**Haar Cascade là gì?**

Về cơ bản là sử dụng các đặc trưng loại **Haar** và sau đó sử dụng thật nhiều đặc trưng đó qua nhiều lượt (**cascade**) để tạo thành một cỗ máy nhận diện hoàn chỉnh.

Ở đặc trưng **Haar**, bộ lọc chỉ chiếm một phần trong cửa sổ trượt thôi. Điều đó được minh hoạ trên ảnh sau:

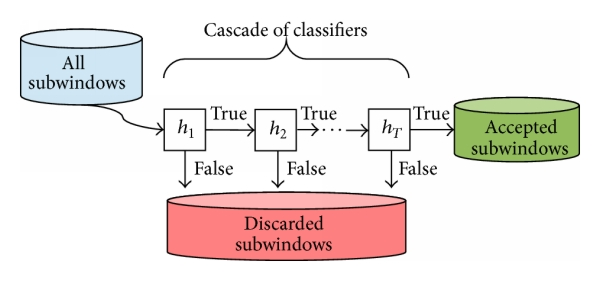


Trong hình trên, cửa sổ trượt được đặt ngay ngắn vừa gọn để nhìn được toàn bộ ảnh.

Bộ lọc **Haar** chỉ nhìn cụ thể vào một vùng trong cửa sổ để tìm thôi: trong khuôn mặt thì mũi lúc nào cũng ở chính giữa chứ không ở các góc, nên không cần nhìn các góc để làm gì cả.

**Các bộ lọc Haar** kể cả sau **Adaboost** vẫn chỉ bắt được những đặc trưng rất cơ bản, và để nhận ra một khuôn mặt thì chúng ta cần tầm 6000 các đặc trưng như vậy. Vậy chúng ta cần có một cách để vote xem cửa sổ đó có chứa mặt không, mà vẫn phải xử lý đủ nhanh cho cả 6000 đặc trưng đó: câu trả lời là **Cascade**.

Việc **cascade** được thiết kế như sau: trong 6000+ đặc trưng đó, chia chúng ra thành rất nhiều bước. Trong đó, mỗi lần cửa sổ trượt qua một vùng bước ảnh, từng bước một sẽ được xử lý: nếu bước 1 nhận đó là mặt, chúng ta chuyển qua bước 2; và nếu không thì chúng ta bỏ qua vùng đó và trượt cửa sổ đi chỗ khác. Nếu một vùng pass toàn bộ các bước test mặt đó thì cửa sổ đó có chứa mặt người.



Các vùng không chứa mặt sẽ bị vứt vào hộp đỏ kia và không bao giờ được nhớ tới nữa, và các vùng có mặt sẽ được đưa vào hộp xanh để xử lý tiếp.

Chúng ta có thể chia kết quả ra làm 4 loại:

| **Tên** | **Đáp án đúng** | **Đáp án của chúng ta** |
| --- | --- | --- |
| True positive | True | True |
| True negative | False | False |
| False positive | False | True |
| False negative | True | False |

Trong đó, nếu dự đoán đúng rồi thì không cần bàn cãi, nhưng nếu chúng ta dự đoán sai một bước sẽ có thể không qua các bước còn lại. Vì vậy, trong quá trình train, mô hình sẽ lựa chọn các classifier tốt nhất với độ tự tin hợp lý để ưu tiên việc False negative không được phép tồn tại — vì nếu chúng ta gặp False positive, các bước sau sẽ loại ví dụ đó cho chúng ta sau. Trong mô hình nhận mặt người, bước đầu tiên đúng có 2 bộ lọc trên, với [tỉ lệ false negative rất gần 0, và tỉ lệ false positive là 40%](http://research.microsoft.com/~viola/Pubs/Detect/violaJones_IJCV.pdf).

Sau đó các cửa sổ nằm trong ô xanh sẽ được trả lại làm các ô chứa mặt: nếu có các ô đè lên nhau cùng chứa một mặt, các toạ độ các góc sẽ được cộng vào lấy trung bình.

