TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

KHOA CÔNG NGHỆ THÔNG TIN

ĐỒ ÁN NHẬN DẠNG:

FACIAL LANDMARK DETECTION

MSSV	Họ Tên
1712159	Nguyễn Đỗ Chí Thảo
1712202	Nguyễn Trọng Văn
1712209	Lê Quang Vũ

Năm học: 2019 -2020

Mục lục

I/ Giới thiệu:

Phát hiện và nhận dạng khuôn mặt là một trong những lĩnh vực sinh – tin học vô cùng quan trọng trong suốt 20 năm qua. Nhiệm vụ của phát hiện khuôn mặt là tìm kiếm khuôn mặt trong hình ảnh và trả về vị trí của khuôn mặt. Những nghiên cứu gần đây đã đem lại sự cải thiện về độ chính xác và thời gian thực hiện.

Tuy nhiên, nếu chỉ dừng lại ở việc phát hiện khuôn mặt thì chúng ta sẽ không có được những thông tin, những đặc trưng của khuôn mặt: viền mắt, góc miệng, lông mày... Đó là lý do Facial Landmark recogniton ra đời.

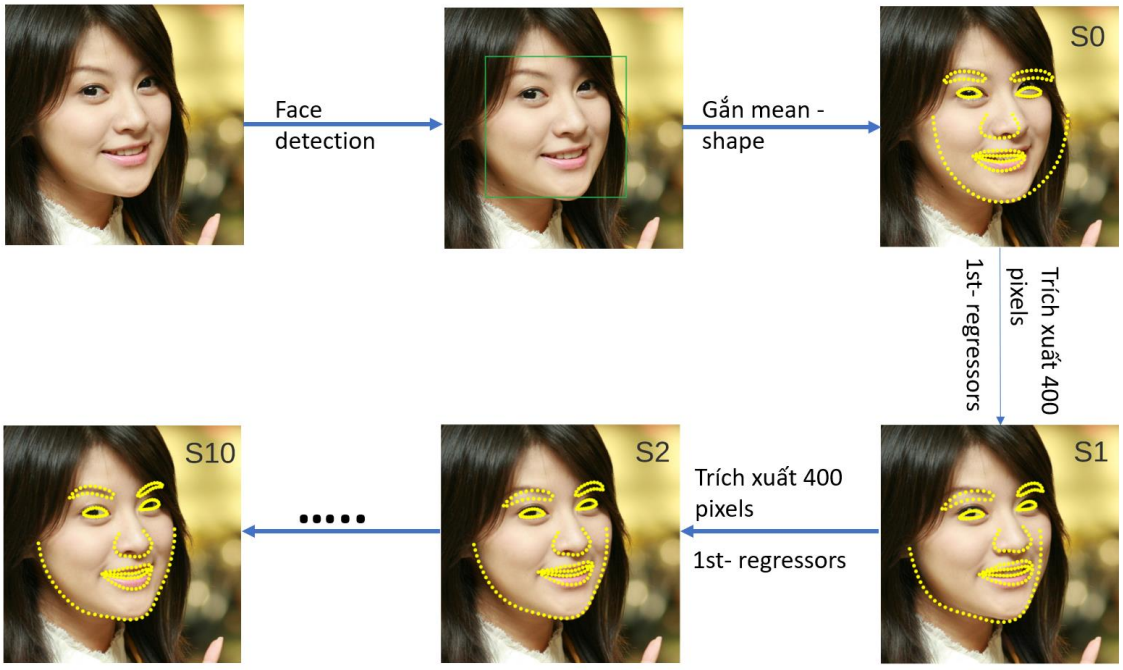
II/ Phương pháp:

1/ Tổng quan:

Có một số phương pháp để giải quyết bài toán này như: Active Shape Model, Active Appearance Model ... Trong bài báo cáo này, nhóm xin trình bày phương pháp **Face Alignment by Shape Regression** (Tạm dịch: Căn chỉnh khuôn mặt dựa trên hồi quy về hình dáng)

Tương tự như những phương pháp khác, để có thể huấn luyện mô hình dựa trên phương pháp Face Alignment by Shape Regression cần phải trải qua hai quá trình: quá trình huấn luyện và quá trình kiểm tra

2/ Quá trình huấn luyện:



Nguồn tham khảo: One Millisecond Face Alignment with an Ensemble of Regression Trees, Vahid Kazemi and Josephine Sullivan

Một face shape (tạm dịch: hình dạng mặt) $S = \left(x_1, y_1, x_2, y_2, ..., x_{N_{LM}}, y_{N_{LM}}\right)$ bao gồm N_{LM} facial landmark (tạm dịch: mốc khuôn mặt). Trong đó (x_i, y_i) tương ứng với tọa độ của facial landmark thứ i.

Với một bức ảnh khuôn mặt, quá trình tìm được vị trí chính xác các facial landmark là tìm shape S sao cho gần nhất với shape chính xác $\hat{S}$. Đây là quá trình làm minimize độ lỗi:

$$|| S - \hat{S} ||_2 \quad (1)$$

Trong phương pháp Face Alignment by Shape Regression, chúng ta sử dụng phương pháp hồi quy tăng cường để kết hợp T "week regressors" (tạm dịch: hồi quy yếu): (R^1, R^2 ... R^T) để tìm ra hình dạng chính xác $\hat{S}$

Gọi face shape ban đầu là S^0. S^0 có thể là "mean-shape":

$$S_0 = \frac{1}{N}\sum_{i=1}^{N} S_i$$

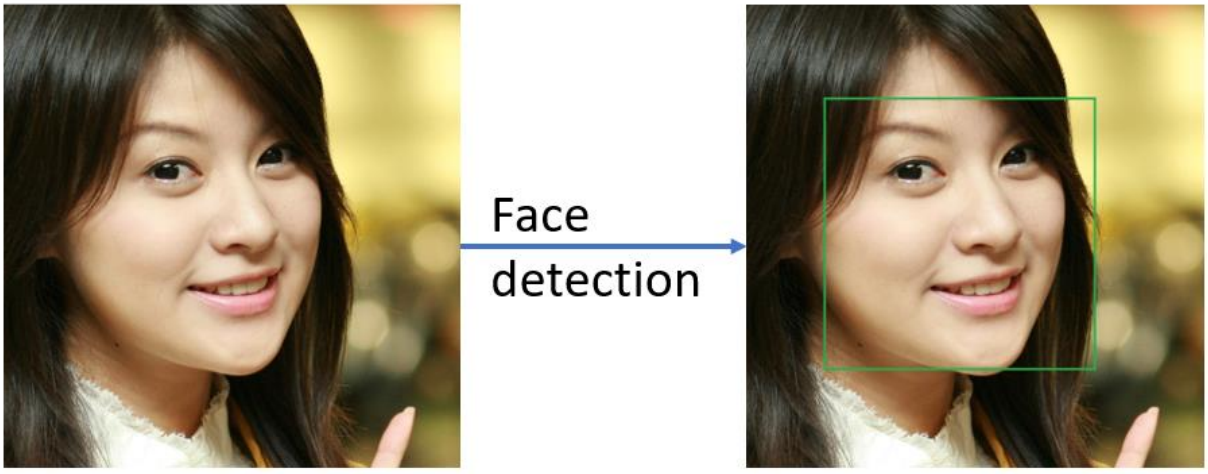
Với hình ảnh khuôn mặt I đầu vào và shape ban đầu S^0, mỗi regressor sẽ tính một "shape increment" δS từ bức hình I và "current shape" S^{t-1}:

$$S^t = S^{t-1} + R^t\left(I, S^{t-1}\right)(2) \text{ với t = 1, 2 ... T}$$

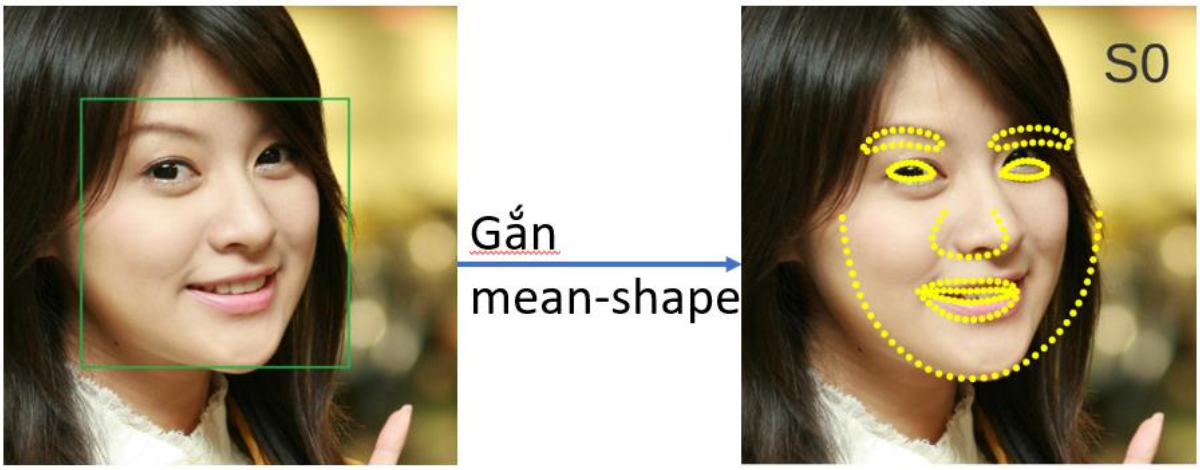
Trong đó, R^t: regressor thứ t, phụ thuộc vào ảnh I và "previous shape" S^{t-1}.

Quá trình này diễn ra như sau:

Đầu tiên, từ hình ảnh dữ liệu huấn luyện đầu vào, việc đầu tiên cần thực hiện xác định vị trí khuôn mặt trong bức hình để phát hiện được vị trí khuôn mặt. Chúng ta có thể thực hiện việc này bằng một số thuật toán, chẳng hạn như **AdaBoost**, ta có thể đánh dấu lại bằng một khung hình chữ nhật.

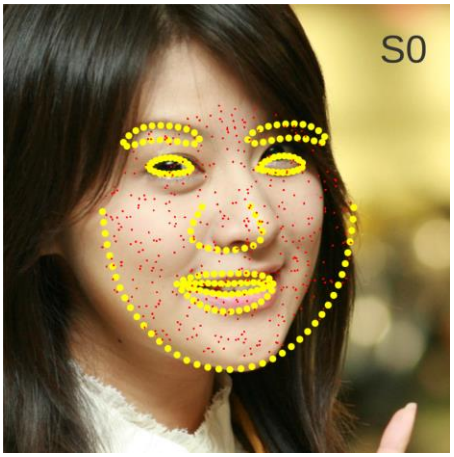


Sau khi đã tìm ra được vị trí khuôn mặt, chúng ta sẽ gắn shape S^0 lên khuôn mặt.



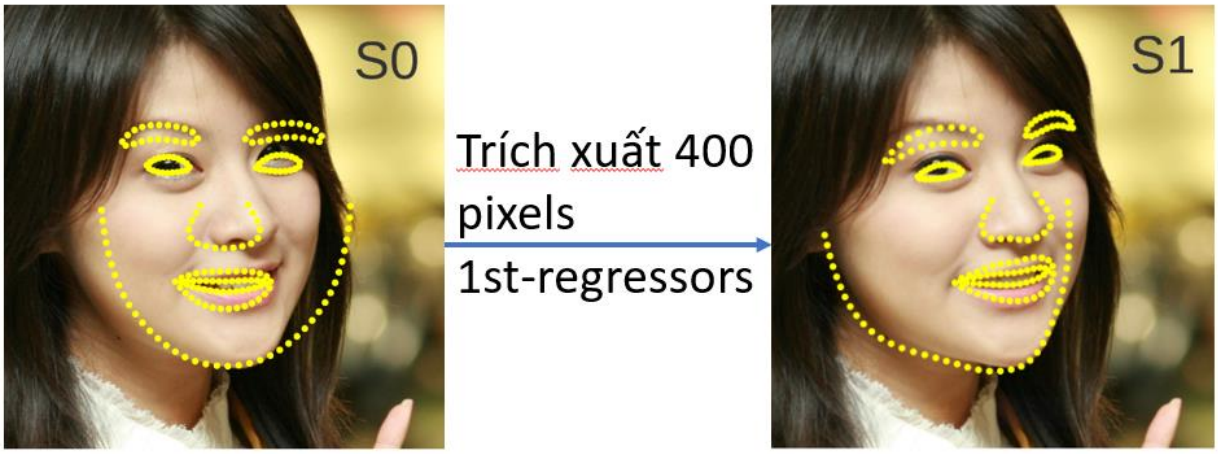
Tuy nhiên, vị trí của shape S^0 vẫn chưa được chính xác cho lắm, so với vị trí chính xác thì vẫn bị "lệch". Do đó, chúng ta cần thực hiện một số bước biến đổi.

+ Đầu tiên, từ mỗi bức hình I_i (trong tập dữ liệu huấn luyện) và "current shape" S chúng ta cần phải trích xuất N_p pixel.



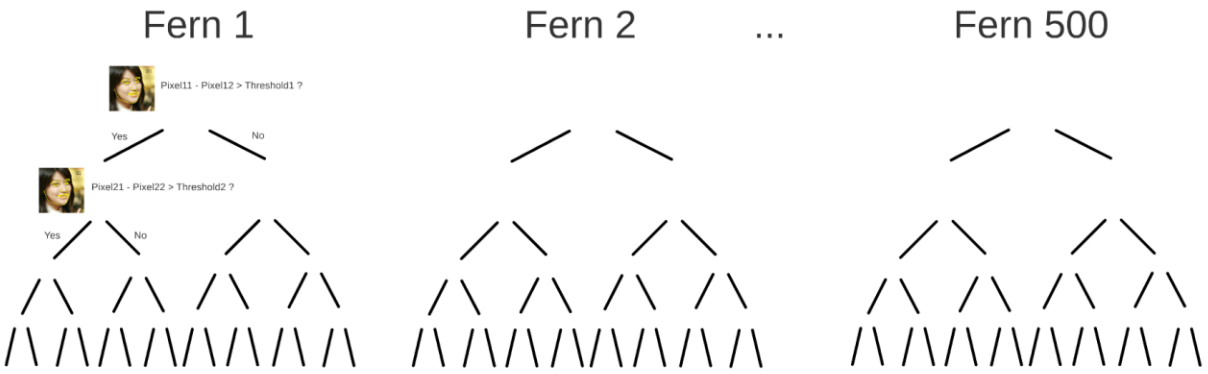
+ Từ thông tin của những pixel đã lấy, ta sẽ xây dựng regressor t: R^t. Sau đó, đưa tập dữ liệu huấn luyện vào, với mỗi bức hình I_i và current shape S^{t-1} ta sẽ thu được S^t tương ứng

Ví dụ, xét “current shape” S^0:

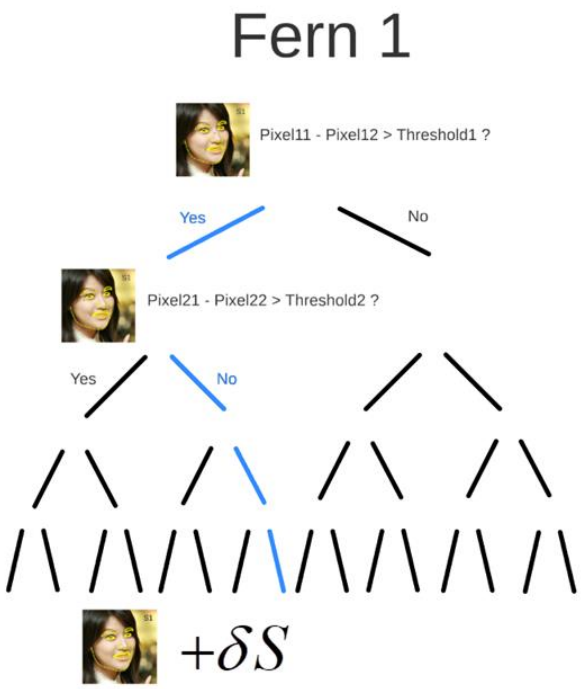


+ Đầu tiên việc chúng ta cần làm là trích xuất N_p pixel từ S^0. Trong ví dụ này, chúng ta sẽ lấy N_p=400. Để lấy 400 pixels này, ta có thể ưu tiên lấy nhiều hơn ở các vùng: mắt, mũi, miệng và lấy ít hơn ở hai bên má hay trán.

+ Sau đó, chúng ta sẽ xây dựng regressor R^1 từ thông tin của 400 pixels này. Thông tin ở đây chính là cường độ sáng của mỗi pixel. Regressor $R^1 = \left(r^1, r^2, ..., r^k\right)$ sẽ bao gồm nhiều fern r^i. Mỗi một fern là một cây nhị phân, trong đó, tại mỗi node sẽ thực hiện một binary test, các node thuộc cùng một tầng sẽ thực hiện cùng một binary test. Các fern trong cùng một regressor có cấu trúc tương tự nhau (độ cao).



Xét Fern 1:



Tại node root của Fern 1 thực hiện một binary test:

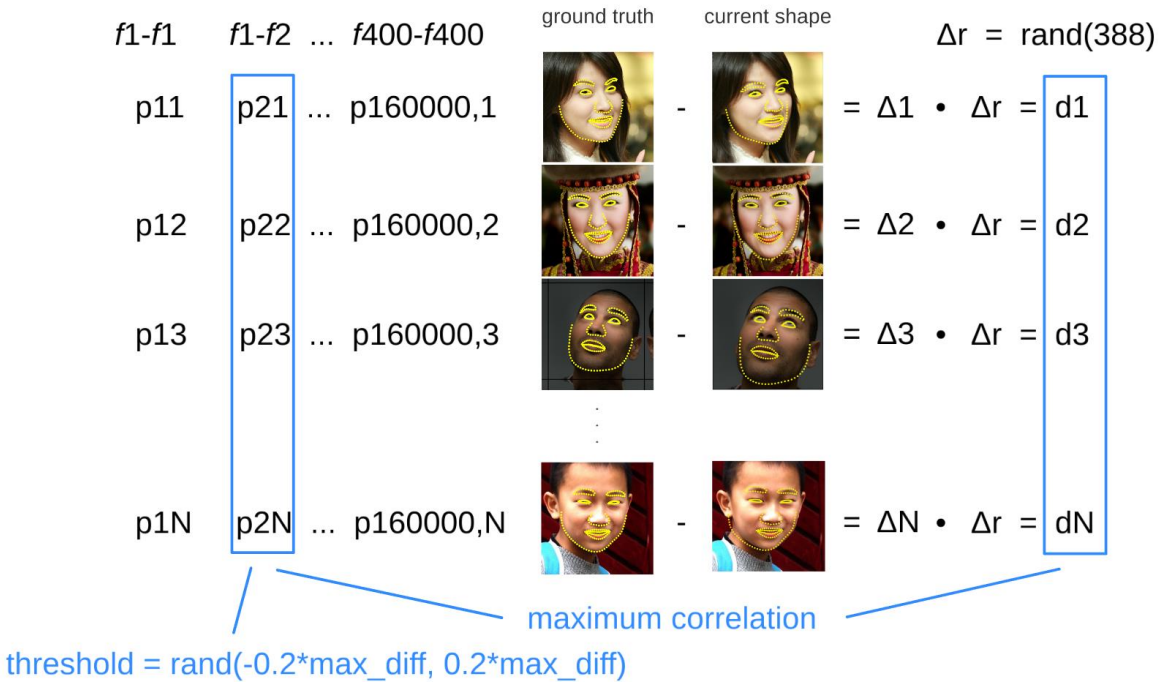
$$Intensity\ (pixel\ 1)\ -\ Intensity\ (pixel\ 2)\ >\ Threshold$$

Rẽ nhánh xuống tầng tiếp theo, chúng ta sẽ sẽ thực hiện một binary test khác. Làm tương tự cho các tầng cho đến khi đến nút lá. Qua mỗi fern, từ tập dữ liệu ban đầu sẽ được chia thành k giỏ khác nhau tương ứng với mỗi node lá. Tại nút lá, ta sẽ thực hiện một số thao tác tính toán để thu được δS_b tương ứng với lá đó, rồi lấy δS_b đem cộng cho current shape.

Để có thể thực hiện binary test tại mỗi node:

$$Intensity\ (pixel\ 1)\ -\ Intensity\ (pixel\ 2)\ >\ Threshold$$

Chúng ta cần phải tìm được một cặp pixel và ngưỡng threshold. Chúng ta có thể thực hiện bằng cách:



Với những bức ảnh trong thuộc cùng một lá, chúng ta sẽ tính sự chệnh lệch cường độ sáng giữa từng pixel trong cùng một bức ảnh: $|pixel\ 1 - pixel\ 1|$, $|pixel\ 1 - pixel\ 2|$ ... thực hiện trên toàn bộ tập dữ liệu thuộc cùng một node lá, ta sẽ có được một ma trận độ chênh lệch cường độ sáng $M_{intensity}$. Trong đó, mỗi dòng là độ chênh lệch cường độ sáng giữa các pixel trong một bức ảnh.

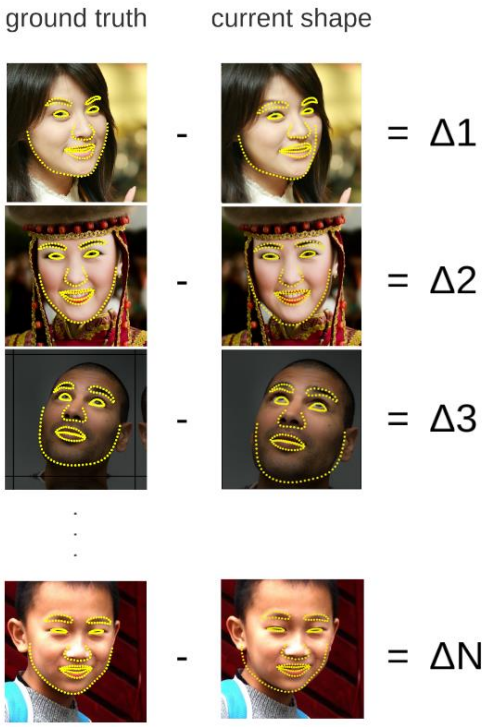
Tiếp theo, ta sẽ tính độ chênh lệch giữa "current shape" và shape chính xác đối với mỗi bức hình: $\Delta 1$, $\Delta 2$ ... ΔN. Khởi tạo ngẫu nhiên Δr. Lấy Δr nhân lần lượt cho: $\Delta 1$, $\Delta 2$ ... ΔN ta sẽ thu được các giá trị vô hướng: d_1, d_2 ... d_N. Lấy những giá trị d_1, d_2 ... d_N này tạo thành một vector d. Chúng ta sẽ xét xem cột nào của $M_{intensity}$ có độ tương quan mạnh nhất với d thì cặp pixel tạo thành cột đó sẽ được lấy để thực hiện binary test. Ngưỡng Threshold được tính bằng cách lấy một giá trị trong khoảng: $(-0.2max\_diff, 0.2max\_diff)$, max_diff là giá trị lớn nhất trong cột chúng ta vừa xét.

Để tính được δS_b tương ứng tại mỗi node lá, ta có thể tính bằng:

$$\delta S_b = \frac{\sum_{i=1}^{|\Omega_b|}\left(\hat{S}_i - S_i\right)}{|\Omega_b|}$$

Trong đó, $|\Omega|$ là kích thước tập dữ liệu tương ứng với mỗi node lá

$\hat{S}_i, S_i$ lần lượt là shape chính xác và current shape của bức hình thứ i trong giỏ b



Để cho mô hình ổn định, chúng ta sẽ lấy δS đem nhân với "damping factor":

$$\delta S_b = \frac{1}{1+\beta/|\Omega_b|}\frac{\sum_{i=1}^{|\Omega_b|}\left(\hat{S}_i - S_i\right)}{|\Omega_b|}$$

Hệ số này nhằm giúp cho mô hình có thể đạt được vị trí chính xác của các facial landmark hay shape chính xác $\hat{S}$.

Sau khi đã tính được δS, ta sẽ lấy δS đem cộng cho "current shape".

Thực hiện quá trình này cho tất cả các fern trong cùng một R^T, ta sẽ thu được δS_b tương ứng tại mỗi lá: $\left(\delta S_{b_1}, \delta S_{b_2}, ..., \delta S_{b_k}\right)$. Cộng tất cả cho "current shape":

$$S^t = S^{t-1} + R^t\left(I, S^{t-1}\right)$$

$$\Leftrightarrow S^t = S^{t-1} + \sum_{i=1}^{k}\delta S_{b_i}$$

Sau khi đã thực hiện xong 1st-regressor, từ shape S^0 ta sẽ thu được S^1. Tuy nhiên, chúng ta cần phải căn chỉnh lại vị trí của N_{LM} pixel đã lấy. Đối với mỗi pixel P, chúng ta sẽ căn chỉnh theo vị trí của landmark L gần nó nhất ở S^0.

Làm tương tự cho: regressor R^2, regressor R^3 ... regressor R^T, ta sẽ thu được S^T gần đúng nhất với $\hat{S}$.

Chúng ta có thể thay đổi một số tham số trong mô hình: độ cao của mỗi fern, số lượng regressor hay là số pixel được lấy ở mỗi shape S và bức hình I. Tuy nhiên sẽ có sự đánh đổi giữa thời gian huấn luyện và độ chính xác.

2/ Quá trình kiểm tra:

Nhìn chung quá trình này tương đối giống quá trình huấn luyện. Tuy nhiên, chúng sẽ sử dụng mô hình mà đã được huấn luyện với những tham số đầu vào, chẳng hạn như: binarry test ở mỗi node hay vị trí các pixel cần lấy... Tất cả đã có sẵn trong mô hình. Từ tập dữ liệu hình ảnh đầu vào, chúng ta đưa vào mô hình sẽ có được face shape tương ứng.

III/ Đánh giá hiệu suất mô hình:

1/ Theo lý thuyết:

Chúng ta sẽ dựa vào kết quả thực nghiệm của một số bài báo khoa học.

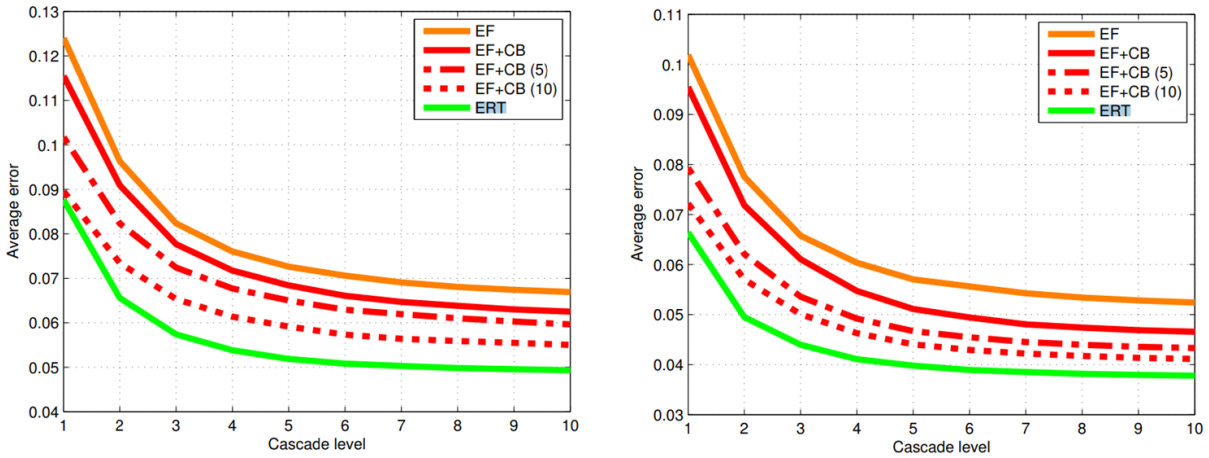
Bảng kết quả về thời gian huấn luyện mô hình (trên 2000 ảnh) tương ứng với số lượng facial landmark:

Landmarks	5	29	87
Training (mins)	5	10	21
Testing (ms)	0.32	0.91	2.9

Table 1. Training and testing times of our approach, measured on an Intel Core i7 2.93GHz CPU with C++ implementation.

Nguồn tham khảo: Face Alignment by Explicit Shape Regression - Xudong Cao, Yichen Wei, Fang Wen, Jian Sun

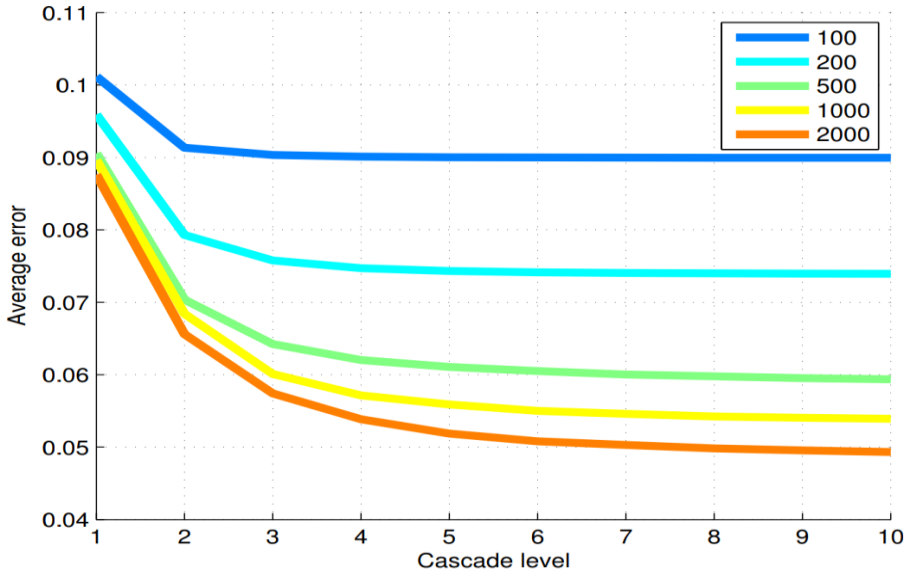
Đồ thị so sánh mối quan hệ độ lỗi trung bình và số cascade giữa các thuật toán:



(a) HELEN (b) LFPW

Nguồn tham khảo: *One Millisecond Face Alignment with an Ensemble of Regression Trees, Vahid Kazemi and Josephine Sullivan*

Đồ thị so sánh mối quan hệ độ lỗi trung bình và số cascade giữa các kích thước của tập dữ liệu huấn luyện:



Nguồn tham khảo: *One Millisecond Face Alignment with an Ensemble of Regression Trees, Vahid Kazemi and Josephine Sullivan*

2/ Theo thực nghiệm:

a/ Quá trình huấn luyện

Mô hình được huấn luyện trên bộ dataset: Ibug-300W.

Cấu hình: Google colab, GPU 12 GB.

Kích thước: 31.5 Mb.

Mô hình xác định 68 điểm trên khuôn mặt (tự huấn luyện) với công cụ từ thư viện Dlib, với các thông số sau:

Cascade Depth = 10,

Tree Depth = 3,

Feature Pool Size = 150,

Fern per cascade =500.

b/ Quá trình kiểm tra:

Độ lỗi trên mô hình tự huấn luyện: error = 0.15

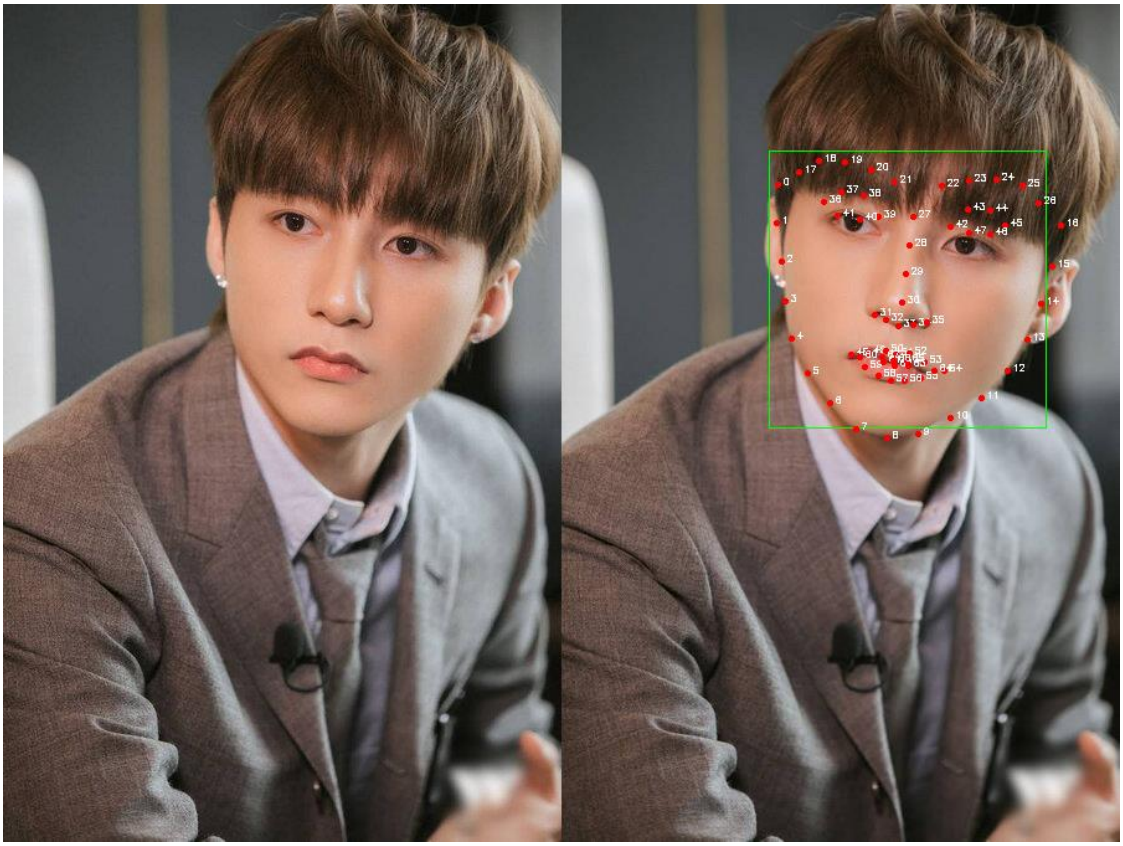
Độ lỗi trên mô hình cung cấp sẵn của Dlib: error = 0.065

Cấu hình: Google colab, GPU 12 GB.

Lý do:

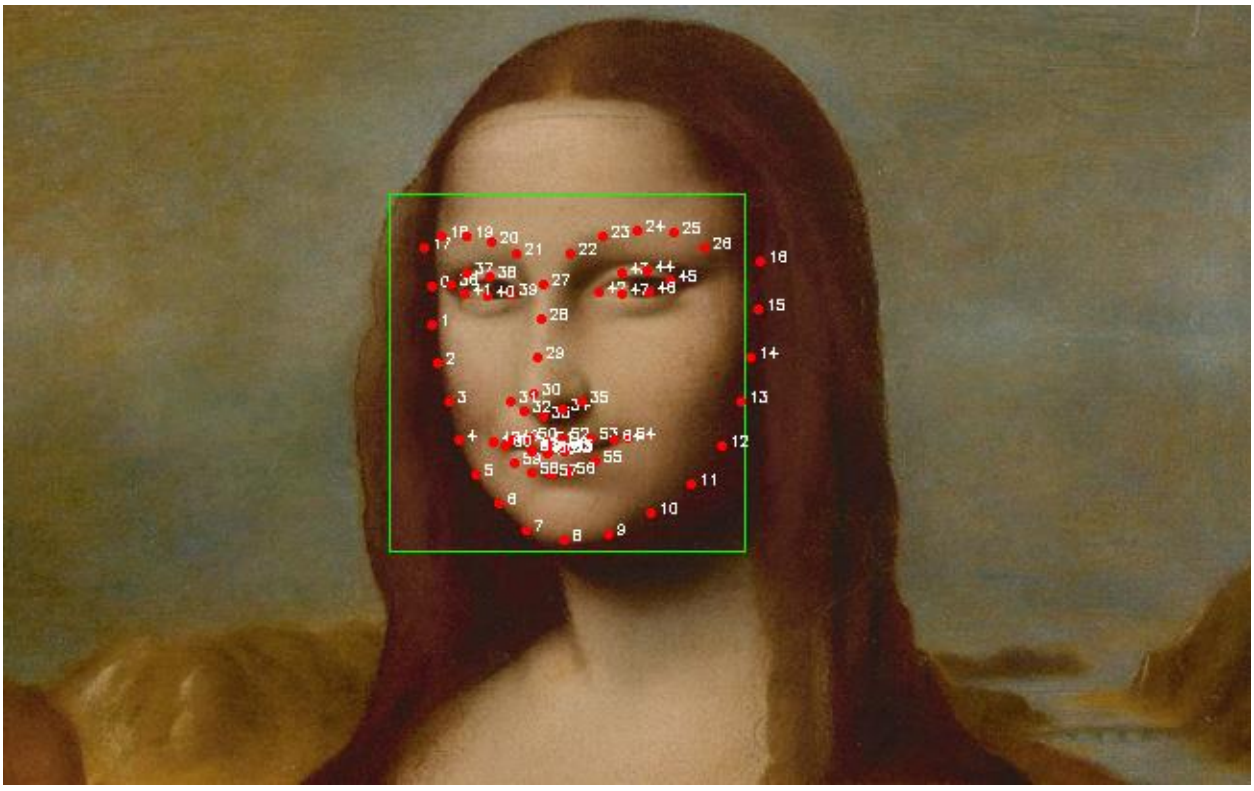
- Bộ dữ liệu dung để huấn luyện mô hình nhỏ hơn so với pretrained - model vì hạn chế về phần cứng.

Một số kết quả của mô hình:

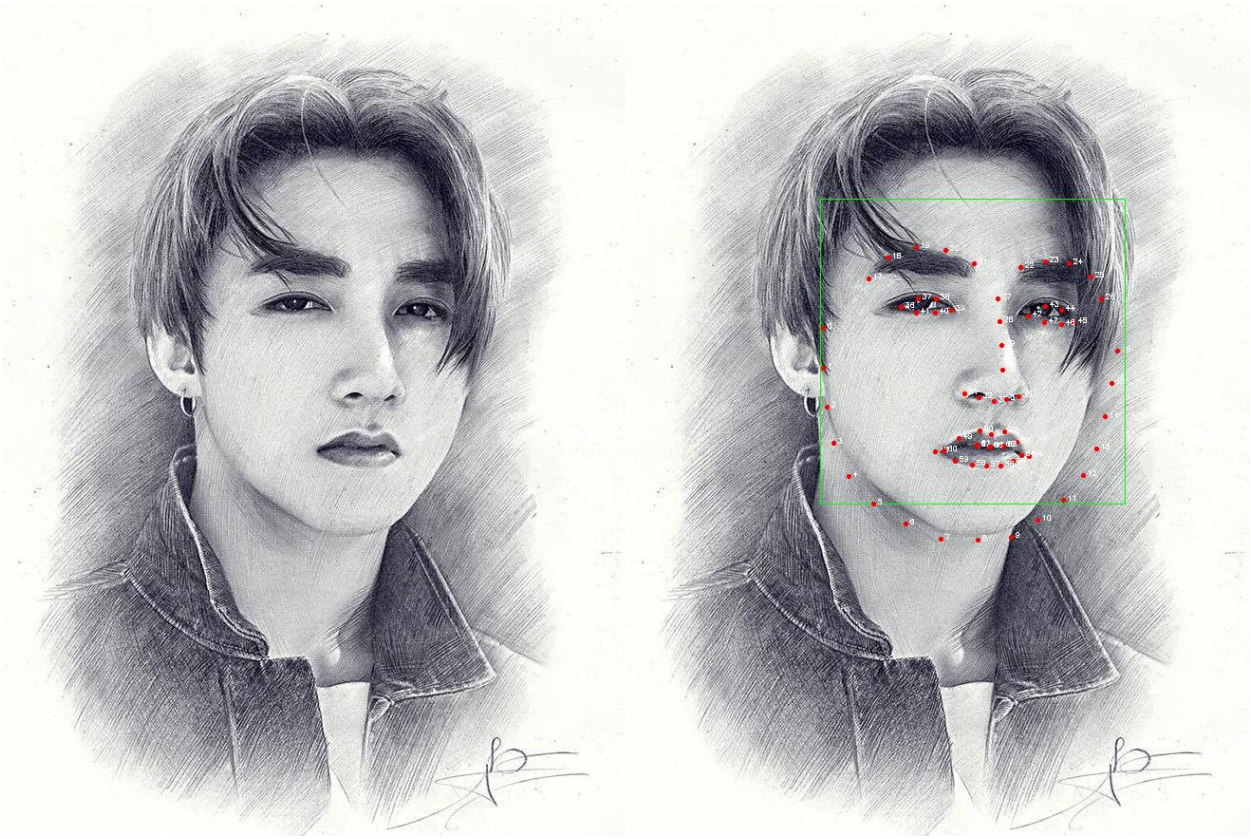


Mô hình có thế nhận dạng facial landmark cho nhiều khuôn mặt trong cùng một bức hình:

Mô hình còn có thể nhận dạng facial landmark trong cả trong những bức tranh. Chẳng hạn như bức họa nổi tiếng Mona Lisa của Leonardo da Vinci:



Hay thậm chí, trong tranh chì mô hình vẫn có thể nhận dạng facial landmark:



IV/ Nhận xét:

- Mô hình Landmark detection do nhóm huấn luyện gặp một số hạn chế:

+ Không thể nhận dạng những hình ảnh trong gương

+ Bị hạn chế bởi: chế độ chiếu sáng, góc mặt …

+ Không thể nhận dạng khi những vị trí landmark bị chắn bởi: tóc, kính mắt …

+ Nhận dạng không chính xác đối với trường hợp khi vị trí khuôn mặt quá nhỏ so với bức ảnh

V/ Tham khảo:

- One Millisecond Face Alignment with an Ensemble of Regression Trees, Vahid Kazemi and Josephine Sullivan

- Face Alignment by Explicit Shape Regression - Xudong Cao, Yichen Wei, Fang Wen, Jian Sun

- Face Alignment, Peter Zhang