CẢI THIỆN HIỆU QUẢ CỦA PHÁT HIỆN NGƯỜI TRONG ĐÁM ĐÔNG BẤT THƯỜNG SỬ DỤNG CHẮT LỌC TRI THỨC CHO MẠNG YOLO

IMPROVEMENT OF PERFORMANCE OF HUMAN DETECTION IN ABNORMAL CROWD USING KNOWLEDGE DISTILLATION FOR YOLO NETWORKS

Đoàn Thị Hương Giang[1,*], Hồ Anh Dũng[2], Nguyễn Ngọc Trung[3], Nguyễn Trung Hiếu[4]

DOI: http://doi.org/10.57001/huih5804.2024.124

TÓM TẮT

Mạng nơ ron học sâu ngày càng thể hiện rõ ưu điểm vượt trội trong nhiều lĩnh vực khác nhau, như bài toán phát hiện đối tượng trong ảnh với mô hình mạng YOLO. Mạng YOLO ngày càng được cải tiến để nâng cao hiệu quả và khắc phục nhược điểm của các phiên bản trước đó. Tuy nhiên, một vấn đề đặt ra là để mạng có thể đạt kết quả cao hơn thì cấu trúc mạng phức tạp hơn, số lượng tham số nhiều hơn và điều đó khiến thời gian đáp ứng lâu hơn. Khi đó, để triển khai các bài toán có cấu hình máy đơn giản sẽ đối mặt với nhiều hạn chế về tốc độ và độ chính xác. Đặc biệt với bài toán phát hiện người trong đám đông bất thường sẽ có số lượng người trong khung hình khá nhiều, dày đặc và tốc độ di chuyển nhanh nên hệ thống phát hiện người trong ngữ cảnh này sẽ có độ chính xác thấp. Để đạt độ chính xác cao đòi hỏi mô hình có kiến trúc phức tạp lại dẫn tới tốc độ xử lý chậm và yêu cầu máy tính có cấu hình cao. Trong nghiên cứu này, mô hình học chuyển giao tri thức mới từ nhiều mô hình YOLO cấu hình cao sang mô hình YOLO cấu hình đơn giản hơn sử dụng cơ chế học chuyển giao có chắt lọc tri thức để mạng có kiến trúc nhẹ có thể học được các kiến thức của mạng có kiến trúc phức tạp. Giải pháp đề xuất được thử nghiệm trên hai cơ sở dữ liệu đám đông bất thường. Kết quả đạt được cho thấy hệ thống của chúng tôi đạt kết quả tốt hơn với giao thức đánh giá trên từng cơ sở dữ liệu riêng biệt và giao thức đánh giá chéo giữa các cơ sở dữ liệu khác nhau. Độ chính xác thử nghiệm cao hơn từ 1% đến 6,8% trong khi thời gian đáp ứng nhanh hơn 9,16ms khi chạy trên GPU.

Từ khóa: *Mạng nơ ron tích chập, học sâu, chắt lọc kiến thức, mô hình giáo viên - sinh viên, học chuyển giao.*

ABSTRACT

Deep Neural Networks has been achieved outstanding advantages in many different fields, especially object detection using the YOLO networks. The YOLO models have increasingly improved to obtain efficiency and overcome shortcomings of previous versions. However, in order to obtain better performance that the network structure is more complex, the number of model parameters is larger, and requires the longer response time and inverse. Especially, human detection problem in abnormal crowds that faces to some problems such as the high density of people in a frame and the lager speed of human movements. To achieve high accuracy in human detection of anormaly crowd context, the model requires a complex architecture, which leads to high time cost. In this study, the knowledge distillation from multiple higher configurations of the complex YOLO models to a simpler configuration of YOLO model. Experimental results performed on two unusual crowd databases show that our propose framework achieves better results on both single database evaluation and cross database evaluation from 1% to 6.8% higher in accuracy, the response time of our method reduces 9.16ms than YOLO V8 when it tested on GPU.

Keywords**:** *Convolution newral network, deep learning, knowledge distillation, teacher-student model, transfer learning.*

[1]Khoa Điều khiển và Tự động hóa, Trường Đại học Điện lực
[2]Khoa Công nghệ thông tin, Trường Đại học Công nghệ Đông Á
[3]Phòng Tổ chức Cán bộ, Trường Đại học Điện lực
[4]Công ty cổ phần giải pháp công nghệ thông tin và truyền thông MQ
[*]Email: giangdth@epu.edu.vn
Ngày nhận bài: 10/01/2024
Ngày nhận bài sửa sau phản biện: 26/2/2024
Ngày chấp nhận đăng: 25/4/2024

1. GIỚI THIỆU

Mạng nơ ron tích chập (Convolution Neural Networks - CNNs) ngày càng tập trung sự quan tâm của nhiều nhà khoa học do những kết quả vượt trội của chúng so với các giải pháp học máy truyền thống (Machine Learning) như SVM, kNN, Random Forest, Naïve Bayes,… kết hợp với các bộ trích chọn đặc trưng truyền thống như SIRF, SURF, KDES… hoặc các bộ trích chọn đặc trưng tự thiết kế cho từng bài toán cụ thể. Đặc

biệt trong lĩnh vực phát hiện và nhận dạng đối tượng trong ảnh thì mạng YOLO (You Only Look Once) đem đến những bước tiến vượt trội so với những giải pháp truyền thống như Dlib, FastRCNN. Các phiên bản YOLO được các nhà nghiên cứu liên tục cải tiến nâng cấp từ phiên bản đầu tiên YOLO V1 [1] vào năm 2015, YOLO V2 [2] vào năm 2016, YOLO V3 [3] vào năm 2018, YOLO V4 [4] vào năm 2019, YOLO V5 [5] vào năm 2020, YOLO V6 [6] vào năm 2022, YOLO V7 [7] vào năm 2022, YOLO V8 [8] vào tháng 3 năm 2023. Thông thường, mỗi phiên bản YOLO thường có các kích thước mô hình khác nhau như YOLO V-t (tiny), YOLO V-n (nano), YOLO V-s (small), YOLO V-m (medium), YOLO V-l (large) và YOLO V-x (extra large) được đề xuất tương ứng với các phiên bản nguyên gốc nhằm giảm kích thước mô hình ví dụ như YOLO V5 với YOLO V5 nano, hoặc YOLO V3 với YOLO V3 tiny. Mô hình kiểu -t, -n, s có kiến trúc rút gọn hơn so với mô hình -m, -l, -x trong cùng một phiên bản. Hơn nữa, các phiên bản mạng YOLO công bố sau có kiến trúc phức tạp hơn nhằm khắc phục nhược điểm của phiên bản mạng YOLO công bố trước đó như phiên bản YOLO V4 ra đời trước YOLO V5 và do đó nó đạt hiệu quả cao hơn về độ chính xác nhưng thời gian đáp ứng cao hơn, mô hình chứa nhiều tham số hơn và yêu cầu về cấu hình máy tính cao hơn. YOLO V7 khắc phục nhược điểm của các phiên bản trước đó nhưng phiên bản này vẫn gặp khó khăn trong việc phát hiện các đối tượng nhỏ, các đối tượng có tỷ lệ khác nhau, các đối tượng có sự thay đổi về ánh sáng hoặc các điều kiện môi trường khác nhau. Do đó, phiên bản YOLO V8 có hai cải tiến chính trong phát hiện Anchor-Free và tăng cường Mosaic. Phiên bản YOLO V8 có nhiều tham số mô hình hơn các phiên bản YOLO V1 đến V5 nhưng ít tham số hơn YOLO V6 và YOLO V7. Để triển khai ứng dụng thực tế thường hướng tới mô hình nhỏ gọn, đáp ứng thời gian thực và hiệu quả hệ thống đạt được cao hơn. Do đó, việc kế thừa tri thức từ các mô hình phức tạp nhất của mạng YOLO là hai phiên bản YOLO V7 và YOLO V8 để triển khai trên các mô hình YOLO nhỏ gọn hơn như YOLO phiên bản thấp và/hoặc mô hình thu gọn như -s, -n, hay -t là cần thiết.

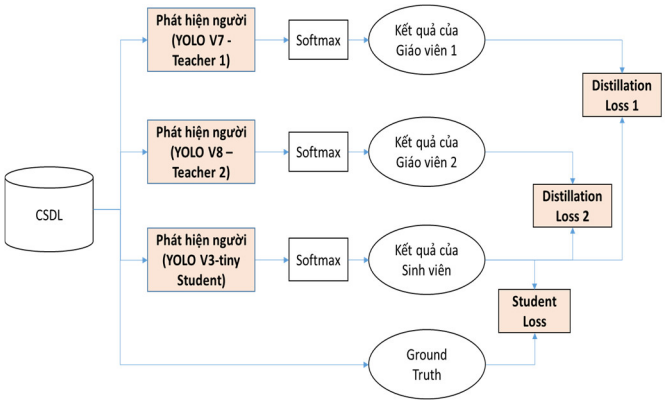
Phát hiện người trong đám đông được áp dụng trong nhiều lĩnh vực khác nhau như trong lĩnh vực cảnh báo an ninh, an toàn như phát hiện sớm đám đông bất thường [15, 16], trong lĩnh vực giáo dục như phát hiện người và đánh giá hiệu quả của lớp học[19],… Tuy nhiên, trong các bài toán trên thường tồn tại một số vấn đề cần giải quyết liên quan đến (1) Số lượng người trong khung hình khá nhiều và dày đặc; (2) Camera thường để cố định trên cao chiếu xuống nên hình trạng người sẽ có thể hiện khác với các tập dữ liệu thu chính diện và thông thường các bộ CSDL đã huấn luyện mô hình được thực hiện với định dạng người được chụp chính diện; (3) Người di chuyển tốc độ nhanh hơn trong ngữ cảnh của đám đông bất thường hoặc đứng yên nhưng bị chồng lấp trong ngữ cảnh của lớp học; (4) Đối tượng phát hiện là người và chỉ là một trong rất nhiều lớp đối tượng đã được huấn luyện trước đó. Khi sử dụng mạng YOLO cho phát hiện người trong đám đông và đặc biệt là phát hiện người trong đám đông bất thường phải đối mặt với các vấn đề cơ bản là bị bỏ sót người không phát hiện được khi sử dụng phiên bản thấp hơn và gây ảnh hưởng rất lớn đến kết quả phát hiện sự kiện bất thường. Để đạt độ chính xác cao thì cần sử dụng các phiên bản cao hơn và do vậy mô hình lớn và phức tạp hơn và thời gian đáp ứng cao. Để có thể cải thiện độ chính xác của các mô hình nhỏ và đơn giản có thể sử dụng cách thức chắt lọc tri thức (knowledge distillation) [9] từ mô hình có kiến trúc lớn và phức tạp khi chuyển giao tri thức (transfer learning) của mạng nơ ron nhân tạo. Hiện nay, các mô hình chuyển giao tri thức có sử dụng mô hình chắt lọc tri thức thường được nghiên cứu trên các mô hình mạng tích chập hai chiều đơn giản như Resnet [10], Densenet [11], Mobilenet [12],… Trong nghiên cứu này, chúng tôi sẽ thực hiện học chuyển giao và chắt lọc tri thức trên mạng YOLO và trên bộ cơ sở dữ liệu (CSDL) phát hiện người trong đám đông bất thường. Giải pháp đề xuất được thực hiện đánh giá trên một bộ CSDL do nhóm tác giả tự thu thập EPUAbnormal [13] và một bộ CSDL được các nhà nghiên cứu công bố cho cộng đồng dùng chung Crow-11 [14]. Các đánh giá cho thấy, giải pháp chắt lọc tri thức do nhóm tác giả đề xuất đạt hiệu quả cao hơn cả về độ chính xác và thời gian đáp ứng trong việc phát hiện người trong ngữ cảnh của các CSDL đám đông bất thường.

Phần tiếp theo của bài báo gồm: Phần 2 mô tả chi tiết giải pháp đề xuất. Kết quả thử nghiệm được trình bày trong Phần 3. Phần 4 là mục cuối cùng sẽ trình bày kết luận và hướng phát triển trong thời gian tiếp theo.

2. GIẢI PHÁP ĐỀ XUẤT

2.1. Mô hình chắt lọc tri thức

Trong nghiên cứu này, chúng tôi đề xuất mô hình chắt lọc và chuyển giao tri thức như mô tả trong hình 1. Trong đó, CSDL sử dụng là các bộ CSDL về đám đông bất thường. Trước hết, hai mô hình YOLO phiên bản mới nhất là YOLO V7 và YOLO V8 được sử dụng huấn luyện (retrain) và tinh chỉnh lại tham số (finetune) mô hình trên CSDL đám đông bất thường. Sau đó, mô hình đã tinh chỉnh này được đóng băng các lớp mạng và sử dụng với vai trò là hai giáo viên (teacher 1 và teacher 2). Mô hình YOLO V3-tiny với vai trò là sinh viên (student) được thiết kế song song với hai mô hình giáo viên là YOLOV7 và YOLOV8. Đây là hai mô hình được lựa chọn vì là các mô hình mới nhất của mạng YOLO đến thời điểm hiện tại.



Hình 1. Sơ đồ khối hệ thống chắt lọc tri thức

Mô hình chắt lọc tri thức chuyển giao trong hình 1 được thực hiện thông qua hàm mất mát là sự kết hợp của hàm mất

mát của sinh viên L_s (Student Loss) và hàm mất mát kết hợp giữa hai giáo viên và học sinh L_{KD1} (Distillation Loss 1) và L_{KD2} (Distillation Loss 2) như mô tả trong công thức (1) sau đây:

$$L(y, \hat{y^s}, \hat{y^t}) = L_s(y, \hat{y^s}) + \alpha L_{KD1}(\hat{y^{t_1}}, \hat{y^s}) + \beta L_{KD2}(\hat{y^{t_2}}, \hat{y^s}) \quad (1)$$

Trong đó, α và β là các hệ số thể hiện sự ảnh hưởng của hàm mất mát của sinh viên L_s và hàm mất mát kết hợp giữa hai giáo viên L_{KD1} và L_{KD2} và mối liên hệ được thể hiện như trong công thức (3). Sự ảnh hưởng của tham số này đối với mô hình đề xuất trong ngữ cảnh với hai bộ CSDL đám đông bất thường được đánh giá trong Phần 3. Ngoài ra, cả hai hàm mất mát đều sử dụng là CE (Cross Entropy) [17] như mô tả trong công thức (2):

$$L(y, \hat{y^s}, \hat{y^t}) = \sum_i^N (y_i \log \hat{y_i^s}) + \alpha \sum_i^N (\hat{y_i^{t_1}} \log \hat{y_i^s}) + \beta \sum_i^N (\hat{y_i^{t_2}} \log \hat{y_i^s}) \quad (2)$$

Trong đó: $\alpha + \beta = 1$.

2.2. Cơ sở dữ liệu

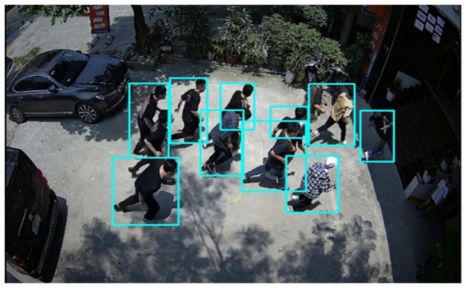
(a) EPUAbnormal

(b) Crow-11

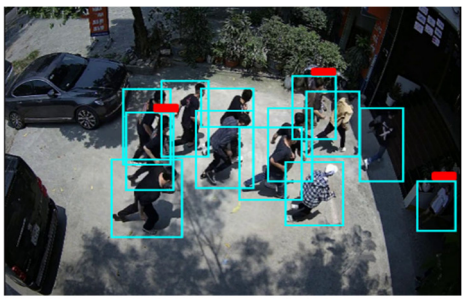
Hình 2. Minh họa CSDL đám đông bất thường: (a) CSDL Crow-11, (b) CSDL EPUAbnormal

Trong nghiên cứu này, chúng tôi sử dụng hai bộ cơ sở dữ liệu về đám đông bất thường như minh họa trong hình 2, gồm EPUAbnormal[13] (hình 2(a)) và Crow-11[14] (hình 2(b)). Đặc điểm của cả hai CSDL này đều có nhiều người xuất hiện trong các khung hình với cả những hoạt động đi lại bình thường và cả các hoặc động bất thường khiến cho người trong khung hình di chuyển hỗn loạn với tốc độ khác nhau. Mỗi cơ sở dữ liệu dược thu thập tại một ngữ cảnh hoàn toàn khác nhau. Trong đó, CSDL EPUAbnormal [13] có 150 video bất thường và 150 video bình thường do nhóm tác giả thực hiện thu thập tại sân của Trường Đại học Điện lực. Máy quay sử dụng là HiK-Vision DS-2CD2643G2-IZS cung cấp ảnh màu RGB với tốc độ thu thập 30 (fps), độ phân giải ảnh 2688x1520 (pixels). Mỗi video có độ dài 3 phút đến 5 phút. Bộ CSDL Crow-11 [14] có 11 video ảnh màu RGB có độ phân giải là 320 x 240 (pixels), mỗi video chứa trung bình 14 phút, tốc độ 30 (fps) và được quay với ngữ cảnh tại tại bãi cỏ, trong sảnh,... Hai bộ CSDL này được sử dụng trong hai trường hợp đơn đánh giá trên từng bộ CSDL và đánh giá chéo giữa các bộ CSDL như mô tả trong mục 2.3.

(a) Hình ảnh đám đông bất thường



(b) Kết quả với YOLO V3 tiny



(c) Kết quả với YOLO V3 tiny KD

Hình 3. Kết quả phát hiện người sử dụng mô hình YOLO V3 tity và YOLO V3 tiny KD

Hình 3 minh họa kết quả của giải pháp đề xuất sử dụng chắt lọc tri thức từ mô hình mạng YOLO V7 và YOLO V8 để huấn luyện cho mô hình YOLO V3 tiny. Hình 3 (a) là ảnh gốc. Hình 3 (b) minh họa kết quả khi phát hiện người trên mô hình YOLO V3 tiny gốc. Hình 3 (c) minh họa kết quả khi phát hiện người của mô hình YOLO V3 tiny KD có thực hiện chuyển giao tri thức từ hai giáo viên là mô hình YOLO V7 và mô hình YOLO V8. Kết quả cho thấy khi sử dụng mô hình YOLO V3 tiny có ba người bị phát hiện thiếu (hình 3 (b)). Trong khi mô hình có YOLO V3 tiny KD thu được kết quả tốt hơn với tất cả người trong khung hình đều phát hiện được (hình 3 (c)). Trong hình 3 (c) này có thêm ba người so với hình

3 (b) được phát hiện và ba người này gắn hình chữ nhật màu đỏ trên box tương ứng. Việc đánh giá định lượng của giải pháp chúng tôi đề xuất sẽ được thực hiện và trình bày chi tiết trong phần 3.

2.3. Giao thức và thang đo đánh giá

2.3.1. Thang đo đánh giá (Metric evaluation)

Để đánh giá hiệu quả của mô hình YOLO, chúng tôi sử dụng các chỉ tiêu độ chính xác P (Precision), độ triệu hồi R (Recall), và độ chính xác trung bình mAP (Mean Average Precision). Trong đó, độ chính xác thu được trên tập dữ liệu thử nghiệm là thước đo đánh giá độ hiệu quả của mô hình qua các thông số độ chính xác P và độ triệu hồi R. Độ chính xác P được định nghĩa là tỉ lệ của số người phát hiện đúng trên tổng số người được phát hiện minh họa bởi công thức sau đây:

$$P = \frac{TP}{TP+FP} \quad (3)$$

Độ triệu hồi R được định nghĩa là tỉ lệ của số người phát hiện đúng trên tổng số người có trong ảnh như minh họa bởi công thức sau đây:

$$R = \frac{TP}{TP+FN} \quad (4)$$

Độ chính xác trung bình mAP của N lớp và được định nghĩa theo công thức sau đây:

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \quad (5)$$

Trong đó, TP (True positive) là tổng số dự đoán đúng được phân lớp vào đúng (positive); FP (False positive) tổng số dự đoán sai được phân lớp vào đúng (positive); FN (False negative) tổng số dự đoán sai được phân lớp vào sai (negative); và TN (True negative) là tổng số dự đoán đúng được phân lớp vào sai, N là tổng số lớp. Kết quả đánh giá hệ thống sẽ được thực hiện và trình bày chi tiết trong mục 3.

2.3.2. Giao thức thử nghiệm (Protocol Evaluation)

Trong bài báo này, chúng tôi sử dụng hai cách thức đánh giá gồm: Đánh giá trên đơn CSDL (SDE- Single Dataset Evaluation); Đánh giá chéo giữa các CSDL (CDE - Cross Dataset Evaluation).

Với đánh giá SDE, chúng tôi thực hiện thử nghiệm trên từng bộ CSDL. Mỗi bộ CSDL được chia thành 10 phần bằng nhau. Sau đó, chúng tôi sử dụng phương pháp "*Leave-one-subject-out*" [18] để chia toàn bộ dữ liệu thành 10 phần hoàn toàn khác nhau, mỗi phần được xem như một đối tượng (subject). Mỗi bộ CSDL được thử nghiệm 10 lần trong đó 8 phần được sử dụng để huấn luyện mô hình, một phần để validation và một phần để đánh giá mô hình. Tại mỗi lần thử nghiệm này sẽ có độ chính xác P_j, R_j và mAP_j (j = (1-M), M = 10). Độ chính xác của một CSDL đạt được bằng cách lấy trung bình của 10 lần đánh giá mô hình với độ chính xác P (Precision) như trong công thức (6) sau:

$$P = \frac{\sum_{j=1}^{M=10} P_j}{M} \quad (6)$$

Độ triệu hồi (recall)của cả cơ sở dữ liệu R được tính trung bình như trong công thức (7) sau:

$$R = \frac{\sum_{j=1}^{M=10} R_j}{M} \quad (7)$$

Độ chính xác trung bình mAP được tính trung bình như trong công thức (8) sau:

$$mAP = \frac{\sum_{j=1}^{M=10} mAP_j}{M} \quad (8)$$

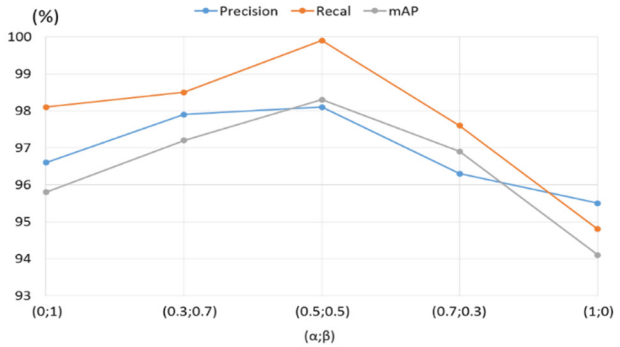
Với đánh giá CDE, hai bộ CSDL được sử dụng với một bộ CSDL được sử dụng để huấn luyện mô hình và một bộ CSDL còn lại được sử dụng để thử nghiệm mô hình. Độ chính xác, độ triệu hồi và độ chính xác trung bình mAP của đánh giá CDE được thực hiện như trong các công thức (3), (4) và (5).

3. KẾT QUẢ THỬ NGHIỆM

Chúng tôi thực hiện các thử nghiệm trên máy tính máy tính sử dụng CPU Intel Core i5-11400H, GPU NVIDIA GeForce GTX 1650, bộ nhớ Ram 8GB. Ngôn ngữ lập trình sử dụng là ngôn ngữ Python. Các thử nghiệm được thực hiện trên hai bộ CSDL gồm EPUAbnormal [13], và Crow-11 [14] như mô tả chi tiết trong mục 2.2. Trong đó, các thử nghiệm cài đặt với kích thước mẻ (batch size) là 64; tốc độ huấn luyện mô hình (learning rate) là 10^{-4}; số lần lặp để huấn luyện mô hình (epochs) là 100. Các thử nghiệm thực hiện sử dụng cho giáo viên 1 là mô hình YOLO V7 và giáo viên 2 là mô hình YOLO V8, sinh viên là mô hình YOLO V3 tiny.

Các thử nghiệm được tiến hành trong nghiên cứu này gồm: (1) Đánh giá độ chính xác phát hiện người với các giá trị khác nhau của các tham số **α** và **β** của mô hình KD trên giao thức SDE; (2) Đánh giá đơn (SDE) độ chính xác phát hiện người của mô hình KD trên từng CSDL riêng lẻ; (3) Đánh giá chéo (SDE) độ chính xác phát hiện người của mô hình KD giữa các CSDL khác nhau; (4) Đánh giá thời gian đáp ứng của mô hình KD (Time cost).

3.1. Khảo sát tham số mô hình KD



Hình 4. Độ chính xác (P-Precision), độ triệu hồi (R-Recall) và độ chính xác trung bình mAP với các giá trị α và β khác nhau của mô hình KD trên CSDL EPUAbnormal

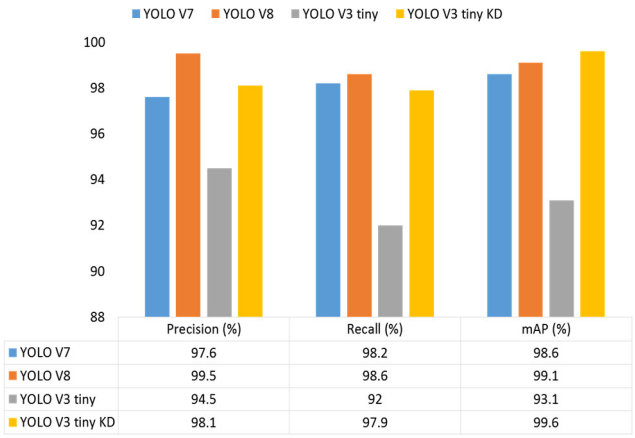
Trong phần này, chúng tôi sẽ tiến hành khảo sát sự ảnh hưởng của các giá trị α và β của mô hình KD trong công thức (3) trên bộ CSDL EPUAbnormal với giao thức đánh giá SDE. Trong đó, các giá trị α và β lần lượt thay đổi từ 0 đến 1 và ngược lại nhưng vẫn luôn đảm bảo tổng của chúng luôn bằng 1. Cụ thể chúng tôi thực hiện tính toán tại 05 điểm tương ứng với các các cặp giá trị của α và β lần lượt tại (0; 1), (0,3; 0,7), (0,5; 0,5), (0,7; 0,3), (1; 0) lần lượt thể hiện tỷ trọng

giá trị hàm mất mát tham gia trong mô hình chuyển giao tri thức (KD) của các mạng YOLO V7 và YOLO V8. Kết quả đánh giá thể hiện trong hình 4.

Hình 4 cho thấy tại α = 1 và β = 0 (quá trình học kiến thức từ mô hình YOLO V7 được sử dụng hoàn toàn và YOLO V8 không tham giá vào quá trình chuyển giao) thì kết quả mô hình KD đạt được cao hơn tại α = 0 và β = 1 (quá trình học kiến thức từ mô hình YOLO V8 được sử dụng hoàn toàn và YOLO V7 không tham giá vào quá trình chuyển giao). Điều đó thể hiện rằng mô hình có YOLO V8 có tri thức tốt hơn sẽ làm cho kết quả mô hình KD tốt hơn. Ngoài ra, khi tăng dần vai trò YOLO V8 thì kết quả mô hình KD tăng dần. Ngoài ra, mô hình KD đạt kết quả tốt nhất tại vị trí cân bằng cả hai mô hình KD với α = 0,5 và β = 0,5 với độ chính xác là 98,1%, 99,9% và 98,3% cho các giá trị P, R và mAP. Giá trị các tham số cân bằng với α = 0,5 và β = 0,5 sẽ được sử dụng trong các đánh giá sau trong mục 3.2, 3.3 và 3.4.

3.2. Độ chính xác P, R và mAP với chế độ SDE

Trong mục này, chúng tôi thực hiện thử nghiệm trên CSDL EPUAbnormal. Tham số mô hình chắt lọc tri thức khi học chuyển giao YOLO V3 tiny KD cài đặt với α = 0,5 và β = 0,5. Kết quả thử nghiệm gồm độ chính xác (Precision), độ triệu hồi (Recall), và độ chính xác trung bình (mAP) được trình bày như trong hình 5. Trong đó, các cột màu xanh lá cây, cam và xám lần lượt biểu diễn ba kết quả của ba mô hình YOLO V7, YOLO V8 và YOLO V3 tiny. Cột màu vàng biểu diễn kết quả của mô hình chắt lọc tri thức khi học chuyển giao sử dụng mô hình YOLO V3 tiny KD do nhóm tác giả đề xuất.



	Precision (%)	Recall (%)	mAP (%)
YOLO V7	97.6	98.2	98.6
YOLO V8	99.5	98.6	99.1
YOLO V3 tiny	94.5	92	93.1
YOLO V3 tiny KD	98.1	97.9	99.6

Hình 5. Kết quả phát hiện người sử dụng mô hình YOLO V3 tiny và YOLO V3 tiny KD với CSDL EPUAbnormal

Kết quả trong hình 5 cho thấy, phiên bản YOLO V3 tiny KD đạt kết quả cao hơn tại các giá trị 98,1%; 97,9%; 99,6% so với kết quả khi sử dụng với mô hình YOLO V3 tiny tại các giá trị 94,5%; 92%; 93,1% tương ứng với độ chính xác (P), độ triệu hồi (R), và độ chính xác trung bình (mAP). Điều đó cho thấy hiệu hiệu quả của quá trình học chuyển giao từ nhiều giáo viên có tri thức tốt sẽ có mô hình sinh viên tốt hơn mô hình tự học chuyển giao trên cả hai thang đo P và R. Thậm chí với giá trị mAP thì mô hình được chuyển giao còn đạt kết quả 99,9% cao hơn cả hai mô hình giáo viên 1 (YOLO V7) với 98,6% và giáo viên 2 (YOLO V8) với 99,6%.

3.3. Độ chính xác P, R và mAP với chế độ CDE

Trong bài báo này, chúng tôi thực hiện giao thức đánh giá chéo (CDE) với một CSDL sử dụng để huấn luyện mô hình YOLO V3 tiny KD và một CSDL còn lại được sử dụng để thử nghiệm. Hai cơ sở dữ liệu được sử dụng trong thử nghiệm này là CSDL EPUAbnormal [13], và Crow-11 [14]. Có hai thử nghiệm được tiến hành như kết quả của hai hàng đầu tiên của bảng 1. Trong đó, CSDL EPUAnnormal được sử dụng để huấn luyện mô hình và CSDL Crow-11 được sử dụng để thử nghiệm mô hình. Hai hàng sau của bảng 1 biểu diễn kết quả khi sử dụng CSDL Crow-11 để huấn luyện mô hình và CSDL EPUNormal được sử dụng để thử nghiệm mô hình. Trong cả hai thử nghiệm này, chúng tôi sử dụng hai mô hình gồm YOLO V3 tiny và YOLO V3 tiny KD (sử dụng hai giáo viên là YOLO V7 và YOLO V8, sinh viên là YOLO V3 tiny). Kết quả thử nghiệm gồm độ chính xác (Precision), độ triệu hồi (Recall), và độ chính xác trung bình (mAP) được trình bày như trong bảng 1.

Bảng 1. Kết quả phát hiện người khi sử dụng mô hình YOLO V3 tiny KD với giao thức CDE

CSDL huấn luyện	CSDL thử nghiệm	Mô hình	Precision (%)	Recall (%)	mAP (%)
EPUAbnormal	**Crow-11**	**YOLO V3 tiny**	91,8	88,6	90,3
		YOLO V3 tiny KD	97,8	96,1	98,4
Crow-11	**EPUAbnormal**	**YOLO V3 tiny**	89,2	87,5	88,9
		YOLO V3 tiny KD	95,3	94,7	96,2

Kết quả trong bảng 1 cho thấy, hiệu quả của mô hình chắt lọc tri thức khi học chuyển giao luôn cao hơn quá trình sử dụng học chuyển giao thông thường trong cả hai đánh giá chéo. Trong thử nghiệm đầu tiên khi sử dụng CSDL EPUAnormal để huấn luyện mô hình và thử nghiệm mô hình sử dụng CSDL Crow-11, mô hình YOLO V3 tiny KD đạt kết quả cao hơn tại 97,8%; 96,1% và 98,4% trong khi mô hình YOLO V3 tiny đạt các kết quả tại 91,8%; 88,6% và 90,3% với các giá trị P, R và mAP. Trong thử nghiệm đánh giá chéo thứ hai, mô hình YOLO V3 tiny KD cũng đạt kết quả cao hơn mô hình YOLO V3 tiny (95,3%; 94,7%; 96,2%) so sánh với (89,2%; 87,5%; 88,9%). Ngoài ra, do bộ CSDL EPUAbnormal có số lượng dữ liệu lớn hơn bộ CSDL Crow-11 nên trên cả mô hình YOLO V3 tiny và YOLO V3 tiny KD thì khi sử dụng bộ CSDL EPUAbnormal để huấn luyện mô hình thì thử nghiệm trên Crow-11 luôn đạt kết quả cao hơn (kết quả tại hàng 1 cao hơn hàng 3, kết quả tại hàng 2 cao hơn hàng 4). Ngoài ra, kết quả đánh giá này cũng cho thấy giải pháp đề xuất đảm bảo tính bền vững khi mô hình thử nghiệm với CSDL bất kỳ.

3.4. Thời gian đáp ứng của mô hình KD

Ngoài việc so sánh kết quả độ chính xác như đã trình bày ở các phần trên, trong nghiên cứu này chúng tôi còn tiến hành thực hiện khảo sát về thời gian thực hiện trung bình của các phiên bản YOLO để phát hiện người khi sử dụng CSDL EPUAbnormal. Trong đó, chúng tôi tiến hành đánh giá thời gian đáp ứng của từng khung hình với phiên bản YOLO V7 (hàng 1), YOLO V8 (hàng 2), YOLO V3 tiny (hàng 3) và YOLO V3 tiny KD (hàng 4). Đánh giá về thời gian đáp ứng này

được chạy thử nghiệm trên nền tảng CPU và GPU. Kết quả được trình bày trong bảng 2.

Bảng 2. Tốc độ phát hiện người trên các phiên bản YOLO trên CSDL EPUAbnormal

Mô hình	Tốc độ xử lý một khung hình (CPU)	Tốc độ xử lý một khung hình (GPU)
YOLO V7	84,12ms	23,42ms
YOLO V8	75,21ms	20,38ms
YOLO V3 tiny	26,46ms	14,51ms
YOLO V3 tiny KD	25,86ms	14,26ms

Kết quả trong bảng 2 cho thấy, mạng YOLO V3 tiny KD có thời gian phát hiện khi sử dụng GPU là thấp nhất và tương đương với với YOLO V3 tiny (14ms) và khi sử dụng CPU thì kết quả lần lượt là 26,46ms và 25,86ms. Kết quả này thấp hơn rất nhiều so với việc sử dụng các mô hình YOLO V7 và YOLO V8 có thời gian đáp ứng khi sử dụng CPU lần lượt là 84,12ms, 75,21ms và khi sử dụng GPU lần lượt là 23,42ms, 20,38ms. Trong khi đó, độ chính xác phát hiện của mô hình YOLO V3 tiny KD có độ chính xác cao hơn YOLO V3 tiny và độ chính xác tiệm cận với kết quả của các mô hình phức tạp là YOLO V7 và YOLO V8. Điều này cho thấy giải pháp học chuyển giao có chắt lọc tri thức với mạng YOLO do chúng tôi đề xuất đạt kết quả tốt với mô hình phát hiện và người trong thử nghiệm trên CSDL đám đông bất thường.

4. KẾT LUẬN

Trong bài báo này, chúng tôi đã đề xuất một mô hình học chuyển giao có chắt lọc tri thức từ nhiều giáo viên áp dụng cho bài toán phát hiện người trong đám đông. Hệ thống đề xuất được đánh giá trên cả bộ CSDL đám đông bất thường tự thu thập và trên cả bộ CSDL đám đông bất thường được công bố cho cộng đồng dùng chung. Kết quả thử nghiệm được đánh giá ở chế độ đánh giá đơn lẻ trên từng CSDL và chế độ đánh giá chéo giữa các bộ CSDL hoàn toàn khác nhau để thể hiện tính bền vững của mô hình. Kết quả đánh giá được thực hiện trên các tiêu chí về độ chính xác (Precision), độ triệu hồi (Recal), độ chính xác trung bình (mAP) và thời gian đáp ứng trên từng khung hình. Kết quả thử nghiệm cho thấy giải pháp đề xuất đạt độ chính xác cao CSDL (cao nhất tại các giá trị 98,1%; 97,9%; 99,6% cho P, R và mAP) tiệm cận với các mô hình phức tạp như YOLO V7 và YOLO V8, đảm bảo tính bền vững khi thực hiện đánh giá chéo giữa các bộ CSDL (cao nhất tại các giá trị 97,8%; 96,1% và 98,4% cho P, R và mAP) và đáp ứng thời gian nhanh (xấp xỉ 14ms). Trong thời gian tới, nhóm tác giả mong muốn ý tưởng đề xuất của nghiên cứu này sẽ tiếp tục được triển khai để đánh giá trên hệ thống thực với số lượng CSDL thử nghiệm nhiều hơn nữa. Ngoài ra, kết quả nghiên cứu cũng sẽ tiếp tục phát triển để sử dụng vào hệ thống phát hiện và nhận dạng đám đông bất thường.

TÀI LIỆU THAM KHẢO

[1]. Redmon Joseph, Divvala Santosh, Girshick Ross, Farhadi Ali, *YOLOv1: You Only Look Once: Unified, Real-Time Object Detection*. 779-788. 10.1109/CVPR.2016.91, 2016.

[2]. Redmon Joseph, Farhadi Ali, *YOLO9000: Better, Faster, Stronger*. 6517-6525. 10.1109/CVPR.2017.690, 2017.

[3]. J. Redmon, A. Farhadi, "Yolov3: An incremental improvement," *ArXiv, abs/1804.02767*, 1-6, 2018.

[4]. A. Bochkovskiy, C.Y. Wang, H.Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *ArXiv: 2004.10934*, 1-17, 2004.

[5]. Glenn Jocher, *Yolov5 in pytorch*. https://github.com/ultralytics/yolov5, 06 2020.

[6]. Chuyi Li, Lulu Li, Hongliang Jiang, Kaiheng Weng, Yifei Geng, Liang Li, Zaidan Ke, Qingyuan Li, Meng Cheng, Weiqiang Nie, Yiduo Li, Bo Zhang, Yufei Liang, Linyuan Zhou, Xiaoming Xu, Xiangxiang Chu, Xiaoming Wei, Xiaolin Wei, "YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications," *ArXiv, abs/2209.02976*, 2022.

[7]. Wang Chien-Yao, Bochkovskiy Alexey, Liao Hong-yuan, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," *ArXiv.2207.02696*, 2022.

[8]. Jocher G., Chaurasia A., Qiu J., *YOLO by Ultralytics*. https://github.com/ultralytics/ultralytics.

[9]. G. E. Hinton, O. Vinyals, J. Dean, "Distilling the knowledge in a neural network," *ArXiv, abs/1503.02531*, 2015.

[10]. He K., Zhang X., Ren S., Sun J., "Deep residual learning for image recognition," *ArXiv:1512.03385,* 2015.

[11]. G. Huang, Z. Liu, L. V. D. Maaten, K. Q. Weinberger, "Densely connected convolutional networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2261-2269, 2017.

[12]. M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.C. Chen, "MobileNetV2: inverted residuals and linear Bottlenecks," in *the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 4510-4520, 2018.*

[13]. EPU Abnormal database: htttps://zenodo.org/record/8365955

[14]. Dupont C., Tobias L., Luvison B., "Crowd-11: A dataset for fine grained crowd behaviour analysis," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2184-2191, 2017.

[15]. Mehran R., Oyama A., Shah M., "Abnormal crowd behavior detection using social force model," In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 935-942, 2009.

[16]. Acsintoae A., Florescu A., Georgescu M., Mare T., Sumedrea P., Ionescu R.T., Khan F.S., Shah M., "Ubnormal: New benchmark for supervised open-set video anomaly detection," In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[17]. R. Y. Rubinstein, "Optimization of computer simulation models with rare events," *European Journal of Operational Research*, 99, 1, 89-112, 1997.

[18]. Huong-Giang Doan, Ngoc-Trung Nguyen, "New blender-based augmentation method with quantitative evaluation of CNNs for hand gesture recognition", *Indonesian Journal of Electrical Engineering and Computer Science*, 30, 2, 796-806, 2023. DOI: 10.11591/ijeecs.v30.i2.pp796-806.

[19]. Thi-Oanh Ha, Hoang-Nhat Tran, Hong-Quan Nguyen, Thanh-Hai Tran, Phuong-Dung Nguyen, Huong-Giang Doan, Van-Toi Nguyen, Hai Vu, Thi-Lan Le, "Improvement of People Counting by Pairing Headand Face Detections from Still Images," *The fourth International Conference on Multimedia Analysis and Pattern Recognition (MAPR),* Hanoi, Vietnam, 2021.

AUTHORS INFORMATION

Doan Thi Huong Giang[1], Ho Anh Dzung[2], Nguyen Ngoc Trung[3], Nguyen Trung Hieu[4]

[1]Faculty of Control and Automation, Electric Power University, Vietnam

[2]Faculty of Information Technology, East Asia University of Technology, Vietnam

[3]Department of Personnel and Organization, Electric Power University, Vietnam

[4]MQ Information and Communication Technology Solutions JSC, Hanoi, Vietnam