TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

──────── \* ───────

**BÁO CÁO**

MÔN: PROJECT 2

**XÂY DỰNG MẠNG THẦN KINH SÂU TRONG**

**NHẬN DẠNG KHUÔN MẶT MÈO SỬ DỤNG**

**HÀM MẤT MÁT BỘ BA**

Sinh viên thực hiện : **Lê Thị Kim Anh**

MSSV : 20229001 **– CNTTK67**

Giáo viên hướng dẫn : **Nguyễn Mạnh Tuấn**

***Hà Nội, tháng 12 năm 2023***

**MỤC LỤC**

**[Chương 1: GIỚI THIỆU ĐỀ TÀI 4](#_Toc24495)**

**[1.1. Đặt vấn đề 4](#_Toc22125)**

**[1.2. Mục tiêu và phạm vi đề tài 4](#_Toc27667)**

**[1.3. Định hướng giải pháp 5](#_Toc25777)**

**[1.4. Bố cục báo cáo 5](#_Toc6296)**

**[Chương 2: KIẾN TRÚC MẠNG THẦN KINH SÂU VÀ 6](#_Toc1105)**

**[XÂY DỰNG HÀM MẤT MÁT BỘ BA 6](#_Toc18633)**

**[2.1. Kiến trúc mạng thần kinh sâu 6](#_Toc19109)**

**[2.2. Hàm mất mát bộ ba và kỹ thuật khai thác bộ ba trực tuyến 7](#_Toc25781)**

**[2.2.1. Hàm mất mát bộ ba có chính quy hóa 7](#_Toc20302)**

**[2.2.2. Khai thác bộ ba 9](#_Toc24087)**

**[2.3. Xây dựng hàm đánh giá độ chính xác của mô hình 11](#_Toc6959)**

***[2](#_Toc26615)*[.3.1. Hàm chính xác bộ ba 11](#_Toc26615)**

**[2.3.2. Hàm chính xác nhận diện 11](#_Toc29728)**

**[Chương 3: CHUẨN BỊ DỰ LIỆU VÀ THỰC HIỆN HUẤN LUYỆN 12](#_Toc15381)**

**[3.1. Chuẩn bị dữ liệu 12](#_Toc4889)**

**[3.2. Huấn luyện mô hình nhận diện 13](#_Toc25815)**

**[3.3. Khả năng nhận diện của mô hình sau huấn luyện 14](#_Toc240)**

**[KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 17](#_Toc30619)**

CHI TIẾT CÔNG VIỆC THỰC HIỆN

|  |
| --- |
| **Nội dung công việc:**  - Công việc 1: Tìm hiểu về bài toán nhận diện gương mặt  - Công việc 2: Tìm hiểu về hàm mất mát bộ ba  - Công việc 3: Tìm hiểu về kiến trúc mạng thần kinh sâu Resnet  - Công việc 4: Khảo sát, tìm kiếm bộ dữ liệu phù hợp |
| **Công cụ phát triển:**  - Ngôn ngữ lập trình: Python  - Môi trường phát triển: Tensorflow |
| **Lịch trình thực hiện (Bám sát nội dung công việc):**  - Tuần 1: Tìm hiểu về bài toán nhận diện gương mặt  - Tuần 2-5: Tìm hiểu về hàm mất mát bộ ba, phương pháp chính quy hóa và cách triển khai trên Python  - Tuần 6-8: Tìm hiểu về kiến trúc mạng thần kinh sâu Resnet, quyết định lựa chọn kiến trúc Resnet50  - Tuần 9-10: Khảo sát, tìm kiếm bộ dữ liệu huấn luyện. Thu thập dữ liệu huấn luyện từ các khoản Instagram công khai  - Tuần 11: Tìm hiểu về YOLO và cách sử dụng YOLO  - Tuần 12: Xây dựng mô hình nhận diện gương mặt mèo, đưa vào huấn luyện với bộ dữ liệu đã tiền xử lý bằng YOLO  - Tuần 13: Viết báo cáo kết quả |

# Chương 1: GIỚI THIỆU ĐỀ TÀI

## Đặt vấn đề

Bài toán nhận diện gương mặt có thể coi là một trong những bài toán kinh điển và được ứng dụng rộng rãi của học sâu. Có thể nói, phương pháp học sâu đã mang lại những thành tựu ấn tượng trong nhận dạng và xác minh khuôn mặt con người. Tuy nhiên, có rất ít nghiên cứu về việc áp dụng học sâu vào nhận dạng động vật.

Hiện nay, bên cạnh chó, mèo cũng được coi là thú nuôi phổ biến trong giới trẻ ở Việt Nam do một số lý do như: ***(i)*** phù hợp với không gian nuôi khép kín, ***(ii)*** chi phí nuôi thấp hơn. Đối với nhiều bạn trẻ, mèo đã trở thành một người bạn thân thiết không thể thiếu. Dù vậy, việc quản lý, tìm lại mèo trong trường hợp chúng bỏ trốn, đi lạc hoặc bị bắt trộm ở Việt Nam là rất khó khăn. Ở một số nước phát triển, thú cưng thường được gắn vi mạch để tracking và nhận dạng. Tuy nhiên, ở Việt Nam hiện nay, phương pháp này hầu như chưa được sử dụng. Bên cạnh đó, trong tương lai, dù công nghệ này trở nên phổ biến hơn, chi phí cho việc gắn chip cũng không hề rẻ, đặc biệt với sinh viên và các bạn trẻ vừa ra trường, chưa có nền tảng kinh tế vững vàng.

Do đó, tôi xin đề xuất công nghệ nhận diện gương mặt mèo dựa trên ảnh kỹ thuật số để giúp chủ nhân tìm lại mèo. Tính năng nhận diện gương mặt này được phát triển như phần lõi của ứng dụng tìm mèo đi lạc trong tương lai.

## Mục tiêu và phạm vi đề tài

Hệ thống nhận dạng khuôn mặt hoạt động bằng cách xác định và phân tích các đặc điểm khuôn mặt độc đáo của một cá nhân trong hình ảnh hoặc video kỹ thuật số. Một hệ thống nhận dạng khuôn mặt dựa phương pháp học sâu thường bao gồm 4 module:

1. Phát hiện khuôn mặt—thuật toán nhận diện khuôn mặt được sử dụng để nhận dạng và định vị khuôn mặt trong hình ảnh hoặc video. Điều này cho phép hệ thống xác định vùng hình ảnh chứa khuôn mặt và loại bỏ thông tin không liên quan khác.
2. Căn chỉnh khuôn mặt— từ kết quả phát hiện, hệ thống tiến hành các bước xử lý như crop để đạt được kết quả là hình ảnh chỉ chứa gương mặt cần nhận diện.
3. Nhận diện khuôn mặt: hệ thống trích xuất các đặc điểm khuôn mặt có liên quan từ khuôn mặt được phát hiện. Các đặc điểm được trích xuất sau đó được chuyển đổi thành biểu diễn số, được gọi là embedding vector (hay vector đặc trưng), có thể so sánh với các vector đặc trưng khuôn mặt khác trong cơ sở dữ liệu. Trong các hệ thống nhận diện dựa học sâu, mạng lưới thần kinh sâu được sử dụng để mã hóa ảnh kỹ thuật số đầu vào chưa khuôn mặt cần nhận diện thành vector đặc trưng.
4. So sánh— vector đặc trưng được so sánh với cơ sở dữ liệu về các khuôn mặt đã biết để xác định danh tính của cá nhân trong hình ảnh hoặc video. Thuật toán đưa ra điểm tin cậy, cho biết khuôn mặt đầu vào khớp với khuôn mặt đã biết trong cơ sở dữ liệu đến mức nào. Nếu ở cuối quá trình này, điểm tin cậy vượt quá một ngưỡng nhất định, hệ thống sẽ nhận dạng cá nhân đó và xác nhận danh tính của họ.

Trong dự án này, tôi sẽ chủ yếu tập trung vào module (iii) và module (iv) của hệ thống. Bên cạnh việc xây dựng mạng thần kinh học sâu, xây dựng dữ liệu huấn luyện phù hợp cũng là một khâu quan trọng quyết định hiệu suốt của mô hình. Hiện nay, chưa có bộ dữ liệu sẵn có nào phục vụ tốt mục đích của bài toán này, do đó, tìm kiếm, xây dựng dữ liệu huấn luyện cũng là khâu quan trọng.

## Định hướng giải pháp

Do giới hạn về thời gian và khả năng tính toán (computing capacity), để đảm bảo performance của model ở mức chấp nhận được, dự án sẽ sử dụng một số kỹ thuật sau:

1. Sử dụng transfer learning với mông hình gốc là Resnet
2. Sử dụng YOLOv5 để thực hiện phát hiện khuôn mặt và căn chỉnh khuôn mặt (đây được coi là bước chuẩn bị dữ liệu)
3. Sử dụng làm hàm mất mát bộ ba

**Công trình này có một số đóng góp chính sau:**

1. Cung cấp một bộ dữ liệu mới gồm hơn 320 hình ảnh riêng lẻ của hơn 32 con mèo khác nhau có nguồn gốc từ ảnh công khai trên mạng xã hội Instagram.
2. Báo cáo kết quả về việc đào tạo YOLOv5 để phát hiện và trích xuất khuôn mặt mèo từ hình ảnh nguồn, bỏ qua việc lựa chọn đặc điểm thủ công và căn chỉnh hình ảnh, khác với các phương pháp trước đó là xử lý trước hình ảnh để căn chỉnh các đặc điểm khuôn mặt và tư thế.
3. Xây dựng 2 hàm mới để đánh giá độ chính xác của mô hình.
4. Cuối cùng báo cáo về việc đào tạo Resnet50 cho các nhiệm vụ xác minh và nhận dạng, sử dụng hàm mất mát là hàm mất mát bộ ba được khai thác theo kiểu trực tuyến chính quy.

## Bố cục báo cáo

Phần còn lại của báo cáo này được tổ chức như sau.

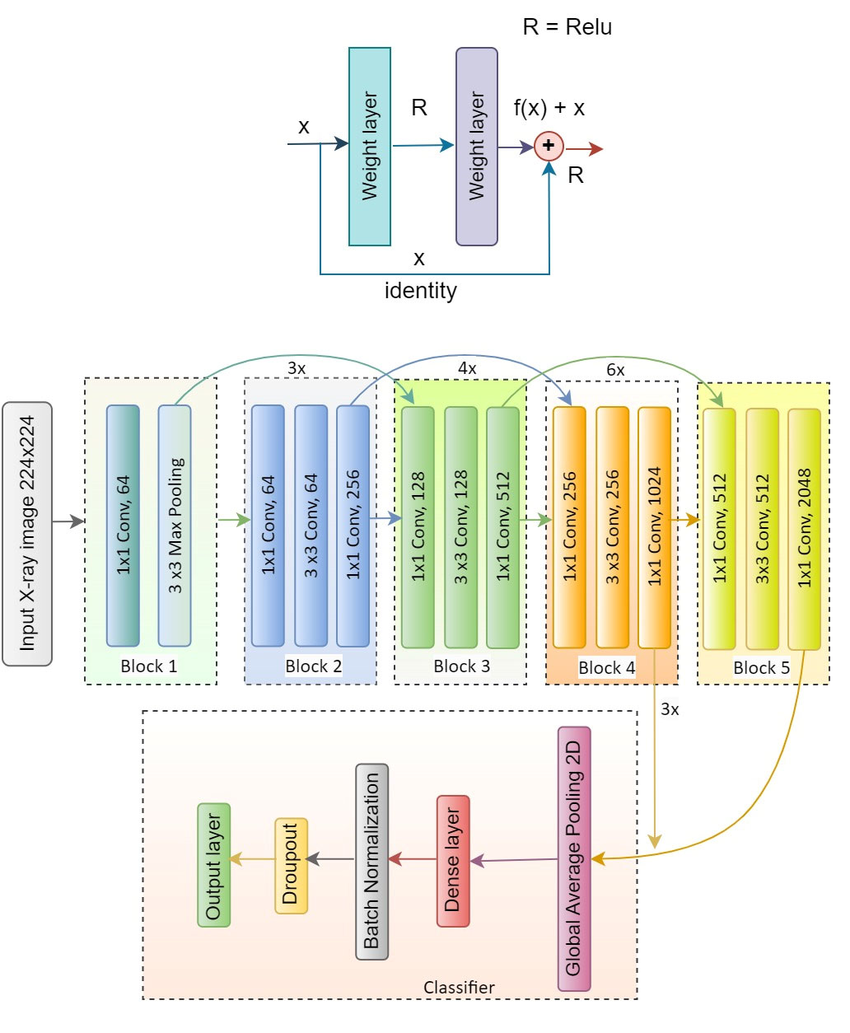
Chương 2 trình bày về kiến trúc của mạng thần kinh sâu sử dụng trong nhận dạng gương mặt và cách hoạt động của hàm mất mát bộ ba

Trong Chương 3, tôi xin giới thiệu về cách thức chuẩn bị dữ liệu cho huấn luyện và kết quả thực tế mô hình đạt được.

# Chương 2: KIẾN TRÚC MẠNG THẦN KINH SÂU VÀ

# XÂY DỰNG HÀM MẤT MÁT BỘ BA

## 2.1. Kiến trúc mạng thần kinh sâu



**Hình ảnh 1. Minh họa kiến trúc Resnet50[[1]](#footnote-0)**

Mô hình sử dụng kỹ thuật transfer learning với mô hình gốc là Resnet50[[2]](#footnote-1) đã được huấn luyện trước trên bộ dữ liệu Imagenet. Resnet đã được chứng minh là một kiến trúc mạng thần kinh sâu mạnh mẽ, hiệu quả; được ứng dụng nhiều trong các bài toán thị giác máy tính như phát hiện vật thể, phân đoạn đối tượng.

Kiến trúc ResNetv50 có 50 lớp tích chập, được chia thành 5 khối. Mỗi khối gồm 3 lớp tích chập 3x3 với cùng số kênh đầu ra. Mỗi tầng tích chập được theo sau bởi một tầng chuẩn hóa theo batch (batch normalization) và một hàm kích hoạt ReLU.

Điểm đặc biệt của kiến trúc ResNetv50 là sử dụng các kết nối tắt (skip connections). Các kết nối tắt này nối trực tiếp đầu vào của một tầng với đầu ra của tầng đó. Điều này giúp cho thông tin từ đầu vào có thể truyền trực tiếp đến đầu ra, giúp cho mạng có thể học tập hiệu quả hơn với các mạng có độ sâu lớn, giảm thiểu đáng kể nguy cơ xảy ra hiện tượng vanishing gradients.

Ở mô hình Resnet50, ta sẽ lấy đến lớp Global Average Pooling 2D và bỏ đi các lớp phía sau. Tiếp đó, ta sẽ kết nối thêm 3 lớp Kết Nối Đầy Đủ (Fully Connected layer) với số lượng hidden unit lần lượt là 512, 128 và 32. Ta sẽ sử dụng hàm kích hoạt ReLU sau mỗi Fully Connected layer (không áp dụng cho lớp cuối), và tiếp đó là lớp Loại Bỏ (Dropout layer) với xác xuất loại bỏ là . Đầu ra của mô hình này trước khi được đưa vào hàm mất mát sẽ được chuẩn hóa theo norm L2 để đảm bảo độ ổn định khi huấn luyện và giúp mạng thần kinh hội tụ nhanh hơn.

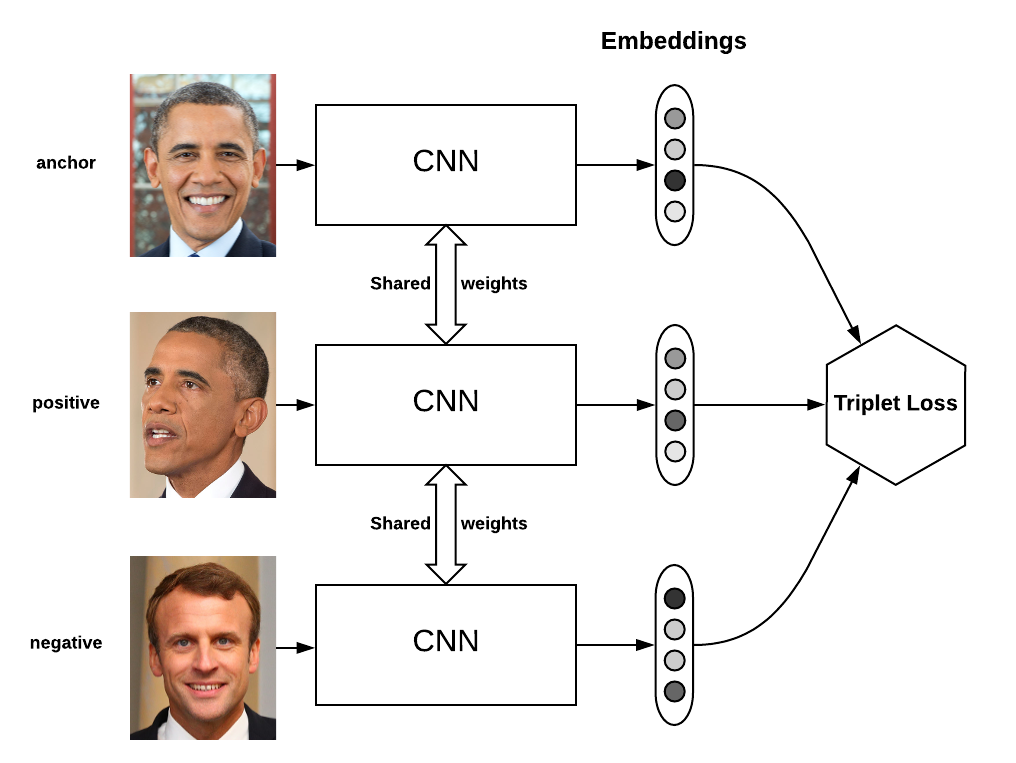
## 2.2. Hàm mất mát bộ ba và kỹ thuật khai thác bộ ba trực tuyến

### **2.2.1. Hàm mất mát bộ ba có chính quy hóa**

*Facenet: A Unified Embedding for Face Recognition and Cluster[[3]](#footnote-2)* là công trình đầu tiên sử dụng triplet loss như một hàm mất mát được để giải quyết bài toán nhận dạng khuôn mặt. Công trình này đã mô tả một cách tiếp cận mới để huấn luyện vector đặc trưng bằng cách sử dụng hàm mất mát triplet loss khai thác theo kiểu trực tuyến.

Thông thường trong học có giám sát, lớp cuối cùng của mạng thần kinh sâu sẽ tạo ra một số lớp cố định và ta sẽ huấn luyện mạng bằng cách sử dụng các hàm mất mát, chẳng hạn như ***mất mát chéo entropy softmax***. Tuy nhiên, trong một số trường hợp, ta không thể biết trước số lớp cần thiết, trong đó nhận dạng khuôn mặt là một ví dụ điển hình. Bài toán nhận dạng khuôn mặt yêu cầu ta không biết trước số gương mặt cần nhận dạng mà vẫn phải đảm bảo mô hình cần có khả năng so sánh hai khuôn mặt chưa từng thấy trước đây và cho biết liệu chúng có phải của cùng một người hay không.

Hàm mất mát bộ ba (triplet loss) là một hàm mất mát được phát triển để giải quyết vấn đề này: triplet loss phục vụ tốt mục tiêu là khuyến khích mạng thần kinh sâu tạo ra các các vector đặc trưng sao cho vector đặc trưng của các trường hợp tương tự (ví dụ: khuôn mặt của cùng một người) sẽ ở gần nhau trong không gian đặc trưng và các trường hợp khác nhau (ví dụ: khuôn mặt của hai người khác nhau) sẽ ở xa nhau.



**Hình ảnh 2: Minh họa cách hoạt động của triplet loss[[4]](#footnote-3)**

Trong triplet loss, quá trình huấn luyện bao gồm bộ ba mẫu: mẫu neo - anchor, mẫu dương -positive (mẫu từ cùng một lớp với mẫu neo) và mẫu âm - negative (mẫu từ một lớp khác). Mục đích của triplet loss nhằm đảm bảo sau quá trình huấn luyện: ***(i)*** Hai mẫu từ cùng một lớp (hay có cùng nhãn) có vector đặc trưng gần nhau trong không gian đặc trưng; ***(ii)*** Hai mẫu từ hai lớp khác nhau có vector đặc trưng xa nhau trong không gian đặc trưng.

Để đo khoảng cách giữa 2 vector, ta có thể sử dụng nhiều metrics khác nhau như khoảng cách euclidean , cosine similarity. Trong bài toán nhận diện khuôn mặt, khoảng cách euclidean được lựa chọn phổ biến và cũng sẽ được sử dụng trong công trình này (khoảng cách euclidean giữa 2 vector a,b được ký hiệu là). Như vậy, hàm mất mát của một bộ ba  được đề xuất trong FaceNet là:



***Trong đó:***

là mẫu neo

 là mẫu dương

  là mẫu âm

 là vector đặc trưng của mẫu 

 là độ chênh lệch tối thiểu quy ước giữa  và

 - một siêu tham số

Tối thiểu hóa L sẽ đẩy  lớn hơn . Khi đạt được điều này, độ mất mát là 0.

Zhang và cộng sự[[5]](#footnote-4) đã giới thiệu một phương pháp chính quy hóa có thể sử dụng với triplet loss là “global orthogonal regularization”. Phương pháp chính quy hóa này trừng phạt các cặp vector đại diện không thuộc cùng một khuôn mặt khi chúng “xa khỏi tiêu chuẩn vuông góc” - hay có thể lý giải một cách dễ hiểu hơn là khi chúng tương tự nhau trong không gian đặc trưng. Trong *Pet Cat Face Verification and Identification[[6]](#footnote-5),* nhóm tác giả đã ghi nhận việc trong quá trình huấn luyện với triplet loss, các vector đặc trưng nhanh chóng trở nên như nhau, đẩy giá trị mất mát của từng bộ ba đúng bằng . Giá trị chính quy hóa này có công thức như sau:







Vậy hàm mất mát được dùng trong công trình là:



### **2.2.2. Khai thác bộ ba**

Dựa trên định nghĩa về hàm mất mát trên, ta có thể phân loại các bộ ba thành 3 loại bộ ba:

1. Bộ ba dễ: bộ ba có độ mất mát là 0, hay ))
2. Bộ ba khó: bộ ba có embedding vector của mẫu âm gần với mẫu neo hơn là mẫu dương, hay ))
3. Bộ ba nửa khó: bộ ba mà embedding vector của mẫu âm không gần mẫu neo hơn mẫu dương, tuy nhiên vẫn có độ mất mát dương, hay



Việc lựa chọn các bộ ba cho huấn luyện ảnh hưởng đáng kể đến hiệu suất mô hình. Trong công trình gốc FaceNet, nhóm tác giả đã chọn các bộ ba nửa khó ngẫu nhiên và huấn luyện trên các bộ ba này. Tuy nhiên, vấn đề đặt ra ở đây là làm sao có thể khai thác được các bộ ba nửa khó này từ dữ liệu huấn luyện.

#### *2.2.2.1. Khai thác bộ ba ngoại tuyến*

Phương pháp đầu tiên và có lẽ là cách dễ đến đầu tiên là Khai thác bộ ba một cách ngoại tuyến (hay Offline Triplet Mining). Ý tưởng chính của phương pháp này là khi bắt đầu mỗi chu kỳ huấn luyện, ta sẽ tính toán tất cả các vector đặc trưng của dữ liệu huấn luyện và chỉ chọn các bộ ba nửa khó để huấn luyện.

Cụ thể, sau khi chọn được các bộ ba nửa khó, ta sẽ đóng gói các bộ ba này thành một mẻ huấn luyện. Như vậy, mỗi mẻ huấn luyện có độ dài B sẽ bao gồm 3B vector đặc trưng. Sau đó, ta sẽ tính toán mất mát bộ ba trên mẻ dữ liệu và thực hiện lan truyền ngược lại vào mạng.

Một cách cơ bản, tính hiệu quả của kỹ thuật này không được đánh giá cao do: ***(i)*** kỹ thuật này đòi hỏi ta phải tính toán tất cả vector đặc trưng trước khi huấn luyện, ***(ii)*** qua mỗi chu kỳ huấn luyện, ta phải tính toán lại các bộ ba nửa khó và thực hiện đóng gói lại vào các mẻ huấn luyện.

#### *2.2.2.2. Khai thác bộ ba trực tuyến*

Do các lý do trên, dù phương pháp Offline triplet mining khá dễ hiểu và dễ thực hiện, trên thực tế, Khai thác bộ ba trực tuyến (hay Online Triplet Mining) mới là kỹ thuật thường được sử dụng. Trong báo cáo của FaceNet cũng ghi nhận tính hiệu quả của kỹ thuật này trong huấn luyện mạng thần kinh sâu nhận dạng khuôn mặt.

Ý tưởng chính của Online triplet mining là tính toán các bộ ba nửa khó dựa trực tiếp ngay trong quá trình huấn luyện. Với một mẻ huấn luyện có độ dài là B gồm B ảnh đầu vào được lựa chọn ngẫu nhiên, ta sẽ tính toán B vector đặc trưng tương ứng và từ B vector đặc trưng này, sẽ có tối đa B3  bộ ba. Tuy nhiên, dễ dàng thấy được phần lớn các bộ ba này đều không hợp lệ (một bộ ba được coi là hợp lệ khi nó thỏa mãn luật có 2 vector của cùng một khuôn mặt và vector còn lại của khuôn mặt khác). Dù vậy, số lượng các bộ ba nửa khó hợp lệ trong mỗi mẻ huấn luyện tạo ra bởi kỹ thuật này cũng nhiều hơn so với Offline Triplet Mining.

Trong công trình *In defense of the Triplet Loss for Person Re-identification[[7]](#footnote-6)****,*** nhóm tác giả đã chứng minh được rằng việc huấn luyện trên các bộ ba khó nhất trong mẻ huấn luyện sẽ khiến mô hình đạt được kết quả tốt nhất. Tuy nhiên, do lượng dữ liệu của công trình này còn tương đối hạn chế nên để tăng thêm số lượng các bộ ba hợp lệ cho mỗi mẻ huấn luyện, ta sẽ chọn các bộ ba nửa khó và khó để huấn luyện. Như vậy, có thể khái quát cách thức sử dụng kỹ thuật Online triplet mining của công trình này như sau:

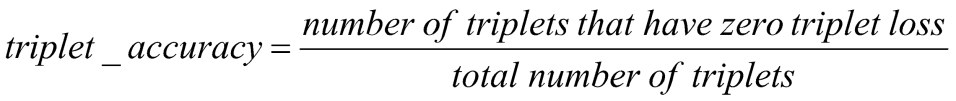
Sau khi lựa chọn tất cả các bộ ba hợp lệ, ta tiến hành:

1. Lựa chọn các bộ ba nửa khó và khó và tính toán giá trị mất mát trung bình trên các bộ ba này. Một điểm cần lưu ý ta cần bỏ qua các bộ ba dễ, nếu không sẽ làm cho giá trị mất mát trung bình trở nên rất nhỏ, dẫn đến mạng thần kinh lâu hội tụ hơn.
2. Thực hiện lan truyền giá trị mất mát trung bình này ngược trở lại mạng.

## 2.3. Xây dựng hàm đánh giá độ chính xác của mô hình

### ***2*.3.1. Hàm chính xác bộ ba**

Đây là hàm đánh giá độ chính xác của mô hình dựa trên trực tiếp kết quả của hàm mất mát bộ ba. Độ chính xác đo bằng hàm này là:



### **2.3.2. Hàm chính xác nhận diện**

Dù hàm chính xác bộ ba có thể đo được mức độ thành công của hàm mất mát bộ ba ứng dụng thực tế của mô hình nhận diện khuôn mặt thường được thiết lập theo cách khác: Đối với 2 hình ảnh đầu vào, mô hình phải quyết định liệu hai hình ảnh đó có đủ tương đồng để thuộc về cùng một khuôn mặt hay không hoặc đủ khác biệt để thuộc về 2 khuôn mặt khác nhau . Để xem xét hiệu suất của mô hình trong nhiệm vụ này, tôi đã xây dựng một hàm để đo độ chính xác này được gọi là Hàm chính xác nhận diện.

Với đầu vào là 2 hình ảnh có thể cùng của 1 gương mặt (cặp dương tính) hoặc của 2 gương mặt khác nhau (cặp âm tính) và nhãn của cặp hình ảnh này (dương tính hay âm tính), hàm chính xác nhận diện sẽ đánh giá khả năng của mô hình trong nhiệm vụ xác xác định 2 bức ảnh đầu vào

- Đủ độ tương đồng để kết luận là của cùng một khuôn mặt:



- Đủ độ khác biện để thuộc về 2 khuôn mặt khác nhau:



# Chương 3: CHUẨN BỊ DỰ LIỆU VÀ THỰC HIỆN HUẤN LUYỆN

## **3.1. Chuẩn bị dữ liệu**

Bộ dữ liệu dùng để huấn luyện cho mô hình cần thỏa mãn: ***(i)*** Mỗi khuôn mặt mèo cần có ít nhất 5 hình ảnh; ***(ii)*** Các hình ảnh phải rõ khuôn mặt mèo. Hiện nay, theo khảo sát, chưa có được API nguồn mở nào cho phép số điện thoại Việt Nam đăng ký đáp ứng được các yêu cầu này.

Theo thống kê 6/2023, Instagram là mạng xã hội có nhiều người dùng thứ 4 thế giới với ước tính 2.35 tỉ người dùng hoạt động. Instagram có định hướng thiên về chia sẻ hình ảnh với nhiều chủ đề đa dạng như thời trang, phong cảnh, thú cưng. Hiện nay, có nhiều tài khoản trên Instagram được lập để chủ tài khoản đăng tải các hình ảnh về thú cưng của mình. Số lượng ảnh trên một tài khoản thường tương đối lớn với chất lượng ảnh tốt. Do đó, đây là nguồn dữ liệu được sử dụng làm dữ liệu huấn luyện cho mô hình.

Tuy nhiên, đầu vào trực tiếp của mô hình nhận diện là các khuôn mặt mèo, không phải hình ảnh của một con mèo. Do đó, cần một công cụ có thể tự động phát hiện khuôn mặt mèo trong ảnh. YOLO là mạng thần kinh sâu ưu việt hiện nay được sử dụng trong bài toán phát hiện vật thể. Với một lượng dữ liệu huấn luyện nhỏ, YOLOv5 cho phép người dùng đạt được kết quả đáng kinh ngạc khi finetune mô hình để phát hiện những vật thể khác với các lớp phân loại ban đầu.

Tôi đã thành công khi sử dụng phương pháp transfer learning để huấn luyện YOLOv5[[8]](#footnote-7) cho nhiệm vụ phát hiện khuôn mặt mèo. Bộ dữ liệu huấn luyện gồm 120 hình ảnh mèo bất kỳ và sử dụng Label Studio để vẽ các hộp giới hạn xung quanh khuôn mặt mèo.



**Hình ảnh 3. Kết quả của YOLOv5 trong phát hiện khuôn mặt mèo**

Dữ liệu được chia thành 2 phần: 100 hình ảnh để đào tạo và 20 ảnh còn lại để xác thực. YOLOv5 được khởi tạo ban đầu với trọng số Darknet chính thức đã được đào tạo trước trên COCO. Trong 100 chu kỳ đầu, chỉ huấn luyện ba lớp cuối cùng cho một nhiệm vụ phát hiện một lớp duy nhất bằng cách sử dụng trình tối ưu hóa Adam với tốc độ học tập 1e-3, sau đó hủy đóng băng tất cả các lớp và huấn luyện cho 50 chu kỳ tiếp theo với tốc độ học tập 1e-4, sử dụng mức giảm tốc độ học tập hệ số 0,1. Mô hình được đào tạo chứng minh 100% mAP trên bộ xác thực với ngưỡng tin cậy 49%. Như có thể thấy trong Hình 2, mô hình YOLO này rất thành thạo về mặt thực nghiệm trong việc phát hiện khuôn mặt mèo, tuy vẫn còn tồn tại một số kết quả dương tính giả khó hiểu.

## 3.2. Huấn luyện mô hình nhận diện

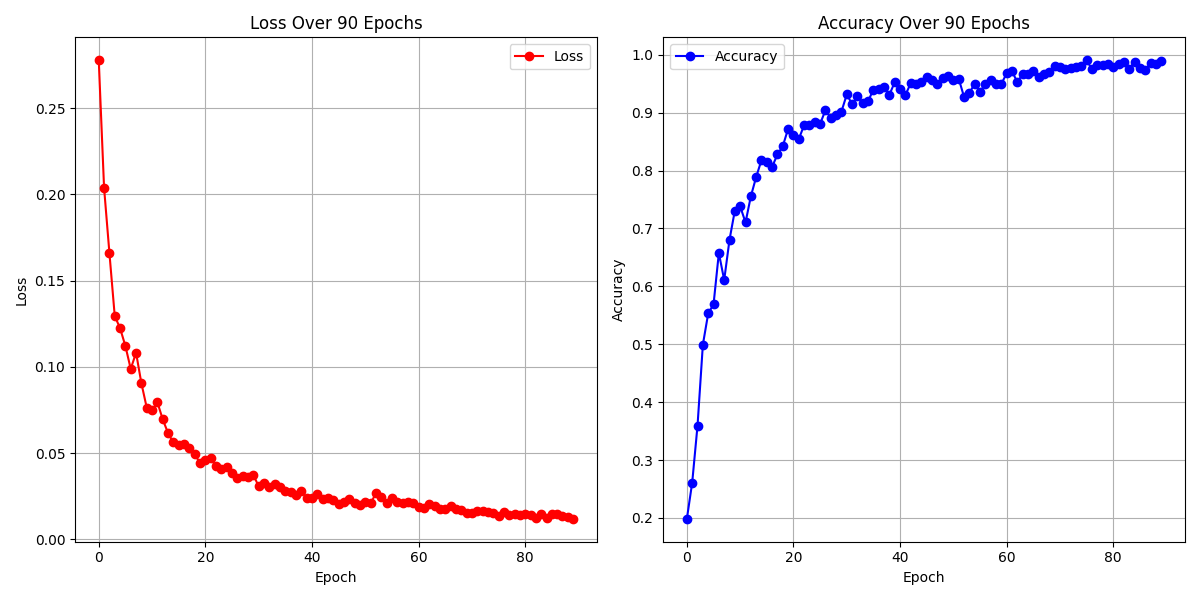
Từ dữ liệu huấn luyện, ta chia chúng thành các mẻ huấn luyện có kích thước mỗi mẻ là 64. Các hình ảnh đầu vào trước khi đưa vào mạng thần kinh sẽ được tiền xử lý: ***(i)*** resize tất cả hình ảnh về kích thước 224x224x3 (đây là kích thước được đề xuất cho Resnet50); ***(ii)*** thực hiện tăng cường dữ liệu (data augmentation) để làm giàu dữ liệu đầu vào, các kỹ thuật cụ thể được dùng là tăng/giảm độ sáng ảnh, xoay ảnh (rotate), lật ảnh.

Theo thiết kế của hàm mất mát sử dụng, các siêu tham số chính của việc huấn luyện được xác định gồm có:

- 

- Trọng số chính quy 

Dựa trên kích thước của vector đặc trưng là 32, được ước tính là 0.2 là con số phù hợp. Đối với trọng số , trên thực tế quá trình huấn luyện đã chọn 3 giá trị của lần lượt là 0, 0.3, 0.5. Điều đáng lưu ý là hiện tượng các vector đặc trưng thu lại thành các vector tương tự nhau không xảy ra, rất có thể do việc lựa chọn dữ liệu huấn luyện của công trình này. Do đó, kết quả thực tế của mô hình huấn luyện với 3 giá trị  không có sự khác nhau đáng kể. Tuy nhiên, 2 mô hình huấn luyện với giá trị  là 0.3, 0.5 vẫn tốt hơn một chút và đạt mức chính xác là 99% trong khi mô hình có giá trị  là 0 đạt được 96% sau 90 chu kỳ huấn luyện (hàm chính xác mất mát bộ ba)



**Hình ảnh 4. Biểu đồ giá trị mất mát và tỉ lệ chính xác của mô hìn**h

Trong 50 chu kỳ huấn luyện đầu

Đối với hàm chính xác nhận diện, mô hình huấn luyện dựa với alpha là 0.3 có kết quả nhỉnh hơn trên các cặp dương tính, đạt tỉ lệ chính xác đạt tới 238/262 với . Một cách dễ hiểu, mạng thần kinh huấn luyện với cũng đạt tỉ lệ chính xác cao hơn trên các cặp âm tính (chính xác 4166/4352 cặp) với .

## 3.3. Khả năng nhận diện của mô hình sau huấn luyện

Để kiểm tra khả năng của mô hình trong bài toán nhận diện, ta sẽ sắp đặt bài toán như sau:

1. Tính toán vector đại diện cho 32 gương mặt mèo trong CSDL. Như đã đề cập, mỗi khuôn mặt có 10 ảnh, vì vậy ta sẽ tính được 10 vector đại diện cho mỗi khuôn mặt. Từ 10 vector này, chọn ra 3 vector có tính đại diện cao nhất cho gương mặt này với tiêu chuẩn 3 vector này có khoảng cách euclidean với các vector còn lại trong cùng một lớp nhỏ nhất (3 vector này được gọi là bộ vector đại diện của một khuôn mặt mèo). Lưu mỗi bộ vector đại diện trong trong một file CSV. Folder lưu tất cả các file CSV này được gọi là CSDL vector đặc trưng.
2. Chọn giá trị threshold, nếu độ chênh lệch giữa 2 vector đặc trưng vượt quá threshold này, 2 bức ảnh tương ứng sẽ được coi là thuộc về 2 khuôn mặt khác nhau; ngược lại, chúng được coi là thuộc về cùng một khuôn mặt
3. Đầu vào của bài toán là ảnh một khuôn mặt mèo bất kỳ. Mô hình sẽ tính toán vector đại diện của ảnh đầu vào và so sánh với các vector đại diện trong CSDL vector đặc trưng. Nếu tồn tại một bộ vector đại diện có ít nhất 2/3 vector thỏa mãn: khoảng cách giữa chúng và vector đại diện của ảnh đầu vào nhỏ hơn giá trị threshold, khuôn mặt trong ảnh đầu vào được xác định là đã tồn tại trong CSDL; ngược lại, khuôn mặt đầu vào chưa được lưu trữ.
4. Đối với trường hợp tồn tại, công trình cho phép người dùng xem một hình ảnh bất kỳ của khuôn mặt đầu vào đã được lưu trữ trong CSDL. Với trường hợp không tồn tại, công trình xác định khuôn mặt có bộ vector đại diện có khoảng cách euclidean với ảnh đầu vào nhỏ nhất và cho phép người dùng xem bức ảnh tương tự nhất này.



**Hình ảnh 5a. Kết quả của mạng thần kinh sâu trong nhận diện khuôn mặt mèo**



**Hình ảnh 5b. Kết quả của mạng thần kinh sâu trong nhận diện khuôn mặt mèo**

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Sau 13 tuần tìm hiểu, triển khai bài toán nhận diện khuôn mặt mèo với hàm mất mát triplet loss, công trình đã cơ bản đạt được những kết quả đặt ra. Tuy nhiên, do hạn chế về thời gian, công trình còn tồn tại nhiều yếu điểm có thể cải thiện trong tương lai như: (i) Bộ dữ liệu còn chưa đa dạng, chưa nghiên cứu nhiều tư thế gương mặt khác nhau như góc nghiêng, bị che khuất một phần, …; (ii) Ảnh đầu vào mô hình còn cần qua tiền xử lý trước bằng YOLO, trong tương lai có thể nghiên cứu tích hợp một mô hình phát hiện gương mặt mèo; (iii) Kích thước của mô hình nhận diện còn tương đối lớn, có thể sẽ thu gọn được trong tương lai khi có bộ dữ liệu huấn luyện lớn hơn.

Cuối cùng, tôi xin cảm ơn Thạc sĩ Nguyễn Mạnh Tuấn đã hướng dẫn tôi hoàn thành nội dung của công trình này!

TÀI LIỆU THAM KHẢO

[1] Chakraborty, Gouri & Batra, Salil & Singh, Aman & Muhammad, Ghulam & Yelamos, Vanessa & Mahajan, Makul. (2023). A Novel Deep Learning-Based Classification Framework for COVID-19 Assisted with Weighted Average Ensemble Modeling. Diagnostics. 13. 1806. 10.3390/diagnostics13101806.

[2] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90

[3] Schroff, F., Kalenichenko, D., & Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 815–823).

[4] Triplet Loss and Online Triplet Mining in TensorFlow. https://omoindrot.github.io/triplet-loss. [Online; accessed 4-December-2019].

[5] Xu Zhang, Felix X. Yu, Sanjiv Kumar, and Shih-Fu Chang. Learning spread-out local feature descriptors. In The IEEE International Conference on Computer Vision (ICCV), Oct 2017.

[6] Adam Klein. Pet Cat Face Verification and Identification

[7] Hermans, Alexander & Beyer, Lucas & Leibe, Bastian. (2017). In Defense of the Triplet Loss for Person Re-Identification.

[8] https://docs.ultralytics.com/yolov5/

1. Chakraborty, Gouri & Batra, Salil & Singh, Aman & Muhammad, Ghulam & Yelamos, Vanessa & Mahajan, Makul. (2023). A Novel Deep Learning-Based Classification Framework for COVID-19 Assisted with Weighted Average Ensemble Modeling. Diagnostics. 13. 1806. 10.3390/diagnostics13101806. [↑](#footnote-ref-0)
2. K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90 [↑](#footnote-ref-1)
3. Schroff, F., Kalenichenko, D., & Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 815–823). [↑](#footnote-ref-2)
4. Triplet Loss and Online Triplet Mining in TensorFlow. https://omoindrot.github.io/triplet-loss. [Online; accessed 4-December-2019]. [↑](#footnote-ref-3)
5. Xu Zhang, Felix X. Yu, Sanjiv Kumar, and Shih-Fu Chang. Learning spread-out local feature descriptors. In The IEEE International Conference on Computer Vision (ICCV), Oct 2017. [↑](#footnote-ref-4)
6. Adam Klein. Pet Cat Face Verification and Identification [↑](#footnote-ref-5)
7. Hermans, Alexander & Beyer, Lucas & Leibe, Bastian. (2017). In Defense of the Triplet Loss for Person Re-Identification. [↑](#footnote-ref-6)
8. https://docs.ultralytics.com/yolov5/ [↑](#footnote-ref-7)