



MỤC LỤC

01

GIỚI THIỆU BÀI TOÁN

02

NHẮC LẠI VỀ CNN

03

EFFICIENT NETS

04

DATASET

05

PHƯƠNG PHÁP THỰC HIỆN

06

KẾT QUẢ MONG MUỐN

07

TÀI LIỆU THAM KHẢO

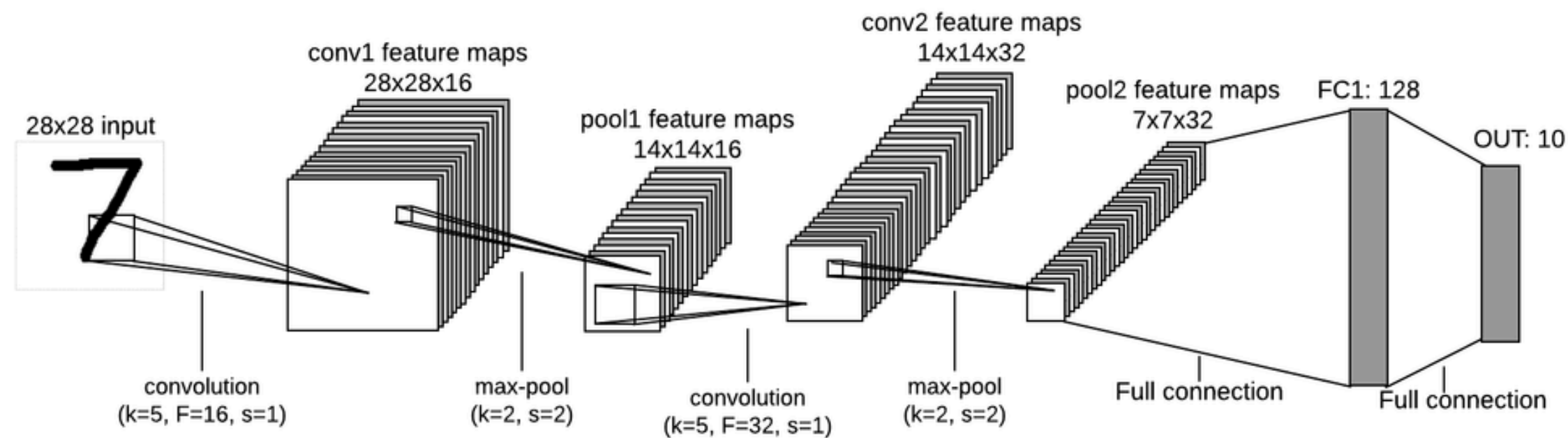


GIỚI THIỆU BÀI TOÁN

Bài toán nhận diện lá cây đã được nghiên cứu rộng rãi với nhiều phương pháp, trong đó CNN truyền thống đạt nhiều kết quả đáng kể.

Đề tài này sử dụng EfficientNet để phát triển ứng dụng nhận diện lá trên thiết bị di động, nhằm so sánh hiệu suất của EfficientNet với CNN truyền thống về tốc độ xử lý, độ chính xác và khả năng triển khai trên nền tảng di động.

NHẮC LẠI VỀ CNN



CNN là một loại mạng nơ-ron chuyên biệt cho việc xử lý dữ liệu lưới như hình ảnh.

CNN được tạo thành từ nhiều lớp khác nhau, bao gồm Convolutional Layer, Pooling Layer và Fully Connected Layer.

Những lớp này giúp mô hình học được các đặc trưng từ hình ảnh đầu vào một cách tự động.

NHẮC LẠI VỀ CNN

01 CONVOLUTIONAL LAYER

Tích chập ảnh đầu vào và kernel tạo ra feature map giữ lại các đặc trưng

$$\begin{array}{|c|c|c|} \hline \text{Input} & & \\ \hline 0 & 1 & 2 \\ \hline 3 & 4 & 5 \\ \hline 6 & 7 & 8 \\ \hline \end{array} * \begin{array}{|c|c|} \hline \text{Kernel} & \\ \hline 0 & 1 \\ \hline 2 & 3 \\ \hline \end{array} = \begin{array}{|c|c|} \hline \text{Output} & \\ \hline 19 & 25 \\ \hline 37 & 43 \\ \hline \end{array}$$

Bộ lọc di chuyển qua hình ảnh và áp dụng các phép nhân tích chập, giúp phát hiện các chi tiết như cạnh, đường viền, màu sắc.

Mỗi bộ lọc học được một loại đặc trưng khác nhau.

NHẮC LẠI VỀ CNN

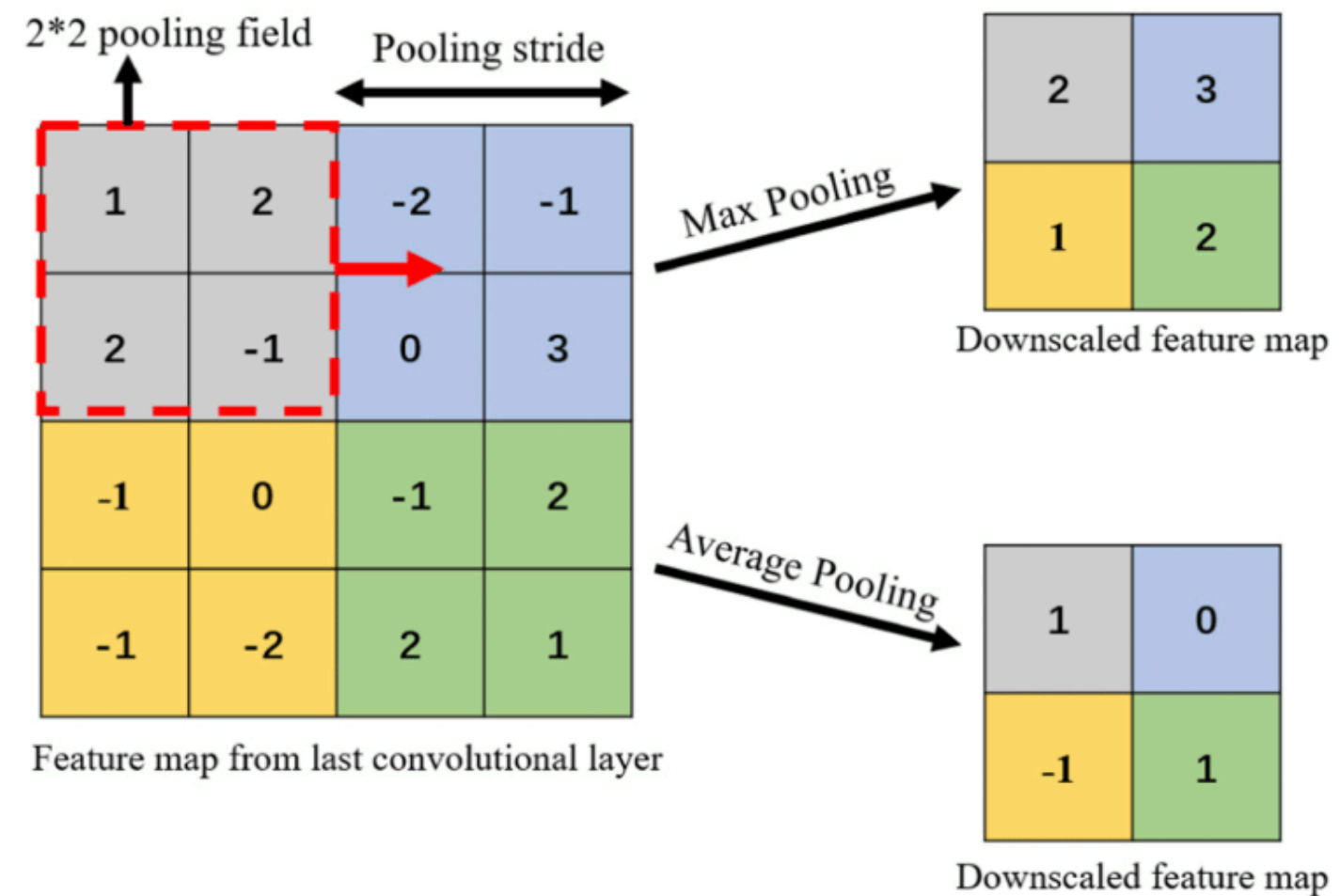
02 POOLING LAYER

Giảm kích thước của đặc trưng, giữ lại những thông tin quan trọng.

Tác dụng của pooling là giảm số lượng tham số và tính toán, đồng thời tránh overfitting.

Max Pooling: Giữ lại giá trị lớn nhất trong vùng chọn.

Average Pooling: Tính giá trị trung bình trong vùng chọn.

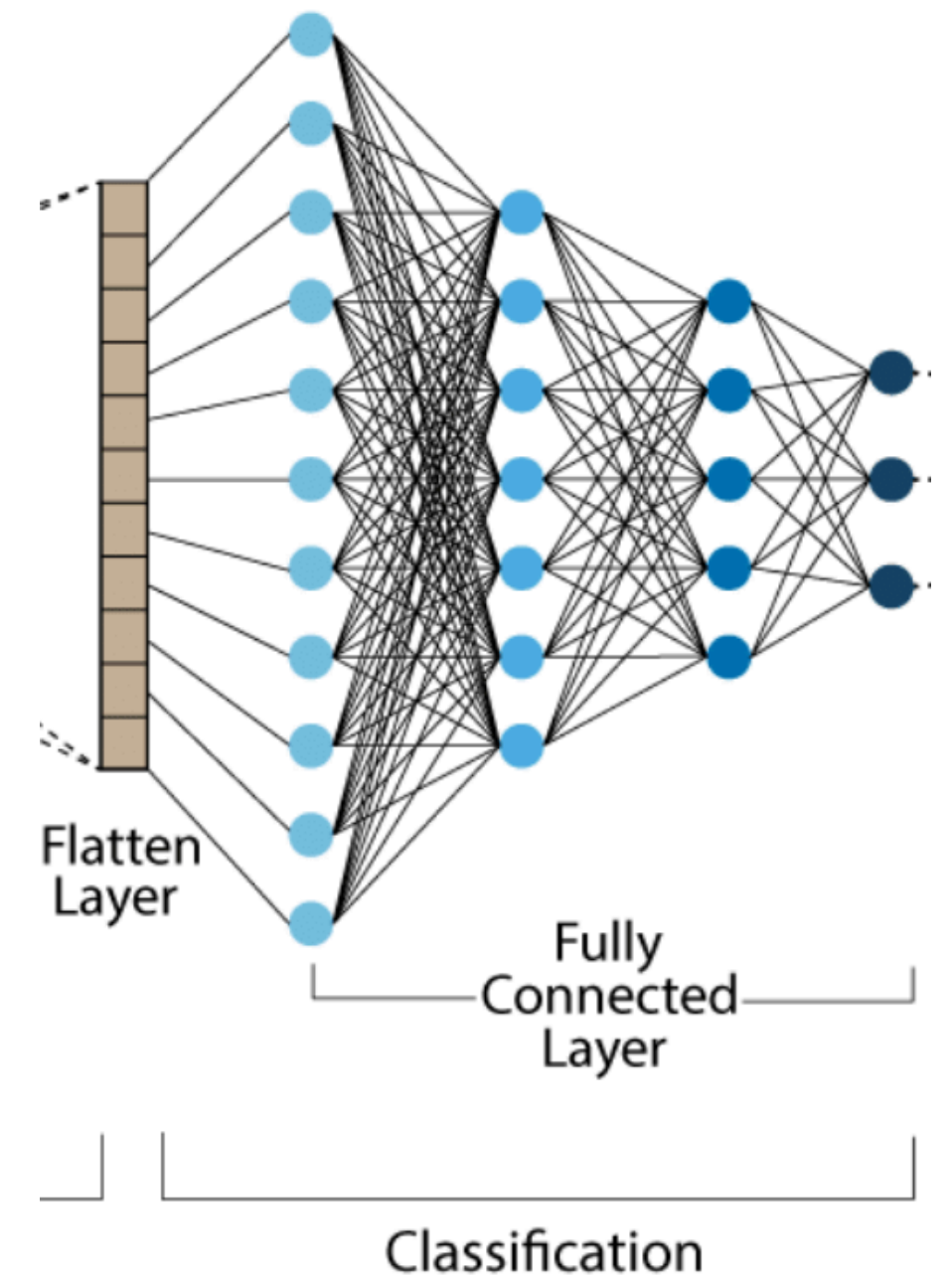


NHẮC LẠI VỀ CNN

03 FULLY-CONNECTED

Kết nối toàn bộ các đặc trưng từ các lớp trước và gộp lại để dự đoán đầu ra.

Đây là nơi toàn bộ các đặc trưng đã được học sẽ được kết hợp để phân loại hoặc dự đoán.



NHẮC LẠI VỀ CNN

Ưu điểm

Học các đặc trưng từ ảnh mà không cần tiền xử lý thủ công.

Xác định mối quan hệ không gian giữa các điểm ảnh, học đặc trưng cục bộ.

Chia sẻ tham số giúp giảm số lượng tham số cần học, tránh overfitting.

Khả năng tổng quát hóa tốt: Nhận diện đối tượng bất kể biến đổi về tỷ lệ, xoay, dịch chuyển.

Ứng dụng rộng rãi: Phân loại ảnh, nhận diện đối tượng, phân đoạn ảnh, nhận diện khuôn mặt

NHẮC LẠI VỀ CNN

Nhược điểm của CNNs truyền thống

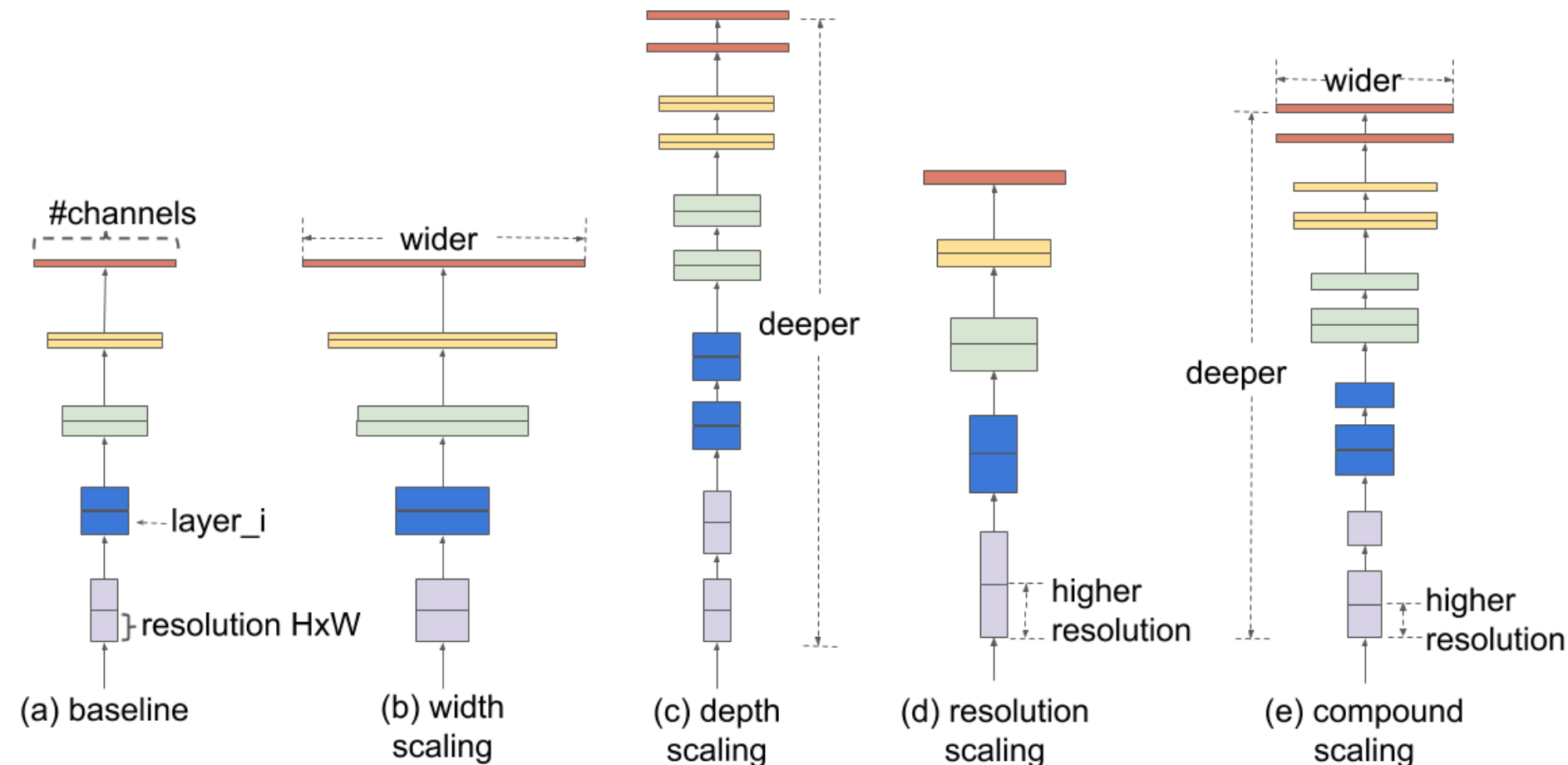
CNN truyền thống như VGG, ResNet yêu cầu nhiều tham số và tài nguyên tính toán để đạt hiệu suất cao.

CNN truyền thống thường phải điều chỉnh kích thước và số lượng lớp thủ công để đạt hiệu quả cao.

CNN truyền thống không tối ưu về sự cân bằng giữa độ rộng, độ sâu, và độ phân giải, dễ dẫn đến overfitting hoặc hiệu suất không ổn định.

CNN truyền thống gặp khó khăn khi triển khai trên các thiết bị di động hoặc nhúng do yêu cầu tài nguyên lớn.

EFFICIENT NETS



EfficientNet là một mạng neural được phát triển bởi Google nhằm giải quyết các vấn đề về hiệu quả tài nguyên trong CNN.

Nó sử dụng một phương pháp có Compound Scaling để mở rộng mô hình một cách thông minh và cân bằng giữa độ sâu, chiều rộng và độ phân giải.

EFFICIENT NETS



Compound Scaling

Các thành phần:

- Depth: Độ sâu của mô hình, liên quan đến số lượng các lớp (layers) của mạng.
- Width: Chiều rộng của mô hình, liên quan đến số lượng neurons trong một lớp.
- Resolution: Độ phân giải của ảnh đầu vào.

Thay vì chỉ tăng một yếu tố, EfficientNet tăng đồng đều cả ba yếu tố này để tối ưu hiệu suất mô hình mà không tốn quá nhiều tài nguyên.



EFFICIENT NETS

Công thức

$$\text{depth: } d = \alpha^\phi$$

$$\text{width: } w = \beta^\phi$$

$$\text{resolution: } r = \gamma^\phi$$

$$\text{s.t. } \alpha \cdot \beta^2 \cdot \gamma^2 \approx 2$$

$$\alpha \geq 1, \beta \geq 1, \gamma \geq 1$$

- d là hệ số mở rộng độ sâu.
- w là hệ số mở rộng chiều rộng.
- r là hệ số mở rộng độ phân giải.
- α, β, γ là các hệ số tỷ lệ
- ϕ một tham số điều chỉnh (compound coefficient) để kiểm soát mức độ mở rộng tổng thể của mô hình.

EFFICIENT NETS

Các phiên bản

EfficientNet-B0: Là phiên bản cơ sở với bộ tham số $\alpha=1, \beta=1, \gamma=1$

EfficientNet-B1: Đặt cố định giá trị của φ bằng 1, nhóm tác giả thu được bộ giá trị tối ưu

$$\alpha=1,2, \beta=1,1, \gamma=1,15$$

Sau khi xác định được bộ giá trị tối ưu, nhóm tác giả cố định α, β, γ là hằng số và mở rộng quy mô mạng cơ sở với φ khác nhau để thu được EfficientNet-B1 đến B7

EFFICIENT NETS

Model	Top-1 Acc.	Top-5 Acc.	#Params	Ratio-to-EfficientNet	#FLOPs	Ratio-to-EfficientNet
EfficientNet-B0	77.1%	93.3%	5.3M	1x	0.39B	1x
ResNet-50 (He et al., 2016)	76.0%	93.0%	26M	4.9x	4.1B	11x
DenseNet-169 (Huang et al., 2017)	76.2%	93.2%	14M	2.6x	3.5B	8.9x
EfficientNet-B1	79.1%	94.4%	7.8M	1x	0.70B	1x
ResNet-152 (He et al., 2016)	77.8%	93.8%	60M	7.6x	11B	16x
DenseNet-264 (Huang et al., 2017)	77.9%	93.9%	34M	4.3x	6.0B	8.6x
Inception-v3 (Szegedy et al., 2016)	78.8%	94.4%	24M	3.0x	5.7B	8.1x
Xception (Chollet, 2017)	79.0%	94.5%	23M	3.0x	8.4B	12x
EfficientNet-B2	80.1%	94.9%	9.2M	1x	1.0B	1x
Inception-v4 (Szegedy et al., 2017)	80.0%	95.0%	48M	5.2x	13B	13x
Inception-resnet-v2 (Szegedy et al., 2017)	80.1%	95.1%	56M	6.1x	13B	13x
EfficientNet-B3	81.6%	95.7%	12M	1x	1.8B	1x
ResNeXt-101 (Xie et al., 2017)	80.9%	95.6%	84M	7.0x	32B	18x
PolyNet (Zhang et al., 2017)	81.3%	95.8%	92M	7.7x	35B	19x
EfficientNet-B4	82.9%	96.4%	19M	1x	4.2B	1x
SENet (Hu et al., 2018)	82.7%	96.2%	146M	7.7x	42B	10x
NASNet-A (Zoph et al., 2018)	82.7%	96.2%	89M	4.7x	24B	5.7x
AmoebaNet-A (Real et al., 2019)	82.8%	96.1%	87M	4.6x	23B	5.5x
PNASNet (Liu et al., 2018)	82.9%	96.2%	86M	4.5x	23B	6.0x
EfficientNet-B5	83.6%	96.7%	30M	1x	9.9B	1x
AmoebaNet-C (Cubuk et al., 2019)	83.5%	96.5%	155M	5.2x	41B	4.1x
EfficientNet-B6	84.0%	96.8%	43M	1x	19B	1x
EfficientNet-B7	84.3%	97.0%	66M	1x	37B	1x
GPipe (Huang et al., 2018)	84.3%	97.0%	557M	8.4x	-	-

We omit ensemble and multi-crop models ([Hu et al., 2018](#)), or models pretrained on 3.5B Instagram images ([Mahajan et al., 2018](#)).

DATASET

FLAVIA

- Số lượng ảnh: 1.907 ảnh lá cây.
- Số lượng loại cây: 32 loài cây khác nhau.
- Kích thước hình ảnh: 1600x1200 pixels.
- Định dạng ảnh: JPEG.
- Hình ảnh được chụp với nền trắng, tạo điều kiện thuận lợi cho việc xử lý và nhận diện.

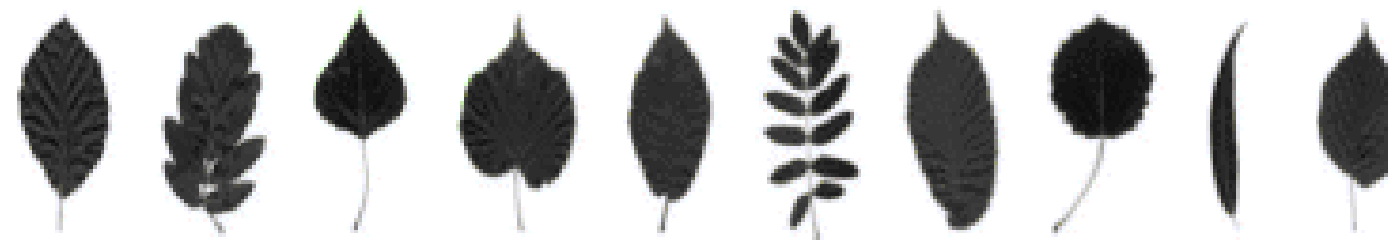
CHIA DỮ LIỆU TRAINING

Chia dữ liệu thành 3 phần:

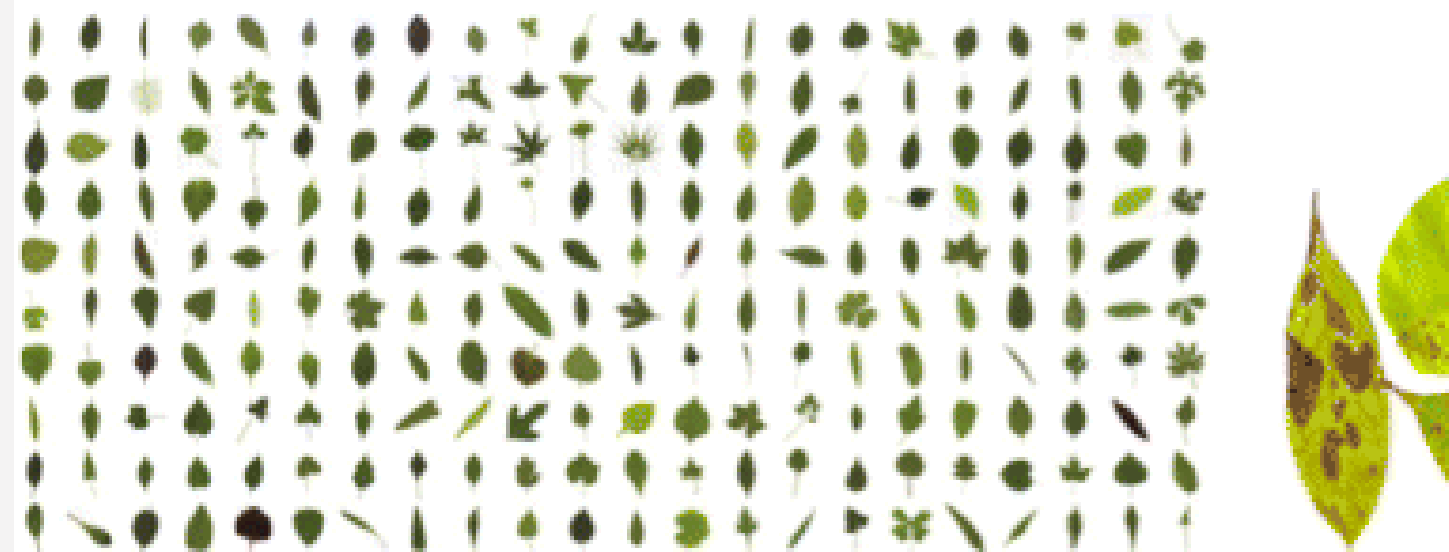
- Tập huấn luyện (Training Set): 70% dữ liệu – Dùng để huấn luyện mô hình.
- Tập xác nhận (Validation Set): 15% dữ liệu – Dùng để tối ưu hóa và điều chỉnh tham số của mô hình.
- Tập kiểm tra (Test Set): 15% dữ liệu – Dùng để đánh giá hiệu năng mô hình sau khi huấn luyện.



(a)



(c)



(e)

PHƯƠNG PHÁP THỰC HIỆN

Kế hoạch dự kiến

Huấn luyện mô hình trên Google Colab

- Sau khi huấn luyện mô hình EfficientNet trên Google Colab (sử dụng TensorFlow/Keras), bạn cần lưu mô hình đã huấn luyện dưới định dạng .h5

Chuyển đổi mô hình sang TensorFlow Lite

- Sử dụng TensorFlow Lite Converter để chuyển đổi mô hình

Tích hợp mô hình TensorFlow Lite vào ứng dụng di động (Android)

- Sử dụng thư viện TensorFlow Lite để chạy mô hình trong ứng dụng

KẾT QUẢ MONG ĐỢI

Triển khai được ứng dụng trên di động

Độ chính xác cao: trên 90%

Khả năng hoạt động tốt trong môi trường thực tế

So sánh độ hiệu quả với các mô hình CNN khác





TÀI LIỆU THAM KHẢO

Mingxing Tan, Quoc V. Le, EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks

Wang-Su Jeon, Sang-Yong Rhee, Plant Leaf Recognition Using a Convolution Neural Network

<https://research.google/blog/efficientnet-improving-accuracy-and-efficiency-through-automl-and-model-scaling/>



THANK YOU