

OPENHUMAN: HỆ THỐNG TỔNG HỢP CỦ CHỈ HỘI THOẠI DỰA TRÊN CẢM XÚC VÀ NGỮ NGHĨA

**OpenHuman: A conversational gesture synthesis system
based on emotions and semantics**

Thanh Hoang-Minh

Giảng viên hướng dẫn:
PGS. TS. Lý Quốc Ngọc



Faculty of Information Technology
VNUHCM-University of Science

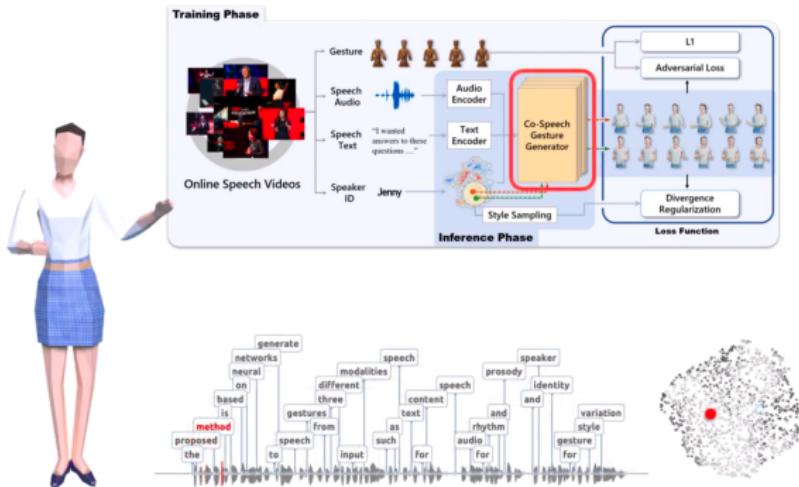
December 17, 2024

Outline

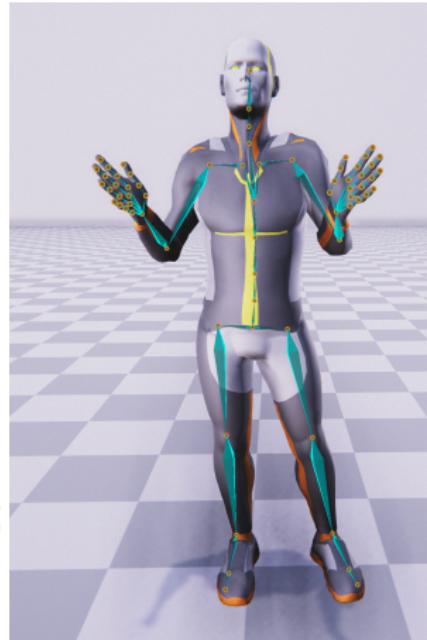
- ① Minh họa
- ② Giới thiệu
- ③ Công trình liên quan
- ④ Mô hình sinh cử chỉ
- ⑤ Thực nghiệm
- ⑥ Kết luận

Minh họa quá trình sinh cử chỉ

- Sinh cử chỉ (Gesture Generation) là gì?



ACM CCS: • Human-centered computing →
Human computer interaction (HCI).



Demo Gesture
Generation: [Youtube](#)

Outline

1 Minh họa

2 Giới thiệu

3 Công trình liên quan

4 Mô hình sinh cử chỉ

5 Thực nghiệm

6 Kết luận

Động lực nghiên cứu

Text-base Deep Learning

- ChatGPT, Alexa, Character.AI,..
- Text to speech, text to text,..

Text/audio to Realistic Digital Human

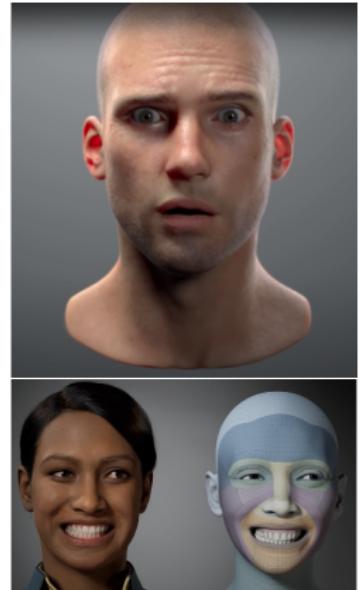
- Video-base (HeyGen, Midjourney,...)
- Rendering-base:
 - Character model (Gaussian Splatting)
 - Character animation: (3D keypoints)



(a)



(b)

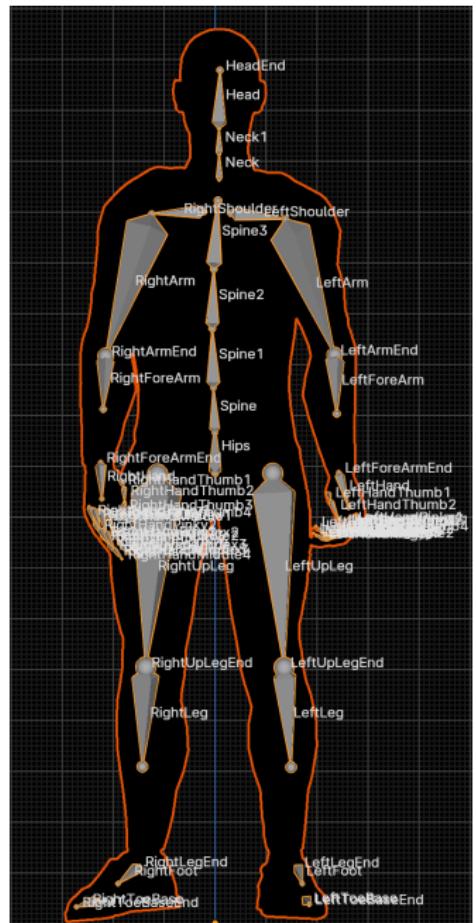


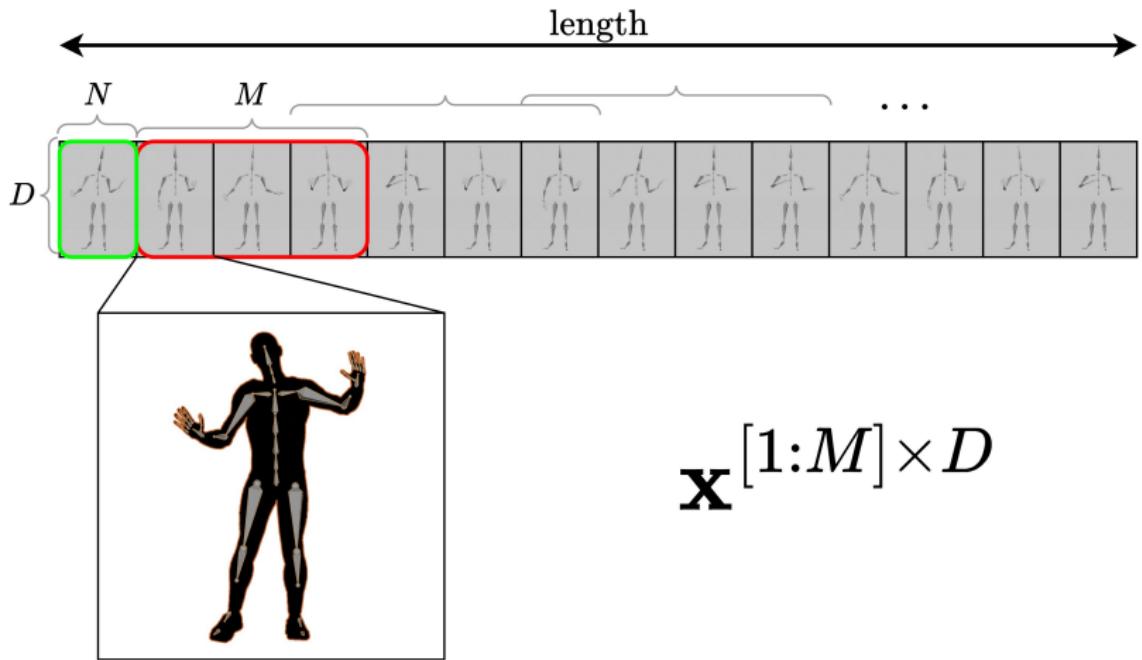
Minh họa người kỹ
thuật số siêu thật
(realistic digital human)

$$\{\mathbf{b}_1, \dots, \mathbf{b}_{75}\}$$



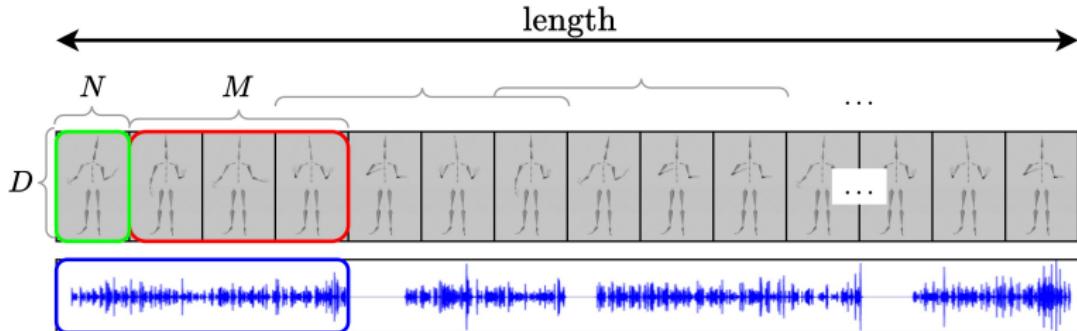
$$\mathbf{b}_i = \{p_x, p_y, p_z, r_x, r_y, r_z\}$$





$$\mathbf{g}^{[1:length] \times 75 \times 6} \xrightarrow{\text{Xử lý}} \mathbf{g}^{[1:length] \times D} \xrightarrow{\text{Cắt}} \mathbf{g}^{[N+M] \times D}$$

Phát biểu bài toán



Input

- Chuỗi cử chỉ khởi tạo: $s \in \mathbb{R}^{[1:N] \times D}$
- Chuỗi âm thanh: a
- Văn bản: v
- Cảm xúc: e
(Happy, Sad, Neutral, Angry, Old, Funny)

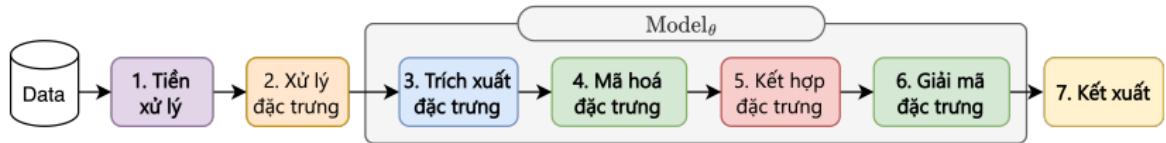
Output

- Chuỗi cử chỉ dự đoán:
 $\hat{x} \in \mathbb{R}^{[1:M] \times D}$

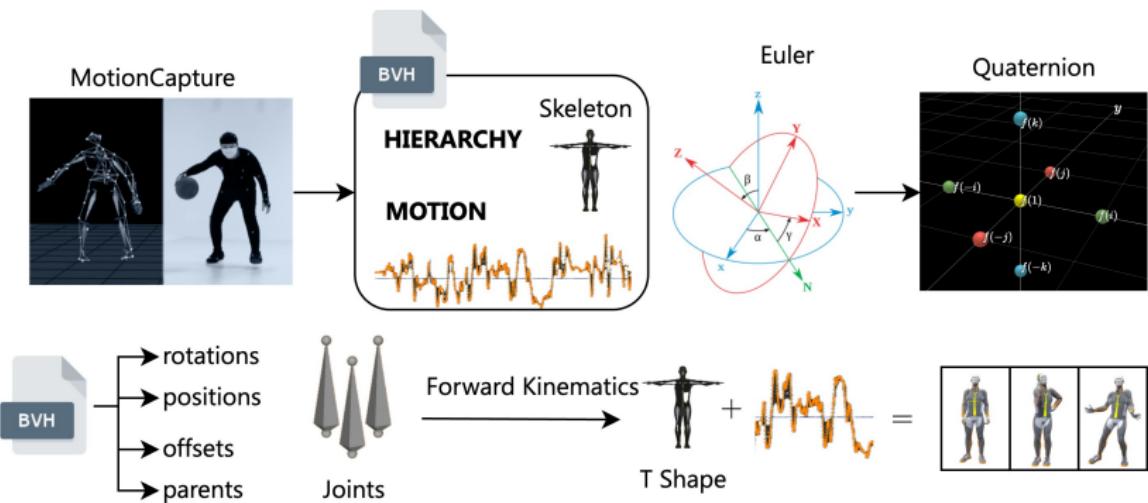
Growth Truth

- Chuỗi cử chỉ gốc:
 $x \in \mathbb{R}^{[1:M] \times D}$

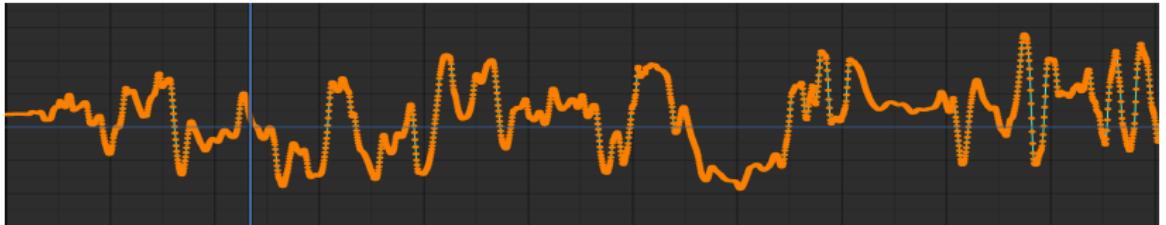
Các công đoạn



Công đoạn 1: Tiền xử lý dữ liệu



Đặc trưng dữ liệu của 1 frame



$$\mathbf{g}^{[1]} = \left[\mathbf{p}_{\text{root}}, \mathbf{r}_{\text{root}}, \mathbf{p}'_{\text{root}}, \mathbf{r}'_{\text{root}}, \mathbf{p}_{\text{joins}}, \mathbf{r}_{\text{joins}}, \mathbf{p}'_{\text{joins}}, \mathbf{r}'_{\text{joins}}, \mathbf{d}_{\text{gaze}} \right] \quad (1)$$

- $\mathbf{p}_{\text{root}} \in \mathbb{R}^3$, $\mathbf{r}_{\text{root}} \in \mathbb{R}^4$: Toạ độ và góc quay của điểm gốc
- $\mathbf{p}'_{\text{root}} \in \mathbb{R}^3$, $\mathbf{r}'_{\text{root}} \in \mathbb{R}^3$: Vận tốc thay đổi của toạ độ và góc quay gốc
- $\mathbf{p}_{\text{joins}} \in \mathbb{R}^{3n_{\text{join}}}$, $\mathbf{r}_{\text{joins}} \in \mathbb{R}^{6n_{\text{join}}}$: Toạ độ và góc quay của các khung xương
- $\mathbf{p}'_{\text{joins}} \in \mathbb{R}^{3n_{\text{join}}}$ và $\mathbf{r}'_{\text{joins}} \in \mathbb{R}^{3n_{\text{join}}}$: Vận tốc thay đổi của toạ độ và góc quay khung xương
- $\mathbf{d}_{\text{gaze}} \in \mathbb{R}^3$: Là hướng nhìn

$$D = 3 + 4 + 3 + 3 + 3 \times 75 + 6 \times 75 + 3 \times 75 + 3 \times 75 + 3 = 1141$$

Thách thức của bài toán

Quan hệ giữa dữ liệu cử chỉ và âm thanh:

- Quan hệ n:n của âm thanh, cử chỉ, cảm xúc.

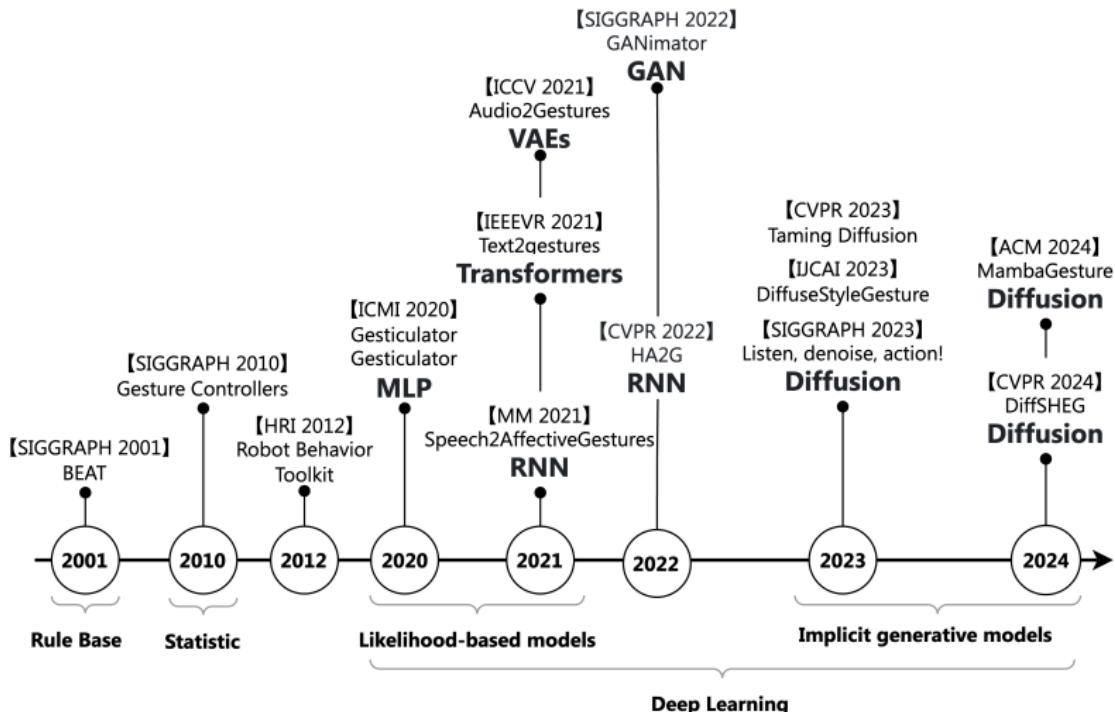
Khó khăn

- ➊ Dữ liệu không đủ nhiều và chất lượng, chi phí cao
- ➋ Không cân xứng về dữ liệu giữa các loại đăng trưng cần học (âm thanh, cử chỉ)
- ➌ Thiếu đồng nhất của về ngữ cảnh của các loại dữ liệu

Outline

- ① Minh họa
- ② Giới thiệu
- ③ Công trình liên quan
- ④ Mô hình sinh cử chỉ
- ⑤ Thực nghiệm
- ⑥ Kết luận

Các phương pháp cho bài toán sinh cử chỉ



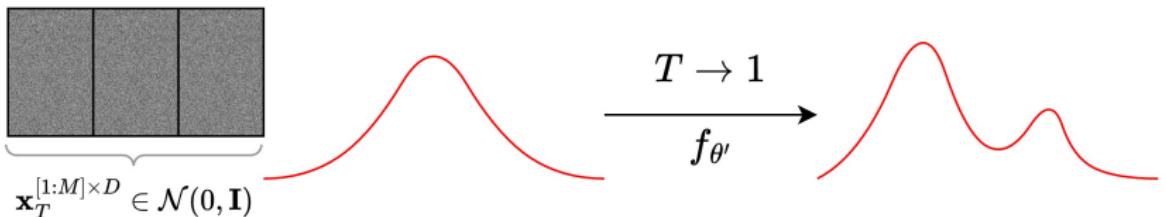
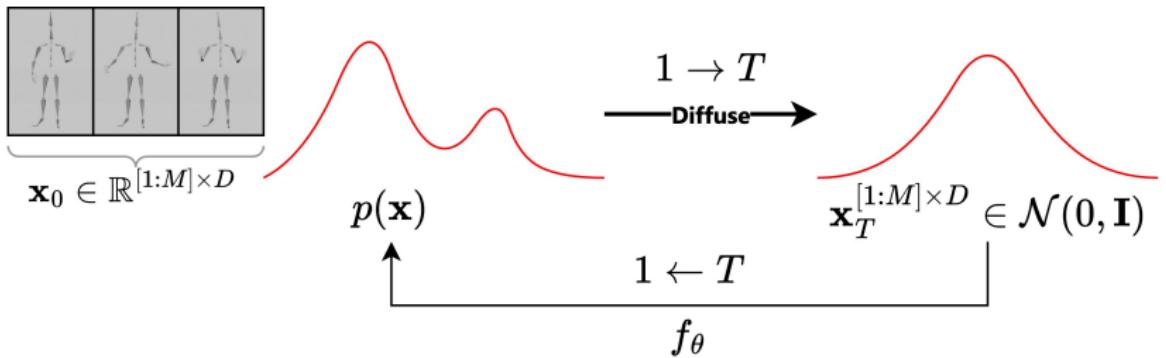
Bảng so sánh các phương pháp

Phương pháp tiêu biểu	Loại	Nguyên lý
Robot behavior toolkit	Rule-Based	Tạo một tập hợp các đơn vị cử chỉ và thiết kế các Rule cụ thể để ánh xạ lời nói thành một chuỗi các đơn vị cử chỉ.
Gesture Controller	Statistical Models	Sử dụng mô hình xác suất như Hidden Markov Models (HMM) để mô hình hóa phân phối cử chỉ bằng cách phân tích các thống kê.
Gesticulator(MLP), HA2G (RNN), Text2Gestures (Transformers)	Likelihood-Based	Sử dụng neuron network (MLP, RNN, Transformers) để học end-to-end quan hệ của cử chỉ và âm thanh. Coi việc sinh cử chỉ như là một quá trình tái tạo (reconstruction) dữ liệu cử chỉ ban đầu.
Diffusion: MDM, Listen, Denoise, Action DiffuseStyleGesture	Implicit Generative	Học cách chuyển từ phân phối chuẩn về phân phối của dữ liệu để ngầm định việc học phân phối của dữ liệu, tạo ra sự đa dạng trong việc sinh cử chỉ.

Đánh giá các phương pháp

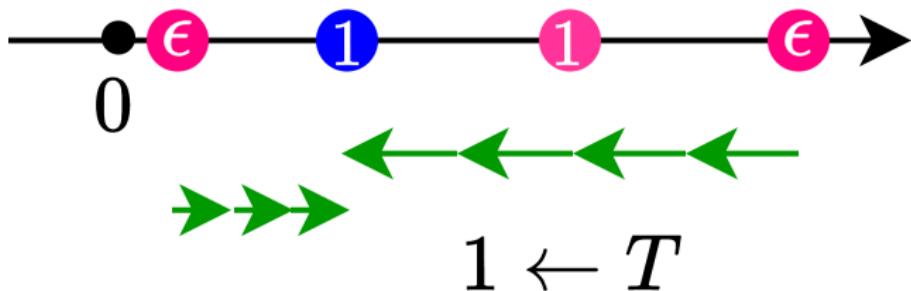
Loại phương pháp	Ưu điểm	Nhược điểm
Rule-Based & Statistical	<ul style="list-style-type: none">- Dễ hiểu và dễ triển khai.- Dễ giải thích và có thể kiểm soát được- Hiệu quả trong các trường hợp đơn giản hoặc dữ liệu nhỏ.	<ul style="list-style-type: none">- Không tổng quát hoá tốt với dữ liệu phức tạp.- Đòi hỏi nhiều công sức trong việc xây dựng quy tắc thủ công.
Likelihood-Based Models	<ul style="list-style-type: none">- Khả năng ước lượng mật độ xác suất của dữ liệu- Có khả năng mở rộng và học từ dữ liệu lớn	<ul style="list-style-type: none">- Dễ bị ảnh hưởng bởi nhiễu- Kết quả thấp ở vùng dữ liệu hiếm- Khả năng sinh không đa dạng
Implicit Generative Models	<ul style="list-style-type: none">- Tạo ra dữ liệu chất lượng cao.- Linh hoạt và đa dạng- Phù được vùng có mật độ dữ liệu thấp	<ul style="list-style-type: none">- Cần cấu hình phức tạp để đạt hiệu năng tốt.- Khó đánh giá do mỗi lần nhiễu là khác nhau.- Quá trình lấy mẫu chậm

Mô hình diffusion



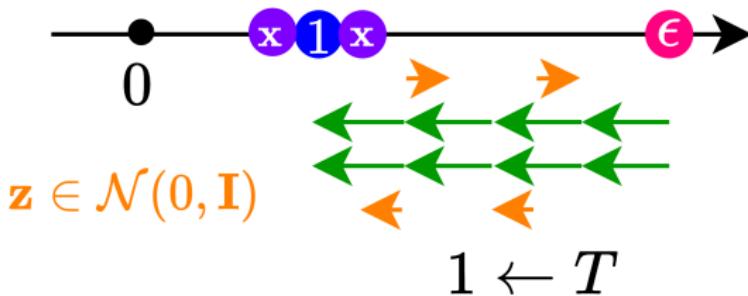
Quá trình huấn luyện (offline)

Drift $\nabla_{\mathbf{x}} \log p(\mathbf{x})$



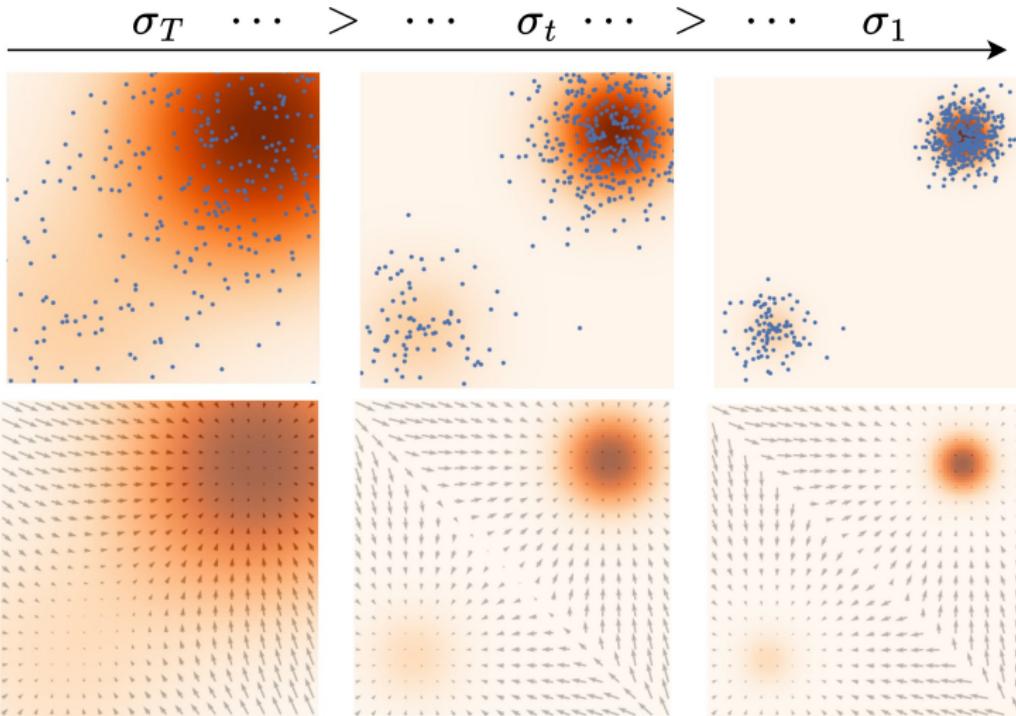
Quá trình suy luận (online)

$$\mathbf{x}_{\text{sample}} = \nabla_{\mathbf{x}} \log p(\mathbf{x}) + \sigma_t \cdot \mathbf{z}$$



Điều khiển σ_t bằng với Langevin dynamics

$T \rightarrow 1 : \sigma_T > \dots > \sigma_2 > \sigma_1$, σ_t sẽ giảm dần nhiều

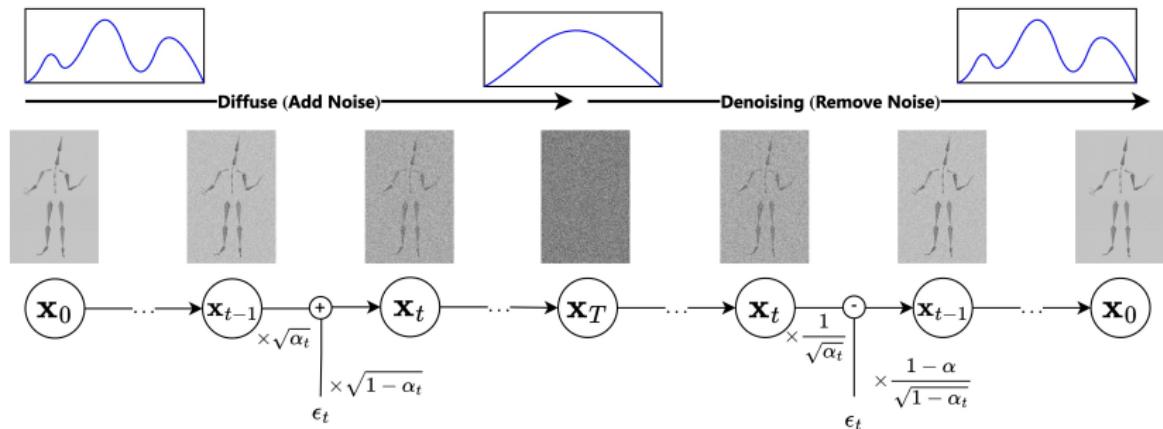


Gây nhiễu và khử nhiễu không huấn luyện

Diffuse (Thêm nhiễu): $\sqrt{\alpha_t}$ giảm dần, $\sqrt{1 - \alpha_t}$ tăng dần.

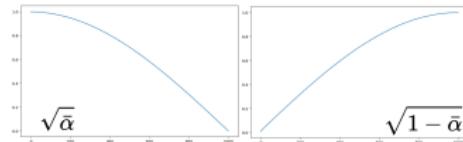
Nhiễu: $\epsilon_0, \epsilon_1, \dots, \epsilon_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, nhiễu ϵ_t cố định.

$$\mathbf{x}_t = \sqrt{\alpha_t} \cdot \mathbf{x}_{t-1} + \sqrt{1 - \alpha_t} \cdot \epsilon \quad (2)$$



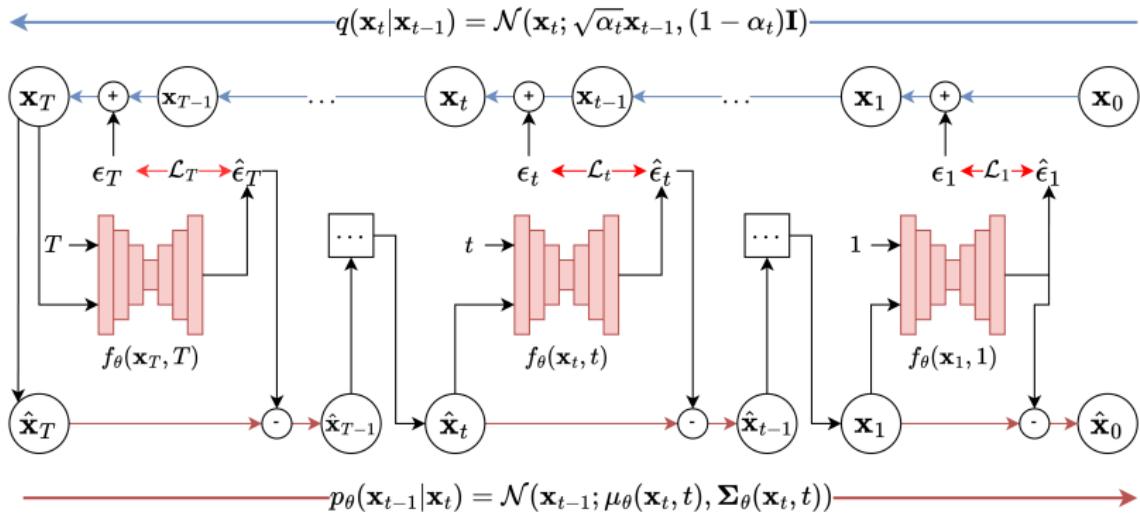
Denoise (Khử nhiễu)

$$\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}} (\mathbf{x}_t - \sqrt{1 - \alpha_t} \cdot \epsilon) \quad (3)$$



Quá trình học Offline (Training DDPM)

Hàm loss: $\mathcal{L} = \sum_{t=1}^T \mathcal{L}_t$. $\mathcal{L}_t = \mathbb{E}_{\mathbf{x}_0, \epsilon_t \sim \mathcal{N}(0, I), t} [\|\epsilon_t - \epsilon_\theta(\mathbf{x}_t, t)\|^2]$

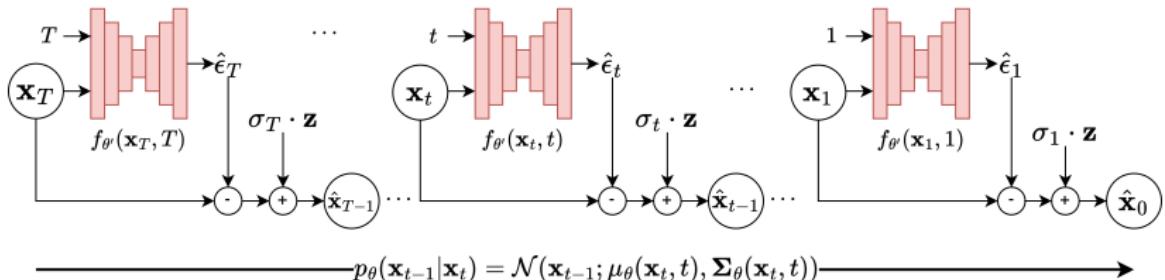


Quá trình học Online (Inference DDPM)

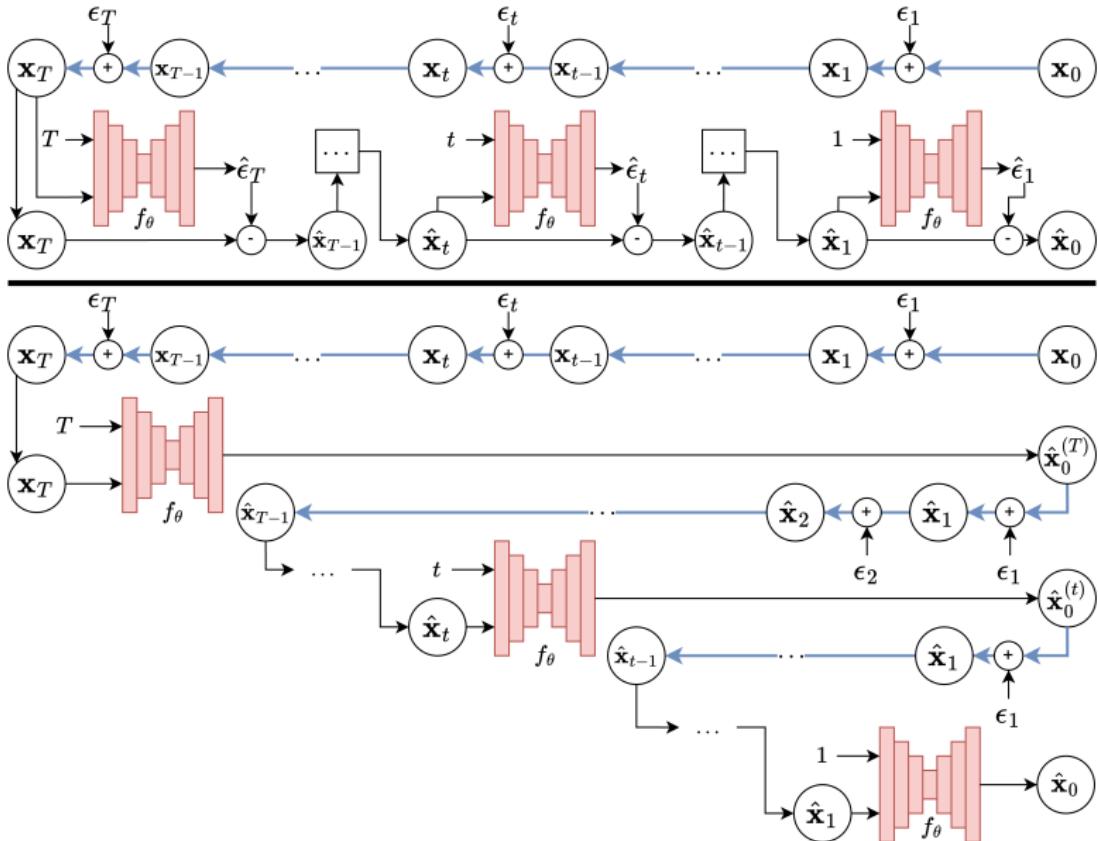
Hàm sampling:

- $\mathbf{x}_T \in \mathcal{N}(0, \mathbf{I})$

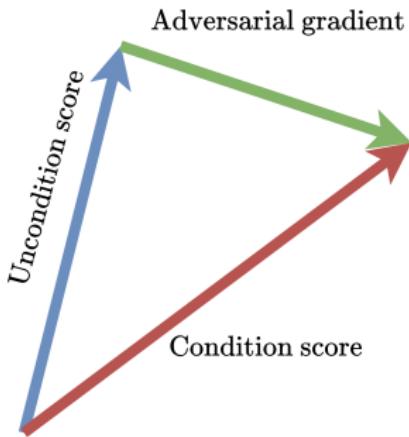
$$\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}} \hat{\epsilon}_{\theta'}(\mathbf{x}_t, t) \right) + \sigma_t \mathbf{z}$$



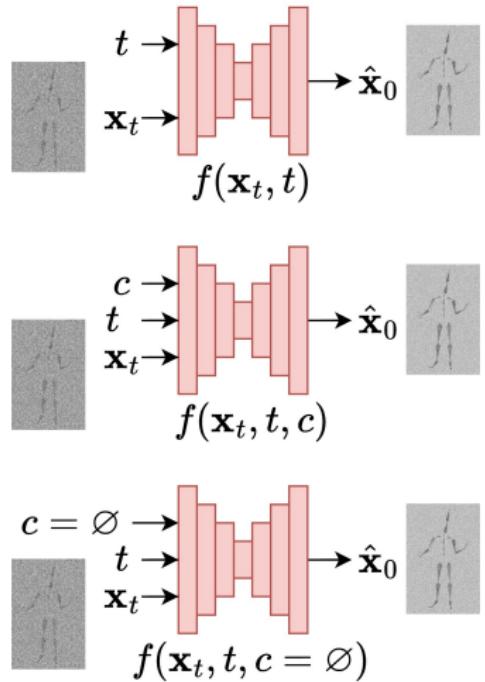
So sánh ϵ objective và x_0 objective



Classifier-Free Guidance



$$\hat{\mathbf{x}}_{0,\gamma,c} = \gamma \cdot f_\theta(\mathbf{x}_t, t, c) + (1 - \gamma) \cdot f_\theta(\mathbf{x}_t, t, c_\emptyset)$$



Outline

- ① Minh họa
- ② Giới thiệu
- ③ Công trình liên quan
- ④ Mô hình sinh cử chỉ
- ⑤ Thực nghiệm
- ⑥ Kết luận

Diffusion cho bài toán sinh cử chỉ

Điểm giống

- Diffusion [3] trên cử chỉ $\mathbf{x}^{1:M \times D}$ (như width và height trong ảnh).
 - M : frame theo thời gian
 - $D = 1141$: đặc trưng khung xương của từng khung hình
- Classifier-Free Diffusion với \mathbf{x}_0 objective. Latent vector 256.

Diffusion cho bài toán sinh cử chỉ

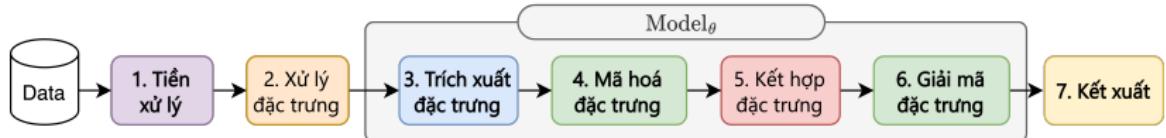
Điểm giống

- Diffusion [3] trên cử chỉ $\mathbf{x}^{1:M \times D}$ (như width và height trong ảnh).
 - M : frame theo thời gian
 - $D = 1141$: đặc trưng khung xương của từng khung hình
- Classifier-Free Diffusion với \mathbf{x}_0 objective. Latent vector 256.

Điểm khác

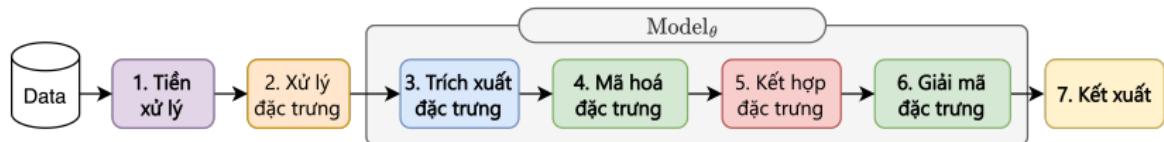
- Sinh có điều kiện:
 - Điều kiện cảm xúc: $c = [\mathbf{s}, \mathbf{e}, \mathbf{a}, \mathbf{v}]$ và $c_\emptyset = [\emptyset, \emptyset, \mathbf{a}, \mathbf{v}]$.
 - Nội suy trạng thái giữa hai cảm xúc $\mathbf{e}_1, \mathbf{e}_2$: $c = [\mathbf{s}, \mathbf{e}_1, \mathbf{a}, \mathbf{v}]$ và $c_\emptyset = [\mathbf{s}, \mathbf{e}_2, \mathbf{a}, \mathbf{v}]$.
- Self-Attention: Mỗi liên hệ giữa các cảm xúc, cử chỉ khởi tạo và từng frame (tương tự DALL-E 2 - mối liên hệ giữa văn bản và ảnh).
- Concat âm thanh và văn bản (Giống ControlNet)

Công đoạn 2: Xử lý đặc trưng



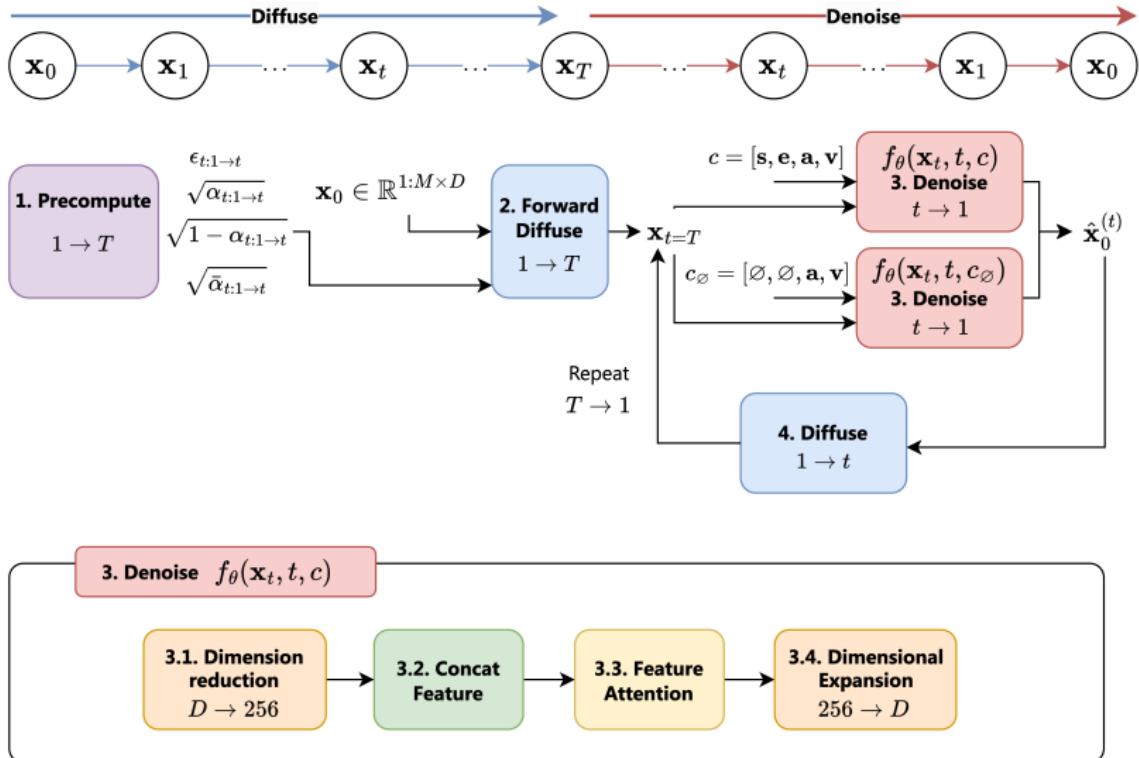
- **Cử chỉ:** Chuẩn hoá dữ liệu. $\mathbf{g}^{\text{length}} \xrightarrow{\text{Cắt thành đoạn}} \mathbf{g}^{[N+M] \times D}$
 - Cử chỉ khởi tạo: $\mathbf{s} \in \mathbb{R}^{[1:N] \times D}$
 - Cử chỉ ground truth: $\mathbf{x} \in \mathbb{R}^{[1:M] \times D}$
- **Âm thanh:** $\mathbf{a}^{\text{length}} \xrightarrow{\text{Cắt thành đoạn}} \mathbf{a}^{64000} \xrightarrow{\text{WavLM}} \mathbf{a} \in \mathbb{R}^{[1:M] \times 1024}$
- **Cảm xúc:** Nhãn "Happy" $\rightarrow \mathbf{e} \in \mathbb{R}^6$
- **Văn bản:** $\mathbf{v}^{\text{length}} \xrightarrow{\text{Cắt theo TextGrid}} \mathbf{v}^{[1:M]} \xrightarrow{\text{FastText}} \mathbf{v} \in \mathbb{R}^{[1:M] \times 300}$

Công đoạn 3. Trích xuất các đặc trưng

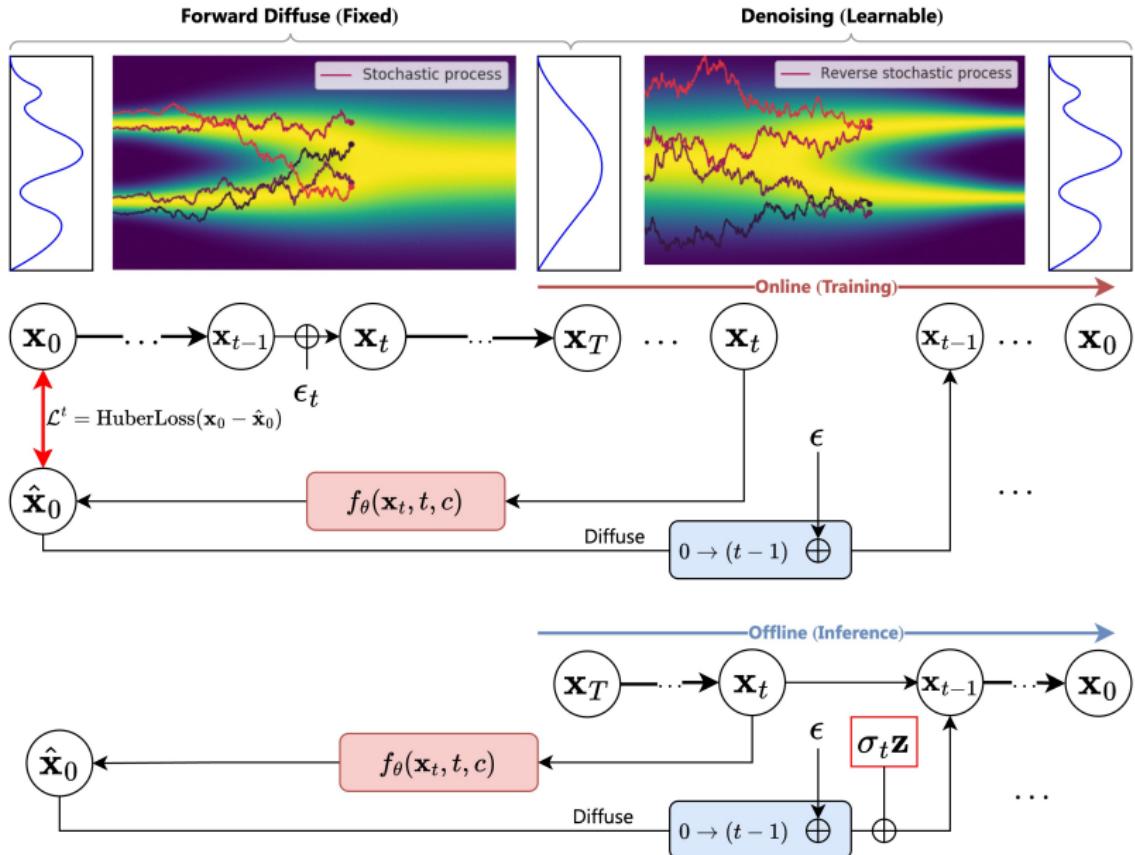


- Âm thanh: $a \in \mathbb{R}^{[1:M] \times 1024} \xrightarrow{\text{Linear}} A \in \mathbb{R}^{[1:M] \times 64}$
- Cử chỉ khởi tạo: $s \in \mathbb{R}^{[1:N] \times D} \xrightarrow{\text{Linear + Random Mask}} S \in \mathbb{R}^{192}$
- Cảm xúc: $e \in \mathbb{R}^6 \xrightarrow{\text{Linear + Random Mask}} E \in \mathbb{R}^{64}$
- Văn bản: $v \in \mathbb{R}^{[1:M] \times 300} \xrightarrow{\text{Linear}} V \in \mathbb{R}^{256}$
- Timestep: $t \in \mathbb{R} \xrightarrow{\text{Linear}} T \in \mathbb{R}^{256}$

Công đoạn 4, 6: Mã hoá và giải mã đặc trưng

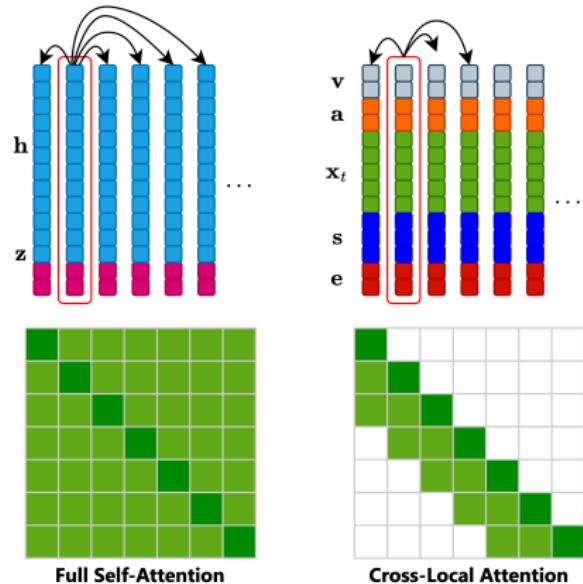


Quá trình học Online và Offline

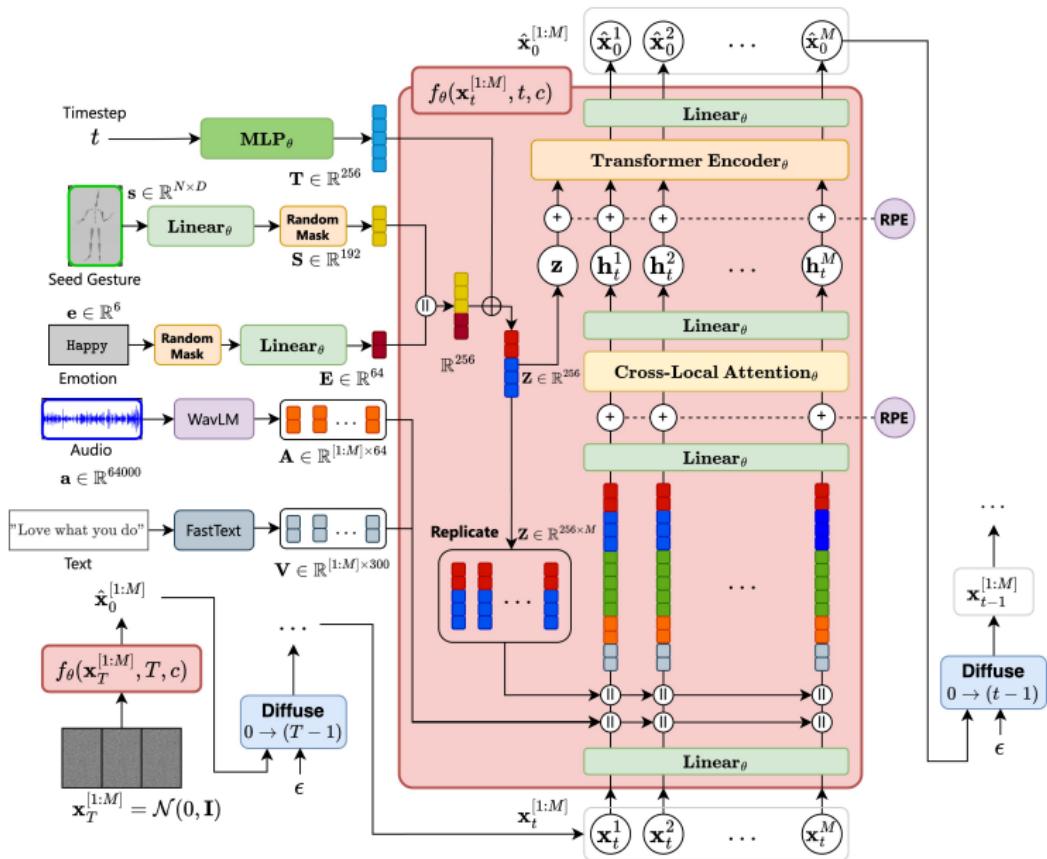


Công đoạn 5: Kết hợp các đặc trưng

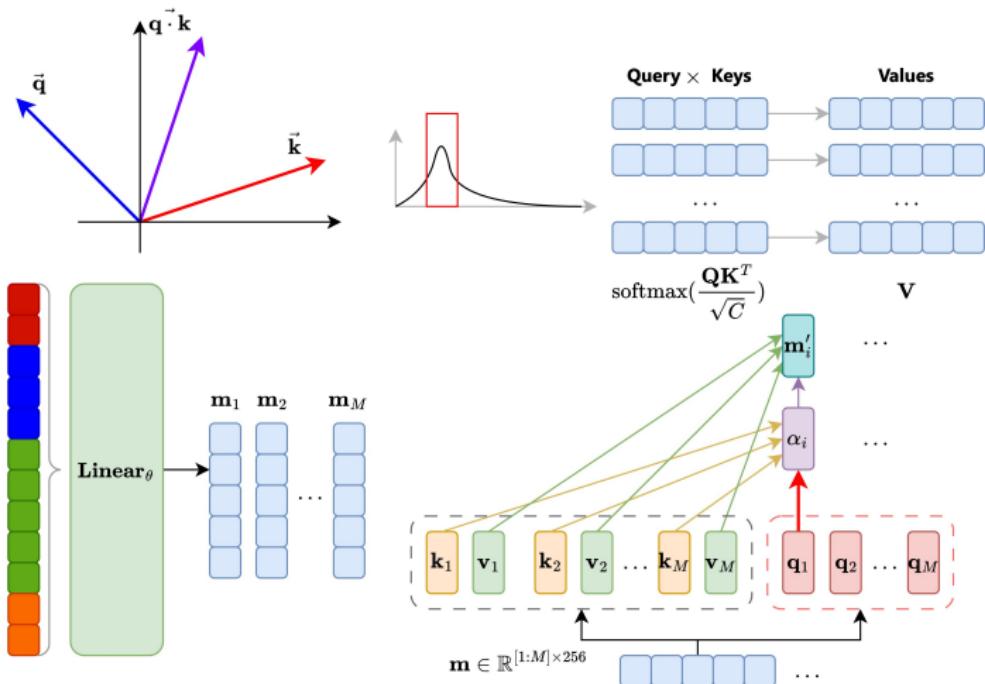
- Concat cảm xúc E và cử chỉ khởi tạo S để được vector latent $Z \in \mathbb{R}^{256}$
- Cộng vector Z với vector timestep T để được vector Z và replicate Z M lần để được $Z \in \mathbb{R}^{[1:M] \times 256}$
- Concat các đặc trưng âm thanh A , văn bản V , cử chỉ x_t và Z để được vector đặc trưng theo từng frame.
- Thực hiện Cross-Local Attention trên vector đặc trưng từng frame để được đặc trưng $h^{[1:M]}$.
- Sử dụng Z làm CLS token cho $h^{[1:M]}$ trước khi đi qua Transformer Encoder.



Mô hình OHGesture



Giải thích cơ chế Attention



Emotion-controllable Gesture Generation

Classifier-free guidance

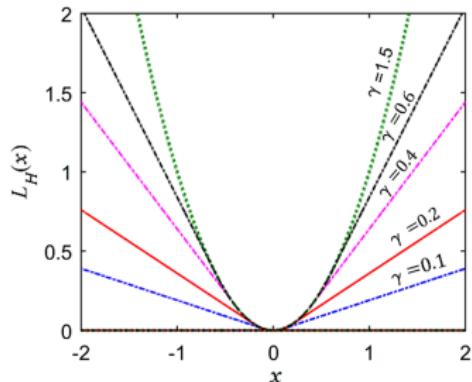
- Điều khiển cảm xúc (Emotion-controllable) Conditional:
 $c_1 = [\mathbf{s}, \mathbf{e}, \mathbf{a}, \mathbf{v}]$, Unconditional: $c_2 = [\emptyset, \emptyset, \mathbf{a}, \mathbf{v}]$
- Nội suy cảm xúc (Emotion-interpolating) Unconditional:
 $c_1 = [\mathbf{s}, \mathbf{e}_1, \mathbf{a}, \mathbf{v}]$, Conditional: $c_2 = [\mathbf{s}, \mathbf{e}_2, \mathbf{a}, \mathbf{v}]$

$$\hat{\mathbf{x}}_{0\gamma, c_1, c_2} = \gamma \cdot f_\theta(\mathbf{x}_t, t, c_1) + (1 - \gamma) \cdot f_\theta(\mathbf{x}_t, t, c_2) \quad (4)$$

Huber Loss

$$\mathcal{L} = E_{\mathbf{x}_0 \sim q(\mathbf{x}_0 | c), t \sim [1, T]} [\text{HuberLoss}(\mathbf{x}_0 - \hat{\mathbf{x}}_0)] \quad (5)$$

- $|y - f(x)| \leq \delta$: Mượt mà và dễ tối ưu hóa khi các lỗi nhỏ.
 $\mathcal{L}_\delta(y, f(x)) = \frac{1}{2}(y - f(x))^2$
- $|y - f(x)| > \delta$: Giảm ảnh hưởng của các giá trị ngoại lai.
 $\mathcal{L}_\delta(y, f(x)) = \delta \cdot |y - f(x)| - \frac{1}{2}\delta^2$



Các bước huấn luyện với OHGesture

- ① Lấy nhãn x_0 từ phân bố của dữ liệu đã chuẩn hóa
- ② Tính sẵn các giá trị, và các siêu tham số: $\gamma, \sqrt{\alpha_t}, \sqrt{1 - \alpha_t}, \sqrt{\bar{\alpha}_t}$,
Random nhiễu ϵ_t ở mọi bước $t : 1 \rightarrow T$. $\{\alpha_t \in (0, 1)\}_{t=1}^T$
- ③ Random Masks $c_1 = [\mathbf{s}, \mathbf{e}_1, \mathbf{a}, \mathbf{v}], c_2 = [\mathbf{s}, \mathbf{e}_2, \mathbf{a}, \mathbf{v}]$ hoặc
 $c_2 = [\emptyset, \emptyset, \mathbf{a}, \mathbf{v}]$
- ④ Forward \mathbf{x}_0 để có cử chỉ nhiễu $\mathbf{x}_t : \mathbf{x}_t = \sqrt{\bar{\alpha}_t} \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \epsilon_t$
- ⑤ for all t , lấy t **ngẫu nhiên** $t \sim [1, T]$
- ⑥ Cho \mathbf{x}_t và t, c, c_\emptyset để dự đoán chuỗi cử chỉ

$$\hat{\mathbf{x}}_{0\gamma, c_1, c_2} = \gamma \cdot f_\theta(\mathbf{x}_t, t, c_1) + (1 - \gamma) \cdot f_\theta(\mathbf{x}_t, t, c_2) \quad (6)$$

- ⑦ Tính loss và đạo hàm để cập nhật trọng số θ

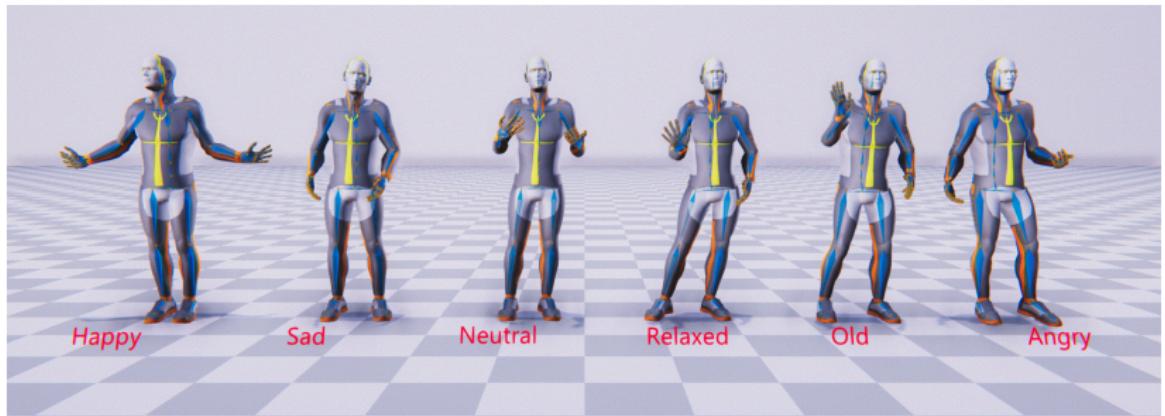
$$\mathcal{L}_t = \mathbb{E}_{t \sim [1, T], \mathbf{x}_0, \epsilon_t} \left[\text{HuberLoss}(\mathbf{x}_0, \hat{\mathbf{x}}_0) \right] \quad (7)$$

- ⑧ Quay lại bước 6 cho đến khi hội tụ để thu được θ'

Các bước lấy mẫu (Sampling) với OHGesture

- ① Bắt đầu với nhiễu: $\mathbf{x}_T \sim \mathcal{N}(0, \mathbf{I})$
- ② Các giá trị $\sqrt{\alpha_t}$, $\sqrt{1 - \alpha_t}$ và $\sqrt{\bar{\alpha}_t}$ có được từ bước huấn luyện, tính sẵn các giá trị σ_t từ α_t ở mọi bước $t : 1 \rightarrow T$
- ③ Cử chỉ khởi tạo s ban đầu là trung bình dữ liệu, sau đó được lấy từ đoạn cử chỉ đã suy luận. Chọn cảm xúc mong muốn, văn bản được lấy từ transcribe speech a và tạo cặp condition $c = [s, e, a, v]$
- ④ for all t , lấy t **tuần tự** $t \sim [T, \dots, 1]$
- ⑤ Random nhiễu $\mathbf{z} \sim \mathcal{N}(0, \mathbf{I})$
- ⑥ Đưa \mathbf{x}_t vào để suy luận $\hat{\mathbf{x}}_0^{(t)} = f_{\theta'}(\mathbf{x}_t, t, c)$
- ⑦ Forward $\hat{\mathbf{x}}_0^{(t)}$ $0 \rightarrow t$ để được $\hat{\mathbf{x}}_{t-1}^{(t)}$
- ⑧ Cộng thêm một lượng nhiễu $\hat{\mathbf{x}}_{t-1} = \hat{\mathbf{x}}_{t-1}^{(t)} + \sigma_t \mathbf{z}$
- ⑨ Quay lại bước 4, khi $t = 1$ ta thu được $\hat{\mathbf{x}}_0$ từ quá trình khử nhiễu

Giai đoạn 7: Kết xuất (Render)



Outline

① Minh họa

② Giới thiệu

③ Công trình liên quan

④ Mô hình sinh cử chỉ

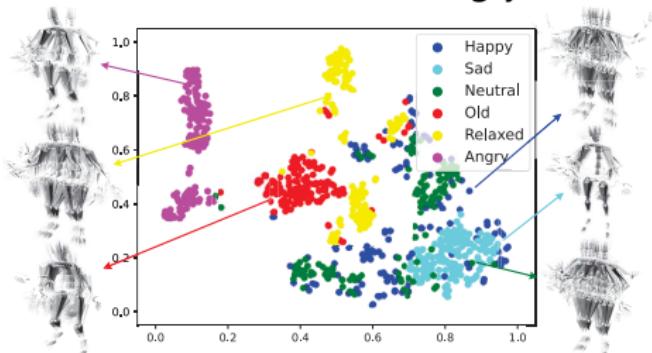
⑤ Thực nghiệm

⑥ Kết luận

Dataset

ZeroEGGs Dataset [4]:

- Bao gồm 67 đoạn độc thoại của diễn viên motion capture nữ
- Độ dài toàn bộ tập dữ liệu là 135 phút.
- Bao gồm 6 cảm xúc: Happy, Sad, Neutral, Old, Relaxed, Angry



Độ đo

Mean opinion scores (MOS)

- Human-likeness
- Gesture-Speech Appropriateness
- Gesture-style Appropriateness

FGD (Fréchet Gesture Distance): Đề xuất GestureScore. Sự tương đồng về phân phối của cử chỉ tạo sinh $g \in \mathbb{R}^{1:M \times D}$ (generated) và $r \in \mathbb{R}^{1:M \times D}$ (real)

$$FGD = \|\mu_r - \mu_g\|^2 + \text{Tr} \left(\Sigma_r + \Sigma_g - 2\sqrt{\Sigma_r \Sigma_g} \right) \quad (8)$$

Kết quả

Name	Human likeness ↑	Gesture-speech appropriateness ↑
Ground Truth	4.15 ± 0.11	4.25 ± 0.09
Ours	4.11 ± 0.08	4.11 ± 0.10
– WavLM	4.05 ± 0.10	3.91 ± 0.11
– Cross-local attention	3.76 ± 0.09	3.51 ± 0.15
– Self-attention	3.55 ± 0.13	3.08 ± 0.10
– Attention + GRU	3.10 ± 0.11	2.98 ± 0.14
+ Forward attention	3.75 ± 0.15	3.23 ± 0.24

Name	FGD trên vector đặc trưng	FGD dữ liệu gốc
Ground Truth	-	-
Ours		
OHGesture (Feature D=1141)	2.058	9465.546
OHGesture (Rotations)	3.513	9519.129

Outline

- ① Minh họa
- ② Giới thiệu
- ③ Công trình liên quan
- ④ Mô hình sinh cử chỉ
- ⑤ Thực nghiệm
- ⑥ Kết luận

Giải quyết các thách thức

- ① Quan hệ $n : n$ cử chỉ, âm thanh → Cross-Local Attention và Self-Attention
- ② Không cân xứng các đặc trưng → Mô hình Diffusion giúp mô hình có thể học được các đặc trưng có phân bố dữ liệu thật thấp.
- ③ Thiếu đồng nhất dữ liệu → sử dụng Huber Loss

Đóng góp chính



- **Sử dụng đặc trưng văn bản:** Transcribe âm thanh để có được văn bản (*Công đoạn tiền xử lý, xử lý đặc trưng và kết hợp đặc trưng*), làm dữ liệu bổ sung trong Diffusion có điều kiện.
- **Mở rộng mã nguồn:** github.com/OHGesture, Mô hình pretrain: [Huggingface.co/openhuman](https://huggingface.co/openhuman)
- **Hệ thống trực quan hoá bằng Unity:** github.com/DeepGesture-Unity.
- **Genea Leaderboard:** Paper xây dựng hệ thống chuẩn hoá
- **Đề xuất GestureScore** (Đánh giá cử chỉ bằng FGD): github.com/GestureScore.

Kết luận

- Mô hình **OHGesture**, có khả năng sinh cử chỉ chân thực, không chỉ trên các mẫu dữ liệu trong tập huấn luyện mà còn mở rộng được với những âm thanh không có trong dữ liệu huấn luyện.
- Sử dụng phương pháp **Classifier-free Guidance**, có thể điều khiển các điều kiện như cảm xúc, cử chỉ khởi tạo, có thể nội suy để suy luận ra giữa các cảm xúc khác nhau.
- Bài toán sinh cử chỉ còn rất mới, chúng tôi dự định sẽ dùng Fourier TT để trích xuất các đặc trưng về pha.