



Đồ Án Cơ Sở Ngành

Công Nghệ Thông Tin

Tên đề tài: Mô Hình CNN cho phân loại động vật

Giáo viên hướng dẫn:
Th.s Ngô Thanh Huy

Sinh viên thực hiện:
Nguyễn Hữu Tiến
MSSV: 110123195
Lớp: DA23TTD

Nội Dung Chính

- 1. Giới thiệu Học Máy**
- 2. Giới thiệu Học Sâu**
- 3. Giới thiệu mạng nơ-ron**
- 4. Giới thiệu về mạng nơ-ron tích chập**
- 5. Mô tả mô hình**

1 Giới thiệu Học Máy

1.1 Khái niệm và nguyên tắc hoạt động

Học máy là một tập con của trí tuệ nhân tạo (AI) cho phép máy tính học từ dữ liệu và đưa ra dự đoán hay quyết định. Nó triển khai các thuật toán nhằm tạo điều kiện thuận lợi cho dự đoán và quyết định này. Các thuật toán được thiết kế với mục đích cải thiện hiệu suất đã trở nên ngày càng chính xác và hiệu quả hơn theo thời gian.

Nguyên tắc hoạt động: - Thu thập dữ liệu

- Tiền xử lý dữ liệu
- Chọn mô hình phù hợp
- Huấn luyện mô hình
- Đánh giá mô hình

Ứng dụng của học máy trong thực tế: - Công nghệ thông tin và Internet.

- Tài chính.
- Y tế.
- Ô tô tự hành và IoT (Internet vạn vật).
- Bán lẻ và Thương mại điện tử.
- Giáo dục.

2. Giới thiệu học sâu

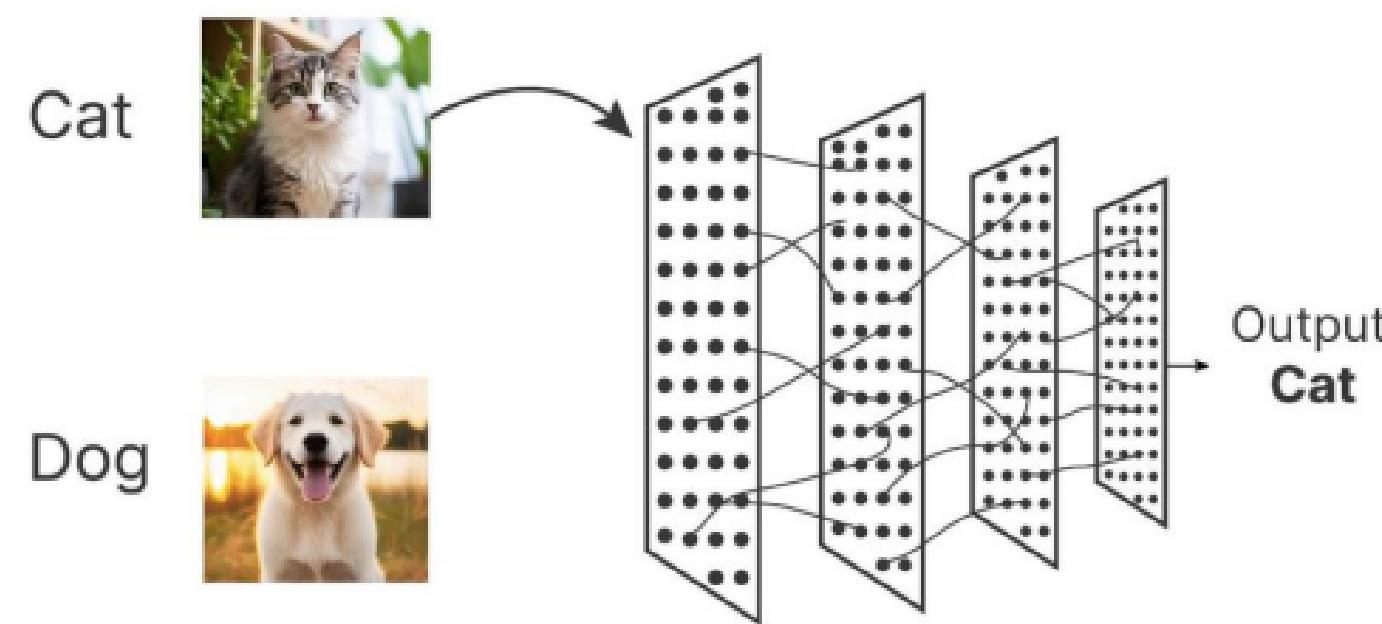
2.1 Khái niệm và nguyên tắc hoạt động

Học sâu là một phần của trí tuệ nhân tạo (AI) mà các mạng nơ-ron sâu được sử dụng để học và hiểu dữ liệu phức tạp. Trong Deep Learning, các mạng nơ-ron được cấu trúc với nhiều lớp, mỗi lớp thực hiện các phép tính phức tạp để tự động rút trích các đặc trưng từ dữ liệu.

Mạng lưới thần kinh học sâu hoặc mạng lưới thần kinh nhân tạo cố gắng bắt chước bộ não con người thông qua sự kết hợp của dữ liệu đầu vào, trọng số và độ lệch. Các phần tử này phối hợp với nhau để nhận dạng, phân loại và mô tả chính xác các đối tượng trong dữ liệu.

Ví dụ:

- Mạng thần kinh tích chập (CNN), được sử dụng chủ yếu trong các ứng dụng phân loại hình ảnh và thị giác máy tính



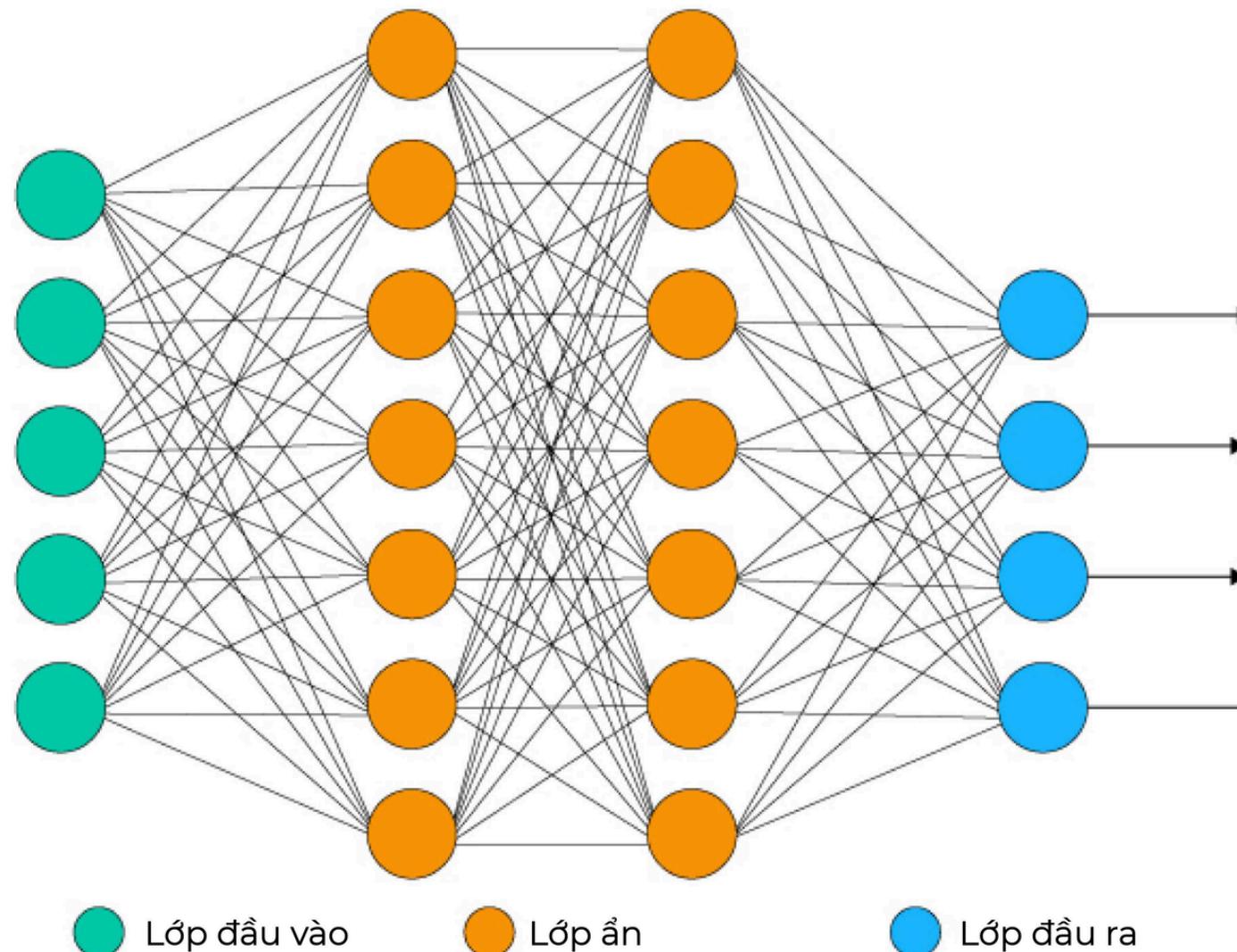
3. Giới thiệu mạng nơ-ron

3.1 Khái niệm và nguyên tắc hoạt động

Neural Network (Mạng nơ-ron) là một mô hình tính toán được lấy cảm hứng từ cách hoạt động của não người. Nó bao gồm một loạt các "nơ-ron" nhân tạo được tổ chức thành các lớp (layers), trong đó mỗi nơ-ron kết nối với nơ-ron trong lớp tiếp theo. Mỗi nơ-ron có thể nhận đầu vào, xử lý thông tin và gửi đầu ra đến các nơ-ron khác.

Mạng nơ-ron hoạt động dựa trên một số nguyên tắc cơ bản. Dưới đây là một phác thảo về cách hoạt động của mạng nơ-ron:

- Nhập dữ liệu
- Tính toán trên nơ-ron
- Tính toán trên các lớp ẩn
- Tính toán đầu ra
- So sánh với kết quả mong muốn
- Đánh giá và điều chỉnh



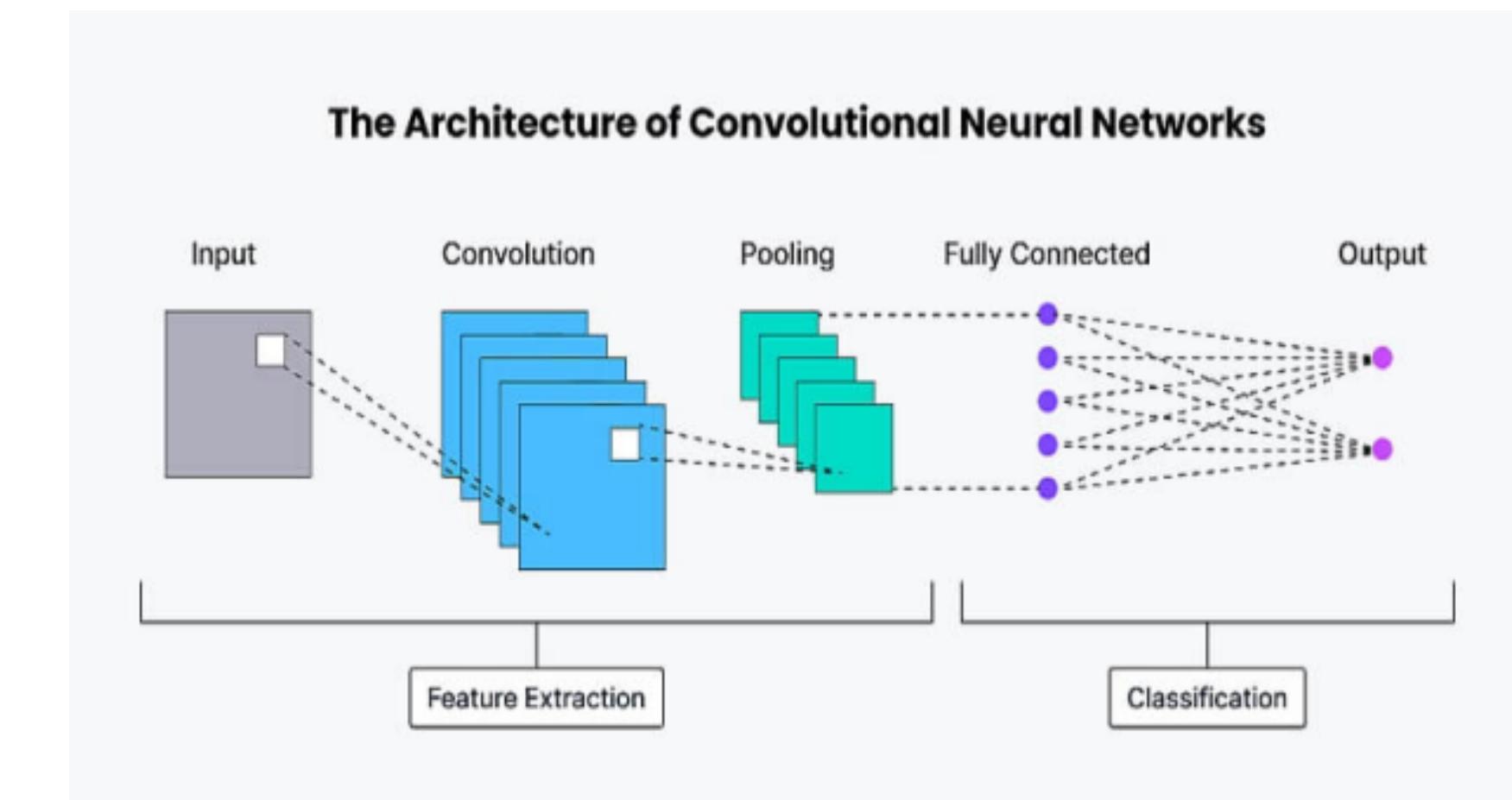
4 Giới thiệu mạng nơ-ron tích chập

4.1 Khái niệm và nguyên tắc hoạt động

Mạng nơ-ron tích chập (CNN) là một loại mạng nơ-ron nhân tạo được thiết kế đặc biệt để xử lý dữ liệu dạng lưới như hình ảnh và video. CNN thường được sử dụng trong các ứng dụng liên quan đến thị giác máy tính và nhận dạng hình ảnh. Đặc điểm chính của CNN là sự áp dụng của các lớp tích chập, lớp gộp (pooling), và các lớp kích hoạt phi tuyến tính, giúp mô hình có khả năng học các đặc trưng cục bộ trong dữ liệu đầu vào một cách hiệu quả.

Nguyên tắc hoạt động của mạng tích chập (CNN) có thể được phân thành các bước chính như sau:

- Lớp tích chập (Convolution)
- Hàm kích hoạt (Activation)
- Lớp gộp (Pooling)
- Bước lặp lại (Repeat)
- Lớp kết nối đầy đủ (Fully Connected – Dense)
- Đầu ra (Output)



5. Mô tả mô hình

5.1 Yêu cầu chức năng

Mô hình CNN phân loại động vật cần đáp ứng các yêu cầu chức năng sau: Nhận đầu vào là hình ảnh động vật có định dạng phổ biến như JPG, PNG.

Thực hiện tiền xử lý ảnh bao gồm thay đổi kích thước, chuẩn hóa giá trị pixel và chuyển đổi dữ liệu phù hợp với mô hình. Tự động trích xuất đặc trưng ảnh thông qua các lớp tích chập (Convolution) và gộp (Pooling). Thực hiện phân loại ảnh và đưa ra nhãn động vật tương ứng.

5.2 Yêu cầu dữ liệu và môi trường

Dữ liệu đầu vào là tập ảnh động vật được phân chia theo từng lớp (loài).

Dữ liệu được chia thành tập huấn luyện và tập kiểm tra theo tỷ lệ xác định.

Ảnh đầu vào được chuẩn hóa về kích thước 128×128 pixel, 3 kênh màu (RGB).

Nhãn dữ liệu được biểu diễn dưới dạng one-hot encoding.

Ngôn ngữ lập trình: Python.

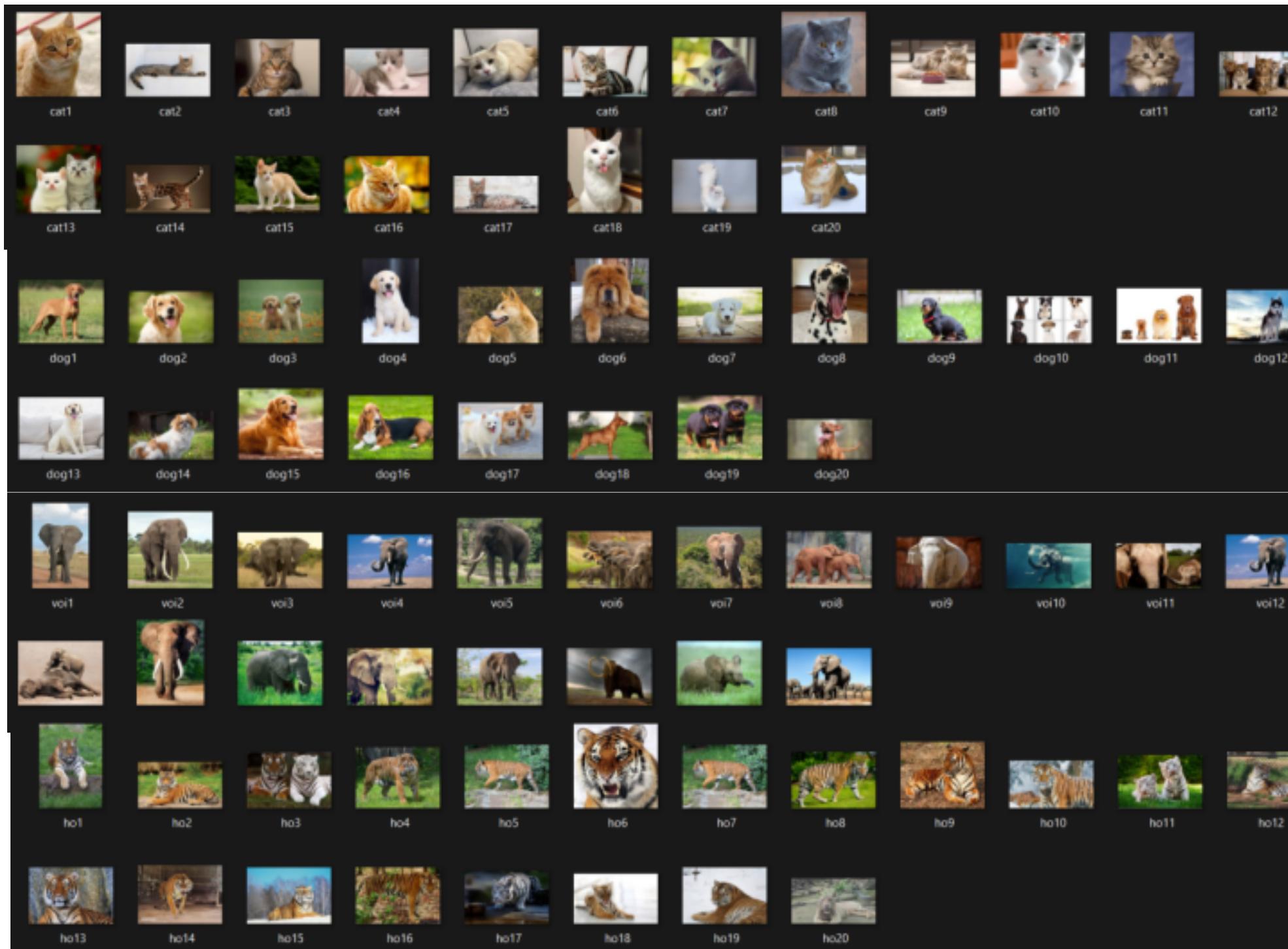
Thư viện sử dụng: TensorFlow, Keras, NumPy, OpenCV, Matplotlib.

Hệ điều hành: Windows hoặc Linux.

Môi trường chạy: CPU (không yêu cầu GPU chuyên dụng).

5.3 Thu thập dữ liệu

Bộ dữ liệu “Dog-Cat-Tiger-Elephant” được lấy ngẫu nhiên trên Internet bao gồm 80 bức ảnh chia thành 2 tập là tập huấn luyện và tập kiểm thử, trong đó 20 bức ảnh là ảnh con mèo, 20 bức ảnh là ảnh con chó, 20 bức ảnh là ảnh con voi, 20 bức ảnh là ảnh con hổ. Ảnh con mèo được gán nhãn thông qua đặt tên file là “cat.x” đặt trong thư mục cat, và ảnh con chó được gán nhãn thông qua đặt tên file là “dog.x” đặt trong thư mục dogs, và ảnh con voi được gán nhãn thông qua đặt tên file là “voi.x”, và ảnh con hổ được gán nhãn thông qua đặt tên file là “tiger.x”



5.4 Xử lý dữ liệu

Tập huấn luyện (train): dùng để huấn luyện mô hình CNN.

Tập kiểm tra (test): dùng để đánh giá hiệu quả của mô hình. Việc chia dữ liệu được thực hiện tự động bằng chương trình split_dataset.py với tỷ lệ 70% cho tập huấn luyện và 30% cho tập kiểm tra, đảm bảo dữ liệu được xáo trộn ngẫu nhiên trước khi chia.

Chuẩn hóa dữ liệu ảnh Các ảnh trong tập dữ liệu có kích thước và giá trị pixel khác nhau. Do đó, ảnh được chuẩn hóa giá trị pixel về khoảng [0,1] bằng cách chia cho 255. Việc chuẩn hóa giúp mô hình hội tụ nhanh hơn và ổn định hơn trong quá trình huấn luyện.

Tên thư mục chính là tên nhãn của ảnh.

Kiểm tra thứ tự nhãn dán print

cat → [1, 0, 0, 0]→0

dog → [0, 1, 0, 0]→1

elephant → [0, 0, 1, 0]→2

tiger → [0, 0, 0, 1]→3

```
# ===== CẤU HÌNH =====
RAW_DIR = "raw_dataset"
OUTPUT_DIR = "animal_dataset"
TRAIN_RATIO = 0.7 # 70% train - 30% test

random.seed(42)

# ===== TẠO THƯ MỤC =====
for folder in ["train", "test"]:
    for cls in os.listdir(RAW_DIR):
        os.makedirs(os.path.join(OUTPUT_DIR, folder, cls), exist_ok=True)
```

```
train_data = train_gen.flow_from_directory(
    TRAIN_DIR,
    target_size=(IMG_SIZE, IMG_SIZE),
    batch_size=BATCH_SIZE,
    class_mode="categorical")
```

5.5 Xây dựng mô hình

Kiến trúc mô hình được thiết kế nhằm trích xuất đặc trưng từ ảnh đầu vào và thực hiện phân loại động vật một cách hiệu quả.

- Tầng đầu vào (Input Layer)
- Khối tích chập thứ nhất (Convolution Block 1)
- Khối tích chập thứ hai (Convolution Block 2)
- Tầng Flatten
- Tầng kết nối đầy đủ (Fully Connected Layer)
- Tầng đầu ra (Output layer)

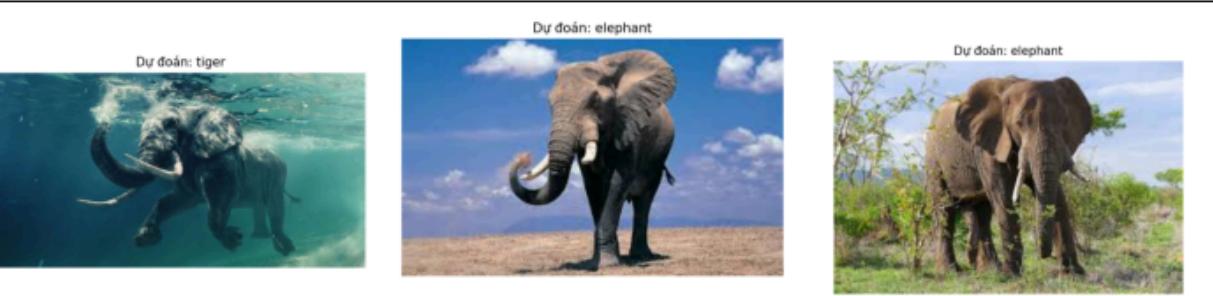
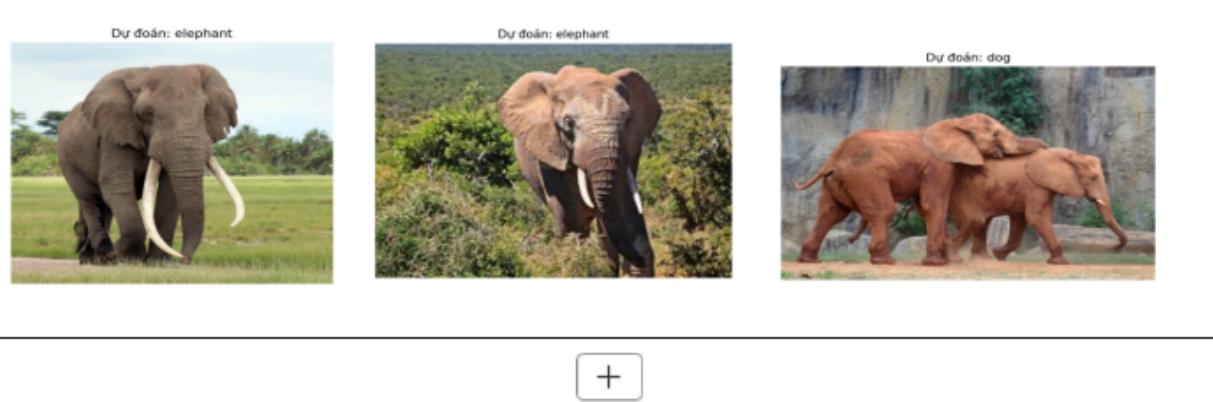
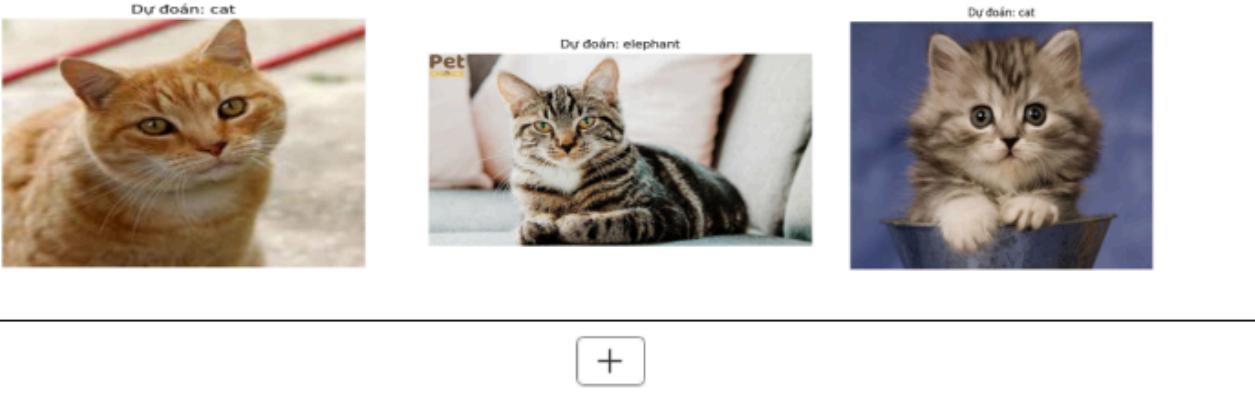
Huấn luyện mô hình

Huấn luyện mô hình CNN phân loại động vật. Sau khi xây dựng kiến trúc mô hình và chuẩn bị dữ liệu, mô hình CNN được tiến hành huấn luyện nhằm học các đặc trưng từ tập dữ liệu ảnh và tối ưu khả năng phân loại các loài động vật.

6. Thực nghiệm với tập ảnh

6.1 Thực nghiệm với ảnh test

Thực nghiệm sẽ tiến hành test lấy 6 ảnh của thư mục cat kết quả như sau:



Bảng thống kê kết quả test:

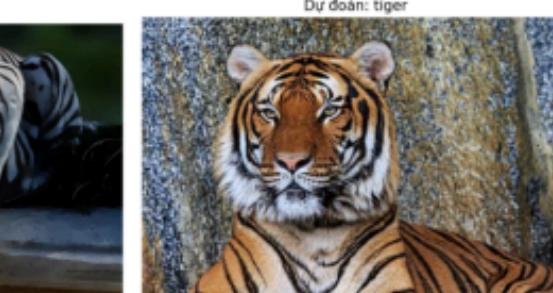
Tên loài	Số ảnh dự đoán đúng	Số ảnh dự đoán sai	Tỷ lệ đúng
Cat	3	3	50%
Dog	4	2	66%
Elephant	4	2	66%
Tiger	2	4	33%

Trong bảng độ chính xác cao nhất là 66% và độ chính xác thấp nhất là 33%

6.2 Thực nghiệm với tập train

Thử nghiệm này được thực hiện trên các ảnh nằm trong tập huấn luyện.

Kịch bản test cũng được thực hiện mỗi test lấy ngẫu nhiên 6 ảnh để dự đoán.



Bảng thống kê kết quả test trên tập train:

Tên loài	Số ảnh dự đoán đúng	Số ảnh dự đoán sai	Tỷ lệ đúng
Cat	6	0	100%
Dog	6	0	100%
Elephant	6	0	100%
Tiger	6	0	100%

Trong bảng, qua các lần test, độ chính xác cao nhất là 100%, độ chính xác thấp nhất là 100%. Độ chính xác trung bình là 100%. So với kết quả test ở kịch bản trước, kịch bản này có độ chính xác rất cao. Điều này là do kiểm thử trên chính dữ liệu huấn luyện, nên kết quả tốt. Trong khi đó, kiểm thử trên dữ liệu mới hoàn toàn (trong kịch bản thứ nhất) thì kết quả chưa tốt do tập dữ liệu huấn luyện chưa đủ lớn để bao phủ, hết các trường hợp. Nếu tăng dữ liệu huấn luyện, độ chính xác của mô hình sẽ được cải thiện. Vấn đề này sẽ được giải quyết trong “hướng phát triển tương lai”.

6.3 Đánh giá mô hình

Kết quả huấn luyện mô hình

Mô hình được huấn luyện trong 10 epoch với batch size = 32

Hàm mất mát categorical_crossentropy và optimizer Adam.

Độ chính xác (accuracy) của mô hình tăng dần qua các epoch.

Giá trị loss giảm ổn định, cho thấy mô hình học hiệu quả và hội tụ tốt.

Điều này chứng tỏ kiến trúc CNN đã lựa chọn là phù hợp với tập dữ liệu và bài toán đặt ra.

Mô hình đạt độ chính xác tương đối tốt trên tập dữ liệu kiểm tra.

Không xuất hiện hiện tượng overfitting nghiêm trọng.

Sai số chủ yếu xuất hiện ở các ảnh có: Góc chụp phức tạp, nhiều vật thể trong cùng một ảnh, ánh sáng kém hoặc nền phức tạp. Điều này phản ánh hạn chế chung của các mô hình CNN khi dữ liệu huấn luyện còn hạn chế.

7. Tổng Kết

Sau quá trình nghiên cứu lý thuyết, tìm hiểu mô hình học sâu, chuẩn bị dữ liệu, xây dựng kiến trúc mạng và huấn luyện mô hình, đề tài “Xây dựng mô hình mạng nơ-ron tích chập (CNN) cho bài toán phân loại động vật” đã được hoàn thành đúng theo mục tiêu ban đầu đề ra. Mô hình CNN được xây dựng nhằm tự động nhận dạng và phân loại các loài động vật dựa trên ảnh đầu vào. Thông qua việc áp dụng các tầng tích chập, tầng gộp và tầng kết nối đầy đủ, mô hình có khả năng trích xuất đặc trưng ảnh và đưa ra kết quả phân loại tương ứng.

Xây dựng thành công mô hình mạng nơ-ron tích chập (CNN) để phân loại động vật dựa trên ảnh đầu vào. Thực hiện đầy đủ các bước tiền xử lý dữ liệu ảnh gồm chia tập dữ liệu, thay đổi kích thước ảnh, chuẩn hóa giá trị pixel và gán nhãn tự động.

Thiết kế kiến trúc mô hình CNN gồm các tầng tích chập, max pooling, flatten và dense phù hợp với bài toán phân loại đa lớp.

Huấn luyện mô hình với các tham số phù hợp và đạt được kết quả phân loại tương đối tốt trên tập dữ liệu kiểm tra.

Triển khai chức năng dự đoán ảnh mới và hiển thị trực quan kết quả phân loại, giúp dễ dàng kiểm tra và đánh giá mô hình.

Cảm ơn thầy cô đã lắng nghe