Khi trích xuất đặc trưng từ các Backbone khác nhau, có một số khía cạnh quan trọng cần hiểu rõ hơn để so sánh và đánh giá hiệu quả của từng backbone. Dưới đây là các yếu tố chi tiết liên quan đến quá trình trích xuất đặc trưng từ các backbone như VGG-Face, Facenet, OpenFace, DeepFace, ArcFace:

**1. Cấu trúc của Backbone và Kiến trúc Mạng**

Mỗi backbone có một kiến trúc mạng khác nhau, và sự khác biệt này quyết định cách mà chúng trích xuất các đặc trưng từ ảnh khuôn mặt.

* + **VGG-Face**:
    - Sử dụng kiến trúc **VGG16** hoặc **VGG19** với các tầng tích chập (convolutional layers) sâu.
    - VGG trích xuất đặc trưng bằng cách sử dụng nhiều tầng tích chập, tiếp theo là các tầng kết nối đầy đủ (fully connected layers). Mô hình học các đặc trưng cấp thấp đến cao từ các chi tiết cơ bản như cạnh, góc đến các đặc trưng phức tạp hơn.
    - **Embedding size**: 4096 chiều.
    - **Nhận định**: VGG-Face học đặc trưng khá mạnh mẽ nhưng số lượng tham số lớn và khả năng xử lý không gian ảnh lớn, thích hợp với các bài toán yêu cầu độ chi tiết cao nhưng có thể chậm hơn so với các backbone khác.
  + **Facenet**:
    - Sử dụng kiến trúc **Inception-ResNet**.
    - Facenet được tối ưu với **triplet loss**, giúp mô hình học cách tối thiểu hóa khoảng cách giữa các đặc trưng của cùng một người và tối đa hóa khoảng cách với các đặc trưng của người khác.
    - **Embedding size**: 128 chiều.
    - **Nhận định**: Facenet tập trung vào việc tạo ra các vector có khả năng phân biệt cao. Nhờ vào triplet loss, các đặc trưng học được giúp giảm nhầm lẫn giữa các khuôn mặt có sự khác biệt nhỏ, phù hợp cho các bài toán yêu cầu độ chính xác cao.
  + **OpenFace**:
    - Dựa trên kiến trúc **nn4** với khả năng huấn luyện nhanh và nhẹ hơn so với VGG-Face và Facenet.
    - Sử dụng mạng neural network đơn giản hơn và được tối ưu hóa cho các ứng dụng nhẹ.
    - **Embedding size**: 128 chiều.
    - **Nhận định**: OpenFace có thể hoạt động nhanh và hiệu quả với tài nguyên tính toán thấp hơn, nhưng độ chính xác không cao bằng các mô hình như Facenet hay ArcFace.
  + **ArcFace**:
    - Sử dụng kiến trúc **ResNet** và tích hợp **ArcMargin loss** thay vì softmax thông thường.
    - ArcFace tạo ra các embeddings có khoảng cách giữa các lớp khác biệt rõ ràng, giúp phân biệt tốt hơn giữa các khuôn mặt khác nhau.
    - **Embedding size**: 512 chiều.
    - **Nhận định**: ArcFace là một trong những backbone tiên tiến nhất, thường cho kết quả tốt hơn trong việc phân biệt các khuôn mặt khó, nhờ vào khoảng cách lớn hơn giữa các đặc trưng.
  + **DeepFace**:
    - Được phát triển bởi Facebook và sử dụng **Deep Neural Network (DNN)** để học đặc trưng khuôn mặt.
    - DeepFace cũng có kiến trúc tương tự VGG-Face nhưng được cải tiến để tăng độ chính xác.
    - **Embedding size**: 4096 chiều.
    - **Nhận định**: DeepFace mạnh mẽ trong việc học các đặc trưng khuôn mặt nhưng cũng có độ phức tạp cao và cần tài nguyên lớn.

**2. Quá trình Trích Xuất Đặc Trưng**

Khi bạn sử dụng các backbone trên để trích xuất đặc trưng (embedding), các bước chính thường bao gồm:

* + **Đầu vào**: Ảnh khuôn mặt đầu vào được đưa qua các tầng tích chập (conv layers), nơi mà các đặc trưng được học dần từ thấp đến cao.
  + **Flatten**: Sau các tầng tích chập, ảnh đầu vào trở thành một tensor với các đặc trưng 2D, nhưng để đưa vào các tầng fully connected, tensor này cần được chuyển thành vector 1D (flatten).
  + **Tầng Fully Connected**: Các tầng fully connected giúp học các đặc trưng tổng hợp cuối cùng và tạo ra **embedding** (vector đặc trưng) đại diện cho khuôn mặt.

Ví dụ:

* + **VGG-Face**: Ảnh được trích xuất thành một vector có kích thước 4096 chiều.
  + **Facenet**: Tạo ra embedding với kích thước nhỏ hơn (128 chiều) nhưng có khả năng phân biệt cao hơn.

**3. So sánh về Kích thước Embedding**

* + **Kích thước embedding**: Kích thước vector đặc trưng (embedding) ảnh hưởng đến cả hiệu suất và khả năng phân biệt.
    - **VGG-Face** và **DeepFace** có embedding lớn (4096 chiều), nghĩa là mô hình học được rất nhiều thông tin từ ảnh, nhưng cũng tốn nhiều tài nguyên hơn.
    - **Facenet** và **ArcFace** có embedding nhỏ hơn (128-512 chiều), giúp mô hình học nhanh hơn và sử dụng ít tài nguyên hơn, nhưng vẫn đảm bảo khả năng phân biệt cao.

=> **Nhận định**: Đối với những ứng dụng yêu cầu tốc độ và tài nguyên tính toán thấp, các mô hình có embedding nhỏ (Facenet, ArcFace) có thể là lựa chọn tốt hơn. Ngược lại, nếu yêu cầu độ chi tiết và phân tích kỹ hơn, VGG-Face hoặc DeepFace có thể phù hợp.

**4. So sánh Khoảng Cách Cosine và Euclidean**

Sau khi trích xuất được embedding, các độ đo như **cosine similarity** hoặc **Euclidean distance** được sử dụng để so sánh các đặc trưng. Mỗi độ đo có một cách đánh giá khác nhau về sự tương đồng giữa hai ảnh:

* + **Cosine Similarity**: Đo góc giữa hai vector đặc trưng trong không gian, giá trị gần 1 có nghĩa là hai vector có hướng giống nhau và do đó hai ảnh có thể là của cùng một người.
  + **Euclidean Distance**: Đo khoảng cách trực tiếp giữa hai điểm trong không gian. Khoảng cách càng nhỏ thì hai ảnh càng giống nhau.

=> **Nhận định**:

* + - Trong một số trường hợp (ví dụ với **ArcFace**), cosine similarity có thể cho kết quả chính xác hơn do kiến trúc ArcMargin loss tối ưu cho cosine similarity.
    - Ngược lại, **Euclidean distance** có thể phù hợp hơn với các mô hình như **VGG-Face**.

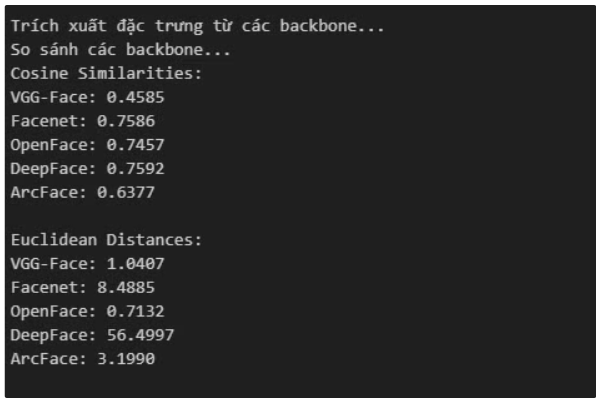
**5. Ứng dụng của các Backbone Khác Nhau**

* + **Facenet** và **ArcFace**: Được sử dụng phổ biến trong các hệ thống nhận diện khuôn mặt yêu cầu độ chính xác cao như bảo mật, kiểm tra danh tính.
  + **VGG-Face** và **DeepFace**: Thích hợp cho các ứng dụng yêu cầu phân tích khuôn mặt chi tiết và không cần tối ưu hóa về tốc độ.
  + **OpenFace**: Phù hợp cho các ứng dụng yêu cầu tài nguyên thấp như nhận diện khuôn mặt trong thời gian thực với các thiết bị di động.

**Kết luận:**

Khi trích xuất đặc trưng từ các backbone khác nhau, bạn sẽ nhận thấy sự khác biệt về khả năng học đặc trưng, kích thước embedding, và hiệu suất tính toán. Mỗi backbone có ưu và nhược điểm riêng, và việc chọn backbone phù hợp phụ thuộc vào yêu cầu cụ thể của ứng dụng như độ chính xác, tốc độ, hay tài nguyên tính toán.

\*\* Thực nghiệm \*\*



**1. Cosine Similarities:**

* + **VGG-Face: 0.4585** - Giá trị này thấp hơn so với các backbone khác, cho thấy VGG-Face không nhận diện đặc trưng khuôn mặt của hai ảnh tốt bằng những mô hình khác.
  + **Facenet: 0.7586** - Đây là một trong những giá trị cosine cao, cho thấy sự tương đồng cao giữa hai ảnh trong không gian đặc trưng của Facenet.
  + **OpenFace: 0.7457** - Giá trị này khá gần với Facenet, cho thấy OpenFace cũng nhận diện khá tốt sự tương đồng giữa hai ảnh.
  + **DeepFace: 0.7592** - Tương tự như Facenet, DeepFace có khả năng nhận diện hai ảnh là tương đồng.
  + **ArcFace: 0.6377** - Giá trị thấp hơn so với Facenet và DeepFace, nhưng vẫn cho thấy sự tương đồng giữa hai ảnh.

**Nhận xét chung về cosine similarity:**

* + Các backbone như **Facenet**, **DeepFace**, và **OpenFace** có cosine similarity cao hơn, điều này cho thấy chúng có thể hiệu quả hơn trong việc nhận diện đặc trưng khuôn mặt giữa hai ảnh.
  + **VGG-Face** có giá trị thấp hơn nhiều, điều này có thể ám chỉ rằng mô hình này ít chính xác hơn trong trường hợp này.

**2. Euclidean Distances:**

* + **VGG-Face: 1.0407** - Giá trị tương đối thấp, cho thấy các đặc trưng từ hai ảnh không quá khác biệt.
  + **Facenet: 8.4885** - Giá trị Euclidean khá cao, điều này ám chỉ rằng khoảng cách giữa hai vector đặc trưng của Facenet là khá lớn, có thể do các vector đặc trưng nằm ở các vị trí xa nhau trong không gian.
  + **OpenFace: 0.7132** - Đây là giá trị Euclidean thấp nhất, cho thấy OpenFace nhận ra sự tương đồng giữa hai ảnh nhiều hơn các mô hình khác.
  + **DeepFace: 56.4997** - Giá trị rất cao, cho thấy rằng các đặc trưng được DeepFace trích xuất giữa hai ảnh rất khác nhau.
  + **ArcFace: 3.1990** - Giá trị Euclidean khá cao, nhưng vẫn thấp hơn so với Facenet và DeepFace.

**Nhận xét chung về Euclidean distance:**

* + **OpenFace** có Euclidean distance thấp nhất, cho thấy rằng các đặc trưng trích xuất giữa hai ảnh rất gần nhau, có thể là backbone tốt trong trường hợp này.
  + **DeepFace** và **Facenet** có khoảng cách Euclidean lớn, điều này có thể chỉ ra rằng hai ảnh không giống nhau lắm khi sử dụng các mô hình này.

**3. Tổng kết:**

* + Dựa trên **cosine similarity**, các backbone như **Facenet** và **DeepFace** cho thấy sự tương đồng cao giữa hai ảnh.
  + Tuy nhiên, **Euclidean distance** từ **OpenFace** lại thấp nhất, cho thấy OpenFace có khả năng phát hiện sự tương đồng giữa hai ảnh cao nhất.
  + **ArcFace** có kết quả khá cân bằng giữa cả cosine và euclidean, không quá tốt nhưng cũng không quá kém.

**4. Những điều cần xem xét:**

* + **Cosine similarity** và **Euclidean distance** là hai cách đo lường khác nhau. Cosine similarity chủ yếu đo góc giữa các vector, trong khi Euclidean đo khoảng cách trực tiếp giữa chúng. Các backbone khác nhau sẽ có hiệu suất khác nhau với từng độ đo.
  + **DeepFace** có khoảng cách Euclidean rất cao, có thể do không phù hợp với đặc thù của các ảnh bạn thử nghiệm.

Nếu mục tiêu của bạn là tìm mô hình tốt nhất cho việc so sánh khuôn mặt trong các ảnh, bạn có thể ưu tiên chọn **OpenFace** dựa trên kết quả Euclidean và **Facenet** hoặc **DeepFace** dựa trên kết quả cosine similarity.

* Hàm Cosine, Euclidian

