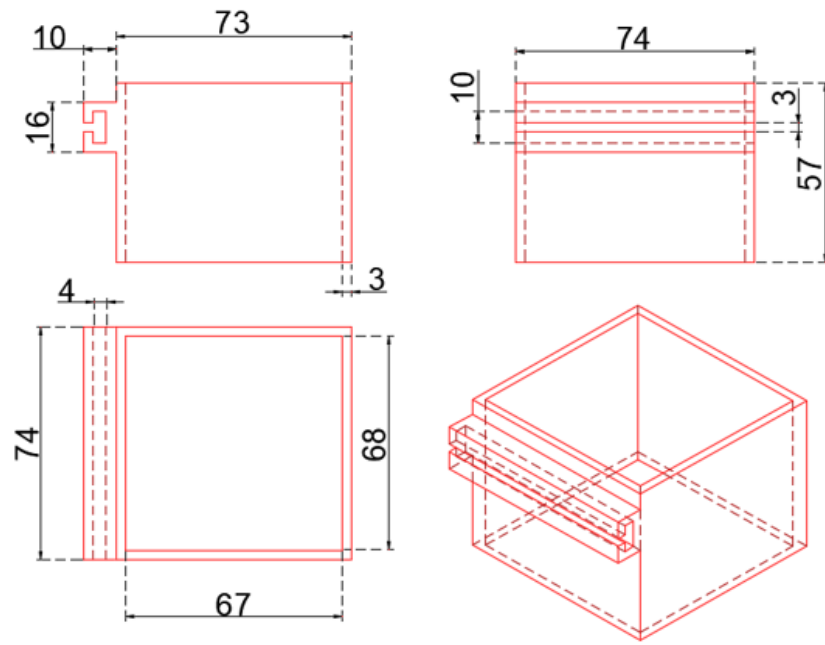
- Tầng 3 là bộ phận gắn ESP32-CAM và 2 dải đèn LED cũng cấp ánh sáng cho camera. Hình 3.5 trình bày kích thước, thiết kế của của tầng 3.
- Tầng 4 là nắp đậy giúp cố định cũng như bảo vệ camera. Các kích thước được trình bày trong Hình 3.6.

Do giá đỡ được thiết kế kín nên khi vận hành không có đủ ánh sáng chiếu vào màn hình máy đo. Vì vậy, trên tầng 3 của giá đỡ có gắn thêm 2 dải đèn LED, gồm 6 bóng LED trên mỗi dải. Nguồn điện cấp cho các dải LED là nguồn điện 5V lấy từ các chân cấp nguồn của module ESP32-CAM.



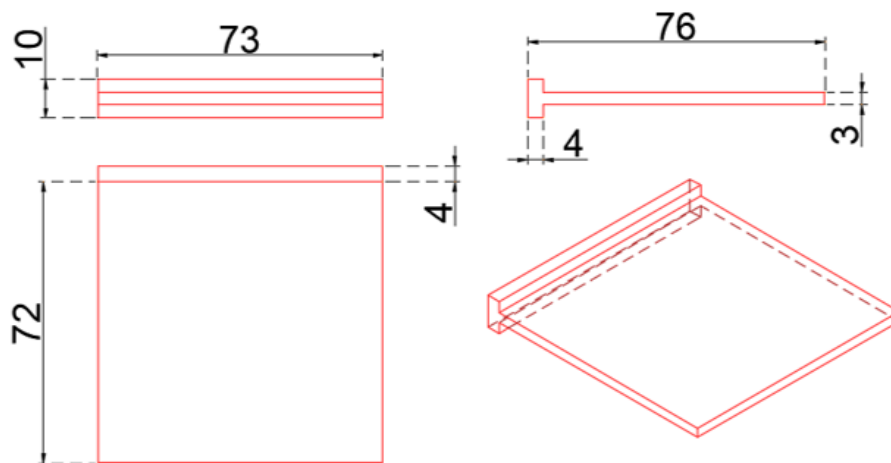
Giá đỡ – tầng 1 (mm)					
Cao	Dài × Rộng (ngoài)	Dài × Rộng (trong)	Dài × Rộng (lỗ trống trên và dưới)	Dài × Rộng (lỗ trống phía trước)	Dày
42	74 × 73	68 × 67	62 × 61	68 × 33	3

Hình 3.2. Bảng vẽ giá đỡ tầng 1



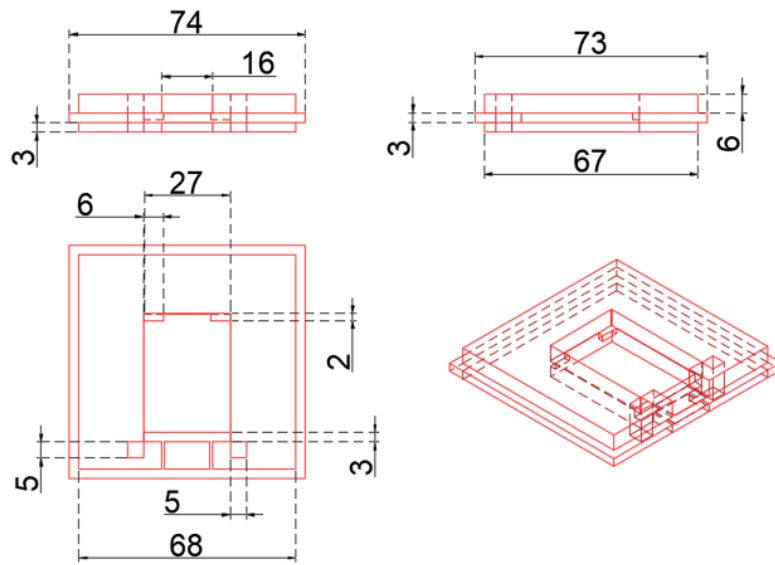
Giá đỡ - tầng 2 (mm)				
Cao	Dài × Rộng	Khe trượt (tổng quan)	Khe trượt (độ rộng)	Dày
57	74 × 73	74 × 16	10 × 4	3

Hình 3.3. Bản vẽ giá đỡ tầng 2



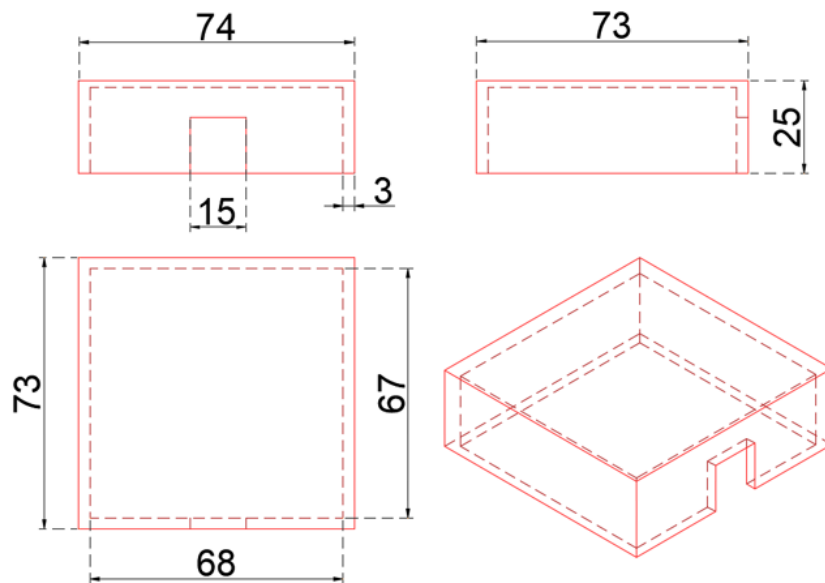
Chân chống (mm)			
Cao	Dài × Rộng	Bề dày chân chống	Bề dày thanh trượt
76	74 × 72	3	4

Hình 3.4. Bản vẽ chân chống



Giá đỡ - tầng 3 (mm)				Ngăn chứa Module ESP32-CAM (mm)			
Cao	Dài × Rộng	Dài × Rộng (trong)	Dày	Thanh đỡ module	Dài × Rộng	Độ rộng khe	Lỗ dẫn dây điện
12	74 × 73	68 × 67	3	2 × 6, 3 × 27	40 × 27	16 × 6	5 × 5

Hình 3.5. Bản vẽ giá đỡ tầng 3

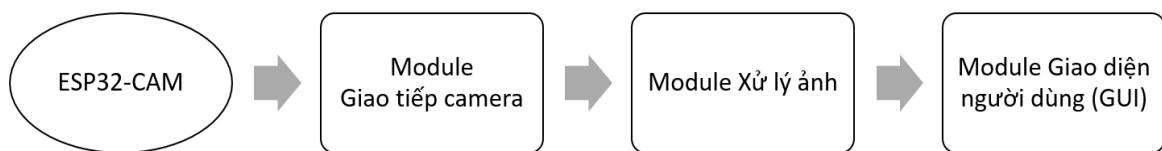


Giá đỡ - tầng 4 (mm)				
Cao	Dài × Rộng	Dài × Rộng (Trong)	Độ rộng khe	Dày
25	73 × 74	68 × 67	15 × 15	3

Hình 3.6. Bản vẽ giá đỡ tầng 4

### 3.2. Thiết kế phần mềm

Hình 3.7 trình bày cấu trúc của phần mềm được phát triển trong đề tài này. Phần mềm bao gồm 3 module chính: module giao tiếp camera, module xử lý ảnh, và module giao diện đồ họa người dùng (GUI). Module giao tiếp camera có nhiệm vụ nhận luồng hình ảnh của ESP32-CAM gửi lên, trích xuất các khung ảnh rồi gửi đến module xử lý ảnh. Tại đây, ảnh được tuần tự đi qua các phân đoạn gồm tiền xử lý ảnh, nhận diện ký tự quang học (OCR), và phân giải kết quả OCR. Kết quả sau đó được truyền đến module GUI phục vụ hiển thị hoặc mục đích lưu trữ. Các tiêu mục dưới đây sẽ lần lượt trình bày chi tiết về các module này.

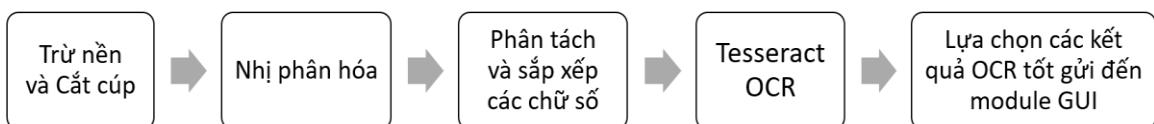


Hình 3.7. Cấu trúc của phần mềm

#### 3.2.1. Module giao tiếp camera

Module giao tiếp camera nhận hình ảnh liên tục theo thời gian thực (được gọi là luồng ảnh). Từ module ESP32-CAM, luồng ảnh này thu thập rất nhiều hình trong một giây. Tốc độ mặc định của camera là tạo ra 10 hình ảnh trên một giây (fps – frame per seconds). Các hình ảnh này được đẩy liên tục lên socket nên sẽ làm quá tải các module khác. Module giao tiếp camera có nhiệm vụ là giới hạn hình ảnh thu thập được bằng cách lấy mẫu trên luồng hình ảnh với tốc độ khoảng 2 hình mỗi giây. Tuần tự từng hình ảnh này sẽ được chuyển đến module tiếp theo để xử lý.

#### 3.2.2. Module xử lý ảnh



Hình 3.8. Cấu trúc của module xử lý ảnh

Module xử lý ảnh là trái tim của cả hệ thống phần mềm. Nó chịu trách nhiệm chuyển đổi số đọc dạng hình ảnh (hiển thị trên màn hình của máy đo RAR) thành số đọc dạng số bằng cách sử dụng thuật toán nhận diện ký tự quang học Tesseract OCR. Tuy nhiên, việc sử dụng trực tiếp hình ảnh gốc (từ module giao tiếp camera) để chạy thuật toán Tesseract OCR sẽ cho kết quả rất tệ vì ảnh đầu vào chứa nhiều chi tiết

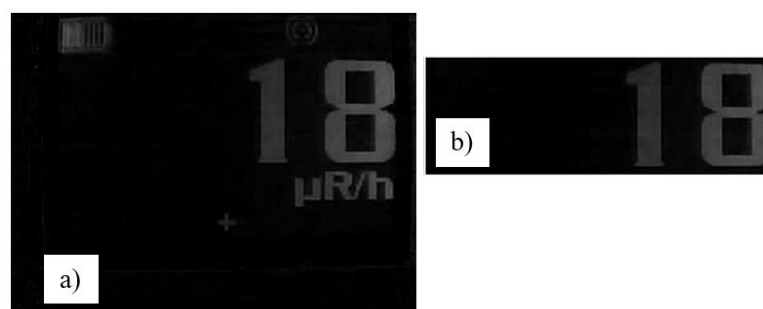
nhiều và có chất lượng kém. Do đó, hình ảnh gốc cần phải được tiền xử lý qua nhiều phân đoạn gồm: trừ nền (background subtract), cắt cúp (crop), nhị phân hóa (binarize), phân tách và tái sắp xếp các chữ số như hình 3.8. Ngoài ra để tăng độ chính xác cho module xử lý ảnh, một bước hậu xử lý được thực hiện để chỉ chấp nhận những kết quả OCR tốt được trả ra từ thuật toán Tesseract OCR.

- **Trừ nền và cắt cúp**



Hình 3.9. Ảnh nền

Ảnh nền (hình 3.9) là ảnh mà chỉ hiển thị màn hình LCD của máy đo RAR khi chưa hoạt động. Ảnh nền sẽ được chụp một lần duy nhất trong cùng điều kiện ánh sáng và vị trí của camera so với khi vận hành, sau đó được lưu trữ như một dữ liệu tiêu chuẩn để chạy phần mềm. Trừ nền là thao tác trừ ảnh của số đọc với ảnh nền nhằm loại bỏ phần hình ảnh cố định của màn hình LCD và phần khung của máy đọc, chỉ giữ lại phần hình ảnh động của các chữ số và một vài chi tiết khác (đơn vị đo, ký hiệu dung lượng pin,...). Minh họa ảnh của số đọc sau khi trừ nền ở hình 3.10 a.



Hình 3.10. a) Ảnh thu được sau khi trừ nền, b) Ảnh cắt cúp từ ảnh đã trừ nền

Tuy việc trừ nền đã loại bỏ rất nhiều các chi tiết không cần thiết, nhưng một số chi tiết “động” vẫn còn, chẳng hạn như đơn vị đo, ký hiệu dung lượng pin, biểu tượng kết nối bluetooth, biểu tượng bật/tắt âm thanh,... Các chi tiết này là không cần thiết khi thực hiện OCR và có khả năng gây ra lỗi. Do đó, cần cắt cúp (crop) ảnh để khu trú vào vị trí của các chữ số. Do vùng hiển thị các chữ số trên màn hình của máy đo là cố định nên ta chỉ cần xem xét một vài hình ảnh tiêu biểu để tìm ra vị trí của các đường cắt một cách thủ công. Vị trí của các đường cắt này được lưu trữ như một tham số tiêu chuẩn của thuật toán, và được sử dụng cho tất cả các ảnh khác của số đọc trong quá trình vận hành. Hình 3.10 b minh họa ảnh sau cắt cúp.

Việc trừ nền và cắt cúp ảnh được triển khai chủ yếu bằng các hàm trong thư viện numpy. Các hàm được trình bày trong hình 3.11.

```
def isub_image(frame, img_bckg):  
    i_bckg = img_bckg[:, :, 1]  
    i1 = i_bckg.astype(np.int16)  
    i = frame[:, :, 1]  
    i2 = i.astype(np.int16)  
    isub = i1 - i2  
    isub[isub < 0] = 0  
    isub2 = isub.astype(np.uint8)  
    cropisub = isub2[80:150, 60:300]  
    return cropisub
```

Hình 3.11. Hàm sử dụng cho mục đích trừ nền và cắt cúp

- **Nhị phân hóa**

Thao tác trừ nền và cắt cúp tuy giúp loại bỏ các chi tiết thừa và khu trú vào khu vực các chữ số, nhưng nó cũng làm suy giảm độ tương phản của các chữ số so với nền. Nhị phân hóa sẽ giúp phân đoạn ảnh thành hai vùng riêng biệt (vùng chữ số và vùng nền) với độ tương phản tối đa.

Để nhị phân hóa, cần đặt ra một mức ngưỡng (threshold) trên dải 256 giá trị, tối thiểu là 0 và tối đa là 255. Độ xám của từng pixel sẽ được so sánh với ngưỡng này. Nếu độ xám nhỏ hơn ngưỡng, pixel đó được gán độ xám mới bằng 0. Ngược lại, nếu độ xám lớn hơn hoặc bằng ngưỡng, pixel đó được gán độ xám mới bằng 255. Kết quả

là bức ảnh ban đầu, với các pixel có độ xám rất đa dạng trải dài từ 0 đến 255, đã được biến thành bức ảnh mới mà trong đó độ xám của các pixel chỉ có thể là một trong hai giá trị là 0 hoặc 255, tương ứng với màu đen hoặc trắng trên ảnh. Vấn đề quan trọng là cần chọn ngưỡng sao cho các pixel thuộc vùng chữ số sẽ được gán giá trị mới là 255, còn các pixel thuộc vùng nền sẽ được gán giá trị mới bằng 0 để tách chúng thành hai vùng ảnh có độ tương phản tối đa. Trong đề tài này, ngưỡng tối ưu được chọn là 25 sau nhiều lần thử đặt các ngưỡng khác nhau. Hình 3.12 a minh họa ảnh sau khi được nhị phân hóa.



Hình 3.12. a) Ảnh sau nhị phân hóa, b) Ảnh nhị phân hóa được mở rộng biên

Ảnh sau khi nhị phân hóa đã có độ tương phản tốt hơn. Tuy nhiên, quá trình này cũng làm tăng độ tương phản của nhiễu xung quanh các ký tự. Các nhiễu này cần được xử lý để tránh làm dính các ký tự lại với nhau dẫn đến thất bại khi đưa vào Tesseract OCR. Kỹ thuật Morphology được sử dụng trong trường hợp này. Trước khi có thể áp dụng được các thuật toán Morphology, cần phải mở rộng biên của ảnh vì thao tác cắt cúp trước đó đã làm làm ảnh nhỏ hơn so với kích thước ban đầu. Việc này chỉ đơn giản là “gắn” thêm các mảng hình ảnh giống với nền (màu đen) vào các cạnh của ảnh sau khi nhị phân hóa. Ảnh sau khi mở rộng biên được minh họa trên hình 3.12 b.

Thuật toán Closing trong Morphology sẽ được sử dụng trên ảnh đã mở rộng biên nhằm loại bỏ các chi tiết nhiễu nhỏ và làm mượt các cạnh của chữ số. Thuật toán này là tổ hợp của thuật toán Erosion (co) sau đó là Dilation (giãn). Dilation và Erosion là 2 thuật toán xử lý ảnh nhị phân dựa trên phương pháp tích chập (convolution) ảnh với một kernel. Dilation có khả năng tăng số lượng pixel trắng trong ảnh, tức là “làm to” các ký tự lên (như Hình 3.13 a). Erosion thì ngược lại, nó sẽ có khả năng làm giảm số lượng các pixel trắng, tức là “làm nhỏ” các ký tự lại (như trong Hình 3.13 b).

Thuật toán Dilation được thực hiện với kernel có kích thước 2x1 và số lần lặp là 2. Erosion được thực hiện với kernel có kích thước 2x2 và số lần lặp là 5. Đây là những lựa chọn tối ưu sau nhiều lần thử nghiệm. Hình 3.14 là kết quả sau khi áp dụng thuật toán Closing.



Hình 3.13. a) Hình ảnh sau khi dùng thuật toán Dilation, b) Hình ảnh sau khi dùng thuật toán Erosion



Hình 3.14. a) Ảnh sau khi nhị phân hóa và mở rộng biên, b) Ảnh sau khi dùng thuật toán Closing.

Các thuật toán nhị phân hóa và Morphology được triển khai bằng cách sử dụng các hàm có sẵn trong thư viện OpenCV-Python như trong hình 3.15

```
def threshold_image(cropisub):
    ret, th1 = cv2.threshold(cropisub, thresh: 20, maxval: 255, cv2.THRESH_BINARY)
    d = np.zeros(shape: [90, 260], np.uint8)
    d[5:75, 10:250] = th1

    dilation = cv2.dilate(d, np.ones(shape: (2, 1), np.uint8), iterations=2)
    erosion = cv2.erode(dilation, np.ones(shape: (2, 2), np.uint8), iterations=5)
    return erosion
```

Hình 3.15. Thuật toán nhị phân hóa (threshold) và hàm sử dụng trong morphology (dilation, erosion)

- **Phân tách và tái sắp xếp các chữ số**



Hình 3.16. a) Các số hiển thị có khoảng cách không đều, b) Các số hiển thị có khoảng cách ba sau khi phân tách và tái sắp xếp

Trên màn hình của máy đo RAR, các ký tự được hiển thị ở vị trí cố định. Trong nhiều trường hợp, khoảng cách giữa các chữ số có thể chênh lệch rất lớn như minh họa trên Hình 3.16 a. Vấn đề này ảnh hưởng xấu đến kết quả nhận diện ký tự quang học. Vì vậy, phân tách, sau đó tổ hợp các chữ số lại để đồng nhất khoảng cách giữa chúng, như minh họa trên hình 3.16 b là phân đoạn cần thiết trong bước tiền xử lý.

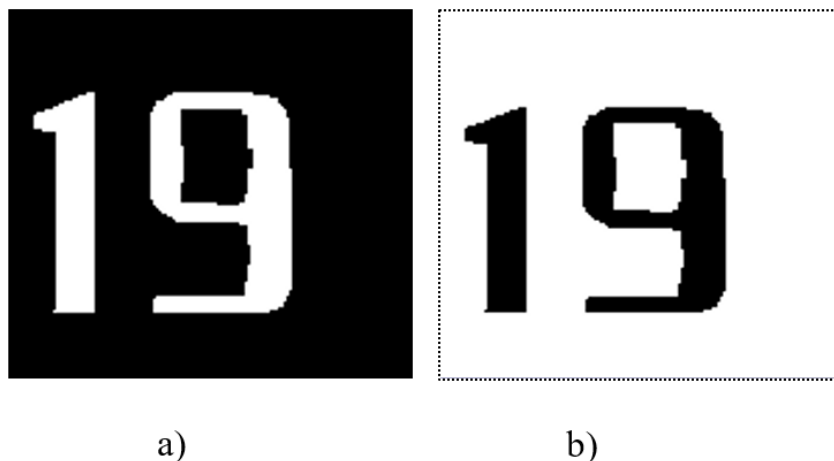
Phân đoạn này gồm 2 bước: xác định khung bao của mỗi chữ số và tái sắp xếp chúng. Bước 1 được bắt đầu bằng việc sử dụng hàm có sẵn trong thư viện OpenCV-Python để tìm đường bao của mỗi chữ số như trong hình 3.17. Dựa vào đường bao này để tạo ra khung bao hình chữ nhật của mỗi chữ số. Ở bước 2, khung bao sẽ được sử dụng để cắt các chữ số ra khỏi bức ảnh, sau đó sắp xếp chúng lại theo thứ tự. Trong lúc sắp xếp, khoảng cách giữa các ký tự có thể được chọn sao cho chúng cách đều nhau.



Hình 3.17. Hình ảnh bao gồm các đường Contour vẽ quanh các ký tự trong ảnh

Đến đây thì chất lượng ảnh đã tốt lên rất nhiều và sẵn sàng đưa vào Tesseract OCR. Tuy nhiên, theo khuyến cáo thì hình ảnh đưa vào Tesseract OCR nên có dạng nền màu trắng và các ký tự màu đen, tức là ngược lại với ảnh hiện có sau phân đoạn phân tách và tái sắp xếp các chữ số như hình 2.18 a. Do đó, cần thực hiện thêm thao

tác đảo ngược (invert) ảnh để các pixel có độ xám 255 trở thành 0 và ngược lại (hình 3.18 b).



Hình 3.18. a) Ảnh sau khi phân tách,  
b) Ảnh phân tách sau khi đảo ngược giá trị pixel

- **Tesseract OCR**

Ảnh sau khi được tiền xử lý sẽ được truyền vào thuật toán Tesseract OCR. Các thông số thiết lập cho thuật toán bao gồm: (1) Loại ngôn ngữ (Lang), cho phép nhận dạng các đoạn văn bản với các ngôn ngữ đã có trong bộ thư viện Pytesseract; (2) Tùy chỉnh phân đoạn (Page Segmentation Modes - PSM), cho phép tùy chọn cách thức phân đoạn ký tự, giúp cải thiện độ chính xác kết quả OCR. (3) Tùy chọn dữ liệu đầu ra (Output), cho phép lựa chọn các dữ liệu trả về sau khi OCR. Đặc biệt, để phù hợp cho việc nhận diện ký tự dạng số, một bộ ngôn ngữ đã được huấn luyện dành riêng cho ký tự dạng số được sử dụng. Bộ ngôn ngữ này được tạo ra bởi Shreeshrii [14] bằng cách huấn luyện Tesseract trên bộ dữ liệu gồm 8704 ảnh.

Thuật toán Tesseract OCR sẽ trả về kết quả OCR bao gồm giá trị nhận diện được (OCR value) và độ tự tin của thuật toán đối với giá trị được nhận diện đó (confident factor). Chỉ những kết quả với độ tự tin tối thiểu 0.9 mới được chấp nhận và chuyển tới module GUI.

### 3.2.3. Module giao diện đồ họa người dùng (GUI)



Hình 3.19. Giao diện phần mềm “RadOCR” được lập trình bằng thư viện tkinter và custom\_tkinter

Module này nhận kết quả OCR được chấp nhận từ module xử lý ảnh để phục vụ mục đích hiển thị và lưu trữ. Giao diện người dùng trong đề tài được đặt tên là RadOCR. Nó được phát triển trên 2 bộ thư viện là Tkinter và CustomTkinter [11]. Hình 3.19 là giao diện của RadOCR. Giao diện được thiết kế đơn giản với các thành phần sau:

- Hai khung hiển thị “Original image” để hiển thị ảnh gốc và “OCR result image” để hiển thị hình ảnh kèm theo theo kết quả OCR. Cả 2 khung hình đều hiển thị liên tục theo thời gian thực.
- Hai hộp văn bản “Value detected” chứa giá trị OCR và “Confident” chứa độ tự tin của thuật toán.

- Các nút bấm: Start – khởi động quá trình, Stop – tạm dừng quá trình, Exit – thoát chương trình, Unit – Hiển thị đơn vị, Export – tải xuống tập tin CSV lưu kết quả đo đặc phóng xạ.

Ngoài chức năng hiển thị kết quả OCR và điều khiển hoạt động của toàn bộ phần mềm, module GUI còn có chức năng thu thập các giá trị OCR kèm theo mốc thời gian rồi lưu trữ chúng vào một tập tin CSV. Tập tin này dễ dàng được tải xuống sau khi quá trình đo đặc kết thúc.

### **3.2.4. Quy trình hoạt động của phần mềm**

Sau khi khởi động phần mềm, quá trình đọc ảnh từ máy đo RAR cũng như các bước xử lý vẫn chưa bắt đầu. Để kích hoạt quá trình này, người dùng phải nhấn vào nút “Start”. Khi đó, ảnh chụp màn hình máy đo RAR sẽ được ESP32-CAM ghi nhận, chuyển cho module giao tiếp camera, rồi module xử lý ảnh, và cuối cùng là module GUI. Quá trình này lặp đi lặp lại với tần suất 1 lần mỗi giây, tương đương với tần suất cập nhật màn hình của máy đo RAR. Do đó, các kết quả hiển thị trên giao diện phần mềm là các số đọc của máy đo trong thời gian thực. Ngoài ra, các kết quả này cũng được lưu trữ trong tập tin CSV kèm theo mốc thời gian tương ứng.

Để dừng quá trình, người dùng phải nhấn nút “Stop”. Khi đó, module giao tiếp camera sẽ ngừng nhận luồng hình ảnh gửi lên từ ESP32-CAM, chương trình chấm dứt việc tuần tự đưa ảnh qua các module, file CSV chứa kết quả đo được kết xuất. Để tải file về thư mục cụ thể trên máy tính, người dùng nhấn nút “Export”, sau đó chọn thư mục cần lưu file.

## **3.3. Phương pháp chạy thử và đánh giá**

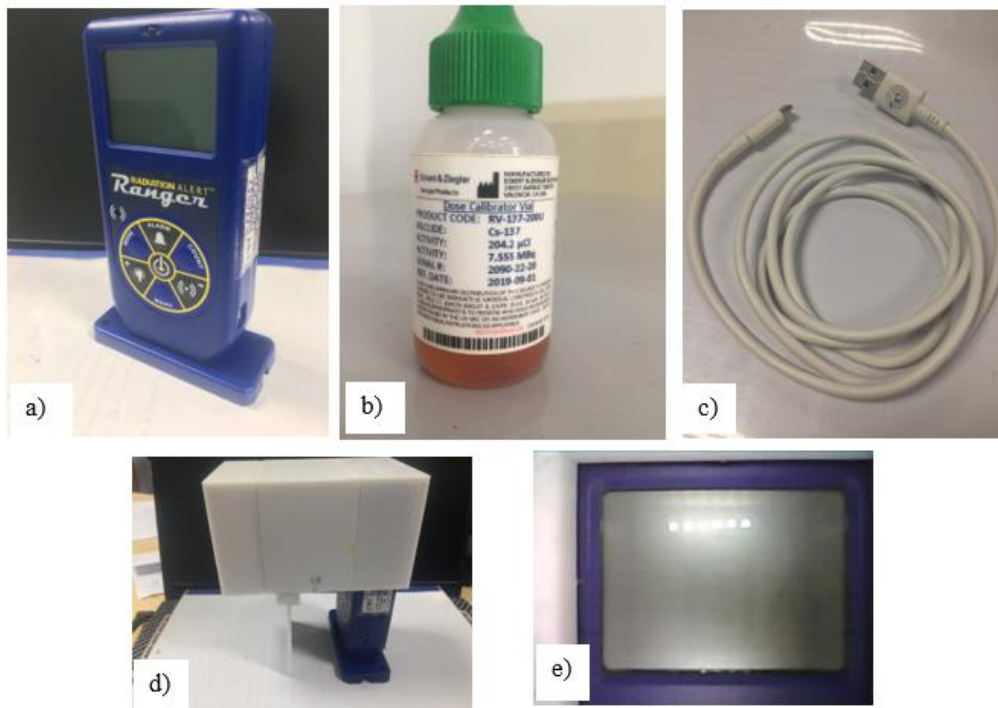
Phần này trình bày về phương pháp chạy thử RadOCR và cách thức đánh giá độ chính xác của phần mềm. Các nội dung trình bày bao gồm: máy đo và vật liệu, tính toán cỡ mẫu, các tham số thực nghiệm và phương pháp lấy mẫu; bố trí thực nghiệm; thu thập, xử lý số liệu và tính toán độ chính xác.

### **3.3.1. Thiết bị và vật liệu**

Các thiết bị sử dụng được thể hiện trong hình 3.20:

- Máy đo bức xạ cầm tay Radiation Alert Ranger®

- Máy tính để chạy phần mềm RadOCR có cấu hình: Ổ đĩa 512 GB; RAM 8GB; Bộ xử lý trung tâm – CPU Intel®: i3-1115G4; Bộ xử lý đồ họa – GPU Intel®: UHD Graphics; Hệ điều hành Windows 10 – 64 bit.
- Cáp sạc Micro – USB dùng cấp nguồn cho camera, nguồn điện 5V
- Ảnh nền, chụp bằng module ESP32-CAM sau khi gắn giá đỡ lên máy đo RAR.
- Nguồn phóng xạ Cs-137, hoạt độ 180.61  $\mu\text{Ci}$  (01-02-2024)
- Thiết bị RadOCR.



Hình 3.20. a) Máy đo RAR, b) Nguồn phóng xạ Cs-137, c) Cáp sạc type micro-USB, d) Thiết bị OCR, e) Ảnh nền

### 3.3.2. Chọn mẫu và các tham số thực nghiệm

Cỡ mẫu ( $n$ ) được tính toán theo công thức [15],

$$n = \frac{t_{\alpha}^2 \times p(1 - p) \times N}{(N - 1) \times \alpha^2 + t_{\alpha}^2 \times p(1 - p)} \quad (3.1)$$

Trong đó,  $N$  là kích thước đám đông,  $p$  là tỉ lệ xuất hiện trong đám đông của đặc trưng cần khảo sát,  $\alpha$  là mức nghĩa thống kê,  $t_{\alpha}$  là giá trị của phân bố chuẩn tương ứng với khoảng tự tin (confident interval,  $CI = 1 - \alpha$ ). Trong khảo sát này, mức ý nghĩa thống kê được chọn là  $\alpha = 0.05$ . Khi đó,  $t_{0.05} = 1.96$ . Đặc trưng cần khảo sát là độ chính xác của phần mềm trong việc trích xuất số đọc dạng ảnh thành số đọc

dạng số. Do tỉ lệ của đặc trưng này chưa biết trước nên được chọn là 0.5. Khi đó, cỡ mẫu  $n$  chỉ còn phụ thuộc vào kích thước đám đông  $N$ .

Trường hợp các mode đo trả về số đọc là dạng số nguyên (không chứa dấu chấm thập phân) như Counts, CPM, màn hình LCD của máy đo RAR hiển tối đa 4 chữ số, tương ứng với các giá trị từ 0 đến 9999. Tổng cộng có 10000 giá trị khả dĩ. Đây chính là kích thước của đám đông trong trường hợp này. Khi đó, cỡ mẫu tối thiểu cần khảo sát là 370. Để tiện tính toán, cỡ mẫu được lấy tròn là 400. Phương pháp chọn mẫu hệ thống (systematic sampling) được sử dụng để đảm bảo mẫu được chọn có tính chất ngẫu nhiên độc lập và phân phối đồng nhất (independent and identical distribution – IID). Cần chọn 400 hình ảnh chứa 400 số đọc khác nhau phân bố rải rác trong tổng số 10000 số đọc khả dĩ. Vì máy đo RAR cập nhật màn hình sau mỗi 1 giây, nên ta chỉ có thể thu được 400 số đọc khác nhau sau ít nhất 400 giây (~ 7 phút). Để tạo ra các số đọc khác nhau, mode Counts được sử dụng để khảo sát vì số đọc là tổng lũy tiến tăng dần. Trong thời gian 400 giây, số đọc phải tăng từ 0 đến 10000. Do đó, tốc độ đếm tại máy đọc phải đạt tối thiểu là

$$r = \frac{10,000}{400} = 25 \text{ cps} \quad (3.2)$$

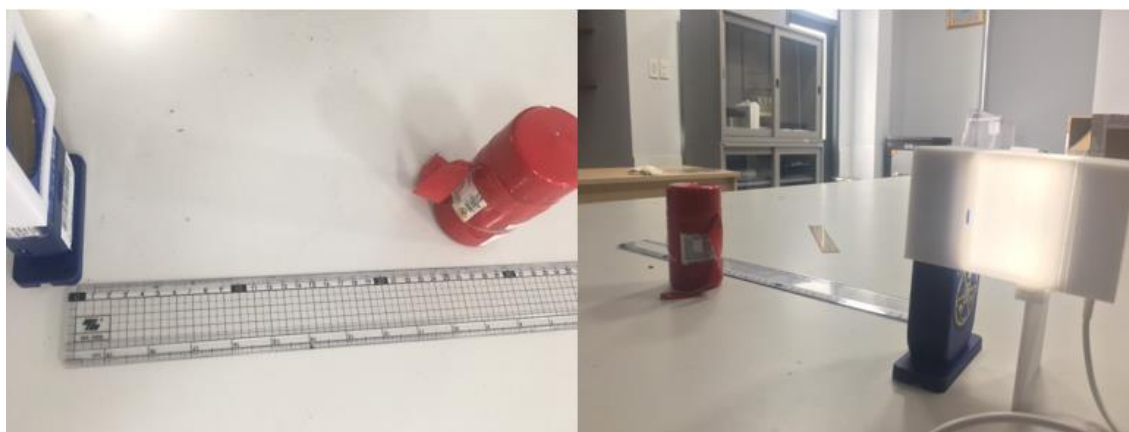
Điều này có thể đạt được bằng cách sử dụng nguồn phóng xạ Cs-137 cùng với việc hiệu chỉnh khoảng cách từ nguồn đến máy đo cho phù hợp, hoặc sử dụng thêm vật liệu che chắn.

Trường hợp các mode đo trả về số đọc là dạng số thập phân (có chứa dấu chấm thập phân) như  $\mu Sv/h$ , CPS, màn hình LCD của máy đo RAR còn có thể hiển thị dấu chấm thập phân ở một trong 3 vị trí nằm giữa 4 chữ số. Do đó, có tổng cộng 30000 giá trị khả dĩ. Đây chính là kích thước của đám đông trong trường hợp này. Khi đó, cỡ mẫu tối thiểu cần khảo sát là 380. Tất cả các mode trong trường hợp này đều trả về số đọc có giá trị tức thời nên việc chọn mẫu thỏa mãn IID là rất khó khăn. Do đó, phương pháp chọn mẫu thuận tiện (convenience sampling) được sử dụng.

### 3.3.3. Bố trí thực nghiệm

Giá đỡ có gắn module ESP32-CAM được gắn vào máy đo RAR. Cấp nguồn cho máy đo. Nguồn Cs-137 được đặt đối diện với máy đo. Mỗi lần chạy thử sẽ thực hiện

trong vòng 10 phút. Để xác định được tốc độ đếm theo yêu cầu là 25 cps, sử dụng mode CPS của máy đo RAR để hiệu chỉnh khoảng cách từ máy đo đến nguồn nhằm đạt yêu cầu này. Thực tế thấy được khoảng cách từ nguồn tới máy đo là khoảng 260 – 270 mm (Hình 3.21).



Hình 3.21. Bố trí thực nghiệm

#### 3.3.4. Thu thập, xử lý số liệu và tính toán độ chính xác

Trong chế độ chạy thử nghiệm, phần mềm RadOCR được xây dựng thêm tính năng lưu lại tất cả các ảnh kết quả OCR vào một thư mục trong suốt quá trình chạy. Sau mỗi lần chạy, các dữ liệu hình ảnh này sẽ được trích xuất dưới dạng một bộ gồm nhiều ảnh. Mỗi bộ ảnh sẽ được lọc thủ công để đảm bảo hai điều kiện: (i) tổng số ảnh trong bộ không nhỏ hơn kích thước mẫu tối thiểu, (ii) không có hai ảnh nào có số đọc hoàn toàn giống nhau trong cùng một bộ.

Mỗi bộ ảnh sau khi lọc sẽ được phân tích thủ công lần lượt từng ảnh bằng cách so sánh giữa số đọc hiển thị trên ảnh và giá trị OCR tìm được. Trường hợp hai số này bằng nhau, ảnh tương ứng được phân vào tập hợp các trường hợp đúng. Ngược lại, nó sẽ bị phân loại vào tập hợp các trường hợp sai. Gọi  $s$  là tổng số các trường hợp sai, khi đó tỉ lệ sai sót của phần mềm ( $e$ ) có thể được tính bằng công thức

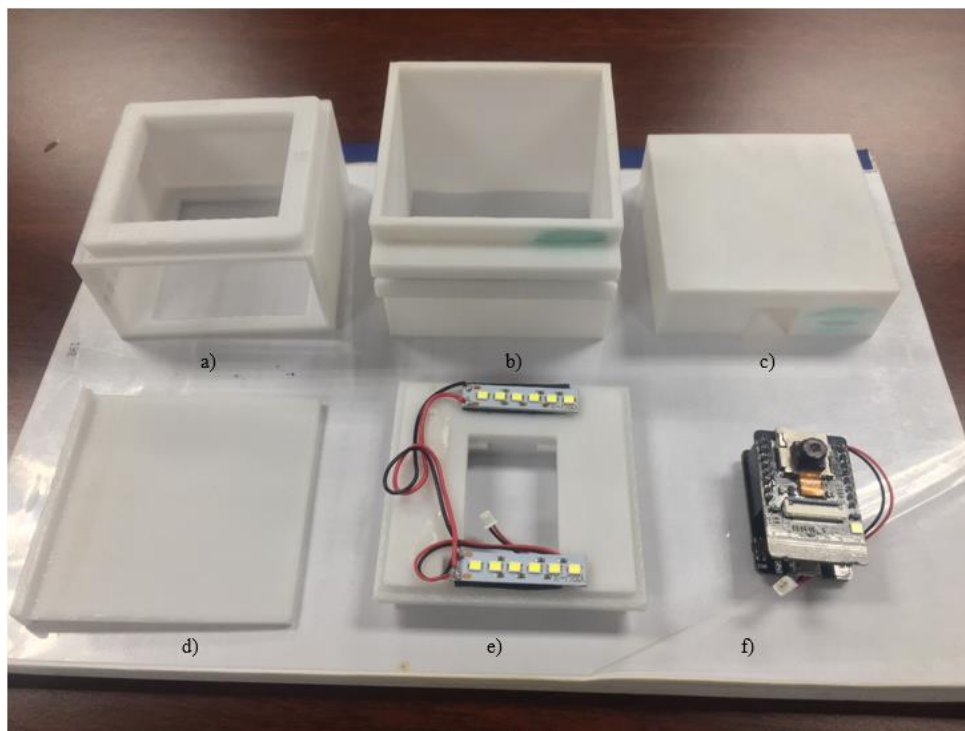
$$e = \frac{s}{T} \times 100\% \quad (3.3)$$

Trong đó,  $T$  là tổng số ảnh có trong bộ ảnh (sau lọc). Khi đó, độ chính xác của phần mềm ( $P$ ) có thể được tính bằng công thức

$$P = 100\% - e \quad (3.4)$$

## CHƯƠNG 4. KẾT QUẢ VÀ BÀN LUẬN

### 4.1. Thiết bị RadOCR



Hình 4.1. Thiết bị RadOCR được tháo rời: a) tầng 1, b) tầng 2, c) tầng 4 – nắp đậy, d) chân chống, e) tầng 3 – trang bị thêm đèn, f) module ESP32-CAM

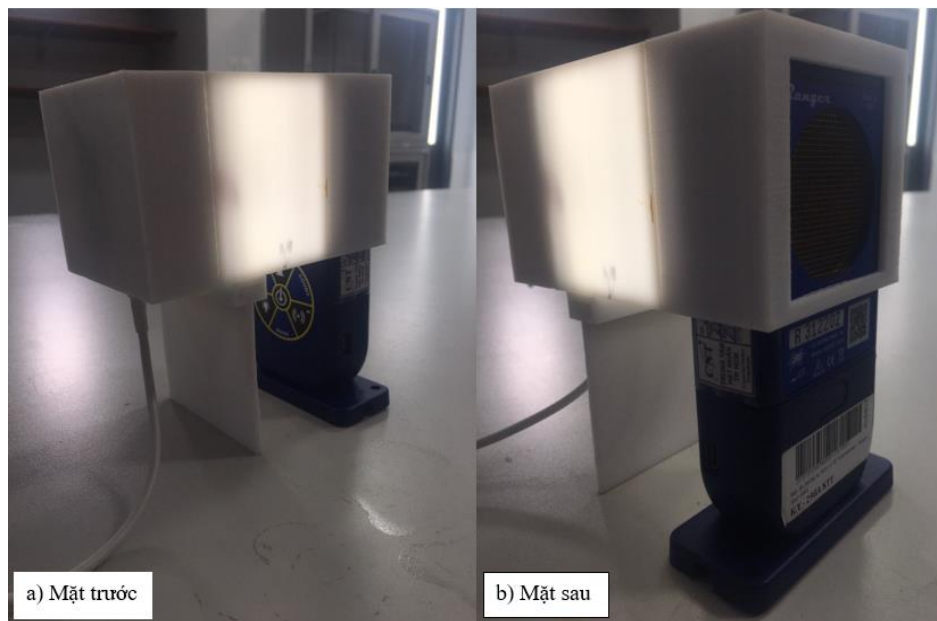
Hình 4.1 là thiết bị RadOCR đã được tháo rời các bộ phận. Phần giá đỡ được gia công bằng phương pháp in 3D trên chất liệu nhựa nhiệt dẻo acrylonitrile butadiene styrene (ABS). Sản phẩm in sau gia công đáp ứng tốt các thông số so với bản thiết kế. Các tầng khớp nối với nhau tốt và chặt chẽ. Hình 4.2 là thiết bị RadOCR đã lắp ráp hoàn chỉnh và được gắn vào máy đo RAR.

Ban đầu khi thiết kế, bộ giá đỡ sẽ không được lắp thêm đèn do đã có đèn flash tích hợp trên mạch của module ESP32-CAM có thể tận dụng để chiếu sáng màn hình LCD của máy đo. Tuy nhiên, khi chạy thử, đèn flash này gây lóa trên màn hình LCD (Hình 4.3). Điều này làm ảnh hưởng tới chất lượng hình ảnh thu được. Bằng cách sử dụng thêm hai dải đèn LED, vấn đề được giải quyết, đồng thời cho khả năng chiếu sáng tốt hơn.

Với mục tiêu là làm ra một bộ phụ kiện cho máy đo RAR, nên thiết bị RadOCR không thiết kế để can thiệp vào bên trong phần cứng của máy đo RAR. Thiết kế thiết bị RadOCR dễ dàng tháo, lắp và thuận tiện di chuyển. Thiết bị có module ESP32-

CAM được tích hợp kết nối mạng không dây như Wifi, mạng di động rất thuận tiện trong kết nối mạng. Chi phí để sản xuất bộ thiết bị thấp.

Module ESP32-CAM là loại giá rẻ nên có đi kèm một vài hạn chế: hình ảnh có độ phân giải thấp, cần phải bảo quản ở những nơi ít bụi bặm, vệ sinh ống kính trước khi đo. Do ống kính có tiêu cự dài làm cho thiết kế giá đỡ của RadOCR trở nên cồng kềnh, nên khi đặt máy đo RAR trên một bề mặt cần phải gắn chân chống đã thiết kế kèm theo bộ RadOCR giữ cho cả bộ máy đo RAR và máy đo RadOCR đứng vững.



Hình 4.2 Thiết bị RadOCR trang bị lên máy đo RAR

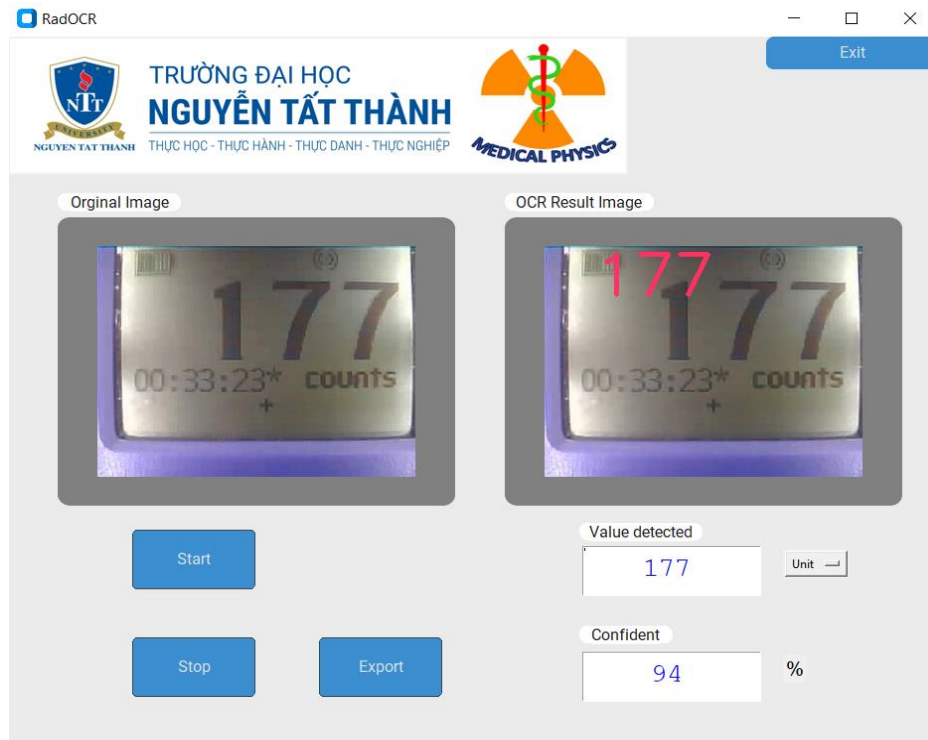


Hình 4.3. Ảnh nền xuất hiện một đốm sáng khi sử dụng đèn flash của module ESP32-CAM

#### 4.2. Phần mềm RadOCR

Hình 4.4 là giao diện của phần mềm RadOCR khi đang làm việc. Các thành phần trên giao diện và các tính năng hoạt động theo đúng ý tưởng thiết kế. Nút Start

sẽ kích hoạt thực hiện các module bên trong chương trình. Hai hộp văn bản “Value detected” và “Confident” hiển thị kết quả OCR cùng lúc với hai khung “Original image” và “OCR Result”.



Hình 4.4. Giao diện RadOCR khi hoạt động

Trong điều kiện kết nối WIFI ổn định, hình ảnh và kết quả OCR được cập nhật tốt theo thời gian thực, không bị đống hoặc mất hình. Trong suốt quá trình vận hành thử nghiệm, phần mềm hoạt động ổn định và chưa phát hiện các lỗi dẫn đến làm ngừng phần mềm đột ngột. Vấn đề quan trọng cần phải lưu ý để đảm bảo hệ thống chạy ổn định là cung cấp một kết nối WIFI đủ tin cậy giữa máy tính chạy phần mềm với module ESP32-CAM.

Một điểm bất tiện trong thuật toán xử lý ảnh là cần dùng ảnh nền của màn hình khi máy đo chưa vận hành. Ảnh này phải được chụp trong cùng điều kiện ánh sáng và vị trí camera so với lúc vận hành. Do đó, khi tháo máy đo hoặc tháo module ESP32-CAM ra khỏi giá đỡ, ta cần phải chụp lại ảnh nền gây bất tiện. Nếu sử dụng ảnh nền không đúng có thể làm phát sinh nhiễu trong phân đoạn trừ nền của bước tiền xử lý ảnh, dẫn đến giảm độ chính xác trong việc trích xuất số đọc của thuật toán. Do bộ thiết bị có thể tháo rời thành các tầng, nên tầng một của bộ giá đỡ có thể gắn cố định

lên máy đo RAR mà không cần tháo khỏi máy như trong hình 4.5, vì vậy không cần phải chụp lại hình nền.



Hình 4.5. Máy đo RAR khi gắn vào tầng 1 của thiết bị RadOCR.  
a) Mặt trước, b) Mặt sau

Quá trình xử lý ảnh cần cắt cúp hình ảnh nhằm khu trú lại các số trên hình. Trong trường hợp tháo gỡ thiết bị RadOCR ra khỏi máy đo RAR và gắn lại, có khả năng đường cắt sẽ không đúng so với vị trí ký tự trên hình ảnh, hình ảnh hiển thị sau đó sẽ có thể bị khuyết một phần của ký tự và dẫn đến kết quả OCR đọc sai. Để tránh trường hợp này, có thể gắn cố định tầng 1 của thiết bị RadOCR như hình 4.5. Giải pháp khác có thể thực hiện được là mở rộng vùng cắt cúp, nhưng không để các ký tự như đơn bị, ký hiệu pin, bluetooth.v.v. vào vùng cắt cúp.

### 4.3. Kết quả kiểm tra độ chính xác của phần mềm

Bảng 2 trình bày kết quả kiểm tra độ chính xác của phần mềm. Tổng cộng có 5 mẫu được chọn khảo sát với 2090 ảnh đã được xử lý. Độ tự tin của Tesseract đối với các ảnh đều nằm trong khoảng 0.90 đến 0.96. Độ chính xác nằm trong khoảng 96.190% đến 98.165%, trung bình là  $(97.365 \pm 0.082)\%$ . Kết quả đánh giá này cho thấy phần mềm hoạt động tốt và ổn định, các lỗi sai xảy ra với tỉ lệ thấp.

Bảng 4.1. Kết quả thu được sau 5 lần chạy thử

Số mẫu thực hiện	Cỡ mẫu	Số trường hợp sai	Số trường hợp đúng	Max Confident	Độ chính xác
Mẫu 1	409	9	400	0.96	97.799 %
Mẫu 2	420	16	404	0.96	96.190 %
Mẫu 3	414	13	401	0.96	96.860 %
Mẫu 4	411	9	402	0.96	97.810 %
Mẫu 5	436	8	428	0.96	98.165 %
Trung bình				0.96	97.365 %
Độ lệch chuẩn					0.832 %

Các kết quả OCR sai được đã được phân tích, qua đó thấy được có hai nguyên nhân chính: số đọc đang trong quá trình cập nhật hoặc lỗi do thuật toán xử lý ảnh.

Do màn hình LCD của máy đo RAR có thời gian chuyển đổi tương đối dài, nên nếu ảnh được ghi nhận trong lúc số đọc đang chuyển từ giá trị này sang giá trị khác thì ảnh thu được sẽ chứa số đọc không rõ ràng như minh họa trong hình 4.6. Ảnh có hiện tượng này sẽ có thể cho kết quả OCR sai. Một số trường hợp vẫn cho kết quả OCR đúng. Đây là hiện tượng khách quan có tính bản chất của máy đo RAR nên rất khó có biện pháp điều chỉnh về phía thuật toán hoặc thiết bị RadOCR.



a)

b)

Hình 4.6 a) Kết quả OCR sai do số đọc chuyển đổi giá trị,  
b) Kết quả OCR đúng dù số đọc đang chuyển đổi giá trị

Nguyên nhân thứ hai gây ra kết quả OCR sai là do lỗi của thuật toán xử lý ảnh. Trong trường hợp này, hình ảnh hiển thị rõ nét nhưng kết quả OCR lại sai (minh họa trong hình 4.7). Có nguyên nhân khả dĩ dẫn đến lỗi của thuật toán xử lý ảnh. Thứ nhất, thuật toán Tesseract chưa được huấn luyện một cách tối ưu, nhất là đối với kiểu chữ số hiển

thị trên màn hình máy đo RAR. Hạn chế này có thể được khắc phục bằng cách huấn luyện lại Tesseract OCR bằng một bộ ảnh đặc trưng hơn. Tuy nhiên, giải pháp này khá phức tạp, tốn nhiều thời gian cũng như công sức. Thứ hai, ảnh có tồn tại nhiễu và khoảng cách giữa một vài số khá sát nhau, dẫn đến hình ảnh nhị phân hóa sẽ dính lại. Có thể khắc phục hạn chế này bằng việc sử dụng thuật toán closing thêm một lần sau khi thực hiện bước phân tách và tái sắp xếp các chữ số, trước khi đưa hình ảnh vào Tesseract OCR.



Hình 4.7. Kết quả OCR sai dù hình ảnh hiển thị các số rõ ràng

Thiết bị RadOCR có thể được phát triển thành một hệ thống đọc dữ liệu đa điểm cho các thiết bị ghi đo phóng xạ có cấu trúc/thiết kế tương tự như máy đo Radiation Alert Ranger® (RAR). Để phát triển cho các máy đo khác, cần điều chỉnh lại thiết kế tầng 1 của giá đỡ sao cho phù hợp với kích thước máy đo của các hãng khác. Về mặt phần mềm, cần tối ưu bước cắt cúp trong xử lý hình ảnh, do màn hình LCD của các máy đo là khác nhau. Ngoài ra, cần điều chỉnh các tham số trong thuật toán Tesseract OCR cho phù hợp với từng máy đo. Để mở rộng thành hệ thống đọc dữ liệu đa điểm, cần cải tiến module giao tiếp camera để truy xuất hình ảnh từ nhiều module ESP32-CAM tại nhiều điểm đo khác nhau. Ngoài ra cần cải tiến module xử lý ảnh và module giao diện để xử lý và hiển thị nhiều số đọc từ nhiều điểm đo khác nhau.

Thiết bị RadOCR được thiết kế như là một bộ phụ kiện rời cho các máy đo như RAR. Với thiết kế không can thiệp vào máy đo nên thiết bị không gây ảnh hưởng đến độ chính xác và kết quả đo của máy. Thiết bị RadOCR cung cấp khả năng kết nối mạng cho máy đo, cho phép trích xuất từ xa dữ liệu của máy đo theo thời gian thực hoặc có thể lưu trữ dưới dạng tập tin.

## CHƯƠNG 5. KẾT LUẬN VÀ KIẾN NGHỊ

### 5.1. Kết luận

Trong đề tài này, chúng tôi đã phát triển một hệ thống thiết bị và phần mềm RadOCR để trích xuất số đo bức xạ từ máy đo Radiation Alert Ranger® (RAR). Những kết quả đã đạt được trong đề tài bao gồm: (1) Thiết kế và gia công thiết bị RadOCR có dạng module lắp ghép có thể gắn vào máy đo RAR; (2) Phần mềm RadOCR với giao diện đồ họa thân thiện và dễ sử dụng. Hệ thống được phát triển dựa trên những công nghệ hiện đại như: lập trình phần mềm bằng ngôn ngữ Python, thiết kế thiết bị trên chương trình AutoCAD®, ứng dụng internet vạn vật. Hệ thống này cho phép số hóa máy đo bức xạ, giúp trích xuất từ xa số đọc của máy theo thời gian thực hoặc lưu trữ dạng tập tin. Bằng cách hiệu chỉnh thiết kế phần cứng và phần mềm, hệ thống có khả năng được mở rộng thành một giải pháp quan trắc phóng xạ đa điểm với đa dạng các loại máy đo bức xạ.

Về cơ bản, đề tài đã đạt được những mục tiêu đề ra ban đầu:

- Nghiên cứu cơ sở lý thuyết liên quan đến đề tài
- Thiết kế phần cứng và phần mềm
- Gia công phần cứng và lập trình phần mềm
- Chạy thử và đánh giá

### 5.2. Kiến nghị

Sau khi hoàn thành nghiên cứu, chúng tôi đề xuất ra một vài kiến nghị như sau:

- Cải biến thiết kế của thiết bị RadOCR phù hợp với các máy đo bức xạ khác
- Tiếp tục cải tiến các thuật toán xử lý ảnh để nâng cao hiệu suất và độ chính xác của phần mềm RadOCR
- Tiếp tục nghiên cứu mở rộng hệ thống RadOCR thành giải pháp quan trắc phóng xạ từ xa đa điểm

## TÀI LIỆU THAM KHẢO

- [1] Zeinab Kamal Aldein Mohammed, Elmustafa Sayed Ali Ahmed, "Internet of Things Applications, Challenges and Related Future Technologies," *World Scientific News*, pp. 127 - 138, 2017.
- [2] Hoàng Anh Tuấn, Đỗ Ngọc Diệp, "Đánh giá hiện trạng và triển vọng ứng dụng công nghệ y học hạt nhân ở Việt Nam," *Tạp chí Khoa học Công nghệ Việt Nam*, pp. 59-64, 2021.
- [3] Ramacos Fardela, Kusminarto, Ahmad Ashari, "Study of Wireless Sensor Network Application for Dosimeter Personal Real Time," *IEEE Xplore*, 2018.
- [4] TS. Đặng Thanh Lương, An toàn Bức xạ - Phần Các kiến thức nền tảng, Vols. Phần 1 - Mô đun 1.4, Thành phố Hồ Chí Minh: Tài liệu nội bộ Đại học Nguyễn Tất Thành, 2020, pp. 108 - 133.
- [5] ThS. Phạm Như Tuyên, "Buồng ion hóa - Ionization Chamber," Tài liệu nội bộ Trường Đại Học Nguyễn Tất Thành, Ho Chi Minh City.
- [6] P. Gobal B.Saha, Physics and Radiobiology of Nuclear Medicine, Vols. Gas-filled Detector, Scintillation and Semiconductor Detector, New York: Springer Science, Business Media New York, 2013, p. 87.
- [7] N. Power, "Nuclear Power," [Online]. Available: Geiger-Mueller Region - Ionization Detector | nuclear-power.com.
- [8] I. S.E. International, "Radiation Alert Ranger - Operation Manual," [Online]. Available:  
[https://seintl.com/media/product\\_document/Ranger%20Operation%20Manual%2008-22-2023\\_230823075637.pdf](https://seintl.com/media/product_document/Ranger%20Operation%20Manual%2008-22-2023_230823075637.pdf).
- [9] Sunfounder, "Github," galaxy-rvr, [Online]. Available:  
[https://github.com/sunfounder/galaxy-rvr/blob/docs/docs/source/hardware/cpn\\_esp\\_32\\_cam.rst](https://github.com/sunfounder/galaxy-rvr/blob/docs/docs/source/hardware/cpn_esp_32_cam.rst).

- [10] Arduino, "Arduino Intergrated Development Enviroment (IDE)," 17 1 2024. [Online]. Available: <https://docs.arduino.cc/software/ide-v1/tutorials/arduino-ide-v1-basics/>.
- [11] TomSchimansky, "GitHub," CustomTkinter, 5 February 2023. [Online]. Available: <https://github.com/TomSchimansky/CustomTkinter/wiki/CTkLabel>.
- [12] NumPy, "NumPy," NumPy, [Online]. Available: <https://numpy.org/>.
- [13] CSV, "Python," CSV, [Online]. Available: <https://docs.python.org/3/library/csv.html>.
- [14] Shreeshrii, "GitHub," Tessdata Shreetest, 29 April 2019. [Online]. Available: [https://github.com/Shreeshrii/tessdata\\_shreetest?tab=readme-ov-file#readme](https://github.com/Shreeshrii/tessdata_shreetest?tab=readme-ov-file#readme).
- [15] Rodríguez del Águila, González-Ramírez, Sample size Calculation, Granada, 2013.