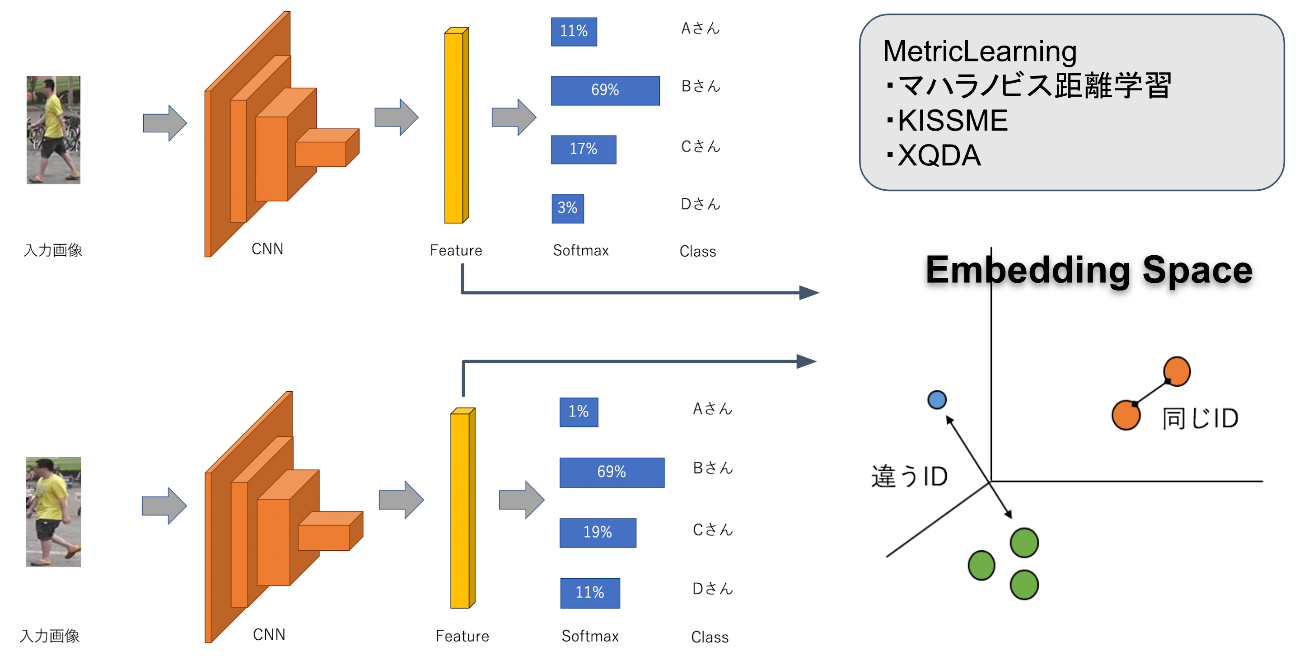
**1. Nếu chỉ sử dụng mạng phân loại đơn giản?**

* Không có sự phân biệt đối với các trường hợp nằm ngoài tập hợp dữ liệu. Cụ thể trong bài toán nhận dạng mặt cho kiểm tra an ninh, trường hợp UNKNOWN sẽ là một mục tiêu.

**2. Metric Learning**

Để giải quyết bài toán phân biệt giữa đối tượng nằm trong và nằm ngoài tập dữ liệu, một phương pháp mới được đưa ra, Metric Learning. Phương pháp này sử dụng **khoảng cách** làm đại lượng để đánh giá sự tương đồng của hai đối tượng.

Khoảng cách ở đây là khoảng cách giữa mảng tính chất của 2 đối tượng sau khi được tính toán qua mạng tích chập trước đó. Thông qua việc học dữ liệu, mảng tính chất (Feature vectors) sẽ có được những giá trị để phân biệt các đối tượng với nhau.



**2.1. Phân biệt giữa học (train) và thực thi (inference)**

Đầu tiên, chúng ta sẽ tìm hiểu về giai đoạn thực thi của mạng Metric Learning.

Giả định:

* Với mỗi đối tượng, tồn tại một mảng tính chất tương ứng mang các giá trị để phân biệt với các đối tượng khác.

Với giả định như trên, chúng ta thử tính khoảng cách với chuẩn hóa L2 của một mảng tính chất với tất cả các mảng còn lại:

Dựa vào biểu đồ:

* Những khoảng cách lớn hơn 1.8 cho chúng ta biết đối tượng nằm ngoài tập dữ liệu.
* Nếu khoảng cách nhỏ hơn 0.6, chúng ta có thể coi đối tượng có nhận dạng là người 0.

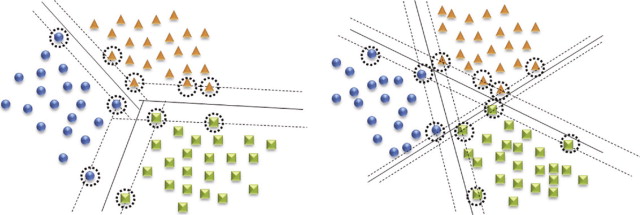
**Vấn đề: Làm sao để có một đáng giá tượng tự đối với cả mười người?**

**Phương pháp**

* Đối với trường hợp nằm ngoài tập dự liệu, chúng ta chọn ngưỡng bằng với khoảng cách lớn nhất.
* Mỗi đối tượng chỉ cần một mảng tính chất.
* Những nhận dạng có khoảng cách ngắn nhất là nhân dạng đúng.

**Phương pháp**

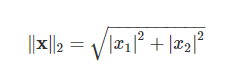
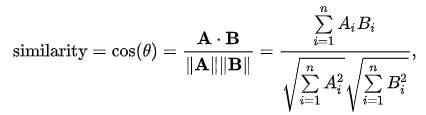
* Đối với trường hợp nằm ngoài tập dự liệu, chúng ta chọn ngưỡng bằng với khoảng cách lớn nhất.
* Mỗi đối tượng cần nhiều hơn một mảng tính chất.
* Sử dụng các thuật toán phân loại để chia vùng của các khoảng cách. Ví dụ như SVM (Support Vector Machine)

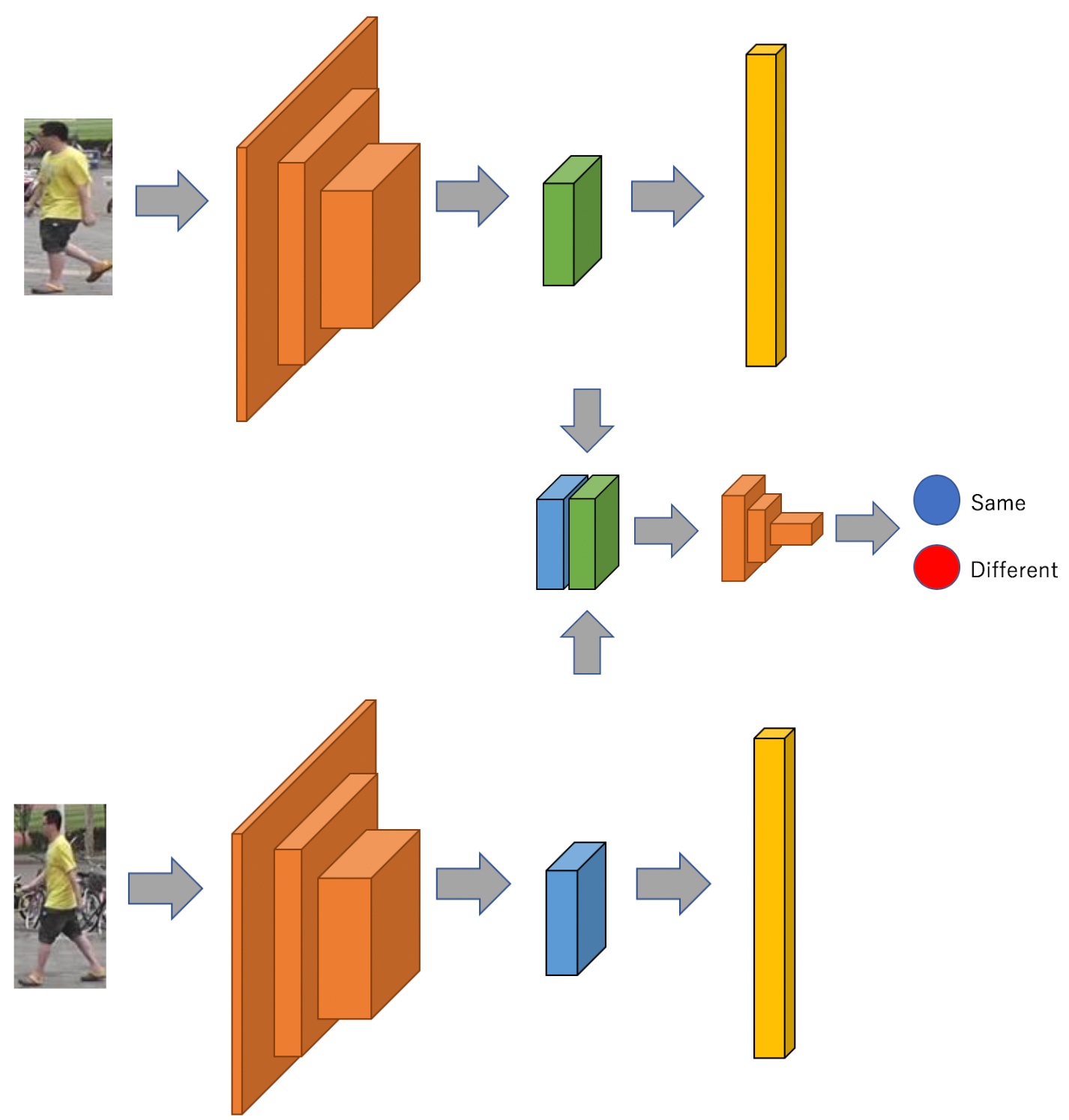


*Support Vector Machine*

**Phương pháp**

Thay đổi cách tính khoảng cách:

* L2 norm: 
* L1 norm: 
* Consine Similiarity: 

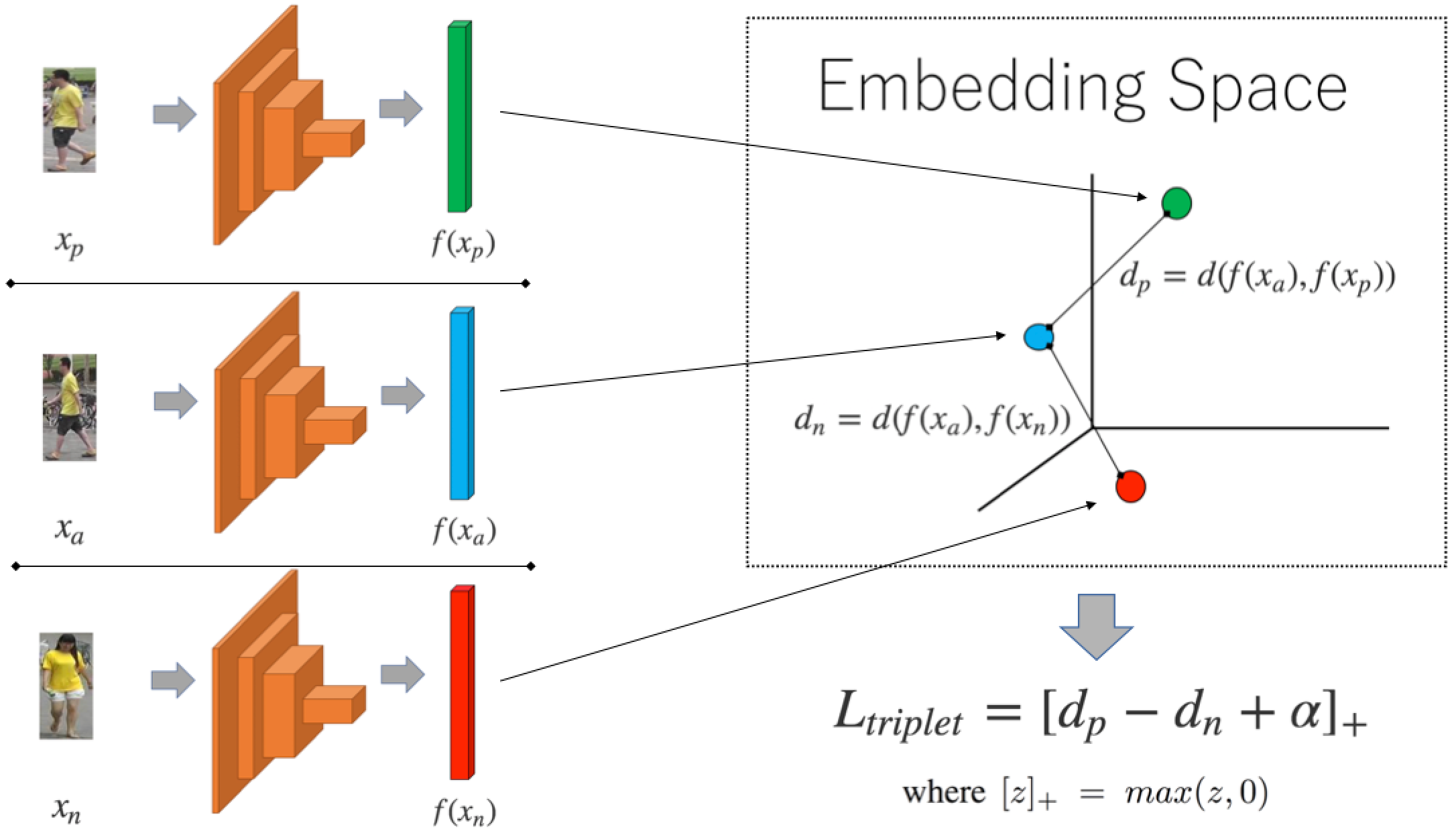


*Hình ảnh tổng quát*

Tiếp theo, chúng ta sẽ tìm hiểu về việc học (train).

Thông thường, trong trường hợp giả định ban đầu về mảng tính chất chưa được đáp ứng, hoặc các mảng tính chất chưa thực sự biểu diễn điểm khác biệt của các đối tượng, chúng ta phải thực hiện việc học để cải thiện những mảng tính chất này.

Dưới đấy là hình ảnh mô tả đơn giản của FaceNet, một mạng dùng để nhận diện mạng với độ chính xác 99.65% trên tập dự liệu Labeled Face in the Wild.



Phần bên trái trong 2 trường hợp rất giống nhau:

**Bức ảnh -> CNN -> Mảng tính chất (Feature vectors)**

Tuy nhiên, để cải thiện các mảng tính chất, chúng ta cần một yếu tố– thành phần đánh giá – hàm mất mát (Loss functions).

Thuật toán về việc học của một vòng lặp bao gồm việc tính toán và đánh giá

**Đánh giá**

**Tính toán**

**Y – Y1**

**AX + B = Y**

https://github.com/davidsandberg/facenet