Intentional Gesture: Deliver your Intentions with Gestures for Speech

Supplementary Material

A Overview

The supplementary material is organized into the following sections:

- Section B: Additional Dataset Analysis
- Section C: Annotation Protocol and Validation
- Section D: Implementation Details
- Section E: Additional Experiments
- Section F: User Study Details
- Section G: Metric Details
- Section H: Ethical Statement
- Section I: Reproducibility statement
- Section J: The Use of Large Language Models
- Section K: Limitations

For more visualization, please see the additional demo videos.

B Additional Dataset Analysis

B.1 Function-to-Gesture Mapping Grounding

Our function-to-gesture mappings derive from established frameworks in gesture pragmatics, particularly McNeill McNeill (1992) and Kendon Kendon (2004). Tab. 5 presents gesture forms associated with each communicative function, which inform our VLM annotation prompt's gesture behavior mapping.

Certain functions correspond to consistent physical gestures (e.g., Deixis to pointing, Emphasis to beat gestures, Negation to head shakes), while others like Modal or Mental State manifest in subtler movements (fist tightening, shoulder shrugs). These literature-backed correspondences ensure interpretable and plausible annotations, providing a bridge between gesture generation and discourse semantics.

Tab. 5 shows the function distribution across dataset splits. Core functions such as Deixis (57-61%), Emphasis (46-51%), Mental State (41%), and Process (26-29%) are well-represented with minimal variation across splits. Less frequent functions like Comparison, Modal, and Valence (5-8%) and specialized functions (Intensifier, Physical Relation, ¡2%) show distributional consistency. Note that these percentages reflect per-sentence function occurrence rather than the cumulative distribution reported in the main paper.

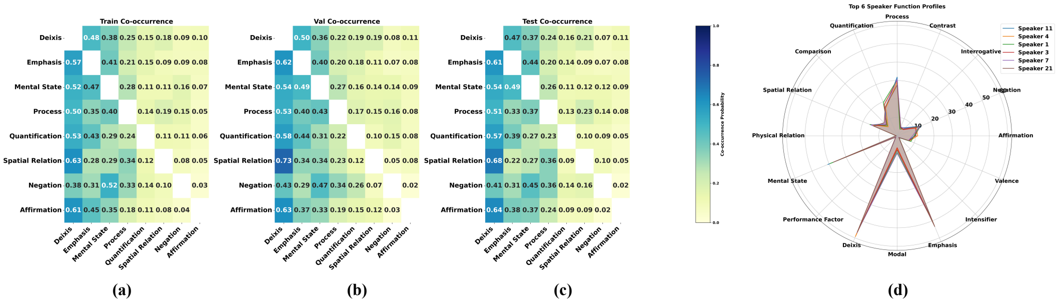
B.2 Co-occurrence Patterns and Speaker-Specific Gesture Profiles.

To further examine the structure of our function annotations, we analyze co-occurrence patterns and speaker-level gesture usage. Figures 8(a–c) present conditional co-occurrence heatmaps for the top 8 gesture functions across train, validation, and test splits. Each cell reflects the probability that function j co-occurs given function i within the same utterance. We observe strong mutual co-occurrence between Emphasis and Deixis, as well as between Mental State and Emphasis, suggesting these functions often emerge in jointly expressive speech segments. These co-occurrence trends remain stable across dataset splits, reinforcing the semantic consistency of our annotations.

Figure 8(d) shows a radar plot of gesture function usage for the top 6 most frequent speakers. While some functions like Deixis and Emphasis are commonly expressed across speakers, other functions (e.g., Contrast, Modal, Quantification) exhibit speaker-specific variability. This aligns with prior

Table 5: Gesture function statistics and mappings. For each function, we report its relative frequency (%) across dataset splits and its typical gestural manifestation.

Function	**Frequency (%)**			**Typical Gesture Mapping**
	Train	Val	Test	
Deixis	57.3	61.8	60.2	Index finger pointing, gaze direction shift
Emphasis	48.3	50.6	46.4	Beat gestures, small head nods
Mental State	42.0	41.1	41.1	Shrug, slow head tilt, hand on chest
Process	29.1	25.6	28.8	Circular motion, continuous hand movement
Quantification	16.7	20.6	17.0	Spread fingers, repeated motions
Spatial Relation	16.1	16.5	18.2	Hands indicating space or depth
Negation	13.2	12.3	11.0	Head shake, subtle hand wave
Affirmation	8.9	10.7	9.9	Big nod, repeated nods
Valence	8.1	7.0	7.1	Open hands (positive), recoiling motion (negative)
Modal	7.6	8.1	5.2	Tight fist, upward palm with tension
Comparison	6.6	7.7	5.6	Left-right hand sweep, comparative spacing
Interrogative	4.6	2.9	3.4	Raised eyebrows, open palms
Contrast	3.9	3.5	3.2	Alternating hand gestures, lateral head tilt
Intensifier	1.4	1.4	1.2	Sharp eyebrow raise, large gesture amplitude
Performance Factor	1.0	1.1	0.9	Gaze aversion, short blink, pause gestures
Physical Relation	0.6	0.6	0.7	Gesture showing size/shape (e.g., distance between hands)



Train Co-occurrence

	Deixis	Emphasis	Mental State	Process	Quantification	Spatial Relation	Negation	Affirmation
Deixis		0.48	0.38	0.25	0.15	0.18	0.09	0.10
Emphasis	0.57		0.41	0.21	0.15	0.09	0.09	0.08
Mental State	0.52	0.47		0.28	0.11	0.11	0.16	0.07
Process	0.50	0.35	0.40		0.14	0.19	0.15	0.05
Quantification	0.53	0.43	0.29	0.24		0.11	0.11	0.06
Spatial Relation	0.63	0.28	0.29	0.34	0.12		0.08	0.05
Negation	0.38	0.31	0.52	0.33	0.14	0.10		0.03
Affirmation	0.61	0.45	0.35	0.18	0.11	0.08	0.04	

Val Co-occurrence

	Deixis	Emphasis	Mental State	Process	Quantification	Spatial Relation	Negation	Affirmation
Deixis		0.50	0.36	0.22	0.19	0.19	0.08	0.11
Emphasis	0.62		0.40	0.20	0.18	0.11	0.07	0.08
Mental State	0.54	0.49		0.27	0.16	0.14	0.14	0.09
Process	0.53	0.40	0.43		0.17	0.15	0.16	0.08
Quantification	0.58	0.44	0.31	0.22		0.10	0.15	0.08
Spatial Relation	0.73	0.34	0.34	0.23	0.12		0.05	0.08
Negation	0.43	0.29	0.47	0.34	0.26	0.07		0.02
Affirmation	0.63	0.37	0.33	0.19	0.15	0.12	0.03	

Test Co-occurrence

	Deixis	Emphasis	Mental State	Process	Quantification	Spatial Relation	Negation	Affirmation
Deixis		0.47	0.37	0.24	0.16	0.21	0.07	0.11
Emphasis	0.61		0.44	0.20	0.14	0.09	0.07	0.08
Mental State	0.54	0.49		0.26	0.11	0.12	0.12	0.09
Process	0.51	0.33	0.37		0.13	0.23	0.14	0.08
Quantification	0.57	0.39	0.27	0.23		0.10	0.09	0.05
Spatial Relation	0.68	0.22	0.27	0.36	0.09		0.10	0.05
Negation	0.41	0.31	0.45	0.36	0.14	0.16		0.02
Affirmation	0.64	0.38	0.37	0.24	0.09	0.09	0.02	

Figure 8: **Co-occurrence and speaker-level analysis of gesture function annotations. (a–c)** show conditional co-occurrence heatmaps of the top 8 gesture functions across the train, validation, and test splits, respectively. Each cell indicates the probability of function j appearing given function i (i.e., $P(j|i)$). Strong pairings (e.g., Emphasis + Deixis, Mental State + Emphasis) reveal compositional gesture semantics. **(d)** presents radar plots of function distribution across the top 6 speakers, revealing shared trends (e.g., high Deixis usage) and speaker-specific variation in gesture behavior.

findings that gesture behavior reflects both discourse demands and speaker idiosyncrasies Kendon (2004). Such variation presents a valuable modeling challenge for systems that aim to personalize or adapt gesture generation to individual styles.

Algorithm 1 Motion Pattern Detection

```
Require: Input data $\mathbf{y} \in \mathbb{R}^T$, thresholds $\epsilon_s$, $\epsilon$
Ensure: Motion statistics and extrema relations
$\mathbf{y} \leftarrow$ reshape to 1D array
if $T \leq 1$ then return insufficient_data
end if
// Extract key statistics
$y_0, y_T \leftarrow \mathbf{y}[0], \mathbf{y}[T-1]$
$i_{max}, i_{min} \leftarrow \arg\max(\mathbf{y}), \arg\min(\mathbf{y})$
$y_{max}, y_{min} \leftarrow \mathbf{y}[i_{max}], \mathbf{y}[i_{min}]$
$\delta \leftarrow y_{max} - y_{min}$, $\Delta \leftarrow y_T - y_0$
// Check if motion is static
if $\delta < \epsilon_s$ then
    return {pattern: 'linear', range: $\delta$, direction: sign($\Delta$)}
end if
// Compute extrema relations
$\mathbf{s} \leftarrow [|y_0 - y_{max}| \leq \epsilon, |y_0 - y_{min}| \leq \epsilon]$        ▷ Start position
$\mathbf{e} \leftarrow [|y_T - y_{max}| \leq \epsilon, |y_T - y_{min}| \leq \epsilon]$        ▷ End position
$\mathbf{in} \leftarrow [i_{max} \notin \{0, T-1\}, i_{min} \notin \{0, T-1\}]$        ▷ Interior extrema
return $\{\mathbf{y}, \Delta, \delta, \mathbf{s}, \mathbf{e}, \mathbf{in}\}$
```

Algorithm 2 Motion Pattern Classification

```
Require: Extrema relations $\mathbf{s} = [s_{max}, s_{min}]$, $\mathbf{e} = [e_{max}, e_{min}]$, $\mathbf{in} = [in_{max}, in_{min}]$
Ensure: Pattern type and description
if $(s_{max} \wedge e_{min}) \vee (s_{min} \wedge e_{max})$ then
    pattern $\leftarrow$ 'round_trip'        ▷ Opposite extremes
else if $(s_{max} \vee s_{min}) \wedge (e_{max} \vee e_{min})$ then
    pattern $\leftarrow$ 'return_to_extreme'        ▷ Same extreme
else if $(s_{max} \vee s_{min}) \wedge \neg(e_{max} \vee e_{min})$ then
    pattern $\leftarrow$ 'peak_at_start'        ▷ Leave from extreme
else if $(e_{max} \vee e_{min}) \wedge \neg(s_{max} \vee s_{min})$ then
    pattern $\leftarrow$ 'peak_at_end'        ▷ Arrive at extreme
else if $in_{max} \wedge in_{min}$ then
    pattern $\leftarrow$ 'peak_between'        ▷ Both extremes inside
else if $in_{max} \oplus in_{min}$ then
    pattern $\leftarrow$ 'single_extreme_inside'        ▷ One extreme inside
else
    pattern $\leftarrow$ 'complex_extrema'        ▷ Boundary-aligned
end if
return pattern
```

Algorithm 3 Helper Functions

```
function GetDirection($\Delta$)
    return $\begin{cases} \text{'positive'} & \text{if } \Delta > 0 \\ \text{'negative'} & \text{if } \Delta < 0 \\ \text{'none'} & \text{otherwise} \end{cases}$
end function

function ClassifyMovement($\Delta, \epsilon_s, \epsilon_{slow}$)
    return $\begin{cases} \text{'static'} & \text{if } |\Delta| < \epsilon_s \\ \text{'slow'} & \text{if } |\Delta| < \epsilon_{slow} \\ \text{'significant'} & \text{otherwise} \end{cases}$
end function
```

Table 6: **Training-time annotation prompt with visual grounding.** Our framework analyzes human gestures by integrating visual keyframes with speech.

System Prompt:
Assume you are the annotator for human gestures. Given images for each word the person speaks, you need to provide fine-grained analysis from motion captions, to Function Derivations, Gesture Behavior Mapping, and finally Inferred Intention. The Definition of Function Derivation & Gesture Behavior Mapping are as follows:

[Function Derivation: 16 classes of Function Derivations]
[Gesture Behavior Mapping: How functions map to physical movements.]

User Prompt:
I will provide you with a transcript of speech, the atomic pose angle movement descriptions and corresponding images showing the speaker's gestures. Please analyze the motion and provide a detailed description as the generation output following this format:

[Format Instruction]
Motion Analysis:

- **Head:** Describe head movements (nodding, shaking, tilting)
- **Hands & Fingers:** Describe hand gestures, positions, finger articulations
- **Arms & Shoulders:** Describe arm movements and shoulder positions
- **Legs & Feet:** Describe lower body movements and weight shifts
- **Torso & Whole Body:** Describe posture and body orientation

Function Derivation: List relevant functions from the prior knowledge
Gesture Behavior Mapping: Map each function to observed gestures
Inferred Intention: Explain overall communicative intent

[One-shot Example:]
Input: "I think this one is much better than the previous one." [Images]
Output: Motion Analysis: [Head, hands, arms, legs, body movements]
Function Derivation: [Comparison, Emphasis, Deixis functions]
Gesture Behavior Mapping: [Function-to-gesture relationships]
Inferred Intention: [Communication intent analysis]

[Data to be Annotated]

C Annotation Protocol and Validation

C.1 Motion Pattern Analysis

We propose a rule-based algorithm for classifying temporal motion patterns by analyzing the geometric relationships between trajectory extrema and boundaries. Given a motion sequence $\mathbf{y} \in \mathbb{R}^T$ (e.g., joint angles or hand positions), our method extracts key statistics and determines the motion pattern through a deterministic decision process, as detailed in Algorithms 1–2.

The algorithm operates in three stages. First, it computes fundamental statistics: boundary values (y_0, y_T), global extrema $(y_{\max}, y_{\min})$ with their indices, and the motion range $\delta = y_{\max} - y_{\min}$ (Algorithm 1, lines 3–6). If δ falls below a static threshold ϵ_s, the motion is classified as linear/static, avoiding misclassification of noise as complex patterns (Algorithm 1, lines 8–10).

For non-static motion, the algorithm analyzes **extrema-boundary relations** by computing boolean indicators for whether the start/end positions are near (within tolerance ϵ) the global extrema, and whether extrema occur in the trajectory interior (Algorithm 1, lines 12–14). These geometric relations capture motion characteristics invariant to scale and translation.

Finally, pattern classification applies hierarchical logical rules based on these relations (Algorithm 2). For instance, if the trajectory starts near one extreme and ends near the opposite $(s_{\max} \wedge e_{\min}) \vee (s_{\min} \wedge e_{\max})$, it's classified as a "round trip" pattern (Algorithm 2, lines 2–3). Other

Table 7: **Test-time annotation prompt without visual grounding.** To prevent data leakage in evaluation, test-time annotations deliberately exclude visual information, requiring functions and intentions to be inferred solely from linguistic content.

System Prompt:
Assume you are the annotator for human speech. Without access to gesture images, you need to infer likely communicative functions and intentions from linguistic content alone. Based on Function Derivations, analyze the words and its durations within the transcript. Then analyze the Inferred Intention. The Definition of Function Derivation are as follows:

[Function Derivation: 16 classes of Function Derivations]

User Prompt:
I will provide you with:

- Previous two sentences for context
- Current sentence to be annotated
- *No visual information or keyframes*

Please analyze the linguistic content and provide predictions as follows:

Linguistic Analysis:

- Identify key words and phrases that typically trigger gestures
- Note speech elements that commonly correlate with specific movements
- Analyze the syntactic and semantic structure that implies gesture potential

Function Derivation: Infer likely functions based solely on linguistic content
Predicted Gesture Types: Suggest probable gesture categories without seeing actual movements
Inferred Intention: Predict the likely communicative intent based on linguistic cues

[One-shot Example for In-Context Learning without visual data]
[Data to be Annotated - transcript only]

patterns include “return to extreme” (starting and ending at the same extreme), “peak between” (both extrema in the interior), and “single extreme inside” (one interior extreme), among others.

The algorithm employs context-aware thresholds that adapt based on motion type (e.g., different sensitivity for hand positions vs. joint angles) and achieves $\mathcal{O}(T)$ complexity through efficient single-pass operations (Algorithm 2). This deterministic approach provides interpretable pattern detection without requiring training data, making it suitable for real-time motion analysis applications where understanding the type of movement (cyclic, monotonic, or complex) is crucial for downstream tasks.

C.2 Training-Time Annotation Protocol (With Motion Frames)

To construct training annotations, we prompt GPT-4o-mini with both linguistic and visual inputs. Each prompt includes: **(1)** The two previous sentences spoken by the speaker, serving as linguistic context. **(2)** The current sentence to annotate, segmented into word units with corresponding timestamps. **(3)** The sampled starting and ending keyframe image for each word, together with the rule-based motion description annotation for the poses. We show the prompt template in Tab.6. The model is instructed to generate a structured analysis with the following outputs:

Motion Analysis: Detailed natural language description of body movements, including head motion, arm/shoulder gestures, finger positions, torso orientation, and stance.

Function Derivation: Identification of pragmatic functions (e.g., Emphasis, Deixis, Negation) that are linguistically relevant to the current sentence.

Gesture Behavior Mapping: Mapping between derived functions and observable gesture types (e.g., pointing, nodding, brow raise) following established gesture theory.

Inferred Intention: A communicative goal inferred from the alignment of motion and function (e.g., emphasizing contrast, directing attention, expressing uncertainty).

Table 8: **Baseline annotation prompt without structure.** This naive protocol excludes gesture theory or function derivation, asking the model to directly infer the speaker's communicative intent. This leads to overgeneralized or underspecified outputs.

System Prompt:
You are an assistant that helps interpret the meaning behind a speaker's body language and words. Given the speaker's sentence and gesture images for each word, describe what the speaker is trying to express overall. Do not break the task into components; simply provide an intention summary based on what you perceive.

[No prior gesture theory, no function derivation definitions]

User Prompt:
I will give you:

- A transcript of the speaker's sentence
- An image for each word the speaker says

Please describe what the speaker is trying to express or communicate. Use natural language, and focus on the overall message or feeling you perceive.

Output:

- One or two sentences summarizing the speaker's communicative intention
- Do not perform motion breakdown or gesture labeling
- Do not mention gesture function classes or mappings

[Example:]
Input: "I think this one is much better than the previous one." [Images]
Output: The speaker is expressing a strong preference for a current choice, likely implying confidence or satisfaction.
[Data to be Annotated]

This protocol captures visually grounded, multi-level annotation aligned with both motion and speech.

C.3 Test-Time Annotation Protocol (Transcript Only)

To avoid potential data leakage in test annotations, we exclude visual motion input from the VLM prompts during test set annotation. Each test prompt contains the two prior sentences for context and the current sentence to be annotated. No keyframes or motion descriptions are provided. The VLM is instructed to: **(1)** Infer likely communicative functions based solely on linguistic content. **(2)** Derive high-level communicative intent without visual grounding, as shown in Tab.7.

This simulates the actual evaluation scenario, where gesture models must predict motion solely from speech, and prevents the test set annotations from being conditioned on ground-truth poses.

C.4 Baseline Annotation Protocol (No Structured Prompt)

To examine the importance of structured reasoning, we design a baseline annotation protocol that omits the function derivation and gesture behavior mapping stages. In this setting, GPT-4o-mini is prompted with the current sentence and visual frames for each word, but is asked only to provide an inferred intention directly—without performing intermediate motion analysis or reasoning about communicative function. We present the prompt example in Tab.8.

This resembles a generic captioning-style instruction (e.g., "Describe what the speaker is trying to express"), lacking any prior definitions or decomposition of gesture semantics. While this setup may yield fluent outputs, it often results in: **(1) Overgeneralization:** Outputs tend to collapse nuanced signals (e.g., emphasis, negation, deixis) into vague descriptions such as "the speaker is sharing a thought." **(2) Hallucination:** In the absence of reasoning stages, the model may infer incorrect intentions (e.g., persuasive intent where none exists). **(3) Loss of Interpretability:** Since outputs are not grounded in functional structure, they cannot be mapped to gesture execution in a controllable or compositional way. This baseline highlights the necessity of structured prompting in generating

Table 9: **Comparative annotation outputs across two utterances.** Structured annotations include function derivation and gesture mapping. Improper annotations suffer from overgeneralization, hallucination, or lack of compositionality.

Utterance A: ”I think watching anime is helpful for me”	
Training-Time (w/ motion)	**Function Derivation:** *Deixis* (“me”), *Mental State* (positive belief). **Gesture Mapping:** Deixis → hand at chest, Mental State → relaxed stance. **Inferred Intention:** The speaker reflects personally on the benefit of anime. Gestures reinforce introspection and confidence.
Test-Time (transcript-only)	**Function Derivation:** *Deixis*, *Mental State*. **Gesture Mapping:** [Not available] **Inferred Intention:** The speaker shares a personal viewpoint with implied conviction, likely supported by subtle gestures.
Improper: Flat Intent Only	**Inferred Intention:** The speaker is talking about anime. *[Missing: No function derivation, no motion context, no gestural insight.]*
Improper: Hallucinated Purpose	**Inferred Intention:** The speaker is encouraging the audience to try watching anime as a productivity tool. *[Issue: Adds persuasive intent not supported by transcript or body motion.]*
Improper: Misaligned Gesture Mapping	**Inferred Intention:** The speaker is contrasting anime with something unhelpful. *[Issue: Misinterprets positive reflection as contrast/negation.]*
Utterance B: ”I always try to move as much as I can when I’m not working”	
Training-Time (w/ motion)	**Function Derivation:** *Emphasis* (“working”), *Negation* (“not working”), *Modal* (“can”). **Gesture Mapping:** Emphasis → steady hands reinforce commitment; Negation → assertive fist posture; Modal → gestural space around ”can”. **Inferred Intention:** Speaker emphasizes an active lifestyle outside of work. Gestures signal assertion and capability.
Test-Time (transcript-only)	**Function Derivation:** Same as above (*Emphasis*, *Negation*, *Modal*). **Gesture Mapping:** [Omitted] **Inferred Intention:** The speaker frames movement as a conscious, empowering action. Likely gestures reinforce contrast and agency.
Improper: Flat Intent Only	**Inferred Intention:** The speaker is saying that they move around a lot. *[Issue: No deeper intent, no gesture mapping, missing compositional structure.]*
Improper: Misaligned Functions	**Inferred Intention:** The speaker is unsure whether they move enough and seems to compare working vs. resting. *[Issue: Misses clear assertion and negation. Misreads modality.]*
Improper: No Composition	**Inferred Intention:** The speaker likes to be active. *[Issue: Oversimplifies the sentence; collapses nuanced components (modal vs. negation vs. emphasis) into a flat label.]*

interpretable and semantically grounded gesture annotations. We include comparative examples in Tab. 9 to illustrate these failure modes in context.

C.5 Annotation Validation and Human Preference Study

To assess the reliability of our annotation pipeline, we randomly sampled 100 utterances from the training set. Each sample was annotated using both the training protocol (with-motion) and the test protocol (transcript-only). Separately, expert annotators were provided with: **(1)** The utterance and its transcript. **(2)** The full sequence of rendered motion frames.

Experts then independently labeled: **(1)** The communicative function(s) present. **(2)** The inferred intention based on motion and speech. **(3)** The gesture types observed in the motion.

We then presented annotators with three candidate annotations for each sample (training VLM, test VLM, and human-generated), blinded and randomized. Annotators were asked to rate: **(1)** Which annotation most accurately reflected the speaker's intent. **(2)** Which annotation was most clearly and consistently reasoned.

Results, shown in main paper Fig.4, indicate that the training-style annotation (with visual grounding) achieved the highest human preference. However, the transcript-only test-style annotations also achieved strong scores, outperforming human-generated annotations in clarity and structural alignment. This validates the effectiveness of our prompt design and supports the use of VLM-generated labels for both training and evaluation.

C.6 VLM Consistency and Hallucination Audit

To ensure the reliability of our VLM-based annotation pipeline, we performed two targeted sanity checks: a consistency audit and a hallucination spot check.

Consistency Under Repeated Prompts. We randomly selected 100 utterances from the dataset and re-prompted GPT-4o-mini three times each under the same configuration. We examined the stability of the output across three categories: (i) function derivations, (ii) inferred intentions, and (iii) gesture behavior mappings. Across the 300 trials: 93% of the outputs maintained consistent function derivation labels. 84% preserved consistent gesture mappings across trials. These results suggest that the model exhibits stable behavior under repeated prompting, with low variance in the output of structural annotations.

Hallucination Spot Check. To assess the faithfulness of annotation outputs to visual evidence, we conducted an expert spot check on 50 randomly sampled annotation instances. Each instance included three components: **(1) Motion Descriptionz**, **(2) Function–Gesture Mapping**, and **(3) Inferred Intention**. For motion descriptions, 4 out of 50 samples (8%) were flagged for partial inconsistencies. These typically involved subtle over-interpretations—e.g., stating a "brow raise" when the face appeared neutral in the keyframe. No instances of fully fabricated or unrelated gestures were identified. For Function–Gesture Mapping, only 1 sample (2%) was marked as problematic, where a mapping relation (e.g., from a deictic phrase to a pointing gesture) was missing. The issue stemmed from under-specification rather than misalignment. For intention inference, 3 samples (6%) were flagged for slight exaggerations—such as over-interpreting neutral tones as emphasizing emotion. These were still broadly reasonable within the context of the utterance, and none were deemed to be outright hallucinations. Overall, the hallucination rate was low, and all identified issues were minor and recoverable. Importantly, no samples exhibited completely incorrect reasoning or disjointed alignment. This suggests the annotations are well-grounded and highlights the strong prompt-following and contextual inference abilities of the VLM. We also observe that minor hallucinations in motion description do not meaningfully degrade the accuracy of intention inference, supporting the robustness of our pipeline.

C.7 Human Study Instructions

We present the details how we conducted the manual hallucination checking from the users as follows.

Study 1: Function–Gesture Mapping Coherence **Objective:** Evaluate whether gestures are appropriate and coherent realizations of their corresponding communicative functions.

Instructions to Annotators: You are provided with a communicative function label (e.g., "Emphasis") and a corresponding gesture description (e.g., "Right hand performs rhythmic beat"). Please assess whether the described gesture appropriately fulfills or expresses the given function.

- Q1: Is this mapping coherent? (Yes / No)
- Q2 (Optional): If you selected "No", briefly explain why.

Evaluation Protocol: We randomly selected 50 samples and recruited 2 expert annotators. Final coherence score is computed as the average percentage of "Yes" responses across raters.

Study 2: Motion Description–Keyframe Fidelity **Objective:** Determine whether the motion description accurately reflects the visible pose and dynamics presented in the keyframes.

Instructions to Annotators: You are shown a short video segment (or sequence of static keyframes) and a motion description (e.g., “Left hand slowly rises while the head turns right”). Please judge whether the described motion is clearly and accurately visible in the keyframes.

- Q1: Does the motion description match the keyframes? (Yes / Partially / No)
- Q2 (Optional): If “Partially” or “No”, please explain which aspects were inaccurate or missing.

Evaluation Protocol: We used the same 50 annotated samples and had each rated by 2 human experts. Final scores are reported as the percentage of samples rated “Yes” (fully correct) and “Partially”.

Study 3: Inferred Intention Plausibility **Objective:** Assess whether the inferred communicative intention is a reasonable high-level summary of the utterance and accompanying gesture behavior.

Instructions to Annotators: You are shown a spoken utterance (text transcript) and a corresponding intention inference (e.g., “The speaker is attempting to reassure the listener about a concern”). Please judge whether the intention is plausible based on the content and tone of the utterance.

- Q1: Is the inferred intention plausible given the utterance? (Yes / Somewhat / No)
- Q2 (Optional): If “Somewhat” or “No”, please describe why the inference may be overstated or misaligned.

Evaluation Protocol: Each of the 50 samples was evaluated by 2 annotators. We report the percentage of “Yes” and “Somewhat” responses to quantify plausibility and over-interpretation.

D Implementation Details

Hierarchical Audio-Motion Modality Alignment. We adopt a dual-tower CLIP-based contrastive framework inspired by Tango Liu et al. (2024a), trained using a global InfoNCE loss. A key design choice for handling audio-motion modality alignment is the separation into low-level and high-level encoders.

For the audio stream, we represent input as raw waveforms and apply a 7-layer CNN (low-level) followed by a 3-layer Transformer (high-level), following the design of Wav2Vec2 (Baevski et al., 2020). For motion, we use a 15D representation and employ a 3-layer residual CNN (adapted from the Momask Motion Tokenizer (Guo et al., 2024)) and a 3-layer Transformer.

We use a projection MLP to process low-level features and another projection MLP with mean pooling for high-level features. Both audio and motion streams are temporally downsampled by a factor of 4.

Local and Global Contrastive Loss. We retain the InfoNCE loss over CLS tokens for global alignment, and additionally introduce a frame-level local contrastive loss. We treat frames within a temporal window ($i \pm t$) as positives and distant frames ($i - kt$, $i - t$, $i + t$, $i + kt$) as negatives, with $t = 4$ and $k = 4$ under a 30 FPS setting. This localized loss encourages robustness to minor temporal misalignments common in natural talking scenarios.

Stop-Gradient on Low-Level Encoders. To jointly optimize both low- and high-level representations, we stop the gradient flow from the global InfoNCE loss to the low-level encoders, as in Tango Liu et al. (2024a). This design promotes complementary feature learning across hierarchy levels.

Intentional Gesture Tokenization. We design the motion tokenizer using a simplified version of the encoder architecture above, followed by a decoder that mirrors its structure. To stabilize training, we reduce both to a single Transformer layer but maintain the same residual CNN blocks. The latent feature dimension is set to 512.

We apply a self-attention layer to project the 512-dimensional encoding to 32 dimension for quantization. The quantizer comprises 8 codebooks, with a dimension 32 and 8192 codes. For post-quantization, another attention layer maps the 32D features back to 512D for decoding.

Intentional Gesture Generator. The generator operates on token sequences produced by the tokenizer. It uses a Transformer with DiT Peebles & Xie (2023) architecture with 8 layers, a hidden dimension of 256, and a feedforward dimension of 1024, and number of head to be 4. In each layer, there is one self-attention, one cross-attention and followed with the feed-forward layer. For the cross-attention layer, due to two levels of audio conditioning, we design the structure of **Decoupled Cross-Attention.** Rather than forcing a single attention over mixed features, we apply two cross-attention branches separately. Given a shared query Q, we compute:

$$\mathcal{Z}_r = \mathrm{SoftMax}\left(\frac{QK_r^\top}{\sqrt{d}} + \mathbf{P}\right)V_r, \quad \mathcal{Z}_i = \mathrm{SoftMax}\left(\frac{QK_i^\top}{\sqrt{d}} + \mathbf{P}\right)V_i, \tag{4}$$

where (K_r, V_r) and (K_i, V_i) are key-value pairs from rhythmic and intentional features, respectively. The outputs $\mathcal{Z}_r$ and $\mathcal{Z}_i$ are summed to form the final conditioning representation.

This design introduces only a minimal overhead—adding separate key and value projections (only adding 2% parameters) for each cross-attention layer—yet yields consistent improvements of 0.01–0.03 in FGD across validation runs. This demonstrates the benefit of explicitly modeling disentangled prosodic and semantic cues during gesture generation.

Optimizer Settings. All modules are trained using the Adam optimizer Kingma (2014), with a learning rate of 1×10^{-4}, $\beta_1 = 0.5$, and $\beta_2 = 0.999$. We utilize a liner schedule with constant decay for the learning rate for the model learning. The generator is trained on 800 epoches for both single speaker setting and multi-speaker setting.

E Additional Experiments

Baseline Methods. We compare against a comprehensive set of recent gesture generation approaches Habibie et al. (2021); Liu et al. (2022a;b; 2023); Chen et al. (2024b); Yi et al. (2023); Liu et al. (2024b); Xu et al. (2024); Liu et al. (2025a), all evaluated under the **1-speaker setting** for fair comparison. This setting is used by most prior works and allows precise alignment with publicly reported results on BEAT-2.

Full Generation Results. Table 10 presents the quantitative results on the BEAT-2 benchmark. Our model, **Intentional-Gesture**, achieves state-of-the-art performance across all key metrics. Notably, our method obtains the lowest FGD (**0.379**), indicating the highest overall realism, while maintaining strong beat consistency (0.690) and natural motion diversity (11.00). These results demonstrate the benefit of our intentional alignment and conditioning mechanisms in generating gestures that are both semantically expressive and rhythmically precise.

Table 10: The quantitative results on BEAT-2. We bold the best results.

Methods	FGD (↓)	BC (→)	Diversity (→)
Ground-Truth	–	0.703	11.97
HA2G Liu et al. (2022c)	1.232	0.677	8.626
DisCo Liu et al. (2022a)	0.942	0.643	9.912
CaMN Liu et al. (2022b)	0.664	0.676	10.86
DiffSHEG Chen et al. (2024b)	0.714	0.743	8.21
TalkShow Yi et al. (2023)	0.621	0.695	13.47
ProbTalk Liu et al. (2024b)	0.504	0.771	13.27
EMAGE Liu et al. (2023)	0.551	0.772	13.06
Audio2PhotoReal Ng et al. (2024)	1.02	0.550	**12.47**
MambaTalk Xu et al. (2024)	0.536	0.781	13.05
SynTalker Chen et al. (2024a)	0.469	0.736	12.43
GestureLSM Liu et al. (2025a)	0.409	0.714	13.24
Intentional-Gesture	**0.379**	**0.690**	11.00

Results on Audio2PhotoReal. Table 11 presents the quantitative results on the Audio2PhotoReal Ng et al. (2024) benchmark. Our model, **Intentional-Gesture**, achieves state-of-the-art performance across all key metrics. These results demonstrate the benefit of our intentional alignment and conditioning mechanisms in generating gestures can also be generalizable to dyadic conversational speaking and listening settings.

Table 11: The quantitative results on Audio2PhotoReal. We bold the best results.

Methods	FGD (↓)	Diversity (→)
Ground-Truth	–	2.50
EMAGE Liu et al. (2023)	4.43	2.13
Audio2PhotoReal Ng et al. (2024)	2.94	2.36
GestureLSM Liu et al. (2025a)	2.64	2.34
Intentional-Gesture	**2.21**	**2.43**

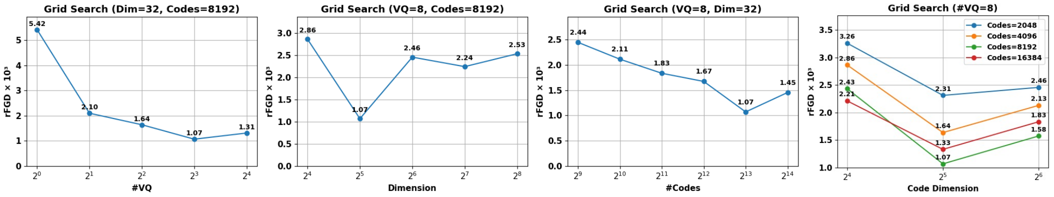
Effect of Speaker Diversity on Retrieval. To examine how speaker diversity influences model generalization, we fix the total number of training

Table 12: **Ablation on Speaker Diversity.** Increasing speaker diversity consistently boosts retrieval for both seen (*Known*) and unseen (*Unknown*) speakers, indicating better generalization.

	Known			**Unknown**		
Num	**R@1↑**	**R@5↑**	**R@10↑**	**R@1↑**	**R@5↑**	**R@10↑**
1	20.63	40.34	50.67	1.03	1.95	2.56
2	29.41	57.63	60.61	1.44	2.31	2.78
3	31.37	60.42	63.39	1.67	2.49	2.92
4	33.52	63.52	66.87	1.87	2.64	3.01

samples and vary only the number of distinct speakers contributing data. As shown in Tab. 12 (right), increasing the number of training speakers from 1 to 4 significantly improves retrieval performance across both **in-domain** (seen speakers) and **out-of-domain** (unseen speakers) settings.

Notably, for in-domain cases, Recall@1 rises from 20.63% (1 speaker) to 33.52% (4 speakers), while for out-of-domain speakers, Recall@1 improves from 1.03% to 1.87%. These gains indicate that speaker diversity not only enriches the representation space but also enables more robust cross-speaker generalization. We hypothesis that training with a wider range of gestural patterns allows the model to better disentangle speaker-specific motion from shared semantic-rhythmic alignment.



(a) (b) (c) (d)

Figure 9: **Tokenizer ablation.** We perform both global and local grid searches to study the effects of codebook design choices on rFGD ($\times 10^3$). (a)–(c): Global sweeps varying one factor at a time; (d): Local grid search over code count and dimension. All results confirm consistent trends: 8 codebooks, a code dimension of 32, and 8192 codes yield optimal or near-optimal performance.

Design Analysis. We ablate design choices of the tokenizer, including the number of codebooks, code dimension, and code size. Fig. 9 shows that (1) 8 codebooks outperform fewer or more, balancing representational capacity and model compactness; (2) a code dimension of 32 achieves the best trade-off between expressiveness and compression; and (3) increasing code size improves rFGD up to 8192 codes, with diminishing returns beyond. These trends are consistent across global and local grid searches. For architecture design, we discover CNN presents better reconstruction quality, but the transformer presents better generation FGD. Our hybrid design takes the advantage of both variants.

Long Sequence Generation Quality. In the main paper, the experiment setting were conducted to generate sequences for the whole testing sequence. Specifically, we follow the existing works Liu et al. (2023; 2025a); Chen et al. (2024b) to utilize a sliding window for long sequence generation (with an average of 65.66 seconds). Each time, we provide the previous 2.13 seconds (a sequence length of 16 for neural representation) generated from the previous generated segment as the condition for the current time segment. Naturally, this setting is easy to encounter the error propagation issue (if the sequence from the previous generation present low quality, this error will be propagated to the current time segment). To understand this effect, we further design the new setting that replicate the inference setting of the same inference audio length as that utilized during training (8.633 seconds). We present the comparison setting between EMAGE, GestureLSM and Intentional-Gesture for single speaker setting in Tab.13. On long sequences, our model achieves the best performance (FGD = 0.379, BC = 0.690, Div. = 11.00). Under short-sequence inference, our FGD further improves by 0.133 (to 0.246), closely matching the improvements of 0.140 and 0.107 seen for EMAGE and GestureLSM, respectively—indicating a consistent FGD gap of 0.12 across models. Note that BC is not reported (–) for 8.633 s segments, due to the tricky implementation to select the precise audio segments from full ground-truth sequences with the generation segments. These results underscore the impact of error accumulation in sliding-window co-speech gesture generation and motivate future work on mitigating segment-wise propagation.

Table 13: Comparison of long-sequence (full test sequences) vs. short-sequence (8.633 s) inference on the single-speaker setting.

	Long-seq Generation			Short-seq Generation		
	FGD↓	BC→	Div.→	FGD↓	BC→	Div.→
GT		0.703	11.97		0.703	11.97
Single-speaker						
EMAGE Liu et al. (2023)	0.570	0.793	11.41	0.430	-	9.57
GestureLSM Liu et al. (2025a)	0.408	0.714	13.24	0.301	-	**12.12**
Ours	**0.379**	**0.690**	**11.00**	**0.246**	-	10.21

Quantizer Comparisons Analysis To isolate the influence of architecture on tokenizer performance, we standardized all encoder–decoder backbones to our CNN+Transformer design, which we found consistently outperforms alternatives across various quantizers. Specifically:

(1) EMAGE Liu et al. (2023) originally uses separate VQ quantizers for upper body, lower body, and hands. We replace its CNN encoders with our ResNet-style CNN blocks and normalize codebook embeddings rather than using raw outputs. We keep the original codebook size and dimensionality to demonstrate how reducing dimension and increasing code count affects performance.

(2) For RAG-Gesture Mughal et al. (2025), we re-implement their encoder and decoder based on Latent Motion Diffusion from MotionLCM codebase Dai et al. (2024). The comparions indicates for the continuous representation, it is hard to present the motion latent with a compressed latent mean prediction from VAE encoder to ensure it is synchronized with the audio for generation.

(3) For ProbTalk Liu et al. (2024b), we maintain their design of product quantization while improve the encoder and decoder with our design. This comparison indicates the product quantization, while present a latent codebook split, unlike our codebook design of separate latent motion representation, presents an inferior performance.

(4) For GestureLSM Liu et al. (2025a), we maintain the design of 6 layers of codebooks for each body region (upper, lower and hands), which leads to 18 codebook in total. While this multi-codebook approach achieves competitive reconstruction, its reliance on separate decoders for sequential region generation reduces efficiency and harms overall motion quality.

F User Study Details

For user study, we recruited 20 participants with good English proficiency. To conduct the user study, we randomly select videos from GestureLSM Liu et al. (2025a), EMAGE Liu et al. (2023), CAMN Liu et al. (2022b) and ours. Each user works on 8 videos. The users are not informed of the source of the video for fair evaluations. A visualization of the user study is shown in Fig 10.

G Metric Details

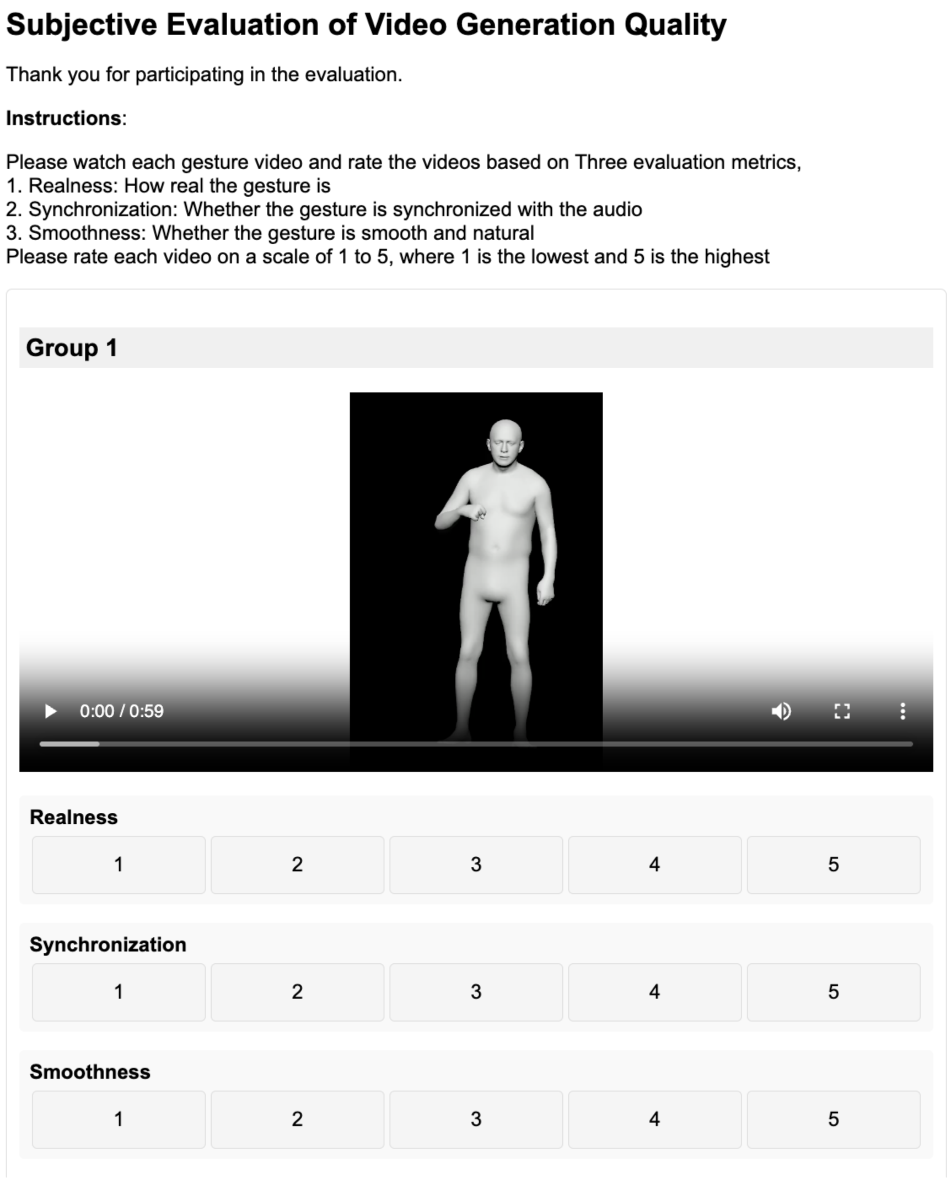
Fréchet Gesture Distance (FGD). We adopt Fréchet Gesture Distance Yoon et al. (2020) to quantify the distributional similarity between real and generated gestures. Inspired by FID in image generation, FGD compares mean and covariance statistics of latent features extracted from a pretrained network:

$$\mathrm{FGD} = \left\|\mu_r - \mu_g\right\|^2 + \mathrm{Tr}\left(\Sigma_r + \Sigma_g - 2(\Sigma_r\Sigma_g)^{1/2}\right), \tag{5}$$

where (μ_r, Σ_r) and (μ_g, Σ_g) are the empirical means and covariances of real and generated gesture embeddings, respectively. Lower FGD indicates better realism and distributional alignment.

L1 Diversity (Div.). To assess sample-level variation, we compute L1 Diversity Li et al. (2021a), defined as the average pairwise L1 distance across N generated sequences:

$$\text{L1 Diversity} = \frac{1}{2N(N-1)} \sum_{t=1}^{N} \sum_{j=1}^{N} \left\| p_t^i - \hat{p}_t^j \right\|_1, \tag{6}$$



Subjective Evaluation of Video Generation Quality

Thank you for participating in the evaluation.

Instructions:

Please watch each gesture video and rate the videos based on Three evaluation metrics,
1. Realness: How real the gesture is
2. Synchronization: Whether the gesture is synchronized with the audio
3. Smoothness: Whether the gesture is smooth and natural
Please rate each video on a scale of 1 to 5, where 1 is the lowest and 5 is the highest

Group 1

Realness

Synchronization

Smoothness

Figure 10: User Study Screenshot

where p_t^i and $\hat{p}_t^j$ denote the joint positions at frame t for the i-th and j-th sequences. To focus on local articulation, global translation is removed before computing distances.

Beat Constancy (BC). Beat Constancy Li et al. (2021b) measures rhythmic alignment between gesture dynamics and speech. Motion beats are detected as local minima in upper body joint velocity, while speech onsets define audio beats. BC is computed as:

$$\text{BC} = \frac{1}{|g|} \sum_{b_g \in g} \exp \left(-\frac{\min_{b_a \in a} \|b_g - b_a\|^2}{2\sigma^2} \right), \tag{7}$$

where g and a are the sets of gesture and audio beats, respectively. BC closer to ground-truth implies stronger gesture-speech synchronization.

H ETHICAL STATEMENT

While our work is centered on generating human motion videos, it raises ethical concerns due to its potential misuse for photorealistic human motion retargeting. We emphasize the importance of responsible use and recommend implementing practices such as watermarking and deepfake detection to mitigate the risks involving deepfake videos and animated representations.

I REPRODUCIBILITY STATEMENT

We have provided the code of algorithmic annotation for the motion pattern analysis in the supplementary material together with the code for the whole system.

J THE USE OF LARGE LANGUAGE MODELS

We utilize Large Langauge Models for the dataset annotation and paper polishing.

K LIMITATIONS

While our framework demonstrates strong performance across alignment, tokenization, and gesture generation, several limitations remain.

First, our method relies on pre-annotated sentence-level intention descriptions to guide semantic learning. This setup assumes that such annotations are either available or can be reliably extracted, which may not hold in less curated or low-resource scenarios. Future work could explore unsupervised or weakly supervised intention discovery to broaden applicability.

Second, while the multi-codebook tokenizer introduces structure into the latent space, it does not guarantee complete disentanglement between semantic and rhythmic dimensions. Investigating more principled inductive biases or factorized token learning may improve interpretability and controllability.

Third, as shown in Sec.E, we discover that existing methods present error propagation issues for long-sequence generation settings. We would like to highlight this issue and hope future works can propose solutions for this fundamental issue for the co-speech gesture generation domain.

Fourth, in this work, while the motion description annotation, gesture-behavior function mapping are intermediate outputs during the annotation procedure, they are not input as variables for the motion control but only intention annotations were utilized. We build this simple baseline because during inference procedure, we are not able to obtain these motion relationed analysis. However, we argue that the values of these annotations should not be ignore and hope future works can further explore the use cases of these annotations as well for motion control and inspire the analysis of the relationships between gesture motion patterns and linguistic cues from speech context.

Finally, although our hierarchical alignment improves generalization across speakers, domain shifts—such as significant accent variation, disfluency, or cultural gesture norms—remain challenging. Incorporating domain adaptation techniques or cross-cultural gesture modeling could enhance robustness in real-world deployments.