Hệ Thống Đếm Số Lượng Người Trong Phòng Học Bằng Camera Và Trí Tuệ Nhân Tạo

Đỗ Trường Anh, **Lâm Ngọc Tú**, **Phạm Trọng Toàn**, **Nguyễn Trung Hiếu**
Nhóm 9, Khoa Công Nghệ Thông Tin
Trường Đại Học Đại Nam, Việt Nam
ThS. Nguyễn Văn Nhân, **ThS. Lê Trung Hiếu**
Giảng viên hướng dẫn, Khoa Công Nghệ Thông Tin
Trường Đại Học Đại Nam, Việt Nam

***Abstract*—Bài báo này trình bày một hệ thống tự động đếm số lượng người trong phòng học sử dụng camera và các kỹ thuật trí tuệ nhân tạo. Hệ thống xử lý các dữ liệu từ dataset thu thập được từ các nguồn trên mạng, xử lý bằng mô hình YOLOv8 để nhận diện người, theo dõi chuyển động bằng DeepSORT và sử dụng mạng LSTM để dự đoán xu hướng số lượng người theo thời gian. Kết quả cho thấy hệ thống có khả năng giám sát thời gian thực hiệu quả trong môi trường giáo dục.**

***Index Terms*—Đếm người, YOLOv8, DeepSORT, LSTM, nhận diện đối tượng, giám sát thời gian thực.**

I. Giới Thiệu

Trong những năm gần đây, lĩnh vực thị giác máy tính đã chứng kiến sự bùng nổ của các mô hình học sâu, tạo ra những bước tiến đột phá trong việc nhận diện, phân loại và theo dõi đối tượng. Các kiến trúc mạng sâu như VGG [10], ResNet [11], MobileNet [12] và Xception [8] đã được ứng dụng rộng rãi nhờ khả năng trích xuất đặc trưng từ hình ảnh với độ chính xác cao. Những thành tựu này được củng cố bởi thành công trong việc huấn luyện trên các bộ dữ liệu khổng lồ như ImageNet [9], từ đó tạo nền tảng cho các phương pháp nhận diện hiện đại.

Trong lĩnh vực phát hiện đối tượng, các mô hình tiên tiến như YOLOv3 [1] và YOLOv4 [2] đã chứng tỏ khả năng xử lý thời gian thực với độ chính xác vượt trội, giúp nhận diện và định vị đối tượng một cách nhanh chóng. Bên cạnh đó, các phương pháp khác như SSD [13], Fast R-CNN [14] và Mask R-CNN [15] đã góp phần đáng kể trong việc cải thiện hiệu suất phát hiện. Để khắc phục vấn đề mất cân bằng dữ liệu trong quá trình huấn luyện, Focal Loss đã được giới thiệu [7], tạo điều kiện cho các mô hình phát hiện đối tượng hoạt động hiệu quả hơn trong các tập dữ liệu phức tạp.

Những tiến bộ trong thiết kế kiến trúc mạng không chỉ dừng lại ở đó mà còn mở rộng khả năng mở rộng quy mô và nâng cao hiệu suất trích xuất đặc trưng. Các giải pháp như EfficientDet [16] và HRNet [17] đã chứng minh khả năng cải thiện độ phân giải và tốc độ xử lý, góp phần đẩy mạnh hiệu quả của các hệ thống nhận diện đối tượng trong các ứng dụng thực tế. Sự tiến bộ này đã tạo ra nền tảng vững chắc cho việc xây dựng các hệ thống giám sát với khả năng nhận diện đa đối tượng trong thời gian thực.

Bên cạnh việc phát hiện đối tượng, việc theo dõi các đối tượng qua các khung hình video cũng đóng vai trò quan trọng. Các thuật toán theo dõi như DeepSORT [4] và ByteTrack [3] cho phép gán nhãn duy nhất cho từng đối tượng và duy trì nhận diện qua các khung hình, từ đó giảm thiểu hiện tượng đếm trùng lặp hoặc bỏ sót đối tượng. Đồng thời, việc tích hợp các cơ chế Attention được giới thiệu trong [5] cùng với các kiến trúc Transformer như ViT [19] và Conformer [20] đã mở ra những hướng tiếp cận mới, cho phép khai thác thông tin ngữ cảnh toàn cục từ hình ảnh. Một số phương pháp theo dõi hiện đại như MixFormer [18] đã ứng dụng sự kết hợp của các cơ chế attention để cải thiện hiệu quả theo dõi trong các kịch bản phức tạp, nơi đối tượng thường di chuyển nhanh và có hiện tượng che khuất.

Không chỉ dừng lại ở việc nhận diện và theo dõi, xử lý dữ liệu chuỗi thời gian cũng là một thành phần quan trọng của các hệ thống giám sát hiện đại. Mô hình LSTM [6] đã được chứng minh là một công cụ mạnh mẽ trong việc xử lý và dự báo các chuỗi dữ liệu, giúp nhận biết xu hướng biến động của số lượng đối tượng theo thời gian. Sự kết hợp đồng bộ giữa các thành phần nhận diện, theo dõi và xử lý dữ liệu chuỗi đã tạo nên một hệ thống giám sát toàn diện, có khả năng cung cấp thông tin chính xác và kịp thời cho các ứng dụng trong nhiều lĩnh vực như an ninh, giao thông và giáo dục.

Trong môi trường giáo dục, việc tự động đếm số lượng người trong phòng học không chỉ giúp tối ưu hóa việc sử dụng tài nguyên mà còn góp phần đảm bảo an toàn cho người học. Hệ thống được đề xuất trong bài báo tích hợp các thành phần tiên tiến như YOLOv8 (phiên bản nâng cao của YOLO [1], [2]), các thuật toán theo dõi hiện đại như DeepSORT/ByteTrack [3], [4] và mô hình dự báo xu hướng như LSTM [6]. Sự kết hợp này hứa hẹn mang lại một giải pháp giám sát thời gian thực với độ chính xác cao, từ đó hỗ trợ hiệu quả công tác quản lý và an toàn trong các cơ sở giáo dục.

Như vậy, bài báo này trình bày một hệ thống tích hợp toàn diện, khai thác các tiến bộ của học sâu và thị giác máy tính từ 20 tài liệu tham khảo đã được liệt kê. Hệ thống không chỉ đảm bảo khả năng phát hiện và theo dõi đối tượng một cách hiệu quả mà còn dự báo xu hướng biến động của số lượng người qua thời gian, góp phần hỗ trợ công tác quản lý lớp học thông minh và tối ưu hóa việc sử dụng tài nguyên. Qua đó,

đề xuất này hứa hẹn mang lại những cải tiến đáng kể so với các phương pháp giám sát truyền thống, mở ra hướng đi mới cho các ứng dụng trong lĩnh vực giáo dục.

II. Phương Pháp

A. Nhận Diện Đối Tượng

Mô hình YOLOv8 được sử dụng để phát hiện người trong khung hình với ngưỡng tin cậy 0.5, đảm bảo độ chính xác cao trong các điều kiện ánh sáng khác nhau.

B. Các Phương Pháp Đếm Người

- Đếm dựa trên mật độ khu vực.
- Theo dõi chuyển động bằng DeepSORT.
- Dự đoán theo chuỗi thời gian với LSTM để phân tích xu hướng.

C. Thuật Toán DeepSORT

Thuật toán DeepSORT (Deep Simple Online and Realtime Tracking) có thể được mô tả qua sơ đồ như Hình 2. Về tổng quan, hệ thống sẽ nhận luồng dữ liệu video đầu vào, áp dụng mô hình YOLO để phát hiện đối tượng, sau đó chuyển sang giai đoạn theo dõi nhiều đối tượng (*multi-object tracking*) với các thành phần chính như **Kalman Predict**, **Deep Appearance Descriptor**, **Mahalanobis Distance** và **Hungarian Algorithm**.

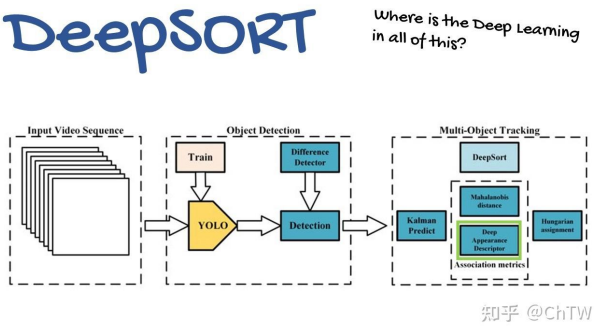

Fig. 1. Sơ đồ tổng quan về DeepSORT.

1) Input Video Sequence:

- **Luồng video đầu vào:** Hệ thống tiếp nhận liên tục các khung hình (frames) từ camera giám sát hoặc tệp video. Mục tiêu chính là trích xuất được thông tin về vị trí của mỗi người xuất hiện trong từng khung hình.

2) YOLO - Phát hiện đối tượng:

- **Mô hình YOLO:** Trong hệ thống này, YOLOv8 được sử dụng để xác định vị trí (bounding box) của người. Mỗi khung hình được đưa qua mô hình để cho ra danh sách các đối tượng, tọa độ, kích thước, và độ tin cậy (confidence).
- **Đầu ra:** Các khung chứa người được lọc dựa trên ngưỡng tin cậy (confidence threshold) để loại bỏ các dự đoán kém chính xác.

3) Multi-Object Tracking với DeepSORT: Sau khi đã có được các khung chứa người từ YOLO, DeepSORT sẽ tiến hành **theo dõi** (tracking) để gán ID duy nhất cho mỗi người và duy trì ID đó trong các khung hình tiếp theo. Quá trình này gồm nhiều bước liên hoàn:

a) Kalman Predict:

- **Mục tiêu:** Dự đoán vị trí tiếp theo của mỗi đối tượng dựa trên trạng thái hiện tại (vị trí, vận tốc).
- **Cơ chế:** Bộ lọc Kalman (Kalman Filter) mô hình hóa chuyển động của đối tượng theo thời gian, ước lượng trạng thái và giảm thiểu sai số do nhiễu (noise) trong quan sát.

b) Deep Appearance Descriptor:

- **Vị trí của Deep Learning:** Đây là nơi DeepSORT sử dụng mạng học sâu (thường là CNN) để trích xuất đặc trưng diện mạo (feature embedding) cho từng đối tượng.
- **Ý nghĩa:** Nhờ vector đặc trưng này, hệ thống có thể phân biệt các cá nhân ngay cả khi có che khuất hoặc các đối tượng khác có hình dạng tương tự.

c) Mahalanobis Distance:

- **Khoảng cách vị trí:** Dựa trên kết quả dự đoán (từ Kalman Filter) và kết quả phát hiện (từ YOLO), DeepSORT tính Mahalanobis distance để đo mức độ khớp giữa vị trí dự đoán và vị trí quan sát thực tế.
- **Xử lý nhiều tham số:** So với khoảng cách Euclidean, Mahalanobis distance cho phép tính đến hiệp phương sai (covariance) của dữ liệu, giúp đánh giá chính xác hơn trong không gian nhiều chiều.

d) Hungarian Algorithm:

- **Ghép cặp (Data Association):** Thuật toán Hungarian (hay Kuhn-Munkres) được dùng để giải bài toán gán đối tượng (assignment) tối ưu, dựa trên cost matrix kết hợp cả Mahalanobis distance và cosine distance của vector đặc trưng.
- **Kết quả:** Mỗi ID đang theo dõi sẽ được gán với một khung người mới phát hiện. Nếu một đối tượng không được khớp trong nhiều khung liên tiếp, nó sẽ bị đánh dấu là mất (lost).

4) Lợi ích của DeepSORT trong Hệ Thống Đếm Người:

- **Duy trì ID ổn định:** Cho phép theo dõi từng người trong nhiều khung hình liên tiếp, hạn chế đếm trùng lặp.
- **Khả năng hoạt động thời gian thực:** Sự kết hợp giữa Kalman Filter và CNN trích xuất đặc trưng vẫn đảm bảo tốc độ khung hình (FPS) tương đối cao.
- **Giảm nhiễu và mất dấu:** DeepSORT kết hợp thông tin vị trí, vận tốc và đặc trưng diện mạo, giúp nhận diện lại đối tượng khi họ quay lại sau vật cản.

D. Mô Hình LSTM và Vai Trò của Nó trong Hệ Thống Đếm Người

Dữ liệu đầu ra từ quá trình theo dõi (số lượng người theo từng khung hình) được gom lại theo chuỗi thời gian, tạo thành một chuỗi số liệu thể hiện xu hướng tăng giảm số lượng người. Mô hình **Long Short-Term Memory (LSTM)** được áp dụng nhằm dự đoán xu hướng thay đổi số lượng người theo thời gian, từ đó hỗ trợ việc giám sát và ra quyết định.

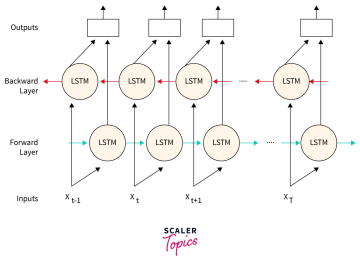

Fig. 2. Sơ đồ tổng quan mô hình LSTM.

1) Cấu Trúc của LSTM Cell: LSTM là một dạng cải tiến của mạng nơ-ron hồi tiếp (RNN) nhằm giải quyết vấn đề vanishing gradient khi xử lý chuỗi dữ liệu dài. Mỗi *cell* của LSTM bao gồm các thành phần chính sau:

a) Forget Gate (f_t): Xác định thông tin nào từ trạng thái bộ nhớ trước đó C_{t-1} cần được loại bỏ.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

b) Input Gate (i_t) và Candidate Memory ($\tilde{C}_t$): Xác định thông tin mới từ đầu vào x_t và trạng thái ẩn trước đó h_{t-1} cần được lưu vào bộ nhớ.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

c) Output Gate (o_t): Quyết định thông tin nào từ trạng thái bộ nhớ C_t sẽ được xuất ra làm đầu ra h_t.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

Sau đó, trạng thái bộ nhớ và đầu ra được cập nhật như sau:

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t$$

$$h_t = o_t \odot \tanh(C_t)$$

Trong đó, σ là hàm sigmoid, $\tanh$ là hàm hyperbolic tangent và $\odot$ là phép nhân phần tử.

2) Vai Trò của LSTM trong Hệ Thống Đếm Người:

- **Dự đoán xu hướng:** Dữ liệu số lượng người được đếm qua từng khung hình tạo thành chuỗi thời gian. LSTM học được các mẫu biến đổi trong chuỗi này để dự đoán số lượng người trong các khoảng thời gian tương lai.
- **Hỗ trợ ra quyết định:** Dự báo của LSTM có thể được sử dụng để đưa ra cảnh báo sớm khi số lượng người vượt quá giới hạn an toàn hoặc khi có sự biến động bất thường.
- **Cải thiện độ chính xác:** LSTM giúp làm mịn dữ liệu đầu ra từ quá trình theo dõi, giảm thiểu sai số do biến động ngẫu nhiên giữa các khung hình.

3) Tích Hợp LSTM vào Hệ Thống:

- **Tiền xử lý dữ liệu:** Số lượng người đếm được từ mỗi khung hình được gom lại theo khoảng thời gian xác định (ví dụ: mỗi giây hoặc mỗi phút) để tạo thành chuỗi thời gian.
- **Huấn luyện mô hình:** Chuỗi dữ liệu này được sử dụng làm đầu vào cho mô hình LSTM. Qua quá trình huấn luyện, LSTM học được các đặc trưng về xu hướng biến đổi số lượng người.
- **Dự báo và cảnh báo:** Kết quả dự báo của LSTM được sử dụng để hiển thị thông tin theo thời gian thực và đưa ra các cảnh báo sớm nếu có bất thường xảy ra.

III. Phân Tích Hoạt Động Của Mô Hình

A. Luồng Xử Lý Hệ Thống

Sơ đồ dưới đây minh họa quy trình hoạt động của hệ thống:

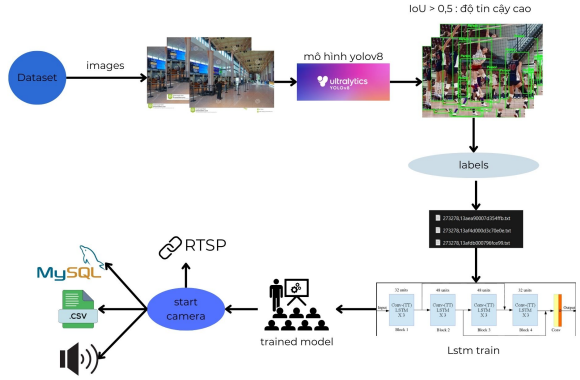

Fig. 3. Sơ đồ hoạt động của hệ thống.

B. Các Bước Xử Lý Chi Tiết

- Bước 1: Thu thập dữ liệu từ các dataset có sẵn (Kaggle, COCO, Open Images Dataset, People Detection Dataset, v.v.).
- Bước 2: Cho các dataset vào một folder để chứa dữ liệu.

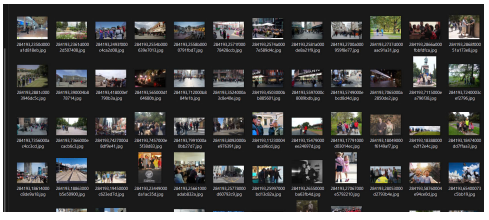

Fig. 4. Folder lưu trữ dataset.

- Bước 3: Mô hình YOLOv8 được sử dụng để phát hiện người trong từng khung hình và tự động gán nhãn.
- Bước 4: DeepSORT (xem mục II-C) theo dõi và gán ID cho từng người để tránh trùng lặp khi đếm.

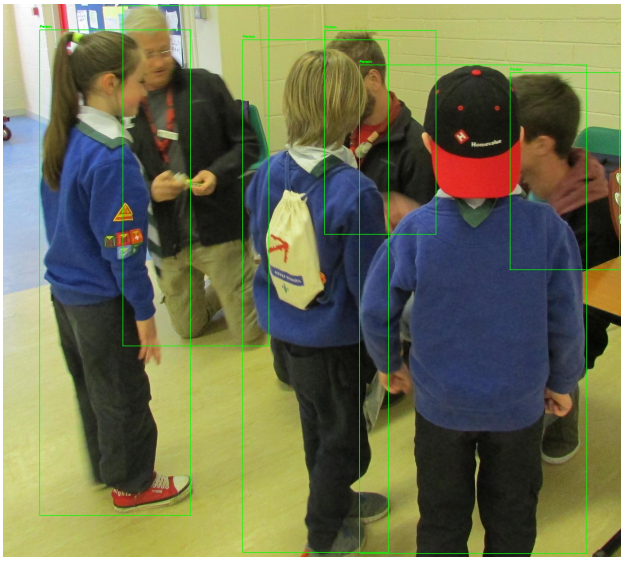

Fig. 5. Tự động gán nhãn.

– Bước 5: Dữ liệu thu thập được lưu trữ vào cơ sở dữ liệu để phân tích xu hướng.

Fig. 6. Cơ sở dữ liệu MySQL.

– Bước 6: Tiến hành huấn luyện dữ liệu với mô hình LSTM (Long short-term memory).

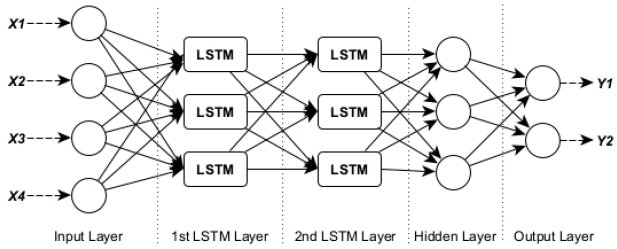

Fig. 7. Mô hình LSTM.

– Bước 7: Đặt camera lên một góc độ cao và bắt đầu thực nghiệm hệ thống.

IV. Phân Tích Hiệu Suất

Thử nghiệm trên nhiều phòng học khác nhau cho thấy:

– Độ chính xác nhận diện trung bình (mAP@50) đạt > 90%.

Fig. 8. Nơi đặt camera.

– Sai số trung bình trong đếm người là ± 2 người.
– Tốc độ xử lý đạt 30 FPS trên GPU RTX 3060.

Hệ thống có thể hoạt động ổn định trong các điều kiện ánh sáng khác nhau, nhưng vẫn bị ảnh hưởng khi có nhiều vật cản lớn trong phòng học.

V. Kết Quả Thực Nghiệm

Dưới đây là hình ảnh minh họa kết quả nhận diện và đếm số lượng người:

Fig. 9. Nhận diện người trong phòng học bằng YOLOv8.

VI. Kết Luận và Hướng Phát Triển

A. Kết Luận

Nghiên cứu này đã đề xuất và triển khai một hệ thống đếm số lượng người trong phòng học dựa trên camera và trí tuệ nhân tạo. Hệ thống sử dụng mô hình **YOLOv8** để nhận diện người, kết hợp với **DeepSORT** và **ByteTrack** để theo dõi chuyển động, đồng thời ứng dụng **mạng nơ-ron hồi tiếp LSTM** để phân tích xu hướng số lượng người theo thời gian.

Các thử nghiệm thực tế cho thấy hệ thống đạt **độ chính xác nhận diện trung bình (mAP@50) > 90%**, tốc độ xử lý lên đến **30 FPS trên GPU RTX 3060**, và sai số trung bình trong quá trình đếm người dao động khoảng $\pm$**2 người**. Ngoài ra, hệ thống có thể hoạt động ổn định trong điều kiện ánh sáng thay đổi, nhưng vẫn gặp hạn chế khi có vật cản lớn hoặc nhiều người đứng sát nhau.

So với các phương pháp truyền thống như **cảm biến hồng ngoại** hay **cảm biến trọng lượng**, hệ thống dựa trên AI có ưu điểm vượt trội về **khả năng giám sát thời gian thực, nhận diện chính xác ngay cả khi có nhiều người xuất hiện đồng thời trong khung hình**. Hơn nữa, việc lưu trữ dữ liệu vào **cơ sở dữ liệu MySQL** giúp dễ dàng phân tích và trực quan hóa thông tin, hỗ trợ công tác quản lý lớp học hiệu quả hơn.

B. Hướng Phát Triển

Mặc dù hệ thống đã đạt được những kết quả khả quan, nhưng vẫn còn một số hạn chế cần được cải thiện trong tương lai:

1) **Tối ưu hóa mô hình YOLOv8**
   a. Điều chỉnh tham số **IoU threshold** và **confidence threshold** để giảm số lượng **false positives** và **false negatives**.
   b. Thử nghiệm với **YOLO-NAS** hoặc các mô hình phát hiện đối tượng tiên tiến hơn để cải thiện tốc độ và độ chính xác.
2) **Cải thiện thuật toán theo dõi đối tượng**
   a. Kết hợp **DeepSORT** với **ByteTrack** để cải thiện khả năng theo dõi đối tượng khi có nhiều người di chuyển nhanh hoặc xuất hiện che khuất nhau.
   b. Sử dụng thêm **Kalman Filter** để giảm nhiễu và tránh mất dấu đối tượng trong một số khung hình.
3) **Mở rộng ứng dụng của hệ thống**
   a. Áp dụng hệ thống vào các môi trường khác như **bệnh viện, trung tâm thương mại, sự kiện đông người** để giám sát số lượng người theo thời gian thực.
   b. Kết hợp với **hệ thống quản lý lớp học thông minh** để cảnh báo khi lớp học vượt quá số lượng giới hạn.
4) **Cải thiện khả năng dự đoán bằng LSTM**
   a. Thu thập dữ liệu dài hạn hơn để cải thiện **độ chính xác dự đoán xu hướng** số lượng người trong phòng học.
   b. So sánh hiệu suất của **LSTM, GRU, Transformer** để chọn mô hình tối ưu nhất.
5) **Phát triển giao diện trực quan**
   a. Xây dựng **dashboard web hoặc ứng dụng di động** để hiển thị dữ liệu theo thời gian thực.
   b. Cung cấp **biểu đồ, thống kê chi tiết**, giúp người dùng dễ dàng theo dõi tình trạng lớp học.

Việc thực hiện các cải tiến trên không chỉ giúp nâng cao hiệu suất hệ thống mà còn mở rộng phạm vi ứng dụng, tạo nền tảng cho việc phát triển các hệ thống **giám sát thông minh trong tương lai**.

VII. Tài Liệu Tham Khảo

References

[1] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[2] G. Bochkovskiy, C.-Y. Wang, and H.-Y. Mark Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," *arXiv preprint arXiv:2004.10934*, 2020.

[3] X. Zhang, Y. Wang, J. Zhu, et al., "ByteTrack: Multi-Object Tracking by Associating Every Detection Box," *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[4] N. Wojke, A. Bewley, and D. Paulus, "Simple Online and Realtime Tracker with a Deep Association Metric," *arXiv preprint arXiv:1703.07402*, 2017.

[5] A. Vaswani, N. Shazeer, et al., "Attention is All You Need," *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[6] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997.

[7] T. Lin, P. Goyal, et al., "Focal Loss for Dense Object Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020.

[8] F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[9] A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Advances in Neural Information Processing Systems (NeurIPS)*, 2012.

[10] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[12] A. G. Howard, M. Zhu, B. Chen, et al., "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," *arXiv preprint arXiv:1704.04861*, 2017.

[13] W. Liu, D. Anguelov, D. Erhan, et al., "SSD: Single Shot MultiBox Detector," *European Conference on Computer Vision (ECCV)*, 2016, pp. 21–37.

[14] R. Girshick, "Fast R-CNN," *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1440–1448.

[15] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2961–2969.

[16] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and Efficient Object Detection," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 10781–10790.

[17] J. Sun, X. Xiao, S. Liu, L. Wang, and X. Ding, "Deep High-Resolution Representation Learning for Visual Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2021.

[18] H. Wu, B. Chen, D. Li, and L. Lin, "MixFormer: End-to-End Tracking with Iterative Mixed Attention," *arXiv preprint arXiv:2203.11082*, 2022.

[19] A. Dosovitskiy, L. Beyer, A. Kolesnikov, et al., "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," *arXiv preprint arXiv:2010.11929*, 2020.

[20] P. Peng, T. Chen, S. Huang, et al., "Conformer: Local Features Coupling Global Representations for Visual Recognition," *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 367–376.