

Xây dựng thuật toán bám đối tượng trên cơ sở xử lý ảnh cho UAV quadrotor

Phan Văn Đạt, Lê Bá Chung, Hoàng Quang Chính*

Trường Đại học Kỹ thuật Lê Quý Đôn, 236 Hoàng Quốc Việt, Bắc Từ Liêm, Hà Nội, Việt Nam.

*Tác giả liên hệ: chinhhq@lqdtu.edu.vn

Nhận bài: 04/9/2024; Hoàn thiện: 06/11/2024; Chấp nhận đăng: 15/11/2024; Xuất bản: 06/12/2024.

DOI: <https://doi.org/10.54939/1859-1043.j.mst.FEE.2024.65-71>

TÓM TẮT

Bài báo trình bày kết quả xây dựng thuật toán và thực nghiệm điều khiển bám đối tượng ứng dụng xử lý ảnh cho UAV quadrotor sử dụng mô hình động lực học của UAV quadrotor, mô hình học sâu và được tối ưu hóa cho hệ thống phân cứng. Hệ thống phân cứng sử dụng camera đo sâu, máy tính nhúng để thu thập dữ liệu hình ảnh, thông tin khoảng cách của đối tượng và tính toán, xử lý để đưa ra các tín hiệu điều khiển các động cơ thông qua bộ điều khiển bay nhằm đảm bảo bám và giữ khoảng cách giữa UAV và đối tượng. Kết quả đạt được của bài báo có thể đánh giá khả năng đáp ứng thời gian thực bài toán bám đối tượng dùng xử lý ảnh với phân cứng có tài nguyên hạn chế.

Từ khoá: Thiết bị bay không người lái; Quadrotor; Mạng học sâu.

1. MỞ ĐẦU

Một trong những thiết bị bay không người lái (UAV - Unmanned Aerial Vehicle) được biết đến và sử dụng phổ biến nhất là UAV quadrotor do chúng có cấu tạo điển hình, đối xứng. Chúng được ứng dụng rộng rãi trong nhiều lĩnh vực như công nghiệp [1], giáo dục [2], y tế [3], hàng không vũ trụ [4], quân sự [5],... Cùng với sự phát triển của kỹ thuật thị giác máy tính, ngày càng nhiều sản phẩm tích hợp thị giác máy tính vào để tạo ra tính linh hoạt trong điều khiển của sản phẩm, trong đó có UAV nói chung và UAV quadrotor nói riêng. Thị giác máy tính là kỹ thuật cho phép nhận dạng, định vị đối tượng và từ đó đưa ra các thông tin điều khiển cần thiết giúp cho sản phẩm thực hiện các bài toán khác nhau, cụ thể như bài toán bám đối tượng. Ngày nay, cùng với sự phát triển của công nghệ máy tính và các thuật toán thông minh, những công nghệ này đã được áp dụng vào trong thị giác máy tính và làm cho chất lượng nhận dạng, định vị đối tượng và đưa ra được những thông tin điều khiển chính xác hơn [6-9].

Trong những năm gần đây, có nhiều công trình nghiên cứu về UAV, các nhà nghiên cứu đã tìm giải pháp, thuật toán nhằm nâng cao hiệu suất bám đối tượng dựa trên thị giác máy tính trong các điều kiện khác nhau. Trong công trình [10] các tác giả đã sử dụng phương pháp phân lớp đặc trưng Haar Casade để định vị tâm gương mặt con người, từ đó tính toán các tham số điều khiển cho UAV bám theo gương mặt người, tuy nhiên, phương pháp này gặp khó khăn khi gương mặt bị nghiêng hoặc đột ngột bị che khuất. Công trình [11] các tác giả đã phát triển một phương pháp phát hiện đối tượng di chuyển hiệu quả trong ảnh chụp từ trên cao dựa trên kỹ thuật tách nền và khác biệt khung hình, tuy nhiên, phương pháp này có hạn chế là khi gặp phải rung động tần số cao trên UAV. Ngoài ra, nghiên cứu [12] đã sử dụng mô hình học sâu YoloV3 và thuật toán bám đối tượng KCF (Kernelized Correlation Filters) để tăng hiệu suất cho hệ thống UAV. Tuy nhiên, nghiên cứu này yêu cầu sử dụng phân cứng có cấu hình cao, tương tự như trong công trình [13]. Ngoài ra, mô hình thử nghiệm sử dụng cơ cấu gá cứng camera làm giảm tính linh hoạt và phạm vi quan sát của UAV. Trong nghiên cứu [7] sử dụng YoloV2 để nhận dạng đối tượng người cho UAV quadrotor, tuy nhiên, tốc độ xử lý của thuật toán còn có độ trễ cao. Tại Việt Nam, các công trình nghiên cứu điều khiển UAV bám theo đối tượng cũng đã được công bố. Trong [14], nhóm tác giả sử dụng mô hình nhận dạng tư thế người và mạng nơ-ron LSTM (Long Short Term Memory networks) để xác định hướng di chuyển của con người từ đó điều khiển UAV sao cho người luôn ở trong tâm camera, tuy nhiên, nghiên cứu này chưa đề cập tới khoảng cách cụ thể giữa đối tượng và UAV.

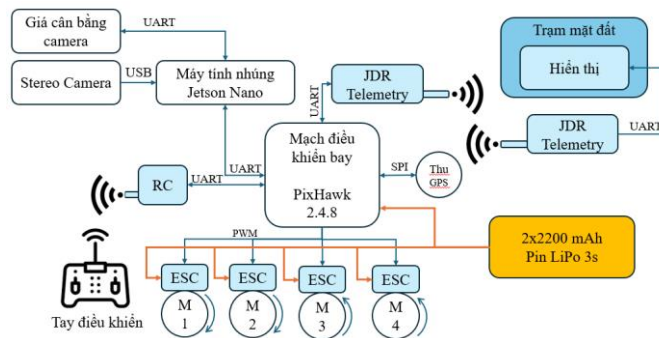
Từ những phân tích trên, có thể đưa ra nhận xét rằng, kỹ thuật phát hiện đối tượng và điều khiển bám đối tượng đã đạt được nhiều kết quả, trong đó có cả việc tích hợp các kỹ thuật AI (Artificial Intelligence) nhằm nâng cao chất lượng nhận dạng đối tượng. Tuy nhiên, vẫn còn những tồn tại như khối lượng tính toán xử lý nhận dạng đối tượng còn lớn, do đó, tiêu chí thời gian thực gặp khó khăn. Do đó, nội dung nghiên cứu của bài báo tập trung vào tích hợp thị giác máy tính với yêu cầu xử lý thời gian thực hoặc gần thực bằng các thuật toán tối ưu cùng với thuật toán điều khiển tự động bám đối tượng cho UAV cấu hình thấp nói chung và UAV quadrotor nói riêng.

Cấu trúc bài báo gồm: Phần khái quát về các vấn đề liên quan đến xây dựng thuật toán xác định vị trí 3 chiều của đối tượng và chương trình sử dụng mô hình học sâu kết hợp thuật toán bám đối tượng được trình bày trong mục 2. Trong mục 3 trình bày kết quả thực nghiệm và đánh giá và cuối cùng là phần kết luận của bài báo.

2. CƠ SỞ XÂY DỰNG HỆ THỐNG BẮM ĐỐI TƯỢNG TRÊN CƠ SỞ XỬ LÝ ẢNH

2.1. Cấu trúc chung hệ thống bám đối tượng trên cơ sở xử lý ảnh

Nhóm tác giả đề xuất sơ đồ cấu trúc của hệ thống bám đối tượng trên cơ sở xử lý ảnh cho UAV quadrotor (hình 1).



Hình 1. Sơ đồ chức năng hệ thống bám đối tượng trên cơ sở xử lý ảnh cho UAV quadrotor.

Cấu trúc hệ thống bao gồm: Giám sát và điều khiển từ xa (gồm trạm mặt đất, tay cầm điều khiển), điều khiển trung tâm (máy tính nhúng), mạch điều khiển bay, khuếch đại công suất, động cơ, hệ thống camera (giá cân bằng và stereo camera) và nguồn.

2.2. Phát hiện, nhận dạng và bám đối tượng

2.2.1. Phát hiện và nhận dạng đối tượng

Thuật toán nhận dạng và bám đối tượng trong bài báo được xây dựng dựa trên mô hình YoloV8 [15] được huấn luyện, tối ưu cho máy tính nhúng Jetson Nano sử dụng GPU của NVIDIA tích hợp và thuật toán ByteTrack [16] giúp tối ưu tốc độ xử lý.

a) Bài toán phát hiện và nhận dạng đối tượng

Mô hình nhận dạng đối tượng được sử dụng trong bài báo được xây dựng dựa trên mô hình YoloV8 nano của Ultralytics [15]. Mô hình YOLO (You Only Look Once) sử dụng các lớp tích chập (convolutional layers) để trích xuất đặc trưng hình ảnh và các lớp mạng nơ-ron được nối đầy đủ để dự đoán xác suất và tọa độ của đối tượng, với đầu vào là một hình ảnh và đầu ra là các khung bao quanh đối tượng và nhãn tương ứng.

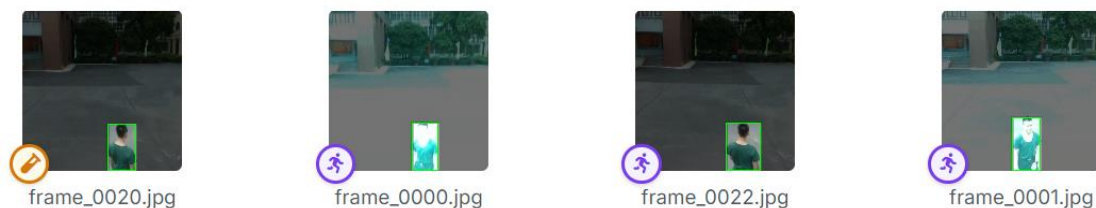
Mục tiêu của nghiên cứu của bài báo là nhận dạng và bám theo đối tượng con người, vì vậy, để tăng độ chính xác cho bài toán nhận dạng, nhóm tác giả xây dựng bộ dữ liệu về đối tượng người với góc nhìn thu được từ camera trên UAV để tiến hành huấn luyện mô hình. Bộ dữ liệu bao gồm 3000 ảnh với các ảnh xoay, nghiêng, mờ tương ứng với các môi trường ánh sáng yếu, đầy đủ ánh sáng giúp mô hình có thể học để cho ra xác suất nhận dạng chính xác cao nhất trong các điều kiện môi

trường khác nhau. Trong quá trình huấn luyện mô hình, dữ liệu được chia thành ba phần chính: bộ dữ liệu huấn luyện, bộ dữ liệu xác nhận và bộ dữ liệu kiểm tra. Trong bài báo này, tỉ lệ ba thành phần này được chọn tương ứng là 70%, 20%, 10% và được gán dẫn và tạo trên RoboFlow [17].

Tính chính xác của mô hình sau huấn luyện được đánh giá qua thông số độ chính xác (Precision). Độ chính xác [18] là thước đo quan trọng trong lĩnh vực phân loại và phát hiện đối tượng trong học máy và được xác định như sau:

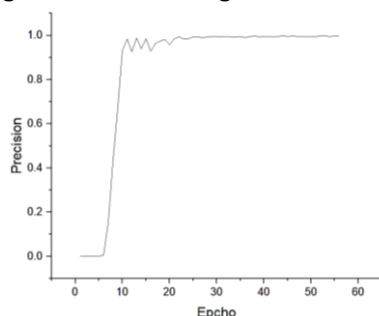
$$Precision = \frac{TP}{TP + FP} \quad (1)$$

Trong đó: True Positives (TP) là số lượng các trường hợp mà mô hình dự đoán đúng; False Positives (FP) là số lượng các trường hợp mà mô hình dự đoán sai.



Hình 2. Bộ dữ liệu huấn luyện mô hình được xây dựng trên RoboFlow.

Trong huấn luyện mô hình toàn bộ quá trình được thực hiện và thể hiện trên nền tảng Comet [19]. Comet là một công cụ mạnh mẽ cho phép quản lý, hiển thị các kết quả trong quá trình huấn luyện và trực quan hóa kết quả trong thời gian thực dưới dạng các biểu đồ (hình 3).



Hình 3. Kết quả độ chính xác sau huấn luyện.

b) Tối ưu mô hình cho phần cứng của Jetson Nano

Nhóm tác giả sử dụng TensorRT [20] nhằm tối ưu hóa mô hình học sâu đã được huấn luyện nhằm tăng khả năng xử lý thời gian thực của hệ thống. TensorRT tối ưu các tham số: hiệu chỉnh độ chính xác (Precision Calibration), kết hợp lớp và tensor (Layer and Tensor Fusion), tối ưu hóa tham số của nhân tính toán trên GPU (Kernel Auto-Tuning), quản lý bộ nhớ động (Dynamic Tensor Memory), thực thi đa luồng (Multiple-Stream Execution) để tối ưu quá trình tính toán trên phần cứng do hãng NVIDIA cung cấp. Các bước tiến hành để tối ưu hóa mô hình được thực hiện như sau:

1. Sử dụng Framework bất kỳ để huấn luyện mô hình YoloV8n.pt nhận dạng đối tượng mong muốn.
2. Chuyển đổi về một mạng nơ-ron trung gian. Để chuyển đổi về định dạng TensorRT, mô hình phải có bước chuyển đổi trung gian về định dạng .uff (thích hợp với các mô hình Tensorflow) hoặc .onnx (thích hợp với nhiều loại mô hình khác nhau).
3. Sử dụng hàm chuyển đổi của TensorRT SDK để chuyển đổi từ .onnx hoặc .uff sang định dạng TensorRT.
4. Thu được mô hình TensorRT, có định dạng .engine.
5. Sử dụng hàm load_model (của thư viện TensorRT) để khởi động mô hình.

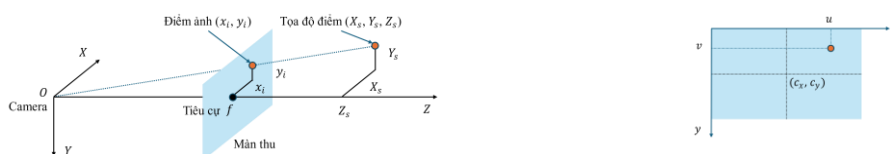
2.2.2. Bài toán bám đối tượng

Bài toán bám đối tượng có thể chia làm hai loại: bám một đối tượng và bám nhiều đối tượng. Nhóm tác giả sử dụng thuật toán ByteTrack để giải quyết bài toán bám nhiều đối tượng trong thời gian thực với việc sử dụng mạng nơron sâu. ByteTrack có thể bám các đối tượng một cách chính xác hơn, ngay cả khi chúng bị che khuất hoặc biến mất khỏi khung hình. Hoạt động dựa trên ba mô hình mạng nơron sâu: Mô hình phát hiện sử dụng YOLOX; Mô hình biểu diễn trích xuất các đặc trưng của đối tượng được phát hiện; Mô hình liên kết liên kết các đặc điểm của các đối tượng được phát hiện trong các khung hình liên tiếp để tạo ra các đường dẫn bám. So với các thuật toán bám nhiều đối tượng, ByteTrack có khả năng xử lý video nhanh hơn tuy nhiên độ chính xác thấp hơn [16].

2.2.3. Cơ sở xây dựng thuật toán xác định tọa độ 3D và bám đối tượng

Xác định tọa độ đối tượng là đầu vào quan trọng của bài toán bám đối tượng, tọa độ của đối tượng được tính toán và chuyển đổi về hệ tọa độ gắn với UAV quadrotor. Hình 4 mô tả cách xác định tọa độ đối tượng từ tọa độ của điểm đối tượng trên màn thu của camera.

Tọa độ của điểm đối tượng được xác định bởi công thức (2) với tham số Z_s được thu từ ảnh độ sâu của camera đo sâu Realsense D435i.



Hình 4. Tọa độ điểm đối tượng trong khung tọa độ camera.

$$\begin{cases} X_s = Z_s \frac{u - c_x}{f} \\ Y_s = Z_s \frac{v - c_y}{f} \end{cases} \quad (2)$$

Trong đó: X_s, Y_s, Z_s [m] là tọa độ của đối tượng trong khung tọa độ camera; u, v, c_x, c_y [m] là tọa độ của đối tượng và tâm ảnh trong khung tọa độ màn thu; x_i, y_i [m] là tọa độ điểm ảnh đối tượng trên màn thu trong khung tọa độ camera; f [m] là tiêu cự của camera.

Từ hình 4, Z_s là khoảng cách giữa đối tượng và camera trong hệ tọa độ camera. Do camera được gắn vào UAV thông qua một giá cân bằng và luôn luôn nghiêng một góc $\alpha=45^\circ$ không đổi so với phương ngang. Do đó, khoảng cách D_s từ camera đến đối tượng theo phương ngang được xác định bởi công thức sau:

$$D_s = Z_s \cdot \cos(\alpha) \quad (3)$$

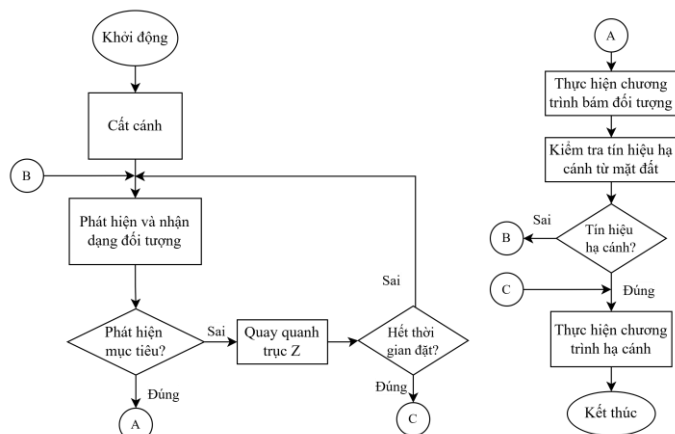
Theo thông số kỹ thuật của Realsense D435i có tốc độ chụp là 30 fps, tuy nhiên, cần nhiều thời gian xử lý ảnh để xác định các thông số đối tượng, qua thực nghiệm của nhóm tác giả, tốc độ chụp giảm xuống còn 10 fps. Trong bài báo này, nhằm đảm bảo không mất thông tin bám đối tượng, các tác giả sử dụng khoảng cách dịch chuyển của đối tượng giữa hai khung hình là 0,15m. Do đó, tốc độ chuyển động của đối tượng lớn nhất được xác định như sau:

$$s = \text{fps} \times e_s \quad (4)$$

Trong đó: s là tốc độ của đối tượng, trong bài báo này $s = 10 \times 0,15 = 1,5$ m/s; fps là tốc độ thực tế của camera khi sử dụng; e_s là khoảng dịch chuyển của đối tượng giữa hai khung hình.

2.3. Xây dựng thuật toán điều khiển bám đối tượng

Thuật toán bám đối tượng (hình 5) được xây dựng dựa trên kết hợp khả năng nhận dạng của mô hình YoloV8n được huấn luyện và thuật toán bám đối tượng ByteTrack.

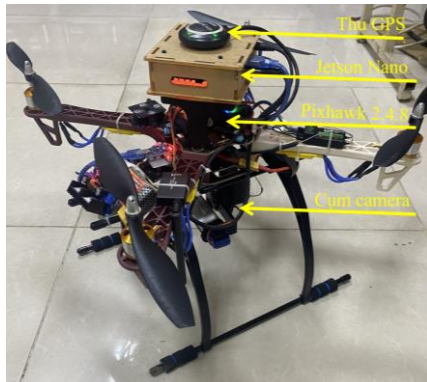


Hình 5. Lưu đồ thuật toán bám đối tượng.

3. THỰC NGHIỆM VÀ ĐÁNH GIÁ

3.1. Thiết bị thử nghiệm

Mô hình thực nghiệm được xây dựng trên khung UAV quadrotor F450 (hình 6, hình 7) gồm các thành phần chính sau: máy tính nhúng Jetson Nano để xử lý thuật toán học sâu và bám đối tượng, bộ điều khiển Pixhawk 2.4.8 điều khiển 04 động cơ, hệ thống cảm biến phục vụ bay gồm cảm biến áp suất, thiết bị thu GPS, bộ thu RF, camera đo sâu Realsense D435i được gắn trên giá cân bằng để nâng cao chất lượng ảnh. Ngoài ra, trạm mặt đất gồm máy tính với phần mềm giám sát bay Mission Planner, tay cầm điều khiển FlySky i6 và bộ thu phát tín hiệu RF.



Hình 6. UAV quadrotor dùng thử nghiệm.



Hình 7. Thiết bị và công cụ đồng bộ.

3.2. Kịch bản thử nghiệm

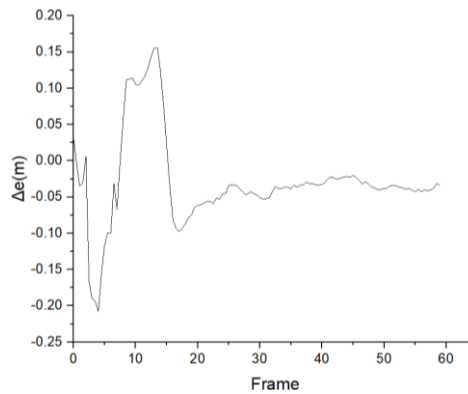
3.2.1. Kịch bản thử nghiệm 1

Thực hiện thử nghiệm UAV quadrotor nhận dạng và bám theo đối tượng đơn, không bị che khuất trong điều kiện đủ ánh sáng. Mục tiêu của thử nghiệm này là đánh giá khả năng nhận dạng đối tượng và điều khiển UAV bám theo đối tượng khi đối tượng di chuyển với tốc độ chậm và duy trì khoảng cách tương đối bằng 4 mét.

Trên hình 8 là ảnh 640x480 pixel được thu về từ camera gắn trên UAV, sau đó được kết hợp với ảnh độ sâu và xử lý bởi thuật toán xác định vị trí đối tượng so với UAV để đưa ra sai lệch giữa đối tượng và tâm khung hình theo trục x cùng khoảng cách tương đối tới đối tượng nhằm điều khiển chuyển động của UAV. Hình 9 trình bày sai lệch bám trong quá trình điều khiển bám đối tượng khi đối tượng di chuyển.



Hình 8. Hình ảnh bám đối tượng.



Hình 9. Sai lệch bám trong quá trình bám đối tượng.

Từ kết quả thực nghiệm của kịch bản 1, UAV đã nhận dạng và bám được đối tượng với độ sai lệch bám khi ổn định nhỏ hơn 0.05m. So sánh với nghiên cứu [13] về thông số sai số xác lập theo phương x, kết quả của thực nghiệm cho thấy thuật toán áp dụng trên phần cứng giá rẻ có kết quả gần bằng nhau, nhưng về thời gian xử lý của hệ thống chậm hơn.

3.2.2. Kịch bản thử nghiệm 2

Kịch bản thử nghiệm 2 nhằm thử nghiệm nhằm đánh giá khả năng tiếp tục bám đối tượng khi xuất hiện các đối tượng tương tự trong khung hình camera của UAV quadrotor.



Hình 10. Thực nghiệm thuật toán điều khiển bám đối tượng kịch bản 2.

Trong hình 10 trình bày kết quả bám đối tượng của UAV quadrotor khi đang bám đối tượng ban đầu, sau đó đối tượng bị che khuất. Kết quả trên hình cho thấy rằng, UAV vẫn duy trì quá trình bám đối tượng và không có sự thay đổi đột ngột khi đối tượng bị che khuất và xuất hiện lại.

Từ những kịch bản trên có thể kết luận rằng, thuật toán bám đối tượng cho UAV quadrotor mà nhóm tác giả xây dựng đã làm việc tốt, cho phép nhận dạng, điều khiển bám đối tượng với độ sai số bám nhỏ hơn 0.05 m và duy trì được quá trình bám sau khi đối tượng đang bám bị che bởi đối tượng tương tự.

4. KẾT LUẬN

Trong bài báo này đã trình bày việc xây dựng hệ thống bám đối tượng dựa trên xử lý ảnh cho UAV quadrotor. Nhóm tác giả đã xây dựng thành công thuật toán điều khiển UAV quadrotor bám đối tượng trên cơ sở mô hình học sâu YoloV8n và thuật toán ByteTrack. Bài báo đã đưa ra phương pháp cụ thể, từng bước xây dựng, tối ưu và đánh giá thuật toán bám đối tượng là người. Kết quả thực nghiệm với hai kịch bản đã chứng minh hoạt động của thuật toán đã được xây dựng. Kết quả nghiên cứu này sẽ làm cơ sở cho việc phát triển các hệ thống bám đối tượng thời gian thực phục vụ trong các bài toán thực tế khác nhau và làm cơ sở cho việc nghiên cứu phát triển các bài toán khác cho UAV với công nghệ thị giác máy tính.

TÀI LIỆU THAM KHẢO

- [1]. M. Dimitris, A. John and P. Nikos, "UAVs for Industrial Applications: Identifying Challenges and Opportunities from the Implementation Point of View", *Procedia Manufacturing Procedia Manufacturing*, v. 55, 183-190, (2021).
- [2]. R. Shadie and Y. Suping, "A systematic review of UAV applications to education", (2022).
- [3]. Fakhar-I-Adil, Qureshi, A. Khan and Deeba, "Applications of Medical Drones in Public Health: An Overview", (2021).
- [4]. A. A. Laghari, A. K. Jumani, R. A. Laghari and H. Nawaz, "Unmanned Aerial Vehicles: A Review", (2023).
- [5]. F. Li, X. Fuhuai, W. Dongzheng and M. Guanglei, "The overview for UAV Air-Combat Decision method", (2014).
- [6]. W. Xudong, L. Bo and Z. Zhang, "Application of UAV Target Tracking Based on Computer," (2021).
- [7]. B. Kamel and R. Naeem, "Human detection based on deep learning YOLO-v2 for real-time," (2021).
- [8]. K. Lesole, W. Wanggen and H. Li, "Analysis Based on Recent Deep Learning Approaches Applied in Real-Time Multi-Object Tracking: A Review," (2021).
- [9]. M. Mandal and S. K. Vipparthi, "An Empirical Review of Deep Learning Frameworks for Change Detection: Model Design, Experimental Frameworks, Challenges and Research Needs," (2022).
- [10]. M. Karahan, H. Kurt and C. Kasnakoglu, "Autonomous Face Detection and Tracking Using Quadrotor UAV," (2020).
- [11]. H. C. Hsien, W. Y. Ta, K. Jau-Hong, S. Ming-Yu and C. Cheng-Chuan, "A hybrid moving object detection method for aerial images," (2010).
- [12]. H. Saribas, B. Uzun, B. Benligiray, O. Eker and H. Cevikalp, "A Hybrid Method for Tracking of Objects by UAVs," (2019).
- [13]. L.-Y. Lo, C. H. Yiu, Y. Tang, A.-S. Yang, B. Li and C.-Y. Wen, "Dynamic Object Tracking on Autonomous UAV System for Surveillance Applications," (2021).
- [14]. M. Phạm Trung and T. Nguyễn Hoàng Thùy, "Mô hình điều khiển drone tự phát hiện và bám theo hướng di chuyển của con người," (2022).
- [15]. G. Wang, Y. Chen, P. An, H. Hong, J. Hu and T. Huang, "UAV-YOLOv8: A Small-Object-Detection Model Based on Improved YOLOv8 for UAV Aerial Photography Scenarios," (2023).
- [16]. Y. Zhang, P. Sun, Y. Jiang, D. Yu, F. Weng, Z. Yuan, P. Luo, W. Liu and X. Wang, "ByteTrack: Multi-Object Tracking by Associating Every Detection Box," (2021).
- [17]. G. Ian, B. Yoshua and C. Aaron, "Deep Learning," (2016).
- [18]. P. Rafael, N. S. L. and S. Eduardo, "A Survey on Performance Metrics for Object-Detection Algorithms," (2020).
- [19]. "Comet ML," [Online]. Available: <https://www.comet.com/site/about-us/>.
- [20]. N. Developer, "NVIDIA TensorRT," [Online]. Available: <https://developer.nvidia.com/tensorrt>.

ABSTRACT

Building an object tracking algorithm based on image processing for UAV quadrotor

The paper presents the results of algorithm construction and the experimental control of object tracking based on image processing for quadrotor UAV using a dynamic model of quadrotor UAV, deep learning model and optimized for hardware system. The hardware system uses a depth camera, embedded computer to collect image data, object distance information and calculate, process to give control signals to the motors through the flight controller to ensure tracking and maintain distance between UAV and object. The results achieved in the paper can evaluate the ability to respond in real time to the problem of object tracking using image processing with hardware with limited resources.

Keywords: Drones; Quadrotors; Deep learning networks.