

PHÁT HIỆN ĐÁM CHÁY SỬ DỤNG PHƯƠNG PHÁP HỌC CHUYỂN TIẾP VỚI MẠNG EFFICIENTNET

FIRE DETECTION BY TRANSFER LEARNING WITH EFFICIENTNET

NGUYỄN HỮU TUÂN*, TRỊNH THỊ NGỌC HƯƠNG

Khoa Công nghệ Thông tin, Trường Đại học Hàng hải Việt Nam

*Email liên hệ: huu-tuan.nguyen@vimaru.edu.vn

Tóm tắt

Các vụ cháy thường gây ra hậu quả nghiêm trọng với các thiệt hại lớn về tài sản và người. Các hệ thống phát hiện và cảnh báo cháy dựa trên các bộ cảm biến nhiệt, khói thường chỉ phát hiện khi đám cháy đã ở giai đoạn nguy hiểm và do đó hạn chế về khả năng dập đám cháy để giảm thiểu thiệt hại. Trong bài báo này, chúng tôi đề xuất một phương pháp phát hiện cháy sớm dựa trên việc áp dụng phương pháp học chuyển tiếp với mô hình mạng CNNs EfficientNet-B3. Từ dữ liệu hình ảnh được thu thập từ camera theo thời gian thực, hệ thống có thể phân tích và đưa ra dự đoán xem có đám cháy hay không một cách nhanh chóng, từ đó đưa ra các cảnh báo sớm và có thể kích hoạt hệ thống dập cháy tự động. Chương trình đã được thực nghiệm với thư viện Tensorflow và cơ sở dữ liệu ảnh công cộng. Kết quả cho thấy độ chính xác mà hệ thống đạt được rất cao (97,5% trên bộ dữ liệu công cộng) và có khả năng triển khai trên thực tế.

Từ khóa: Fre classification, nhận dạng ảnh đám cháy, mạng nơon nhân chập, CNNs, Python, EfficientNet, Tensorflow, Keras.

Abstract

Fires often cause huge damage of wealth and lives. Traditional fire detection systems based on temperature and smoke sensors are able to detect fires when they are in dangerous stages and thus have limitations in extinguishing to minimize damages. This paper proposes a new method for early fire detection by applying transfer learning with EfficientNet-B3, a CNNs model. Images captured from camera are predicted to quickly discover if there is fire or not. If there is fire, then the alarm and fire extinguishing systems might be activated. Our system is implemented with Tensorflow and experimented upon public dataset. The obtained results show that its accuracy is high (97.5%) and can be applied in real life applications.

Keywords: Fire detection, convolutional neural network, CNNs, EfficientNet, Tensorflow, Keras.

1. Đặt vấn đề

Các vụ cháy thường xảy ra bất ngờ và gây ra các thiệt hại lớn về tài sản và tính mạng con người. Theo các quy định của Pháp luật Việt Nam [1], các nhà xưởng, khu nhà ở phức hợp đều có trang bị các hệ thống phát hiện, cảnh báo và dập các đám cháy. Tuy nhiên, các hệ thống phát hiện đám cháy truyền thống này đều dựa trên các bộ cảm biến nhiệt, khói và thường được đặt khá cao so với mặt đất. Điều này dẫn tới một thực tế là hệ thống chỉ có thể phát hiện đám cháy một cách hiệu quả (chính xác) khi đám cháy đã lớn và ở giai đoạn nguy hiểm (khó dập tắt). Kết quả là khi phát hiện ra đám cháy thì việc dập tắt đã trở nên khó khăn và hầu như không thể ngăn chặn được các thiệt hại về tài sản và đôi khi cả tính mạng con người. Một hạn chế nữa của các hệ thống phát hiện đám cháy truyền thống là chúng không cung cấp được hình ảnh và vị trí của các đám cháy cho bộ phận quản lý để có thể xác thực và đưa ra các phương án dập đám cháy, cứu người và tài sản hợp lý nhất.

Trong bài báo này, chúng tôi đề xuất một hệ thống thị giác máy tính sử dụng phương pháp học chuyển tiếp (transfer learning) với mô hình mạng EfficientNet, một mô hình mạng CNNs hiệu quả mới được phát triển trong thời gian gần đây để dự đoán xem một bức ảnh có chứa đám cháy hay không. Mô hình mạng EfficientNet-B3 đã huấn luyện với cơ sở dữ liệu ảnh ImageNet được chúng tôi sử dụng để huấn luyện với cơ sở dữ liệu ảnh công cộng. So sánh kết quả với các công bố gần đây cho thấy hệ thống đạt được độ chính xác cao (97,5%) và có khả năng áp dụng trên thực tế.

Nội dung của bài báo này được bố cục như sau: trong phần tiếp theo chúng tôi sẽ trình về một số cách tiếp cận dựa trên phân tích hình ảnh cho bài toán phát hiện đám cháy, tiếp đến phần 3 của bài báo sẽ trình bày về phương pháp học chuyển tiếp và mô hình mạng EfficientNet, cài đặt và kết quả thực nghiệm của hệ thống đề xuất. Phần 4 là kết luận và hướng phát triển của bài báo.

2. Một số cách tiếp cận hiện đại cho bài toán phát hiện đám cháy

Có hai cách tiếp cận chính trong việc áp dụng các kỹ thuật của lĩnh vực thị giác máy tính vào giải quyết bài toán phát hiện đám cháy dựa trên phân tích hình ảnh: sử dụng các phương pháp phát hiện đối tượng (object detection) và sử dụng các phương pháp phân lớp/nhận dạng hình ảnh (image classification). Trong bài báo này, chúng tôi tập trung vào các phương pháp theo hướng nhận dạng/phân lớp hình ảnh.

Kể từ khi mô hình mạng CNNs đầu tiên, mạng AlexNet [2], thành công trong một bài toán thị giác máy tính quan trọng là phân lớp hình ảnh trên tập dữ liệu ImageNet, rất nhiều mô hình mạng mới đã được phát triển cho nhiều bài toán khác và phát hiện đám cháy cũng không phải là một ngoại lệ. Dunning và cộng sự [3] đã khảo sát sự hiệu quả của các mô hình mạng CNNs như AlexNet, VGG [4], InceptionNetV1 [5] và đề xuất một mô hình mạng FireNet riêng cho bài toán phát hiện đám cháy. Kết quả mà nhóm tác giả nhận được khá khả quan khi độ chính xác cao nhất là 93,4% (với mô hình InceptionNetV1). Nhóm các nhà khoa học khác [6] cũng theo dõi hướng áp dụng các mô hình mạng CNNs cho bài toán phát hiện đám cháy như mô hình ResNet [7], InceptionNetV4 [8] và đã cải thiện được độ chính xác lên 96% với mô hình InceptionNetV4 trên cùng cơ sở dữ liệu hình ảnh đã được sử dụng trong [3].

3. Học chuyển tiếp với EfficientNet cho bài toán phát hiện đám cháy từ dữ liệu hình ảnh

3.1. Học chuyển tiếp

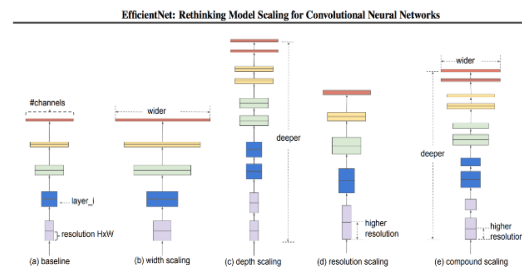
Học chuyển tiếp (transfer learning) [9] là phương pháp học máy cho phép sử dụng lại các tham số của một mạng nơ ron đã huấn luyện trong một bài toán để giải quyết một bài toán mới. Điều này cũng tương tự như việc sử dụng các kiến thức về toán học để làm nền tảng cho việc học vật lý. Trên thực tế, phương pháp học chuyển tiếp khá hiệu quả trong nhiều bài toán nên trong bài báo này chúng tôi đề xuất sử dụng cách tiếp cận này với việc sử dụng các mô hình mạng EfficientNet đã được huấn luyện trên cơ sở dữ liệu ảnh ImageNet [2].

3.2. Mô hình mạng EfficientNet

Kể từ khi mô hình mạng CNNs đầu tiên là AlexNet, các nhà nghiên cứu đã đi theo hướng mở rộng các tham số của mạng nhằm cải thiện độ chính xác, các tham số để mở rộng một mạng CNNs là độ sâu (depth), độ rộng (width) và độ phân giải (resolution). Độ sâu được mở rộng bằng cách thêm vào nhiều lớp hơn. Độ

rộng là số kênh hay số bộ lọc được sử dụng ở mỗi lớp nhân chập của mạng. Mở rộng độ phân giải tương ứng với việc tăng kích thước ảnh đầu vào cho mỗi mạng. Độ phân giải chuẩn của các mạng CNNs là 224x224. Trên thực tế, việc mở rộng các mạng CNNs theo các tham số trên có cải thiện được độ chính xác của mạng lên nhưng cách tiếp cận này cũng có những vấn đề: việc mở rộng mạng sẽ dẫn tới chi phí tính toán tăng lên nhanh chóng (1), mức độ cải thiện về độ chính xác sẽ nhanh chóng bão hoà khi đạt tới ngưỡng hơn 80% do hiện tượng suy giảm giá trị của đạo hàm (gradient vanishing) (2), và nhiều khi việc mở rộng mạng cũng không cải thiện được nhiều (3), ví dụ như độ chính xác của mạng ResNet-1000 cũng không hơn mạng ResNet-101.

Cách tiếp cận của mô hình mạng EfficientNet [10] là cách tiếp cận tổng hợp bằng cách mở rộng cả ba tham số về độ sâu, độ rộng và độ phân giải của mạng CNNs trong khi vẫn giữ cho chi phí tính toán của mạng không tăng lên quá nhiều (xem Hình 1: (a) - mô hình cơ sở, (b) - mở rộng về chiều rộng, (c) - mở rộng về chiều sâu, (d) - mở rộng về độ phân giải và (e) - mở rộng tổng hợp). Điều này tương ứng với số tham số của mạng không quá lớn và sẽ cho phép các mô hình mạng có thể thực hiện trong thời gian thực và triển khai trên các thiết bị di động, thiết bị xử lý tại biên.



Hình 1. Cách tiếp cận của mô hình mạng

Một kỹ thuật nữa được các tác giả của mô hình mạng EfficientNet sử dụng là kỹ thuật đảo ngược khối thặng dư (Inverted Residual Block) [11] với việc kết nối một số lượng nhỏ các kênh và tách phép nhân chập sẽ làm giảm chi phí tính toán xuống đáng kể mà không làm mất tính hiệu quả của khối tính toán như trong đoạn giả mã lệnh sau:

```
from keras.layers import Conv2D, DepthwiseConv2D, Add
def inverted_residual_block(x, expand=64, squeeze=16):
    block = Conv2D(expand, (1,1), activation='relu')(x)
    block = DepthwiseConv2D((3,3), activation='relu')(block)
    block = Conv2D(squeeze, (1,1), activation='relu')(block)
    return Add()([block, x])
```

EfficientNet bao gồm một họ các mô hình mạng được xây dựng từ một mô hình cơ bản gọi là EfficientNet-B0 có kiến trúc như trong Bảng 1 sau:

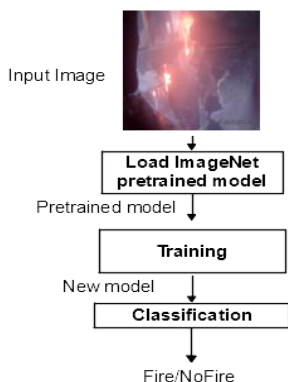
Bảng 1. Kiến trúc mạng EfficientNet-B0

Stage i	Operator \hat{F}_i	Resolution $\hat{H}_i \times \hat{W}_i$	#Channels \hat{C}_i	#Layers \hat{L}_i
1	Conv3x3	224 × 224	32	1
2	MBConv1, k3x3	112 × 112	16	1
3	MBConv6, k3x3	112 × 112	24	2
4	MBConv6, k5x5	56 × 56	40	2
5	MBConv6, k3x3	28 × 28	80	3
6	MBConv6, k5x5	14 × 14	112	3
7	MBConv6, k5x5	14 × 14	192	4
8	MBConv6, k3x3	7 × 7	320	1
9	Conv1x1 & Pooling & FC	7 × 7	1280	1

Trong đó cột F là các thao tác nhân chập (Conv) và khối đảo ngược thẳng dư (Inverted Residual Block) của mạng MobileNetV2 [11], cột Resolution là độ phân giải của các bản đồ đặc trưng đầu ra của các khối (từ cột F), cột #Channels là số bộ lọc, cột L là số lần lặp lại của các khối của cột F.

Các mô hình mạng từ B1 tới B7 sau đó được xây dựng bằng cách mở rộng mạng B0 sử dụng phương pháp kết hợp với việc thay đổi các tham số về độ sâu, độ rộng và độ phân giải phù hợp trong khi vẫn kiểm soát chi phí tính toán của mạng không tăng lên quá nhiều (B1 có chi phí tính toán tăng không vượt quá 2 lần B0, B2 có chi phí tính toán tăng không vượt quá 2 lần B1, ...).

3.3. Hệ thống phát hiện đám cháy



Hình 2. Thuật toán phát hiện đám cháy sử dụng học chuyển tiếp với mô hình mạng EfficientNet

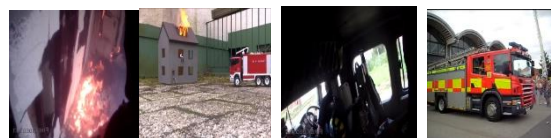
Hình 2 minh họa các bước trong hệ thống phát hiện đám cháy đề xuất: hình ảnh từ camera giám sát sẽ được thu thập để huấn luyện, việc huấn luyện sẽ bắt đầu bằng thao tác lấy các hệ số của mô hình mạng đã được huấn luyện trước trên tập dữ liệu ImageNet, sau đó quá trình huấn luyện với các ảnh có đám cháy và

không có đám cháy sẽ được thực hiện. Sau khi huấn luyện, mô hình mạng mới sẽ được sử dụng để phát hiện đám cháy từ các ảnh test.

3.4. Cài đặt và thực nghiệm

Chương trình demo được cài đặt trên nền tảng thư viện Tensorflow 2.4 và ngôn ngữ Python 3.8. Hệ thống được chạy thử nghiệm trên một máy trạm Z820 với 1 card GPU 1080 Ti 11 GB bộ nhớ.

Dữ liệu thử nghiệm là cơ sở dữ liệu công cộng được công bố trong [3], chúng tôi sử dụng cùng tập huấn luyện có kích thước 224x224 bao gồm 23408 ảnh huấn luyện và 2931 ảnh test. Hệ thống thực hiện huấn luyện với 50 epoch. Một số ảnh trong tập dữ liệu sử dụng được minh họa trong Hình 3.



Hình 3. Một số ảnh minh họa

Kết quả thực nghiệm với các mô hình mạng EfficientNet khác nhau được thể hiện trong Bảng 2.

Bảng 2. Kết quả so sánh các mô hình mạng EfficientNet

TT	Mô hình mạng	Độ chính xác trên tập huấn luyện	Độ chính xác trên tập test
1	B0	98,8	95,5
2	B1	98,5	95,7
3	B2	98,8	95,8
4	B3	99,2	97,5
5	B4	98,7	96,1
6	B5	97,9	96,6
7	B6	97,8	97
8	B7	98,4	96,7

Từ Bảng 2 có thể thấy rằng độ chính xác lớn nhất đạt được của hệ thống là khi sử dụng mô hình mạng EfficientNet-B3 với 97.5% trên tập test.

Bảng 3. Kết quả so sánh với các phương pháp khác

TT	Mô hình mạng	Độ chính xác trên tập test
1	InceptionV1 [3]	93,4
2	InceptionV4 [6]	96,0
3	Hệ thống đề xuất	97,5

Từ Bảng 3 có thể thấy hệ thống đề xuất có độ chính xác cao hơn các cách tiếp cận được công bố gần đây.

Điều này khẳng định cách tiếp cận mà bài báo đề xuất là đúng đắn và kết quả này hoàn toàn có khả năng triển khai trong các ứng dụng thực tế.

4. Kết luận

Với mục tiêu xây dựng một hệ thống phát hiện đám cháy từ hình ảnh, bài báo đã đề xuất sử dụng phương pháp học chuyển tiếp và sử dụng mô hình mạng EfficientNet-B3. Mô hình huấn luyện trước được thực hiện trên cơ sở dữ liệu công cộng ImageNet. Hệ thống được cài đặt bằng ngôn ngữ Python với thư viện Tensorflow 2.4 và chạy thử nghiệm trên hệ điều hành Windows 7 64 bit với 1 máy trạm Z820 được trang bị 1 card GPU Geforce 1080 Ti có 11 GB Ram. Kết quả thực nghiệm trên cơ sở dữ liệu ảnh công cộng cho thấy hệ thống đã đạt được độ chính xác cao (97.5%) và tốt hơn so với một số cách tiếp cận được công bố gần đây. Điều này chứng tỏ cách tiếp cận của bài báo là đúng đắn và hiệu quả.

Trong tương lai nhóm tác giả sẽ tiếp tục thực nghiệm với các cơ sở dữ liệu ảnh lớn hơn, sử dụng thêm các kỹ thuật tăng cường dữ liệu nhằm xác thực sự hiệu quả của hệ thống đề xuất và tăng độ chính xác lên cao hơn.

Lời cảm ơn

Nghiên cứu này được tài trợ bởi Trường Đại học Hàng hải Việt Nam trong đề tài mã số: **DT20-21.52**.

TÀI LIỆU THAM KHẢO

- [1] Nghị định 79/2014/NĐ-CP hướng dẫn Luật Phòng cháy và chữa cháy.
<https://luatvietnam.vn/an-ninh-trat-tu/ngphi-dinh-79-2014-nd-cp-chinh-phu-88482-d1.html>
(accessed Apr. 23, 2021).
- [2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, *ImageNet classification with deep convolutional neural networks*, Commun. ACM, Vol.60, No.6, pp.84-90, May 2017.
doi: 10.1145/3065386.
- [3] A. J. Dunning and T. P. Breckon, *Experimentally Defined Convolutional Neural Network Architecture Variants for Non-Temporal Real-Time Fire Detection*, in 2018 25th IEEE International Conference on Image Processing (ICIP), pp.1558-1562, Oct. 2018.
doi: 10.1109/ICIP.2018.8451657.
- [4] K. Simonyan and A. Zisserman, *Very Deep Convolutional Networks for Large-Scale Image Recognition*, *arXiv:1409.1556 [cs]*, Apr. 2015,

Accessed: Apr. 22, 2021. [Online]. Available: <http://arxiv.org/abs/1409.1556>.

- [5] C. Szegedy *et al.*, *Going deeper with convolutions*, in 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, pp.1-9, Jun. 2015.
doi: 10.1109/CVPR.2015.7298594.
- [6] G. S. C.A., N. Bhowmik, and T. P. Breckon, *Experimental Exploration of Compact Convolutional Neural Network Architectures for Non-Temporal Real-Time Fire Detection*, in 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA), pp.653-658, Dec. 2019.
doi: 10.1109/ICMLA.2019.00119.
- [7] K. He, X. Zhang, S. Ren, and J. Sun, *Deep Residual Learning for Image Recognition*, in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.770-778, Jun. 2016.
doi: 10.1109/CVPR.2016.90.
- [8] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, *Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning*, *arXiv:1602.07261 [cs]*, Aug. 2016, Accessed: Apr. 22, 2021. [Online]. Available: <http://arxiv.org/abs/1602.07261>.
- [9] D. Sarkar, R. Bali, and T. Ghosh, *Hands-on transfer learning with Python: implement advanced deep learning and neural network models using TensorFlow and Keras*. 2019.
- [10] M. Tan and Q. V. Le, *EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks*, *arXiv:1905.11946 [cs, stat]*, Sep. 2020, Accessed: Apr. 22, 2021. [Online]. Available: <http://arxiv.org/abs/1905.11946>.
- [11] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, *MobileNetV2: Inverted Residuals and Linear Bottlenecks*, *arXiv:1801.04381 [cs]*, Mar. 2019, Accessed: Apr. 22, 2021. [Online]. Available: <http://arxiv.org/abs/1801.04381>.

Ngày nhận bài:	23/4/2021
Ngày nhận bản sửa:	02/5/2021
Ngày duyệt đăng:	09/5/2021