

# Unsupervised Learning of Symmetric 3D Faces with PyTorch3D

Nguyễn Minh Vũ  
21120369

Ngành Trí tuệ Nhân tạo  
Trường đại học Khoa học Tự nhiên, ĐHQG-HCM  
21120369@student.hcmus.edu.vn

Nguyễn Minh Tú  
21120586

Ngành Trí tuệ Nhân tạo  
Trường đại học Khoa học Tự nhiên, ĐHQG-HCM  
21120586@student.hcmus.edu.vn

Phùng Hoài Thi  
21120558

Ngành Trí tuệ Nhân tạo  
Trường đại học Khoa học Tự nhiên, ĐHQG-HCM  
21120558@student.hcmus.edu.vn

## I. GIỚI THIỆU

Bài toán tái tạo khuôn mặt người 3D (3D face reconstruction) từ hình ảnh 2D đã và đang là vấn đề rất được quan tâm trong lĩnh vực thị giác máy tính trong những thập kỷ qua vì tính ứng dụng thực tế mà bài toán này mang lại. Có thể kể đến các vấn đề về bảo mật, truy cập quyền kiểm soát thông qua nhận dạng khuôn mặt hay hỗ trợ tác vụ xác định nghi phạm,... Tuy nhiên, không giống như mắt người có thể nhận dạng vật thể 3 chiều từ ảnh 2 chiều mà chỉ cần nhìn qua chúng, việc tái tạo ảnh 3D lại vô cùng phức tạp. Bài toán phải đổi mặt với nhiều thách thức như loại bỏ che khuất, loại bỏ trang điểm và chuyển đổi biểu cảm. Che khuất có thể là nội hoặc ngoại. Một số trong những che khuất nội tiếng là tóc, râu, ria mép và tư thế bên. Che khuất bên ngoài xảy ra khi một số đối tượng/người khác che phần của khuôn mặt, chẳng hạn như kính, tay, chai, giấy và khẩu trang. Việc đó khiến cho bài toán rất khó để xử lý. May mắn thay, trong những năm trở lại đây, sự phát triển mạnh mẽ của các bộ xử lý trung tâm đa lõi (CPU), điện thoại thông minh, đơn vị xử lý đồ họa (GPU) và các ứng dụng đám mây như Amazon Web Services (AWS), Google Cloud Platform (GCP) và Microsoft Azure đã góp phần thúc đẩy cho các mô hình tái tạo khuôn mặt 3D.

## II. CÁC CÔNG TRÌNH LIÊN QUAN

Các kỹ thuật dùng để tái tạo khuôn mặt 3D có thể được phân loại thành 2 nhóm phương pháp chính: phương pháp truyền thống như Định dạng từ đồ bóng (Shape from Shading) [1] và phương pháp Học sâu (Deep Learning).

### A. Shape from Shading

Shape from Shading hay phục hồi hình dạng (Shape recovery) là một tác vụ cơ bản trong thị giác máy tính, xuất phát từ một kỹ thuật định hình theo X (Shape-from-X) với X có thể là đồ bóng, chuyển động hay cấu trúc ảnh. Trong SFS, cho trước một ảnh xám, mục tiêu là trả về phục hồi hướng của nguồn sáng và bề mặt của hình dạng vật thể trong ảnh dưới mức độ pixel. SFS có thể giải quyết tốt đối với các ảnh

có nguồn sáng không chính diện nhưng lại không làm tốt tác vụ vật thể bị che khuất vì hình dạng được xác định dựa trên bóng của vật thể. Có thể kể đến bài báo của BKP Horn [2] lần đầu tiên đề cập đến phương pháp SFS hay các mô hình cải tiến sau này như [3], [4], [5].

### B. Supervised Learning

Một cách tiếp cận khác đó là sử dụng phương pháp học có giám sát trong tái tạo khuôn mặt 3D thu hút sự chú ý nhờ khả năng mang lại kết quả chính xác cao, tuy nhiên cũng tiềm ẩn nhiều hạn chế cần được xem xét kỹ lưỡng [6], [7]. Điểm mạnh chính của phương pháp này nằm ở việc sử dụng hàm mất mát hiệu quả, Chiết lược tập trung vào hình ảnh lõi dựa trên màu da và mạng nơ-ron dự đoán độ tin cậy, nhưng lại vấp phải những khó khăn như yêu cầu dữ liệu 3D thực tế khắt khe, tính thiên vị, khả năng tổng quát hóa thấp. Bên cạnh đó, vẫn đề đạo đức liên quan đến việc thu thập và sử dụng dữ liệu 3D của con người cũng cần được quan tâm. Nhìn chung, phương pháp học có giám sát vẫn là công cụ mạnh mẽ nhưng cần được cải thiện để giải quyết các hạn chế và đảm bảo tính minh bạch, công bằng và đạo đức.

### C. Unsupervised Learning

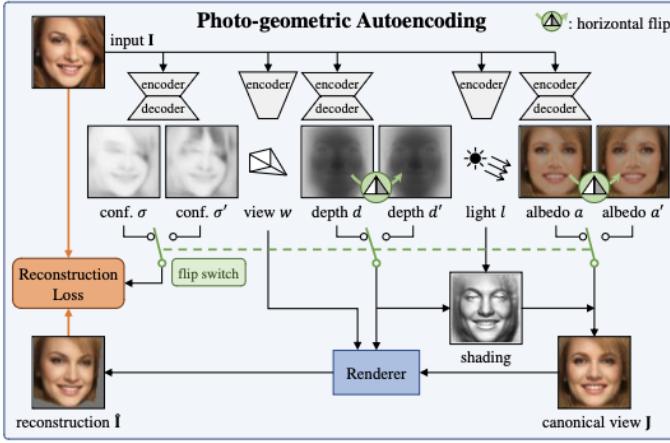
Những phương pháp tái tạo khuôn mặt 3D sử dụng deep learning mang lại hiệu quả đáng tin cậy, ổn định với các thay đổi từ môi trường và bắt được những chi tiết nhỏ. Nhưng việc huấn luyện mô hình có giám sát đòi hỏi tập dữ liệu có nhãn nên có thể gây ra khó khăn trong quá trình thu thập dữ liệu. Những phương pháp học không giám sát có thể khắc phục được han chế trên.

Mô hình UnsupNet [8] có thể tái tạo hình dạng 3D của khuôn mặt qua quá trình học không giám sát. Mô hình sử dụng kiến trúc Encoder-Decoder. Mô hình sẽ thực hiện quá trình encoding và sau đó là quá trình decoding để có được thông tin về độ sâu, ảnh chiếu trực diện, góc nhìn và nguồn sáng. Việc huấn luyện mô hình UnsupNet không cần nhãn dữ liệu hay mô hình hình dạng như 3DMM, đổi lại mô hình có

thể cho hiệu quả không cao trên các khuôn mặt có biểu cảm phức tạp, không đối xứng, bị vật thể khác che khuất.

### III. ĐỀ XUẤT MÔ HÌNH

Để giải quyết bài toán tái tạo khuôn mặt 3D, bài viết này đề xuất lựa chọn kiến trúc Encoder-Decoder. Từ một ảnh ban đầu sẽ được đưa qua những lớp Encoder có thiết kế khác nhau để xây dựng các embedding cho ảnh. Các lớp embedding này sẽ đó được qua các lớp Decoder để trích xuất các đặc trưng của ảnh. Từ các đặc trưng này ta có thể tái tạo lại biểu diễn 3D của khuôn mặt. Sau đó biểu diễn 3D và ảnh gốc ban đầu được cho qua một hàm lỗi để đánh giá sai khác trong quá trình tái tạo. Bằng cách tối ưu hóa hàm lỗi bằng các phương pháp tối ưu gradient sẽ cập nhật lại trọng số của các lớp Encoder và Decoder. Mô hình sau khi được huấn luyện có thể trích xuất các đặc trưng của ảnh từ đó tái tạo lại cách biểu diễn 3D của khuôn mặt. Để làm được điều đó, chúng ta sẽ sử dụng cách cài đặt<sup>1</sup> của [9].



Hình 1. Framework của phương pháp đề xuất

#### A. Kiến trúc Encoder-Decoder trên ảnh hình học

Cho ảnh  $I$  là một hàm ảnh xạ  $\Omega \rightarrow \mathbb{R}^3$  với  $\Omega = \{0, \dots, W-1\} \times \{0, \dots, H-1\}$ . Ta giả định rằng đối tượng được quan tâm trong ảnh  $I$  sẽ nằm ở chính giữa ảnh. Khi này, mục tiêu của chúng ta là cài đặt một hàm học  $\Phi$  cho mạng nơ-ron, sao cho ảnh xạ ảnh  $I$  thành các nhân tố bao gồm:

- Bản đồ chiều sâu  $d : \Omega \rightarrow \mathbb{R}_+$ .
- Ảnh ánh sáng phản chiếu  $a : \Omega \rightarrow \mathbb{R}^3$ .
- Hướng sáng  $l \in \mathbb{S}^2$ .
- Điểm nhìn  $v \in \mathbb{R}^6$ .

Qua đó, ta có thể tái tạo lại ảnh thông qua các yếu tố trên. Quá trình tái tạo ảnh  $I$  gồm 2 bước, *Tạo  $\Lambda$*  và *Lắp  $\Pi$*  theo công thức:

Canonical image  $v = 0$

$$I \approx \hat{I} = \Pi(\Lambda(a, d, l), d, v). \quad (1)$$

Hàm Tạo  $\Lambda$  tạo ra một phiên bản đối tượng dựa trên 3 yếu tố

<sup>1</sup>[https://github.com/hiroharu-kato/neural\\_renderer](https://github.com/hiroharu-kato/neural_renderer)

$a, d, l$  dựa trên điểm nhìn chuẩn (canonical view)  $v = 0$ . Sau đó, hàm Lắp  $\Pi$  sẽ dựa trên điểm nhìn  $v$  và tạo ra ảnh  $\hat{I}$  với yếu tố độ sâu  $d$  và ảnh  $\Lambda(a, d, l)$  đã tạo ở bước đầu. Ở đây, điểm nhìn  $v$  thể hiện sẽ chuyển đổi giữa góc nhìn chuẩn và góc nhìn thực sự của ảnh  $I$ . Để  $I \approx \hat{I}$ , trong quá trình học ta sẽ sử dụng độ lỗi tái tạo.

#### B. Khả năng đối xứng của đối tượng

Trong tác vụ tái tạo 3D, việc tận dụng sự đối xứng của vật thể trong ảnh giúp cải thiện chi phí và kết quả đáng kể. Tuy nhiên điều này đòi hỏi việc xác định những điểm đối xứng của đối tượng. Ở đây, chúng ta giả định rằng độ sâu  $d$  và suất phản xạ  $a$  được tái tạo từ 1 frame tiêu chuẩn, đã được đối xứng qua 1 mặt phẳng đúng cố định. Điều này bên cạnh đó giúp cho mô hình phát hiện ra điểm nhìn chuẩn cho vật thể.

Ta tiến hành lật  $a$  và  $d$  ( $a, d \in \mathbb{R}^{C \times W \times H}$ ) theo chiều ngang sao cho:  $[flip a]_{c,w,h} = a_{c,W-1-w,h}$ . Điều này yêu cầu  $d \approx flip d'$  và  $a \approx flip a'$ . Thay vì áp dụng hàm độ lỗi tương ứng cho việc học này (có thể khá khó để cân bằng), chúng ta chỉ cần thông qua tái tạo một ảnh thứ hai  $\hat{I}'$  từ độ sâu và suất tương phản đã lật trước đó cũng cho ra kết quả tương tự:

$$I \approx \hat{I}' = \Pi(\Lambda(a', d', l), d', v). \quad (2)$$

Reconstruction loss

Bởi vì 2 độ lỗi  $I \approx \hat{I}$  và  $I \approx \hat{I}'$  tương đồng nhau, ta dễ dàng điều chỉnh và huấn luyện cùng nhau. Điều này mang đến cho chúng ta về khả năng đối xứng ảnh. Cụ thể hơn, ảnh gốc  $I$  và ảnh tái tạo  $\hat{I}$  được so sánh thông qua độ lỗi sau:

$$\mathcal{L}(\hat{I}, I, \sigma) = -\frac{1}{|\Omega|} \sum_{w,h \in \Omega} \ln \frac{1}{\sqrt{2}\sigma_{w,h}} \exp -\frac{\sqrt{2}\ell_{1w,h}}{\sigma_{w,h}} \quad (3)$$

Với  $\ell_{1w,h} = |\hat{I}_{w,h} - I_{w,h}|$  là khoảng cách  $L_1$  mật độ giữa pixel tại vị trí  $w$ , và  $\sigma \in \mathbb{R}_+^{W \times H}$  thể hiện cho bản đồ tin cậy (confidence map). Cả 2 công thức trên đều được tính toán bởi mạng  $\Phi$  từ ảnh  $I$ . Độ lỗi ở đây có thể được giải thích như một hàm log-likelihood âm của phân phối Laplace được phân tách trên các sai lệch tái tạo [10].

Trong trường hợp xem xét sự "đối xứng" việc tái tạo  $\hat{I}'$  thì mô hình hoá sự không chắc chắn khá quan trọng vì khi này, ta sử dụng cùng một độ lỗi  $\mathcal{L}(\hat{I}', I, \sigma')$ . Bên cạnh đó, ta dùng mạng nơ-ron nhằm tính toán một bản đồ tin cậy thứ hai, từ cùng một ảnh  $I$  đầu vào. Bản đồ tin cậy này cho phép mô hình học cách phân biệt các thành phần không đối xứng trong ảnh. Chẳng hạn như các bộ phận tóc trên khuôn mặt người sẽ có xu hướng không đối xứng, và  $\sigma'$  sẽ chỉ định ưu tiên tái tạo các vùng không ổn định trên khuôn mặt. Lưu ý rằng điều này phụ thuộc vào các ví dụ cụ thể và có sự quan sát, và được học bởi chính mô hình.

Tổng thể, mục tiêu học của mô hình được cho bởi sự kết hợp giữa hai độ lỗi tái tạo:

$$\varepsilon(\Phi; I) = \mathcal{L}(\hat{I}, I, \sigma) + \lambda \mathcal{L}(\hat{I}', I, \sigma') \quad (4)$$

Trong đó,  $\lambda_f$  là trọng số,  $(d, a, w, l, \sigma, \sigma') = \Phi(I)$  chính là đầu ra của mạng nơ-ron,  $\hat{I}$  và  $I$  được lấy tạo ra qua (1) và (2).

### C. Mô hình tạo ảnh

Chúng ta sẽ đi sâu hơn vào hai công thức (1) và (2). Ta thấy rằng ảnh được hình thành từ việc máy ảnh chiếu vào một vật thể 3D. Ta kí hiệu  $P = P_x, P_y, P_z \in \mathbb{R}^3$  cho một điểm ảnh 3D được thể hiện trong 1 khung ảnh và được ánh xạ đến điểm ảnh  $p = (w, h, 1)$  bởi phép chiếu dưới đây:

$$p \propto KP, K = \begin{bmatrix} f & c_w \\ f & c_h \\ 1 \end{bmatrix}, \quad \begin{cases} c_w = \frac{W-1}{2}, \\ c_h = \frac{H-1}{2}, \\ f = \frac{2W-1}{2 \tanh \frac{\theta_{FOV}}{2}}. \end{cases} \quad (5)$$

Mô hình giả sử một chiếc máy ảnh phôi cảnh với góc độ trường nhắm (FOV)  $\theta_{FOV} \approx 10^\circ$ . Khoảng cách từ  $f$  từ camera đến vật thể là 1m.

Bản đồ chiếu sâu  $d : \Omega \rightarrow \mathbb{R}_+$  tương trưng cho giá trị độ sâu  $d_{w,h}$  cho mỗi pixel  $(w, h) \in \Omega$  với điểm nhìn chuẩn. Bằng cách đảo ngược mô hình máy ảnh, ta nhận ra điều này sẽ tương đương với điểm 3D  $P = d_{w,h} \cdot K^{-1} \cdot p$ .

Điểm nhìn  $v \in \mathbb{R}^6$  sử dụng phép biến đổi Euclide  $(R, T) \in SE(3)$ , với 3 chiều đầu  $v_{1:3}$  là phép xoay và 3 chiều sau  $v_{4:6}$  là phép tịnh tiến dọc theo trục  $x, y$  và  $z$ .

Bản đồ  $(R, T)$  biến đổi các điểm dữ liệu 3D từ điểm nhìn chuẩn thành điểm nhìn thực sự của ảnh. Vì vậy điểm pixel  $(w, h)$  sẽ được ánh xạ đến điểm pixel  $(w', v')$  của điểm nhìn thực sự thông qua một hàm gói  $\eta_{d,v} : (w, h) \rightarrow (w', h')$  được cho bởi:

$$p' \propto K(d_{w,h} \cdot RK^{-1} + T), \quad (6)$$

với  $p' = (w', h', 1)$ .

Sau cùng, hàm chiếu lại  $\Pi$  sẽ lấy đầu vào  $d, v$  và áp dụng kết quả từ công thức (6) để tạo ra ảnh với điểm nhìn thực sự  $\hat{I} = \Pi(J, d, v)$  với  $\hat{I}_{w',h'} = J_{w,h}$ , và  $(u, v) = \eta_{d,v}^{-1}(w', h')$ .

Ảnh chuẩn  $J = \Lambda(a, d, l)$  được tạo ra từ sự kết hợp của suất phản chiếu, ảnh chuẩn hoá và hướng sáng. Đầu tiên, ta cần tìm ảnh chuẩn hoá  $n : \Omega \rightarrow \mathbb{S}^2$  từ bản đồ chiếu sâu  $d$  bằng cách liên kết với mỗi pixel  $(w, h)$  là một vec-tơ chuẩn hoá nằm dưới bề mặt 3D. Ta tính vec-tơ tiếp tuyến  $t_{w,h}^w v t_{w,h}^h$  với bề mặt dọc theo hướng của  $w, h$ . Cụ thể,  $t_{w,h}^w = d_{w+1,h} \cdot K^{-1}(p + e_x) - d_{w-1,h} \cdot K^{-1}(p - e_x)$  với  $p$  được định nghĩa ở (5) và (6) và  $e_x = 1, 0, 0$ . Sau cùng, ta tính tích 2 vec-tơ tiếp tuyến  $n_{w,h} \propto t_{w,h}^w \times t_{w,h}^h$ .  $n_{w,h}$  sẽ được nhân với hướng sáng  $l$ , rồi nhân tiếp với độ phản chiếu  $a$  theo công thức:

$$J_{w,h} = (k_s + k_d \max\{0, \langle l, n_{w,h} \rangle\}) \cdot a_{w,h} \quad (7)$$

Ở đây,  $k_s$  và  $k_d$  là những trọng số vô hướng đối với các môi trường xung quanh và khuếch tán, được dự đoán bởi mô hình trong khoảng từ 0 đến 1 thông qua việc thay đổi đầu ra tanh.

Hướng sáng  $l = \frac{(l_x, l_y, 1)^T}{(l_x^2 + l_y^2 + 1)^{\frac{1}{2}}}$  được mô hình hoá như một khối cầu bằng việc dự tính  $l_x$  và  $l_y$  với hàm tanh.

### D. Độ lỗi perceptual

Hàm lỗi  $L_1$  (3) tương đối nhạy cảm đối với vùng địa nhỏ không hoàn hảo và cho các tái tạo mờ. Vì vậy, ta thêm "độ lỗi perceptual" (Perceptual loss) để giải quyết vấn đề này. Tại lớp thứ  $k$  của bộ encoder ảnh có sẵn (VGG16 trong trường hợp này [12]) dự đoán  $e^{(k)}(I) \in \mathbb{R}^{C_k \times W_k \times H_k}$  với  $\Omega_k = \{0, \dots, W_k - 1\} \times \{0, \dots, H_k - 1\}$  thể hiện miền không gian. Lưu ý rằng các đặc trưng encoder không nhất thiết phải được train với các tác vụ học có giám sát. Tương tự với (3), ta giả định với phân phối Gauss, độ lỗi perceptual được cho như sau:

$$\mathcal{L}_p^{(k)}(\hat{I}, I, \sigma) = -\frac{1}{|\Omega|} \sum_{w,h \in \Omega} \ln \frac{1}{\sqrt{2\pi(\sigma_{w,h}^{(k)})^2}} \exp -\frac{(\ell_{w,h}^{(k)})^2}{2\sigma_{w,h}^{(k)}} \quad (8)$$

trong đó,  $\ell_{w,h}^{(k)} = |e_{w,h}^{(k)}(\hat{I}) - e_{w,h}^{(k)}(I)|$  cho mỗi đơn vị pixel  $w, h$  trong lớp thứ  $k$ . Ta tính độ lỗi cho  $\hat{I}'$  sử dụng  $\sigma^{(k)'}.$   $\sigma^{(k)'}$  và  $\sigma^{(k)}$  là các conf. map được dự đoán bởi mô hình. Trong thực tế, chúng ta có thể thấy rằng điều này hiệu quả để sử dụng các đặc trưng chỉ từ một lớp *relu3\_3* của kiến trúc VGG16.

### E. Cải tiến

Tại thời điểm mô hình của paper gốc [8] được xây dựng differentiable renderer được sử dụng là Neural 3D Mesh Renderer [9]. Renderer này có thể xấp xỉ gradient cho thao tác rasterization từ đó có thể render ảnh 3D. Mặc dù renderer này hỗ trợ đầy đủ các thao tác cần thiết cho mô hình tái tạo ảnh 3D, nhưng vì tác giả không còn cập nhật mã nguồn sau thời điểm xuất bản nên các lỗi về không tương thích phiên bản xuất hiện khi cài đặt các phiên bản PyTorch mới sau này. Vì lý do đó, chúng tôi đã sử dụng một renderer được cộng đồng duy trì và cải tiến thường xuyên là renderer của PyTorch3D [15]. PyTorch3D hỗ trợ nhiều đối tượng như: Camera, Light, Renderer, Rasterizer, Shader được đóng gói thành các đối tượng dễ sử dụng và thao tác. Ngoài ra việc huấn luyện bằng PyTorch3D cũng cho thấy tốc độ nhanh hơn so với Neural 3D Mesh Renderer.

## IV. THỰC NGHIỆM

**Bộ dữ liệu:** Chúng ta thử nghiệm mô hình trên bộ **CelebA** [11]. **CelebA** là một bộ dữ liệu khuôn mặt người quy mô lớn, bao gồm hơn 200.000 hình ảnh của mặt người trong môi trường tự nhiên với 66 chủ thích điểm keypoints, ta dùng các keypoints để đánh giá các dự đoán 3D. Chúng tôi chia dữ liệu theo tỉ lệ 8 : 1 : 1 cho các tập train : dev : test.

**Độ đo:** Vì quy mô của việc tái tạo 3D vốn đã khá mờ hồ [14], chúng tôi đã nối lỏng việc đánh giá lại một ít. Cụ thể hơn, với bản đồ chiếu sâu  $d$  được dự đoán bởi mô hình với tầm nhìn chuẩn, ta gói nó lại thành  $\bar{d}$  ở điểm nhìn thực sự sử dụng điểm nhìn dự đoán và so sánh cái sau với bản đồ độ sâu ground-truth  $d^*$  thông qua **scale-invariant depth error** (SIDE) [13]:

$$ESIDE(\bar{d}, d^*) = (\frac{1}{WH} \sum_{w,h} \Delta_{w,h}^2 - (\frac{1}{WH} \sum_{w,h} \Delta_{w,h})^2)^{\frac{1}{2}} \quad (9)$$

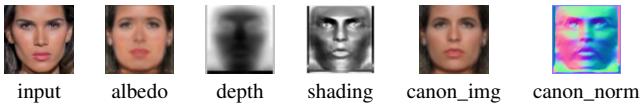
với  $\Delta_{w,h} = \log d_{w,h} - \log d_{w,h}^*$ . Chúng ta chỉ so sánh các pixel chiều sâu khả thi và làm xói mòn mặt nạ foreground bởi 1 pixel để giảm thiểu việc hiển thị các tạo tác tại biên độ vật thể.

## V. KẾT QUẢ

### A. So sánh với các phương pháp gốc

Đối với phương pháp cải tiến, mô hình được huấn luyện nhanh hơn gấp 3,4 lần mô hình gốc. Tuy nhiên, trong mô hình gốc, tác giả đã huấn luyện qua 30 epochs. Đối với phương pháp cải tiến, tuy có phần nhỉnh hơn về mặt thời gian nhưng nhóm chỉ chạy được 14 epochs. Do đó kết quả tuy về phần hình dạng chưa được ổn định nhưng về các đặc trưng của ảnh vẫn được giữ dưới dạng 3D.

Bảng I  
CÁC ĐẶC TRUNG ĐẦU RA



Bảng II  
KẾT QUẢ SAU 5 EPOCHS



Bảng III  
KẾT QUẢ CỦA NHÓM



### B. Hạn chế

Mặc dù mô hình đã được cải thiện về thời gian huấn luyện, nhóm vẫn cần thêm thời gian để kết quả đầu ra ổn định nhất có thể.

## VI. KẾT LUẬN

Chúng tôi đã đề xuất 1 phương pháp mà có thể học mô hình 3D cho các vật thể biến dạng từ tập dữ liệu ảnh dưới 1 góc nhìn. Mô hình này có khả năng học các vật thể cụ thể.

Bảng IV  
KẾT QUẢ CỦA PHƯƠNG PHÁP GỐC



Việc huấn luyện này đều dựa trên độ lỗi tái tạo mà không cần sự giám sát nào, giống như một bộ mã hoá tự động. Bên cạnh đó, chúng tôi đã chỉ ra các thành phần đối xứng và độ sáng có tác động lớn đến hình dáng và giúp mô hình quy tụ về sự tái tạo có ý nghĩa. Đối với hướng cải tiến trong tương lai, các hạn chế như việc mô hình học từ các khuôn mặt đơn giản và điểm nhìn tự nhiên, thay vào đó, ta có thể xây dựng một mô hình áp dụng được cho các mặt bị che 1 phần phức tạp hơn với nhiều điểm nhìn khác nhau hơn.

## TÀI LIỆU

- [1] Zhang, Ruo, et al. "Shape-from-shading: a survey." IEEE transactions on pattern analysis and machine intelligence 21.8 (1999): 690-706.
- [2] Horn, Berthold KP. "Shape from shading: A method for obtaining the shape of a smooth opaque object from one view." (1970)
- [3] Sengupta, Soumyadip, et al. "Sfsnet: Learning shape, reflectance and illuminance of faces in the wild". Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.
- [4] Kemelmacher-Shlizerman, Ira, and Ronen Basri. "3D face reconstruction from a single image using a single reference face shape." IEEE transactions on pattern analysis and machine intelligence 33.2 (2010): 394-405.
- [5] Jiang, Luo, et al. "3D face reconstruction with geometry details from a single image." IEEE Transactions on Image Processing 27.10 (2018): 4756-4770.
- [6] Chen, Yajing, et al. "Self-supervised learning of detailed 3d face reconstruction." IEEE Transactions on Image Processing 29 (2020): 8696-8705.
- [7] Deng, Yu, et al. "Accurate 3d face reconstruction with weakly-supervised learning: From single image to image set." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops. 2019.
- [8] Wu, S., Rupprecht, C., Vedaldi, A.: Unsupervised Learning of Probably Symmetric Deformable 3D Objects from Images in the Wild. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1-10. (2020).
- [9] Kato, Hiroharu, Yoshitaka Ushiku, and Tatsuya Harada. "Neural 3d mesh renderer." Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.
- [10] Kendall, Alex, and Yarin Gal. "What uncertainties do we need in bayesian deep learning for computer vision?." Advances in neural information processing systems 30 (2017).
- [11] Liu, Ziwei, et al. "Deep learning face attributes in the wild." Proceedings of the IEEE international conference on computer vision. 2015.
- [12] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014).
- [13] Eigen, David, Christian Puhrsch, and Rob Fergus. "Depth map prediction from a single image using a multi-scale deep network." Advances in neural information processing systems 27 (2014).

- [14] Luong, Quang-Tuan, and O. D. Faugeras. "The geometry of multiple images." MIT Press, Boston 2.3 (2001): 4-5.
- [15] Ravi, N., Reizenstein, J., Novotny, D., Gordon, T., Lo, W.-Y., Johnson, J., & Gkioxari, G. (2020). Accelerating 3D Deep Learning with PyTorch3D. arXiv preprint arXiv:2007.08501.