



ỨNG DỤNG THỊ GIÁC MÁY TÍNH HUẤN LUYỆN THIẾT BỊ BAY KHÔNG NGƯỜI LÁI NHẬN DIỆN CÁC VẬT THỂ CÓ DẠNG HÌNH HỌC CƠ BẢN

Hoàng Hữu Trung^{1*}, Nguyễn Thị Minh Hương¹, Lê Thị Quỳnh Liên¹, Nguyễn Quang Phú^{2,3}

¹ Trường Đại học Kinh tế, Đại học Huế, Huế, Việt Nam

² Khoa Kỹ thuật và Công nghệ, Đại học Huế, Huế, Việt Nam

³ Phòng Nghiên cứu và Phát triển, Công ty cổ phần dược Medipharco, Huế, Việt Nam

Tóm tắt. Nghiên cứu này đề xuất một phương pháp tổng quan để huấn luyện thiết bị bay không người lái (drone) tự động nhận diện vật thể có hình dạng đơn giản và phổ biến trong thế giới thực. Cụ thể, một mô hình mạng no-ron tích chập (Convolutional Neural Network - CNN) được tích hợp vào hệ thống camera của drone DJI Tello nhằm thực hiện việc nhận diện các vật thể có dạng hình học cơ bản và hình chữ nhật, hình tam giác, hình tròn, hình ngũ giác được lựa chọn để kiểm tra độ hiệu quả nhận diện của mô hình. Trong đó, một bộ dữ liệu ảnh đen trắng được xây dựng tự động với các vật thể có kích thước và vị trí đa dạng để tối ưu và giảm thời gian quá trình thu thập dữ liệu và huấn luyện mô hình. Mô hình CNN đề xuất với cấu trúc gọn nhẹ để đảm bảo drone có thể nhận diện theo thời gian thực được huấn luyện trên bộ dữ liệu này và đạt độ chính xác gần như tuyệt đối trên cả tập huấn luyện và kiểm thử. Sau đó, mô hình được tích hợp vào camera drone và kiểm tra thực nghiệm cho thấy khả năng nhận diện chính xác trong thời gian thực mà không gặp hiện tượng quá khớp (overfitting). Kết quả này khẳng định tính hiệu quả và tiềm năng ứng dụng của phương pháp trong các hệ thống drone thông minh.

Từ khóa: hình học cơ bản, mạng no-ron tích chập, nhận diện vật thể, thị giác máy tính, thiết bị bay không người lái (drone)

Applying computer vision to train unmanned aerial vehicles for recognizing basic geometric objects

Hoang Huu Trung^{1*}, Nguyen Thi Minh Huong¹, Le Thi Quynh Lien¹, Nguyen Quang Phu^{2,3}

¹ University of Economics, Hue University, Hue, Vietnam

² School of Engineering and Technology, Hue University, Hue, Vietnam

³ R&D Department, Medipharco Pharmaceutical Joint Stock Company, Hue, Vietnam

Abstract. This study proposes a generalized methodology for training unmanned aerial vehicles (UAVs) to autonomously recognize objects of simple and common geometries in real-

* Liên hệ: hoanghuutrong@hueuni.edu.vn

world environments. Specifically, a Convolutional Neural Network (CNN) model is integrated into the camera system of a DJI Tello drone to detect basic geometric objects, including rectangles, triangles, circles, and regular pentagons, which are selected to evaluate the model's recognition performance. A grayscale image dataset comprising objects of varying sizes and positions is automatically generated to optimize the data collection and model training process. The proposed CNN model, designed with a lightweight architecture to ensure real-time processing capability on the drone, is trained on this dataset and achieves approximately 100% accuracy on both the training and test sets. Subsequently, the model is integrated into the drone's camera system, and experimental results confirm its ability to perform accurate real-time object detection without overfitting. These findings demonstrate the effectiveness and practical potential of the proposed method for integration into intelligent drone systems.

Keywords: computer vision, convolutional neural network, drone, geometric shapes, object detection

1 Giới thiệu

Cuộc Cách mạng công nghiệp 4.0 đánh dấu một sự thay đổi bước ngoặt trong cách thức sản xuất và quản lý trong nền kinh tế toàn cầu, nơi công nghệ số và tự động hóa trở thành trọng tâm. Tâm điểm của cuộc cách mạng này là trí tuệ nhân tạo (Artificial Intelligence - AI), công nghệ thúc đẩy sự phát triển của đa số các ngành nghề thông qua nhiều ứng dụng, điển hình là về thị giác máy tính (Computer Vision - CV) [1] và xử lý ngôn ngữ tự nhiên (Natural Language Processing) [2]. Với những đổi mới này, cuộc Cách mạng công nghiệp 4.0 đang mở ra các cơ hội mới và thay đổi cơ bản cách con người sống và làm việc.

Cùng với sự phát triển của trí tuệ nhân tạo, thiết bị bay không người lái (Unmanned Aerial Vehicle - UAV, hay còn gọi là drone) đã trở thành một thiết bị không thể thiếu trong nhiều lĩnh vực như tìm kiếm cứu hộ trong thiên tai [3], xây dựng công nghiệp [4], vận tải [5], quản lý, tăng năng suất nông nghiệp [6], công nghiệp khai thác [7]. Tuy nhiên, đa số các ứng dụng này drone vẫn cần con người điều khiển và giám sát trực tiếp. Để cắt giảm sức người, tăng tính tự động hóa và khả năng thực hiện nhiều nhiệm vụ khác nhau, việc tích hợp trí tuệ nhân tạo vào drone là cần thiết và ngày càng phổ biến [8].

Trong trí tuệ nhân tạo, thị giác máy tính là ngành có nhiều ứng dụng nổi bật và đang phát triển mạnh mẽ trong kỷ nguyên số ngày nay [1]. Thị giác máy tính là lĩnh vực giúp máy tính nhìn và hiểu thế giới, với nhiều ứng dụng nổi bật như nhận diện vật thể, theo dõi đối tượng, phân tích ảnh y tế, xe tự hành [1]. Mạng Nơ-ron Tích chập (Convolutional Neural Network - CNN) là một loại mạng nơ-ron nhân tạo được thiết kế đặc biệt cho các nhiệm vụ thị giác máy tính, hoạt động bằng cách trích xuất các đặc điểm từ hình ảnh qua hệ thống các lớp nơ-ron [9]. CNN có nhiều ưu điểm như khả năng học tự động từ dữ liệu, hiệu quả cao và thích ứng với nhiều bài toán thị giác máy tính khác nhau [9].

Ngày nay, việc tích hợp thị giác máy tính vào camera giám sát đã trở nên rất phổ biến trong đời sống hàng ngày. Tuy nhiên, camera giám sát thông thường không có khả năng linh hoạt cao khi chỉ cố định một vị trí và không thể di chuyển. Do đó, việc tích hợp thị giác máy tính vào drone là cần thiết, với khả năng di chuyển theo nhiều phương hướng, drone có thể thực hiện một số tác vụ như nhận diện vật thể và theo dõi đối tượng hiệu quả hơn so với camera cố định. Thị giác máy tính tích hợp vào camera của drone đã giải quyết được nhiều bài toán như nhận diện vật thể [10], theo dõi [11], tránh chướng ngại vật [12], cứu hộ [13].

Trong thị giác máy tính, bài toán nhận diện vật thể là bài toán cơ bản và là nền tảng cho các bài toán chuyên sâu khác. Để theo dõi một vật thể chuyển động, camera của drone cần nhận diện được đúng vật thể sau đó mới có thể theo dõi vật thể mục tiêu. Tương tự như vậy, để tránh chướng ngại vật hay cứu hộ con người, camera của drone cũng cần nhận diện được vật thể ở vị trí nào của camera để tránh hay cần nhận diện đúng con người để có thể trợ giúp và cứu trợ. Chính vì vậy, việc huấn luyện camera của drone nhận diện các vật thể cơ bản trong đời sống hàng ngày là cần thiết để phát triển rộng hơn các bài toán cụ thể khác.

Những đóng góp chính của bài báo được tóm tắt như sau:

- Đề xuất phương pháp tổng quan và khép kín từ việc xây dựng bộ dữ liệu đến huấn luyện và tích hợp mô hình gọn nhẹ cho phép camera của drone thực hiện nhận diện vật thể có hình dạng đơn giản và phổ biến trong thế giới thực. Tính hiệu quả và tiềm năng của phương pháp tổng quan này được chứng minh thông qua bài toán nhận diện 4 dạng hình học cơ bản là hình chữ nhật, hình tam giác, hình tròn, hình ngũ giác đều và cho kết quả là độ chính xác xấp xỉ 100%.

- Xây dựng bộ dữ liệu ảnh với các dạng hình học cơ bản hỗ trợ huấn luyện drone nhận diện các vật thể này. Bộ ảnh có sử dụng hệ màu đen trắng nhằm giảm kích thước ảnh, làm tăng tốc độ huấn luyện so với các bộ ảnh màu có kích thước ảnh lớn. So với các bài toán khác phải thu thập dữ liệu ảnh thủ công gây tốn thời gian, công sức, chi phí thu thập dữ liệu, bài báo đề xuất thuật toán để tạo số lượng lớn ảnh huấn luyện trong một thời gian ngắn.

- Ứng dụng thuật toán CNN huấn luyện drone nhận diện và phân biệt 4 loại vật thể với độ chính xác lên đến xấp xỉ 100%, đây là độ chính xác gần như tuyệt đối. Nghiên cứu lựa chọn mô hình gọn nhẹ để có thể triển khai nhận diện nhanh theo thời gian thực.

- Tích hợp mô hình CNN đã huấn luyện vào camera của drone DJI Tello và có thể sử dụng trực tiếp camera của drone để nhận diện và phân loại 4 loại vật thể đã nêu ở trên. Vật thể có các màu đặc trưng như đỏ, lục, lam để camera của drone có thể nhận diện. Màu sắc của vật thể được chuyển thành màu trắng, những vật thể còn lại trong khung hình camera được chuyển thành màu đen để phù hợp với bộ dữ liệu ảnh huấn luyện có màu đen trắng. Kết quả nhận diện thực tế xấp xỉ 100% cho thấy phương pháp này có thể ứng dụng vào thế giới thực.

Các phần nội dung tiếp theo của bài báo được cấu trúc như sau: Phần 2 giới thiệu các công trình nghiên cứu liên quan đến bài báo, phần 3 trình bày phương pháp huấn luyện thiết bị bay

không người lái nhận diện vật thể có các dạng hình học cơ bản, kết quả nghiên cứu và thảo luận về kết quả nghiên cứu được trình bày trong phần 4, cuối cùng phần 5 trình bày kết luận của bài báo.

2 Công trình nghiên cứu liên quan

Tại Việt Nam, thiết bị bay không người lái ngày càng được ứng dụng và nghiên cứu rộng rãi tại Việt Nam. Thiết bị không người lái được ứng dụng ở nhiều nghiên cứu liên quan đến khảo sát địa hình [14], [15] và tạo bản đồ [16], [17], [18]. Đặc điểm của các nghiên cứu này là ứng dụng những điểm nổi bật của drone như có thể hoạt động trên không và có camera ghi hình để ghi lại đặc điểm địa hình tại nơi muốn theo dõi một cách tổng quát từ trên cao sau đó thu thập dữ liệu ảnh. Tuy nhiên, những ứng dụng này vẫn cần sự điều khiển và giám sát của con người mà chưa có khả năng tự hoạt động từ thiết bị bay không người lái.

Các thuật toán nhận diện áp dụng trí tuệ nhân tạo cũng đã được ứng dụng để phát hiện rác thải nhựa ven biển [19]. Nghiên cứu này thu thập dữ liệu bằng cách sử dụng 95 ảnh chụp từ drone Phantom 4 Pro để đưa vào mô hình mạng nơ-ron tích chập sâu (Deep CNN) và huấn luyện cho ra kết quả. Đề tài đã phần nào phát hiện được các loại rác thải nhựa ven biển Hội An, Quảng Nam. Tuy nhiên, drone chỉ được sử dụng để thu thập dữ liệu chứ chưa thể nhận diện trực tiếp các loại rác thải nhựa khi đi dọc ven biển.

Trên bình diện quốc tế, [20] là một nghiên cứu phát hiện và phân loại các hình dạng đối tượng vật thể như hình tròn, hình chữ nhật, hình tam giác. Mô hình sẽ có khả năng phát hiện nhiều đối tượng từ một hình ảnh, đếm số lượng các đối tượng đã phát hiện, tách những đối tượng này thành các hình ảnh riêng biệt thông qua quá trình chuyển đổi sang mức xám, ngưỡng hóa, phát hiện biên, tìm các điểm góc của đối tượng. Quá trình phân loại được thực hiện thông qua mạng nơ-ron nhân tạo (Artificial neural network - ANN) và máy vectơ hỗ trợ (Support vector machine - SVM) bằng cách sử dụng các tập dữ liệu tổng hợp. Giải pháp đề xuất này đã đạt được độ chính xác phân loại 95% bằng ANN và từ 95% đến khoảng 100% bằng SVM. ANN và SVM đã học cũng có thể phân loại các đối tượng mới được phát hiện từ một hình ảnh. Tuy nhiên, nghiên cứu này chỉ nhận diện vật thể thông qua camera thông thường chứ chưa tích hợp được vào camera của drone.

Trong luận văn thạc sĩ [21], tác giả đã tập trung vào việc sử dụng các máy bay không người lái Tello và Tello EDU, tạo ra một nền tảng kiểm soát chuyển động của máy bay không người lái thông qua nhận diện tư thế cơ thể con người. Nền tảng Tello SDK được sử dụng với nhiều lớp dữ liệu và phương thức nhằm kiểm soát máy bay không người lái Tello và MediaPipe cung cấp nền tảng kiến trúc mạng nơ-ron tích chập cho việc ước tính tư thế con người và tùy chỉnh việc ước tính thời gian thực trên các thiết bị di động và thiết bị có cấu hình thấp. Tuy nhiên, nghiên cứu này sử dụng nền tảng có sẵn MediaPipe Machine Learning của Google AI tập trung nhận

diện tư thế cơ thể con người chứ chưa nghiên cứu về các vật thể cơ bản khác và phụ thuộc vào mức độ phát triển của Google chứ chưa tự phát triển được mô hình riêng phù hợp từng bài toán khác nhau.

Trong nghiên cứu [22], nhóm tác giả đã tập trung vào việc nhận diện đối tượng thông qua video thu được từ thiết bị bay không người lái, ứng dụng mạng no-ron tích chập đã cho phép thiết bị bay không người lái nhận diện một số loại đối tượng như tòa nhà, ô tô, cây cỏ và con người. Nghiên cứu đã cho thấy kết quả độ chính xác của hai mô hình Single Shot Detector (SSD) và Faster R-CNN đối với tòa nhà, cây cỏ, ô tô và con người rất cao, trung bình trên 85% và tối đa là 99%. Các mô hình SSD chú tâm nhiều hơn đến tỷ lệ và vị trí lấy mẫu dự đoán hơn là Faster R-CNN, thời gian trung bình của mỗi khung hình là 115 mili giây (Millisecond - ms), nhưng tỷ lệ nhận diện đối tượng thấp. Trong đó, Faster R-CNN chính xác hơn và tìm thấy nhiều đối tượng hơn từ cảnh quan, gần 95% tất cả các đối tượng trong hình ảnh có thể được nhận biết, nhưng thời gian trung bình của mỗi khung hình ít nhất là 140 ms. Tuy nhiên, nghiên cứu này sử dụng bộ dữ liệu video có sẵn để thực nghiệm và chưa thử nghiệm thực tế.

Ở một nghiên cứu khác [23], nhóm tác giả đã trình bày một phương pháp phát hiện và theo dõi đối tượng mục tiêu đang di chuyển hoặc đứng yên cho máy bay không người lái Parrot AR từ việc ứng dụng mô hình mạng no-ron tích chập. Kết quả nhận diện đối tượng cho thấy rằng CNN phát hiện và phân loại đối tượng với mức độ chính xác cao (98%). Đối với việc theo dõi thời gian thực, thuật toán theo dõi phản ứng nhanh hơn so với các phương pháp thông thường được sử dụng, hiệu quả trong việc theo dõi đối tượng đã được nhận diện mà không mất mục tiêu khỏi tầm nhìn. Các tính toán dựa trên một số lần lặp cho thấy rằng hiệu suất đạt được cho việc theo dõi mục tiêu là 96,5%. Nghiên cứu này cho thấy rằng việc ứng dụng CNN phát hiện đối tượng và tích hợp vào drone đạt hiệu quả cao trong thực tế. Tuy nhiên, nghiên cứu này cần nhiều thời gian để tạo và tiền xử lý dữ liệu thông qua việc tự động chụp 5100 bức ảnh.

3 Huấn luyện thiết bị bay không người lái nhận diện vật thể có dạng hình học cơ bản

3.1 Tổng quan phương pháp

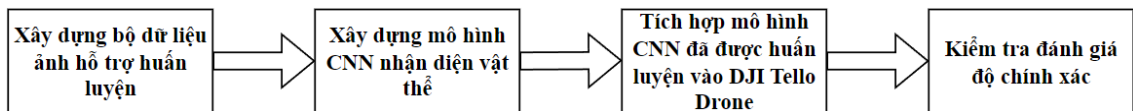
Phương pháp tổng quan của nghiên cứu này được chia thành bốn bước chính được mô tả trong **Hình 1** như sau:

- Xây dựng bộ dữ liệu ảnh hỗ trợ huấn luyện: Bước đầu tiên là xây dựng bộ dữ liệu ảnh để huấn luyện mô hình CNN nhận diện vật thể trên drone DJI Tello. Trong bước này, một bộ dữ liệu chứa các hình ảnh các vật thể có dạng hình học cơ bản như hình chữ nhật, hình tam giác, hình tròn, hình ngũ giác đều sẽ được xây dựng. Bộ dữ liệu ảnh này sử dụng hệ màu đen trắng để giảm kích thước ảnh làm tăng tốc độ huấn luyện.

- Xây dựng mô hình CNN nhận diện vật thể: Sau khi đã có bộ dữ liệu ảnh, một mô hình CNN gọn nhẹ sẽ được đề xuất để học các đặc điểm của 4 dạng hình học cơ bản ở bước 1 trong bộ dữ liệu ảnh.

- Tích hợp mô hình CNN đã được huấn luyện vào drone DJI Tello: Sau khi có mô hình CNN có thể nhận diện 4 dạng vật thể cơ bản, mô hình này sẽ được tích hợp vào camera của drone DJI Tello để có thể nhận diện trực tiếp các vật thể này từ camera của drone.

- Kiểm tra, đánh giá độ chính xác: Để đánh giá độ chính xác của mô hình CNN, một số phương pháp đánh giá bài toán phân loại được sử dụng như độ chính xác (accuracy), hàm mất mát (loss function), precision, recall, F1-score. Ngoài ra, để đảm bảo drone DJI Tello có thể nhận diện được các vật thể có 4 dạng hình học đã nêu, mô hình CNN sẽ được tích hợp vào camera của drone để nhận diện trực tiếp các vật thể để kiểm tra độ chính xác trong thực tế. Việc thực nghiệm được diễn ra trong môi trường trong nhà để đảm bảo điều kiện ánh sáng cho camera của drone có thể nhận diện vật thể.



Hình 1. Sơ đồ tổng thể phương pháp

3.2 Xây dựng bộ dữ liệu hình ảnh vật thể

Tổng quan về bộ dữ liệu hình ảnh vật thể

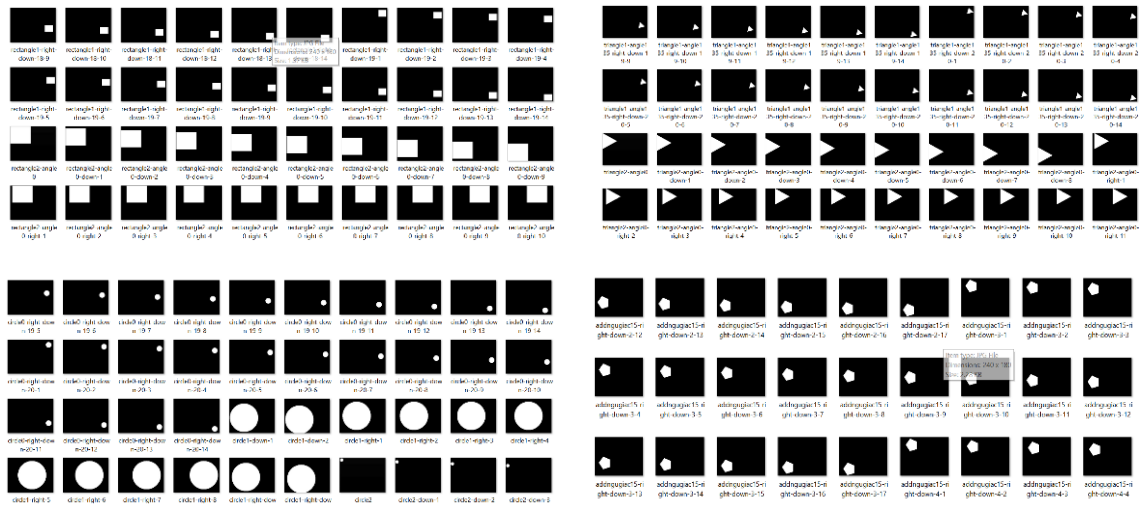
Bộ dữ liệu hình ảnh được xây dựng với kích thước 240 điểm ảnh (pixel) \times 180 điểm ảnh, đây là kích thước ảnh phù hợp, không quá lớn làm tăng kích thước bộ dữ liệu cũng như không quá nhỏ để làm giảm chất lượng hình ảnh. Hệ màu của bộ dữ liệu ảnh là đen trắng để có thể phân biệt vật thể chính với các vật còn lại trong khung hình. Khi camera của drone nhận diện, toàn bộ vật thể muốn nhận diện sẽ được chuyển qua màu trắng, các vật thể còn lại trong khung hình của camera sẽ được chuyển thành màu đen. Bộ dữ liệu bao gồm 4 dạng hình học cơ bản bao gồm:

- Hình chữ nhật: Nhận diện vật thể với nhãn là hình chữ nhật (rectangle). Bộ dữ liệu hình ảnh vật thể có dạng hình chữ nhật bao gồm nhiều dạng hình chữ nhật với tỷ lệ chiều dài và chiều rộng khác nhau, nằm ở nhiều vị trí khác nhau trong khung ảnh.

- Hình tam giác: Nhận diện vật thể với nhãn là hình tam giác (triangle). Bộ dữ liệu hình ảnh vật thể có dạng hình tam giác bao gồm nhiều dạng hình tam giác khác nhau như tam giác nhọn, tam giác tù, tam giác cân, tam giác vuông với nhiều kích thước đa dạng khác nhau từ nhỏ tới lớn và kèm với đó là nhiều góc độ quay và vị trí khác nhau của hình tam giác trong hình ảnh bộ dữ liệu.

- Hình tròn: Nhận diện vật thể với nhân là hình tròn (circle). Bộ dữ liệu hình ảnh vật thể có dạng hình tròn bao gồm các hình tròn có kích thước khác nhau từ nhỏ tới lớn và nằm ở đa dạng các vị trí khác nhau trong ảnh để tối ưu khả năng nhận diện hình tròn chính xác.

- Hình ngũ giác đều: Nhận diện vật thể với nhân là hình ngũ giác đều (pentagon). Bộ dữ liệu hình ảnh vật thể có dạng hình ngũ giác đều bao gồm nhiều hình ngũ giác đều với kích thước khác nhau, với nhiều góc độ quay và vị trí khác nhau trong khung ảnh.



Hình 2. Minh họa về bộ dữ liệu

Xây dựng thuật toán hỗ trợ việc xây dựng bộ dữ liệu hình ảnh

Đầu tiên, một hình ảnh với vật thể màu trắng có dạng hình học cơ bản nằm ở vị trí góc trên bên trái của hình ảnh có kích thước 240 pixel × 180 pixel. So với việc thu thập dữ liệu tốn nhiều công sức, dữ liệu ảnh này ban đầu chỉ cần dùng các ứng dụng vẽ mô hình, đồ thị hoặc các ứng dụng chỉnh sửa ảnh đơn giản là có thể tạo được nhanh chóng, nghiên cứu này sử dụng ứng dụng Draw.io [24] để thực hiện điều này. Kích thước hình ảnh vật thể có thể thay đổi dựa vào khoảng cách giữa camera và vật thể, do đó, vật thể màu trắng sẽ được thay đổi kích thước nhiều lần và vẽ thông qua ứng dụng vẽ mô hình. Từ ứng dụng trên, bộ dữ liệu hình ảnh với dạng hình học hình chữ nhật, hình tam giác, hình tròn, hình ngũ giác đều ban đầu nằm góc trên bên trái được xây dựng.

Sau đó, để tạo hàng loạt ảnh một cách tự động, một thuật toán Python hỗ trợ việc xây dựng bộ dữ liệu ảnh được thực hiện. Đoạn mã này sử dụng thư viện OpenCV và Matplotlib để xử lý, hiển thị hình ảnh, tạo ra các hình ảnh nhằm di chuyển vật thể màu trắng trong khung ảnh từ trái sang phải mỗi lần 10 pixel, sau khi đến sát phải khung hình, vật thể sẽ bắt đầu lại từ bên trái khung hình như trong hình 3 nhưng ở dưới vị trí ban đầu 10 pixel. Thuật toán này sẽ tạo ra bộ

dữ liệu ảnh vật thể nằm mọi vị trí trong khung hình, giúp thuật toán CNN có thể nhận diện tốt hơn. Thuật toán này được biểu diễn bằng các đoạn mã Python được mô tả trong **Thuật toán 1**.

```

Input: Ảnh đầu vào hệ màu đen trắng, kích thước ảnh đầu vào (height, width)
Output: Bộ ảnh sau khi đã di chuyển đối tượng màu trắng trong ảnh
// Đọc ảnh đầu vào
img = cv2.imread("<đường dẫn file ảnh ban đầu>")
// Chuyển ảnh sang hệ ảnh xám
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
// Áp dụng ngưỡng để chuyển đổi sang ảnh nhị phân, hệ màu đen trắng
blackAndWhiteImage = cv2.threshold(gray, 127, 255, cv2.THRESH_BINARY)
// Lấy kích thước của ảnh nhị phân
height, width = blackAndWhiteImage.shape

// Vòng lặp 2 chiều di chuyển các điểm ảnh màu trắng trong ảnh
for i in range(1, 24):
    // Tạo một ảnh mới để lưu trữ ảnh đã được di chuyển
    img_new = blackAndWhiteImage.copy()
    // Duyệt qua từng pixel theo hàng và cột (theo chiều từ phải sang trái)
    for h in range(height):
        for w in range(width - 1, -1, -1):
            // Kiểm tra nếu điểm ảnh không phải là màu đen (tức là điểm ảnh màu trắng)
            if blackAndWhiteImage[h][w] != 0:
                // Kiểm tra xem nếu di chuyển qua phải 10 pixel đã vượt qua giới hạn phải của ảnh chưa
                if w + 10 < width
                    // điểm ảnh hiện tại đổi sang màu đen
                    img_new[h][w] = 0
                    // điểm ảnh ở vị trí dịch chuyển sang 10 pixel sang màu trắng
                    img_new[h][w + 10] = 255
                // nếu di chuyển sang phải 10 pixels là vượt qua giới hạn ảnh
            else:
                // điểm ảnh ở vị trí hiện tại đổi sang màu đen
                img_new[h][w] = 0
// Lưu ảnh
file_name = "<đường dẫn file ảnh lưu>" + str(i) + ".jpg"
cv2.imwrite(file_name, img_new)

```

Thuật toán 1. Thuật toán python tạo bộ ảnh dữ liệu

3.3 Ứng dụng thuật toán mạng nơ-ron tích chập tích hợp vào camera của drone nhận diện các vật thể có dạng hình học cơ bản

a. Đề xuất kiến trúc mạng nơ-ron tích chập nhận diện các vật thể có dạng hình học cơ bản

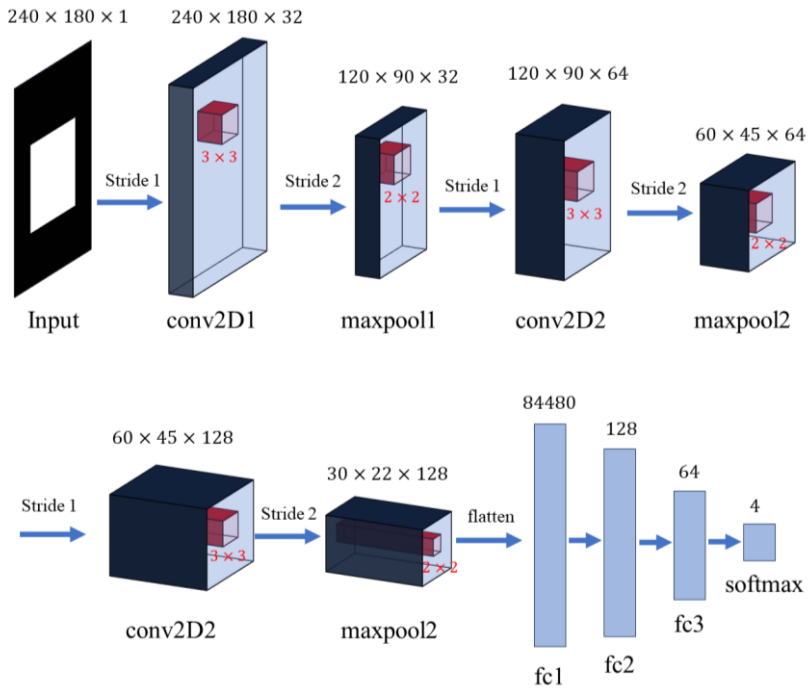
Trong phần này, mô hình CNN được đề xuất thay vì các kiến trúc tân tiến khác như SSD, Faster R-CNN, YOLO vì theo nghiên cứu [22] đã phân tích hiệu năng của SSD và Faster R-CNN trên dữ liệu video từ drone, kết quả cho thấy Faster R-CNN đạt độ chính xác cao hơn (nhận diện gần 95% đối tượng) nhưng có thời gian xử lý trung bình mỗi khung hình tối thiểu là 140 ms. Trong khi đó, mô hình SSD nhanh hơn nhưng lại có tỷ lệ nhận diện thấp hơn. Ở nghiên cứu [23], kết quả nhận diện đối tượng đã cho thấy rằng CNN có thể nhận diện đối tượng với mức độ chính xác cao (98%). Đối với việc theo dõi thời gian thực, thuật toán CNN có thể theo dõi phản ứng nhanh hơn so với các phương pháp thông thường được sử dụng, hiệu quả trong việc theo dõi đối tượng đã được nhận diện. Nghiên cứu này cho thấy rằng việc ứng dụng CNN phát hiện đối tượng và tích hợp vào drone đạt hiệu quả cao trong thực tế.

Từ phân tích trên, trong nghiên cứu này, một kiến trúc CNN được đề xuất để giúp camera của drone có thể nhận diện và phân biệt được các vật thể có dạng hình chữ nhật, hình tam giác, hình tròn, hình ngũ giác đều. Việc sử dụng thuật toán CNN phù hợp và có độ chính xác cao có thể giúp drone nhận diện và phân biệt các vật thể dựa trên các đặc trưng (feature) được trích xuất từ bộ dữ liệu ảnh. Mô hình CNN sẽ học cách nhận biết và phân loại các hình ảnh chứa vật thể với các dạng hình học cơ bản khác nhau.

Mục tiêu của nghiên cứu này là đề xuất một phương pháp hỗ trợ thiết bị bay không người lái nhận diện các vật thể có dạng hình học cơ bản như hình chữ nhật, hình tam giác, hình tròn, hình ngũ giác đều với khả năng được thực hiện trong thời gian thực. Do đó, nghiên cứu này sử dụng kiến trúc CNN ở mức “gọn nhẹ” để giảm mức độ tính toán và thời gian học của CNN.

Tổng thể kiến trúc của mô hình CNN được sử dụng để học từ bộ dữ liệu ảnh được minh họa như trong **Hình 3**, và các chi tiết của cấu hình mô hình cũng như số lượng các tham số được liệt kê trong **Bảng 1**. Hình ảnh có kích thước $240 \times 180 \times 1$ được sử dụng làm lớp đầu vào của mô hình mạng nơ-ron tích chập CNN.

Ảnh đầu vào có kích thước $240 \times 180 \times 1$, trong đó 240×180 là kích thước ảnh, 1 là số kênh màu của ảnh, vì dữ liệu được sử dụng để huấn luyện ở dạng đen trắng nên số kênh chỉ có 1 để giảm kích thước ảnh để giảm thời gian huấn luyện. Mô hình CNN được đề xuất bao gồm ba lớp tích chập hai chiều (conv2D) và ba lớp tổng hợp tối đa (maxpool) xen kẽ nhau. Tiếp theo là dữ liệu sẽ được làm phẳng ở dạng 1D (flatten – fc1) sau đó tiếp tục qua 2 lớp kết nối đầy đủ (fully connected 2 và 3 – fc2 và fc3). Cuối cùng, để có thể đưa ra dự đoán, 1 lớp softmax được đặt ở cuối mô hình để đưa ra dự đoán về dạng hình học xuất hiện trong bức ảnh.



Hình 3. Kiến trúc mô hình CNN đề xuất

Bảng 1. Cấu hình mô hình CNN đề xuất

Lớp	Mô tả mạng	Kích thước đầu ra
Ảnh đầu vào (Input)	Bộ dữ liệu ảnh	240×180×1
Lớp tích chập 1 (conv2D1)	32 filters 3×3, stride 1	240×180×32
Lớp tổng hợp 1 (maxpool1)	pool size 2×2, stride 2	120×90×32
Lớp tích chập 2 (conv2D2)	64 filters 3×3, stride 1	120×90×64
Lớp tổng hợp 2 (maxpool2)	pool size 2×2, stride 2	60×45×64
Lớp tích chập 3 (conv2D3)	128 filters 3×3, stride 1	60×45×128
Lớp tổng hợp 3 (maxpool3)	pool size 2×2, stride 2	30×22×128
Lớp kết nối đầy đủ 1 (fc1)	flatten	84.480
Lớp kết nối đầy đủ 2 (fc2)	fully connected layer	128
Lớp kết nối đầy đủ 3 (fc3)	fully connected layer	64
Lớp softmax	softmax layer	4
Số lượng tham số		10.914.886

Lớp conv2D đầu tiên sử dụng bộ lọc (filter) 32 kênh với bước nhảy (stride) (1, 1) và kích thước bộ lọc là 3 × 3. Sau đó, thông tin đi qua một lớp chuẩn hóa hàng loạt và sử dụng lớp kích

hoạt (activation function) là Rectified Linear Unit (ReLU) để tối ưu. Trong mô hình CNN này, số lượng tham số có thể học được (trainable params) là 10.914.886. Số lượng lần học toàn bộ bộ dữ liệu (epochs) được sử dụng để huấn luyện là 10, số lượng ảnh trong một lần huấn luyện (batch size) là 32. Bộ dữ liệu gốc được chia thành 80% cho bộ dữ liệu huấn luyện (training dataset) và 20% được dùng để làm bộ dữ liệu kiểm thử (validation dataset).

b. Nhận diện vật thể thông qua camera của drone

Hệ thống thu thập hình ảnh từ drone DJI Tello hoạt động thông qua kết nối không dây WiFi. Khi khởi động, drone sẽ tự động phát tín hiệu WiFi, cho phép máy tính thực hiện kết nối và điều khiển từ xa. Để tương tác với drone thông qua giao thức WiFi, DJI cung cấp bộ công cụ phát triển phần mềm Tello SDK (Software Development Kit) [25]. Tello SDK là một nền tảng chuyên dụng được thiết kế nhằm hỗ trợ việc phát triển ứng dụng và tích hợp các tính năng điều khiển với máy bay không người lái Tello.

Để triển khai Tello SDK trong môi trường Python, nghiên cứu sử dụng thư viện DjiTelloPy [26], một thư viện mã nguồn mở được phát triển để điều khiển drone DJI Tello. DjiTelloPy cung cấp giao diện lập trình ứng dụng (API) đầy đủ cho việc điều khiển và tương tác với drone DJI Tello. Thư viện này cho phép việc phát triển các ứng dụng từ cơ bản đến phức tạp, bao gồm điều khiển chuyển động, thu thập dữ liệu cảm biến và xử lý hình ảnh thời gian thực. DjiTelloPy đóng vai trò quan trọng trong việc đơn giản hóa quá trình lập trình và mở rộng khả năng ứng dụng của drone trong các nghiên cứu khoa học.

Hệ thống xử lý hình ảnh được thiết kế dựa trên phương pháp phân tách màu sắc và nhận diện hình học. Đầu tiên, hình ảnh từ camera drone được truyền về máy tính và được chuyển đổi thành hệ ảnh nhị phân (đen trắng) thông qua việc nhận diện một trong ba màu cơ bản: đỏ, xanh lá cây hoặc xanh dương. Các pixel có màu được chọn sẽ được chuyển thành màu trắng, trong khi các pixel còn lại được chuyển thành màu đen. Để đảm bảo độ chính xác trong nhận diện, vật thể cần được thiết kế với một trong ba màu đặc trưng này, đồng thời môi trường xung quanh phải không chứa màu tương ứng.

Sau khi hoàn thành quá trình chuyển đổi ảnh nhị phân, hệ thống sử dụng mô hình mạng nơ-ron tích chập đã được huấn luyện trước để nhận diện các dạng hình học cơ bản của vật thể, bao gồm hình tròn, hình chữ nhật, hình tam giác và hình ngũ giác đều.

4 Kết quả và thảo luận

4.1 Kết quả huấn luyện mô hình CNN nhận diện vật thể

Kết quả huấn luyện mô hình CNN đề xuất sẽ trình bày dưới dạng các chỉ số về độ lỗi (loss) và độ chính xác (accuracy) trên tập huấn luyện và tập kiểm thử, cùng với thời gian xử lý trung bình mỗi mẫu.

Trong **hình 4**, kết quả nhận thấy được:

- Tập huấn luyện bao gồm 101.380 mẫu và tập kiểm thử bao gồm 25.345 mẫu.
- Độ lỗi (Loss): Là điểm số thể hiện mức độ sai lệch giữa dự đoán của mô hình và giá trị thực tế. Thường là mục tiêu để được giảm thiểu trong quá trình huấn luyện.
- Độ chính xác (Accuracy): Tỷ lệ phần trăm các dự đoán đúng trên tổng số mẫu. Đây là một chỉ số quan trọng để đánh giá hiệu suất của mô hình.
- Trong chu kỳ huấn luyện (epoch) đầu tiên, độ lỗi trên tập huấn luyện là 0,2124 và độ chính xác là 0,9132. Điều này thể hiện rằng mô hình đã bắt đầu với một mức độ khá tốt trên tập huấn luyện.
- Qua các epoch (chu kỳ huấn luyện) tiếp theo, độ lỗi trên tập huấn luyện giảm đáng kể xuống 0,0056; trong khi đó, độ chính xác tăng lên 0,9986. Điều này cho thấy mô hình đã học rất tốt và thể hiện khả năng dự đoán rất chính xác.

Từ kết quả trên **hình 4** và **hình 5** cho thấy mô hình CNN đã đạt được hiệu suất đáng kể trong quá trình huấn luyện. Giá trị độ lỗi trên cả tập huấn luyện và tập kiểm tra giảm theo từng epoch, đồng thời độ chính xác tăng lên một cách nhanh chóng. Đặc biệt, mô hình đã đạt đến mức độ chính xác cao hơn 99% trên tập kiểm tra, cho thấy khả năng phân loại chính xác của mô hình trên các dữ liệu mới.

Một điểm đáng chú ý khác là mô hình không có dấu hiệu quá khớp (overfitting), mặc dù độ chính xác trên tập huấn luyện đã đạt mức rất cao. Cách tiếp cận tạo bộ dữ liệu ảnh với vị trí các vật thể màu trắng đa dạng kích thước và vị trí, giúp mô hình học được tốt hơn và tránh overfitting. Điều này cho thấy mô hình có khả năng tổng quát hóa tốt cả trên dữ liệu không được sử dụng trong quá trình huấn luyện.

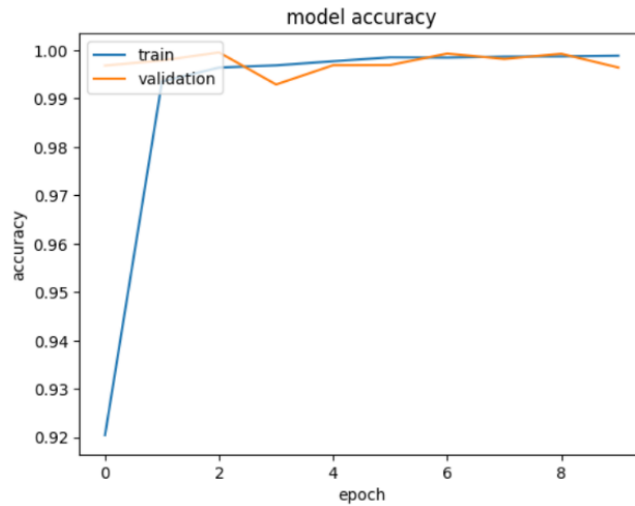
Thời gian trung bình mỗi mẫu xử lý qua mô hình trong mỗi epoch là khoảng 433 microseconds ($\mu\text{s/sample}$). Thời gian này cho thấy mô hình đang được huấn luyện một cách hiệu quả và có khả năng xử lý nhanh chóng.

```

Train on 101380 samples, validate on 25345 samples
Epoch 1/10
101380/101380 [-----] - ETA: 0s - loss: 0.2124 - acc: 0.9132/usr/local/lib/python3.10/dist-packages/keras/engine/training_v1.py:2335:
updates = self.state_updates
101380/101380 [-----] - 56% 555us/sample - loss: 0.2124 - acc: 0.9132 - val_loss: 0.0158 - val_acc: 0.9951
Epoch 2/10
101380/101380 [-----] - 44% 437us/sample - loss: 0.0220 - acc: 0.9931 - val_loss: 0.0246 - val_acc: 0.9930
Epoch 3/10
101380/101380 [-----] - 43% 420us/sample - loss: 0.0148 - acc: 0.9954 - val_loss: 0.0086 - val_acc: 0.9969
Epoch 4/10
101380/101380 [-----] - 44% 435us/sample - loss: 0.0121 - acc: 0.9967 - val_loss: 0.0190 - val_acc: 0.9937
Epoch 5/10
101380/101380 [-----] - 43% 422us/sample - loss: 0.0086 - acc: 0.9975 - val_loss: 0.0055 - val_acc: 0.9991
Epoch 6/10
101380/101380 [-----] - 44% 437us/sample - loss: 0.0094 - acc: 0.9976 - val_loss: 0.0123 - val_acc: 0.9966
Epoch 7/10
101380/101380 [-----] - 43% 421us/sample - loss: 0.0060 - acc: 0.9978 - val_loss: 0.0105 - val_acc: 0.9979
Epoch 8/10
101380/101380 [-----] - 44% 438us/sample - loss: 0.0060 - acc: 0.9983 - val_loss: 0.0108 - val_acc: 0.9976
Epoch 9/10
101380/101380 [-----] - 43% 422us/sample - loss: 0.0042 - acc: 0.9989 - val_loss: 0.0106 - val_acc: 0.9972
Epoch 10/10
101380/101380 [-----] - 44% 433us/sample - loss: 0.0056 - acc: 0.9986 - val_loss: 0.0058 - val_acc: 0.9987
Saved model to disk
dict_keys(['loss', 'acc', 'val_loss', 'val_acc'])
/usr/local/lib/python3.10/dist-packages/keras/engine/training_v1.py:2359: UserWarning: "Model.state_updates" will be removed in a future version. This property
updates=self.state_updates,

```

Hình 4. Kết quả huấn luyện



Hình 5. Đồ thị biểu diễn kết quả huấn luyện của mô hình CNN

Kết quả mô hình phân loại được thể hiện trong **Bảng 2** cho thấy các kết quả sau:

- Chỉ số precision đo lường tỷ lệ các dự đoán đúng về một lớp cụ thể trên tổng số các dự đoán của lớp đó. Trong trường hợp này, tất cả các lớp ("circle", "triangle", "rectangle", "pentagon") đều có giá trị precision được làm tròn bằng 1,00 tức là đạt xấp xỉ 100%, nghĩa là tất cả các dự đoán đều đúng cho tất cả các lớp.

- Chỉ số recall đo lường tỷ lệ các dự đoán đúng về một lớp cụ thể trên tổng số mẫu thực tế thuộc lớp đó. Tương tự như precision, tất cả các lớp đều có recall bằng 1,00, chỉ ra rằng tất cả các mẫu thực tế của các lớp đều được dự đoán chính xác.

- Chỉ số F1-score kết hợp giữa precision và recall và thường được sử dụng để đánh giá hiệu suất của mô hình phân loại. Với precision và recall đạt giá trị 1,00, F1-score cũng sẽ đạt giá trị 1,00 cho tất cả các lớp.

- Chỉ số Support thể hiện số lượng mẫu thực tế thuộc mỗi lớp. Trong trường hợp này, các lớp "circle", "triangle", "rectangle", và "pentagon" có số lượng mẫu tương ứng là 19.411, 36.372, 31.990 và 38.952.

- Chỉ số Accuracy (độ chính xác tổng thể) là tỷ lệ phần trăm các dự đoán đúng trên tổng số mẫu. Trong **Bảng 2**, accuracy xấp xỉ 1,00, tức là tất cả các mẫu gần như được dự đoán chính xác.

Bảng 2. Kết quả mô hình phân loại

	precision	recall	f1-score	support
circle	1.00	1.00	1.00	19411
triangle	1.00	1.00	1.00	36372
rectangle	1.00	1.00	1.00	31990
pentagon	1.00	1.00	1.00	38952
accuracy			1.00	126725
macro avg	1.00	1.00	1.00	126725
weighted avg	1.00	1.00	1.00	126725

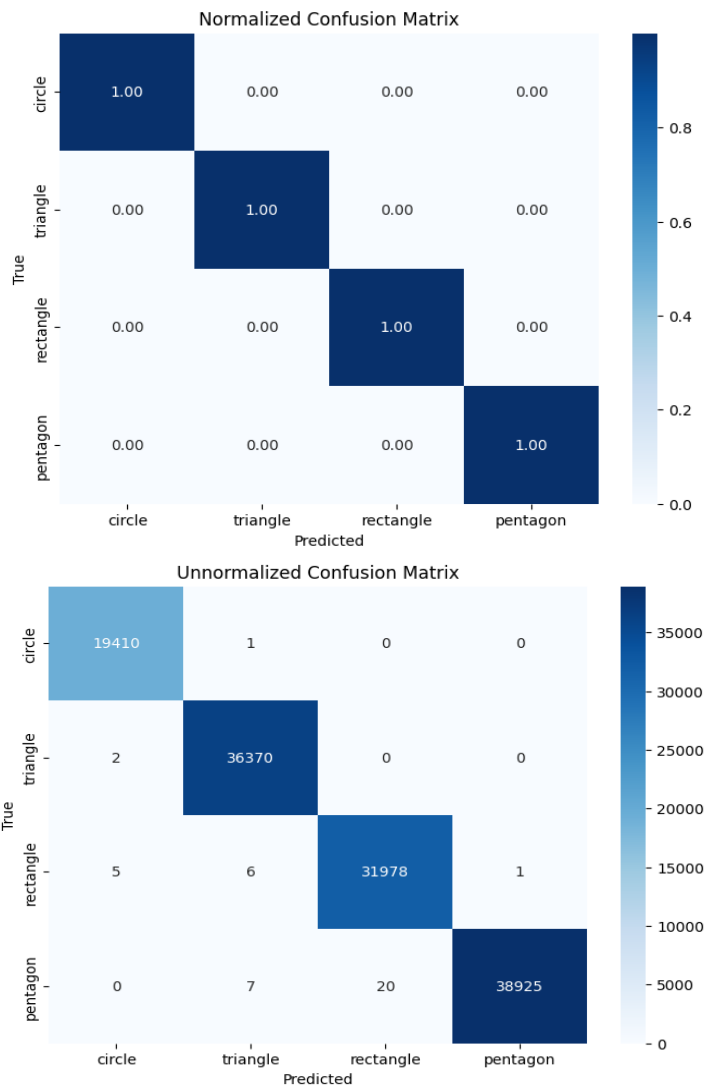
Kết quả đánh giá từ **Bảng 2** cho thấy mô hình phân loại đã thể hiện hiệu suất tốt trên toàn bộ các lớp dữ liệu. Độ chính xác cao của precision, recall và f1-score cho thấy mô hình có khả năng phân loại chính xác mọi trường hợp. Số lượng mẫu thực tế trong từng lớp cũng thể hiện mức độ đa dạng và đại diện của tập dữ liệu. Độ chính xác gần như đạt 100% vì đây là bài toán khá cơ bản trong thị giác máy tính, dữ liệu nhiều, đầy đủ cũng giúp mô hình đạt hiệu suất gần như tuyệt đối.

Trong **Hình 6**, các ma trận nhầm lẫn thể hiện sự tương quan giữa kết quả dự đoán và kết quả thực tế trên tập dữ liệu kiểm tra. Từ việc phân tích chi tiết về biểu đồ màu sắc và đường chéo chính của ma trận, hiệu suất phân loại của mô hình được đánh giá cụ thể.

Đối với ma trận nhầm lẫn không chuẩn hóa (Unnormalized Confusion Matrix), các ô màu đậm biểu thị giá trị cao và màu sắc đậm hơn trên đường chéo chính thể hiện giá trị lớn hơn. Ma trận trên cho thấy các ô kết quả dự đoán chính xác màu đậm đều tập trung trên đường chéo chính, các ô kết quả còn lại có màu sắc nhạt và các dự đoán sai lẫn nhau giữa các lớp là rất ít.

Đối với ma trận nhầm lẫn chuẩn hóa (Normalized Confusion Matrix), các ô trên đường chéo chính được màu sắc đậm thể hiện giá trị làm tròn 1,00, chỉ ra rằng mô hình đã dự đoán gần như chính xác tuyệt đối 100% cho tất cả các mẫu trong từng lớp. Các ô còn lại, có giá trị làm tròn 0.00 và màu sắc nhạt, cho thấy rằng gần như không có dự đoán sai giữa các lớp.

Từ kết quả phân tích ma trận nhầm lẫn trên, mô hình trí tuệ nhân tạo hỗ trợ drone DJI Tello nhận diện vật thể có dạng hình chữ nhật, hình tam giác, hình tròn, hình ngũ giác đều có khả năng dự đoán chính xác và đáng tin cậy các dạng vật thể trên.



Hình 6. Ma trận nhầm lẫn không chuẩn hóa (Unnormalized Confusion Matrix) và ma trận nhầm lẫn chuẩn hóa (Normalized Confusion Matrix)

Kết quả nhận diện thực tế

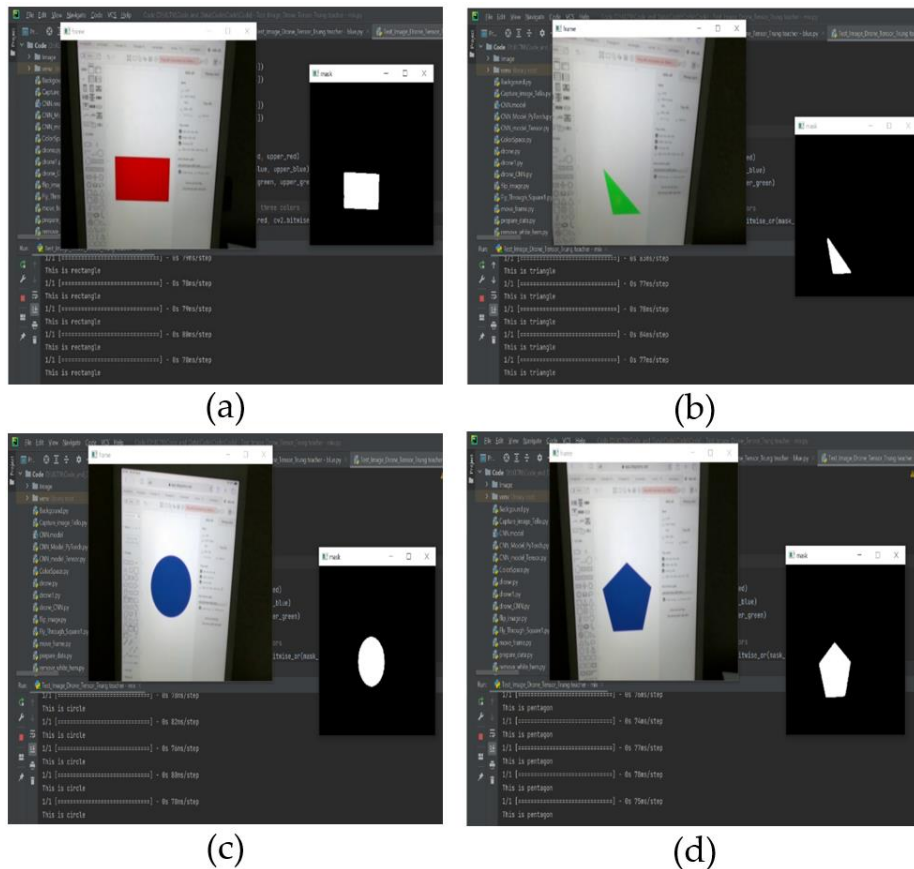
Để kiểm nghiệm độ chính xác của mô hình CNN đã huấn luyện và thực nghiệm kết quả thực tế trên drone DJI Tello, nghiên cứu đã thử nghiệm nhận diện với mỗi loại vật thể với các dạng hình học cơ bản là hình chữ nhật, hình tam giác, hình tròn, hình ngũ giác đều 50 lần, tổng 200 lần thực nghiệm. Mỗi loại vật thể đều luân phiên sử dụng 3 hệ màu khác nhau là đỏ, lục, lam để kiểm tra tính nhận diện màu sắc của mô hình.

Hình 7 thể hiện kết quả việc nhận diện các loại vật thể với 4 dạng hình học khác nhau là hình chữ nhật, hình tam giác, hình tròn, hình ngũ giác đều. Kết quả sau 200 lần thử nghiệm, drone

DJI Tello đều có thể phân biệt và nhận diện được 4 dạng vật thể với độ chính xác gần như tuyệt đối 100% với tốc độ nằm trong khoảng 70 đến 80 ms.

Kết quả nhận diện thực tế chứng minh mô hình CNN được huấn luyện phù hợp với thực tế và có thể sử dụng trong thế giới thực chứ không chỉ là mô phỏng. Điều này cũng chứng minh mô hình CNN đạt độ chính xác cao xấp xỉ 100% và không bị hiện tượng quá khớp (overfitting).

So với các kết quả nghiên cứu trong [22] và [23], mô hình của chúng tôi được thiết kế chuyên biệt cho bài toán nhận diện các hình dạng cụ thể trên nền tảng drone tài nguyên thấp. Bằng cách sử dụng bộ dữ liệu ảnh đen trắng được xây dựng chuyên biệt và kiến trúc CNN gọn nhẹ, hệ thống đạt được độ chính xác xấp xỉ 100% với tốc độ xử lý khoảng 70-80ms mỗi bước (như được ghi nhận trong Hình 7). Độ chính xác và tốc độ này nhanh hơn đáng kể so với Faster R-CNN được báo cáo trong [22], cho thấy sự phù hợp của phương pháp tiếp cận đối với các ứng dụng yêu cầu xử lý thời gian thực.



Hình 7. Kết quả nhận diện thực tế 4 dạng vật thể cơ bản (a) hình chữ nhật, (b) hình tam giác, (c) hình tròn, (d) hình ngũ giác thông qua camera của DJI Tello Drone

Mặc dù đạt độ chính xác gần như tuyệt đối trên bộ dữ liệu được xây dựng, nhưng hiệu suất này có thể thay đổi khi áp dụng vào các môi trường phức tạp hơn với các yếu tố như ánh sáng thay đổi, vật thể bị che khuất một phần hoặc các đối tượng có hình dạng phức tạp hơn. Các hướng nghiên cứu tiếp theo sẽ tập trung giải quyết những thách thức này để cải thiện khả năng tổng quát hóa của mô hình trong thế giới thực.

5 Kết luận

Nghiên cứu đã triển khai thành công mô hình mạng nơ-ron tích chập (CNN) tích hợp vào hệ thống camera của drone nhằm nhận diện và phân loại các vật thể có dạng hình học cơ bản. Kết quả thực nghiệm cho thấy mô hình đạt độ chính xác cao, khả năng phân loại ổn định và không xuất hiện hiện tượng quá khớp, khẳng định tính hiệu quả của phương pháp đề xuất. Nghiên cứu đã thử nghiệm với bốn dạng hình học gồm hình chữ nhật, tam giác, tròn và ngũ giác đều, cho thấy phương pháp có khả năng áp dụng cho nhiều dạng hình học đơn giản khác trong thực tế.

Bên cạnh đó, nghiên cứu xây dựng bộ dữ liệu ảnh chuyên biệt sử dụng hệ màu đen trắng, đồng thời tối ưu kích thước ảnh và tự động hóa quá trình tạo dữ liệu nhằm giảm chi phí và thời gian huấn luyện. Mô hình CNN được thiết kế phù hợp với bài toán, giúp trích xuất đặc trưng hiệu quả và nâng cao độ chính xác phân loại. Việc tích hợp mô hình vào camera của drone DJI Tello cho phép hệ thống nhận diện vật thể trực tiếp theo thời gian thực với hiệu suất xử lý ổn định.

Kết quả nghiên cứu góp phần khẳng định tiềm năng ứng dụng của trí tuệ nhân tạo và thị giác máy tính trong các hệ thống drone thông minh.

Lời cảm ơn

Công trình nghiên cứu này được tài trợ bởi Đề tài Khoa học và công nghệ cấp Đại học Huế, mã số: **DHH2023-06-137**.

Tài liệu tham khảo

1. R. Szeliski, *Computer vision: algorithms and applications*. Springer Nature, 2022.
2. D. Khurana, A. Koli, K. Khatter, and S. Singh, "Natural language processing: state of the art, current trends and challenges," *Multimed. Tools Appl.*, vol. 82, no. 3, pp. 3713–3744, Jan. 2023, doi: 10.1007/s11042-022-13428-4.
3. S. M. S. M. Daud et al., "Applications of drone in disaster management: A scoping review," *Sci. Justice*, vol. 62, no. 1, pp. 30–42, 2022.

4. H.-W. Choi, H.-J. Kim, S.-K. Kim, and W. S. Na, "An overview of drone applications in the construction industry," *Drones*, vol. 7, no. 8, p. 515, 2023.
5. D. Cvitanić, "Drone applications in transportation," in 2020 5th international conference on smart and sustainable technologies (SpliTech), 2020, pp. 1–4.
6. S. Ahirwar, R. Swarnkar, S. Bhukya, and G. Namwade, "Application of drone in agriculture," *Int. J. Curr. Microbiol. Appl. Sci.*, vol. 8, no. 01, pp. 2500–2505, 2019.
7. J. Shahmoradi, E. Talebi, P. Roghanchi, and M. Hassanalian, "A comprehensive review of applications of drone technology in the mining industry," *Drones*, vol. 4, no. 3, pp. 1–25, Sep. 2020, doi: 10.3390/drones4030034.
8. M.-A. Lahmeri, M. A. Kishk, and M.-S. Alouini, "Artificial intelligence for UAV-enabled wireless networks: A survey," *IEEE Open J. Commun. Soc.*, vol. 2, pp. 1015–1040, 2021.
9. Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, "A survey of convolutional neural networks: analysis, applications, and prospects," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 12, pp. 6999–7019, 2021.
10. Y. Akbari, N. Almaadeed, S. Al-maadeed, and O. Elharrouss, "Applications, databases and open computer vision research from drone videos and images: a survey," *Artif. Intell. Rev.*, vol. 54, no. 5, pp. 3887–3938, Jun. 2021, doi: 10.1007/s10462-020-09943-1.
11. D. H. Lee, "CNN-based single object detection and tracking in videos and its application to drone detection," *Multimed. Tools Appl.*, vol. 80, no. 26–27, pp. 34237–34248, Nov. 2021, doi: 10.1007/S11042-020-09924-0.
12. B. Lindqvist, S. S. Mansouri, A. Agha-mohammadi, and G. Nikolakopoulos, "Nonlinear MPC for collision avoidance and control of UAVs with dynamic obstacles," *IEEE Robot. Autom. Lett.*, vol. 5, no. 4, pp. 6001–6008, 2020.
13. B. Mishra, D. Garg, P. Narang, and V. Mishra, "Drone-surveillance for search and rescue in natural disaster," *Comput. Commun.*, vol. 156, pp. 1–10, 2020.
14. L. N. Dũng et al., "Nghiên cứu chế độ bay UAV trong khảo sát địa hình công trình dạng tuyến-ứng dụng cho đoạn đường đê Xuân Quan, Hà Nội," *Tạp Chí Khoa Học Công Nghệ Xây Dựng TCKHCNXD-ĐHXDHN*, vol. 15, no. 7V, pp. 131–142, 2021.
15. Hồ Anh Tuấn, "Nghiên cứu và thiết kế máy bay không người lái nhiều chong chóng mang phục vụ khảo sát địa hình," *Đại học Bách khoa Hà Nội, Hà Nội*, 2019.
16. M. Sỹ, B. Quý, P. Hiệp, and Lê Đình Quý, "Nghiên cứu sử dụng dữ liệu ảnh máy bay không người lái (UAV) trong thành lập bản đồ địa hình tỷ lệ lớn," *Tạp Chí Khoa Học Đo Đạc Và Bản Đồ*, no. 33–9, 2017.
17. Vũ Phan Long, Vũ Văn Chất, and Nguyễn Vũ Giang, "Bay chụp ảnh bằng máy bay không người lái (UAV) thành lập bản đồ không gian 3 chiều (3D)," *Tạp Chí Khoa Học Đo Đạc Và Bản Đồ*, vol. 31, 2017.
18. Đỗ Trọng Tuấn, "Nghiên cứu khả năng sử dụng ảnh hàng không chụp bằng máy bay không người lái (UAV) trong thành lập bản đồ địa chính (Thử nghiệm tại xã Vật Lại, huyện Ba Vì, thành phố Hà Nội)," *Trường Đại học Khoa học Tự nhiên, Hà Nội*, 2015.
19. Đỗ Thị Nhung et al., "Nghiên cứu mô hình phát hiện rác thải nhựa ven biển sử dụng ảnh máy bay không người lái và mạng nơ-ron tích chập sâu," *Tạp Chí Khoa Học Đo Đạc Và Bản Đồ*, vol. 49, 2021.
20. A. C. Roy, K. Hossin, M. P. Uddin, M. A. Al Mamun, M. I. Afjal, and A. M. Nitu, "Detection and classification of geometric shape objects for industrial applications," *Adv. Image Process. Pattern Recognit.*, vol. 1, no. 2, pp. 11–19, 2018.

21. T. Saini, "Manoeuvring drone (Tello and Tello EDU) using body poses or gestures," Polytechnic University of Catalonia, 2021.
22. C. Sun, W. Zhan, J. She, and Y. Zhang, "Object Detection from the Video Taken by Drone via Convolutional Neural Networks," *Math. Probl. Eng.*, vol. 2020, pp. 1–10, Oct. 2020, doi: 10.1155/2020/4013647.
23. A. Rohan, M. Rabah, and S.-H. Kim, "Convolutional neural network-based real-time object detection and tracking for parrot AR drone 2," *IEEE Access*, vol. 7, pp. 69575–69584, 2019.
24. "draw.io." Accessed: Dec. 24, 2024. [Online]. Available: <https://app.diagrams.net/>
25. Ryze Tech, "SDK 2.0 User Guide."
26. Damià Fuentes Escoté and Jakob Löw, "DJITelloPy – Python interface for DJI Tello drones." Accessed: Jun. 13, 2025. [Online]. Available: <https://github.com/damiafuentes/DJITelloPy>