Supplementary Material for "Zero-MELO: Test-Time Evidence Calibration with Multimodal LLMs for Zero-Shot Micro-Gesture Recognition"

Anonymous Author(s)
Submission Id: 6240

A Misaligned Cases in Prompt-Based Free-Form Generation of Close-Source MLLMs

This section provides additional examples showing that, under prompt-based free-form generation, closed-source MLLMs may produce misaligned labels, i.e., the labels whose names do not exactly match the predefined label inventory. We illustrate this phenomenon using GPT-4o mini and Gemini2.5-Flash on both iMiGUE and MA52.

We count a clip as mismatched if at least one returned top-k label is not an exact-match dataset label, without applying alias normalization or fuzzy repair. Even when the prompt explicitly requires the model to output exact strings from the provided label set, such misaligned outputs still occur.

Table 1 summarizes the mismatch statistics on the two datasets. On iMiGUE, GPT-4o mini and Gemini2.5-Flash produce exact-string mismatches on 36 and 280 clips, respectively. On MA52, the corresponding numbers are 441 and 195. These statistics are reported only to show that the phenomenon is systematic rather than anecdotal. Table 2 lists several representative examples.

Table 1: Simple statistics of exact-name mismatches in free-form generation. A clip is counted if at least one returned top-k label is not an exact-string-matched label.

Dataset	Model	Total clips	Clips with mismatched labels
iMiGUE	GPT-4o mini	4534	36 (0.79%)
iMiGUE	Gemini2.5-Flash	4534	280 (6.18%)
MA52	GPT-4o mini	5586	441 (7.89%)
MA52	Gemini2.5-Flash	5586	195 (3.49%)

Table 2: Representative misaligned labels observed in prompt-based free-form generation.

Dataset	Model	Representative misaligned labels
iMiGUE	GPT-4o mini	Touching or scratching jaw; Touching forehead; Touching or adjusting hair
iMiGUE	Gemini2.5-Flash	Holding back arms; Adjusting tie; Touch jaw; Touchng jaw
MA52	GPT-4o mini	Bowling head; Scratching or touching head; Raising up; Touching face
MA52	Gemini2.5-Flash	Scratching face; Touching face; Touching chest; Tipping head

B Implementation Details of Cue-Guided Tree Search

This section supplements Sec. 3.3 with the concrete implementation used to instantiate the anchor-wise search operator: search(g_a, c_i), the retained proposal set: $\hat{\mathcal{B}}_{i,a}$, and the aggregation operator: Agg$(\cdot)$. Since the tree search component is inspired by the generic single-image tree-exploration backbone, we do not restate its generic node bookkeeping or image-tree traversal here. Instead, we focus only on the implementation choices that are specific to our video-based MGR setting.

Anchor-Wise Search on Local Temporal Windows. Since the target is a video clip containing people's movements, search is performed on a small anchor set $\mathcal{A}$ on the sampled timeline. For each anchor $a \in \mathcal{A}$, the search tree itself is built on the corresponding anchor frame g_a, i.e., tree nodes remain spatial boxes on a single image. However, each MLLM confidence probe is evaluated on a short anchor-centered local clip rather than on a single snapshot. This design preserves short-term motion context for transient micro-actions while keeping the number of MLLM queries bounded. Inspired by the idea that the onset, apex, and offset frames can describe a micro-expression, we use three anchors in our experiments.

Semantic Misalignment Mitigation. The spatial search space is discretized by a hierarchical patch tree constructed on each anchor frame. A limitation of rigid, non-overlapping patches is the semantic misalignment: small cue regions may straddle partition boundaries and be partially cropped in every child node, reducing cue detectability. To alleviate this, an *overlapping partition* is adopted when splitting a parent box of width W and height H into $k_w \times k_h$ children. Let the parent box parametrized as $B = [\xi_0, \eta_0, W, H]$, where (ξ_0, η_0) denotes the top-left corner of the paraent box, and W and H denote its width and height, respectively. Then, it is split into a $k_w \times k_h$ grid using edge coordinates,

$$\begin{aligned} \xi_u &= \xi_0 + \text{round}\left(\frac{uW}{k_w}\right),\ u = 0, \ldots, k_w, \\ \eta_v &= \eta_0 + \text{round}\left(\frac{vH}{k_h}\right),\ v = 0, \ldots, k_h, \end{aligned} \tag{1}$$

with $\xi_{k_w} = \xi_0 + W$ and $\eta_{k_h} = \eta_0 + H$. For each cell (u, v), denote its base width and height as $w_b = \xi_{u+1} - \xi_u$ and $h_b = \eta_{v+1} - \eta_v$. With overlap ratio $\rho \in [0, 1)$, the cell is expanded symmetrically and clipped such that it lies within the parent box:

$$\begin{aligned} \tilde{\xi}_1 &= \max(\xi_0, \xi_u - \frac{\rho}{2} w_b),\ \tilde{\xi}_2 = \min(\xi_0 + W, \xi_{u+1} + \frac{\rho}{2} w_b), \\ \tilde{\eta}_1 &= \max(\eta_0, \eta_v - \frac{\rho}{2} h_b),\ \tilde{\eta}_2 = \min(\eta_0 + H, \eta_{v+1} + \frac{\rho}{2} h_b). \end{aligned} \tag{2}$$

In this case, the resulting overlapping child box is

$$\tilde{B}_{u,v} = [\tilde{\xi}_1, \tilde{\eta}_1, \tilde{\xi}_2 - \tilde{\xi}_1, \tilde{\eta}_2 - \tilde{\eta}_1] \subseteq B. \tag{3}$$

The terminal tree patch size is set smaller than the final resized MLLM input, which yields finer localization while leaving the downstream visual encoder unchanged.

Node Ranking, Stopping, and Threshold Relaxation. Tree expansion is conducted in a best-first manner. Each queued node is assigned a fast confidence used only for search priority. At shallow depths, the priority score is computed based on a *latent prompt* that asks whether further zooming-in would likely help. At deeper depths, the priority switches to an *existence prompt* that asks whether the cue is already present in the current region. Denoting the node depth by $d(n)$, the priority score is

$$s(n) = \begin{cases} s_{\text{lat}}(n), & d(n) \le d_0, \\ s_{\text{ex}}(n), & d(n) > d_0, \end{cases} \tag{4}$$

where both $s_{\text{lat}}(n)$ and $s_{\text{ex}}(n)$ are obtained from the binary `Yes`/`No` logits of the MLLM. Nodes are re-ranked by $s(n)$ after each expansion. Stopping is decided by a cue-specific adequacy query, i.e., *cue question* applied to the current zoomed-in view. At inference time, the *latent prompt*, *existence prompt*, or *cue question* is combined with a wrapper prompt before being fed to the model.

Anchor-Wise Proposal Retention and Robust Aggregation. For each anchor a, the search returns a candidate set $\mathcal{B}_{i,a}$. After removing duplicate boxes, we retain only a small subset $\hat{\mathcal{B}}_{i,a} \subseteq \mathcal{B}_{i,a}$ ranked by stopping confidence, with depth used as a tie-breaker. The retained proposals are then aggregated across anchors. Let

$$\mathcal{U}_i = \bigcup_{a \in \mathcal{A}} \hat{\mathcal{B}}_{i,a}. \tag{5}$$

Rather than directly unioning all boxes, we first cluster $\mathcal{U}_i$ by pairwise IoU to suppress anchor-specific outliers. If a dominant IoU-consistent cluster exists, only that cluster is retained; otherwise all available boxes are kept. The surviving boxes are then unioned, padded by a small ratio relative to the union box size, and clipped to the image boundary. The resulting box is used as the final cue region b_i. If no reliable proposal is obtained, or if the aggregated box becomes excessively large relative to the full frame, we conservatively fall back to the root region. This yields a temporally stable tube-like crop while remaining robust to occasional anchor failures.

Rendering of the Local Evidence of Each Cue. Once b_i is obtained, the same spatial box is applied to every sampled frame:

$$z_i = \text{CropResize}(g, b_i). \tag{6}$$

Batched Yes/No Evaluation and caching. All confidence computations in Step 1 are binary `Yes`/`No` probes. Nodes sharing the same prompt type and view format are grouped into batches before being sent to the MLLM. Only the logits of the `Yes` and `No` tokens are used to compute the confidence score.

In addition, decoded sampled frames, anchor-centered local windows, rendered node views, and processor-side visual tensors are cached and reused across repeated queries. When multiple candidate labels invoke the same cue, the corresponding search is executed only once, and the resulting cue-conditioned view is reused in later all-label scoring. These optimizations reduce repeated MLLM forwards without changing the search objective.

C Detailed Derivation of Test-Time Calibration

In this section, the detailed derivation of test-time calibration in Sec. 3.4 is discussed. Based on Eq. (5) in the main paper, let $v(y) = \big(v_1(y), \ldots, v_{|v(y)|}(y)\big)$ denote the tokenized label string for the candidate label y, and let $v_{<j}(y) = \big(v_1(y), \ldots, v_{j-1}(y)\big)$. Under teacher forcing, the conditional likelihood factorizes as

$$p_{\mathcal{M}}(v(y) \mid x, t) = \prod_{j=1}^{|v(y)|} p_{\mathcal{M}}\big(v_j(y) \mid v_{<j}(y), x, t\big), \tag{7}$$

and thus

$$\log p_{\mathcal{M}}(v(y) \mid x, t) = \sum_{j=1}^{|v(y)|} \log p_{\mathcal{M}}\big(v_j(y) \mid v_{<j}(y), x, t\big). \tag{8}$$

In our implementation, only the answer tokens in $v(y)$ are scored, while the prompt prefix is masked out. Therefore, the label-wise score is the mean token-level negative log-likelihood

$$\ell_x(y) = -\frac{1}{|v(y)|} \sum_{j=1}^{|v(y)|} \log p_{\mathcal{M}}\big(v_j(y) \mid v_{<j}(y), x, t\big). \tag{9}$$

The prediction is then obtained by

$$y^* = \arg\min_{y \in Y} \ell_x(y). \tag{10}$$

In practice, we compute token-level cross-entropy, which is equivalent to NLL under teacher forcing. Therefore, the label-wise score is

$$\begin{aligned} \ell_x(y) &= \frac{1}{|v(y)|} \sum_{j=1}^{|v(y)|} \text{CE}\Big(\delta_{v_j(y)}, p_{\mathcal{M}}(\cdot \mid v_{<j}(y), x, t)\Big) \\ &= -\frac{1}{|v(y)|} \sum_{j=1}^{|v(y)|} \log p_{\mathcal{M}}\big(v_j(y) \mid v_{<j}(y), x, t\big), \end{aligned} \tag{11}$$

where $\delta_{v_j(y)}$ denotes the one-hot target distribution of the j-th answer token.

To diagnose the bias, we leverage the introduced x_b and x_r to obtain the corresponding scores $s_{x_b}(y)$ and $s_{x_r}(y)$:

$$s_{x_b}(y) \triangleq -\ell_{x_b}(y), \qquad s_{x_r}(y) \triangleq -\ell_{x_r}(y). \tag{12}$$

The inference based on x_b reflects language prior, while the inference based on x_r captures motion-agnostic appearance prior beyond language prior via

$$\Delta_r(y) = s_{x_r}(y) - s_{x_b}(y). \tag{13}$$

In this sense, blank and lowrank calibrations expose two partially separable prior-driven bias components in label scoring.

Let y_t and y_w denote the ground-truth and baseline prediction, respectively. We quantify bias as

$$A_{\text{blank}}(x) = s_{x_b}(y_w) - s_{x_b}(y_t), \quad A_{\text{lowrank}}(x) = \Delta_r(y_w) - \Delta_r(y_t), \tag{14}$$

revealing language and appearance biases, respectively. In other words, a positive $A_{\text{blank}}(x)$ indicates that the blank reference favors the wrong label over the true label. A positive $A_{\text{lowrank}}(x)$ indicates that, relative to the blank reference, the lowrank reference provides more support to the wrong label than to the true label. These quantities are used in SM F.2 to analyze the separability of the two prior components.

Based on this diagnosis, we introduce the following test-time calibration:

$$\tilde{s}_{x_b}(y) = s_x(y) - s_{x_b}(y), \quad \tilde{s}_{x_r}(y) = s_x(y) - s_{x_r}(y). \tag{15}$$

The prediction under each calibration is given by

$$y_b^* = \arg\max_y \tilde{s}_{x_b}(y), \quad y_r^* = \arg\max_y \tilde{s}_{x_r}(y). \tag{16}$$

Equivalently, in NLL form,

$$\tilde{\ell}_{x_b}(y) \triangleq \ell_x(y) - \ell_{x_b}(y), \qquad \tilde{\ell}_{x_r}(y) \triangleq \ell_x(y) - \ell_{x_r}(y). \tag{17}$$

If a label remains highly preferred even under the blank reference, it is likely dominated by language bias and is penalized by $\tilde{s}_{x_b}(y)$. Likewise, if a label is strongly supported by static appearance alone, it is penalized by $\tilde{s}_{x_r}(y)$. Therefore, the two calibrations correct different bias tendencies in the label scoring space.

D Detailed Derivation of Multi-Cue Fusion

Consider the following approximate conditional factorization:

$$p(y \mid x_1, \dots x_n) \approx \prod_{i=1}^{n} p(y \mid x_i). \tag{18}$$

A Naive Bayes approximation is adopted, meaning that the local views are treated as independent evidence sources conditioned on (y, g):

$$p(z_{1:n} \mid y, g) \approx \prod_{i=1}^{n} p(z_i \mid y, g). \tag{19}$$

By Bayes' rule, the posterior can be factorized as

$$p(y \mid g, z_{1:n}) \propto p(y \mid g) \prod_{i=1}^{n} p(z_i \mid y, g). \tag{20}$$

Based on Bayes' theorem,

$$p(y \mid g, z_i) = \frac{p(z_i \mid y, g) p(y \mid g)}{p(z_i \mid g)}. \tag{21}$$

Substituting Eq. (21) into Eq. (20), and absorbing the terms independent of y (e.g., $\prod_i p(z_i \mid g)$) into the proportionality constant, we obtain:

$$p(y \mid g, z_{1:n}) \propto p(y \mid g) \prod_{i=1}^{n} \frac{p(y \mid g, z_i)}{p(y \mid g)} = p(y \mid g)^{1-n} \prod_{i=1}^{n} p(y \mid g, z_i). \tag{22}$$

Implementation in Calibrated NLL Space. In particular, $\bar{\ell}_x(y)$ denotes $\bar{\ell}_g(y)$ for global view input (whole frame) and $\bar{\ell}_i(y)$ for a zoomed-view input (cropped local evidence).

$$\bar{\ell}_g(y) = -\tau_d \log\left(\pi_g \exp\left(-\frac{\tilde{\ell}_{g_b}(y)}{\tau_d}\right) + (1-\pi_g)\exp\left(-\frac{\tilde{\ell}_{g_r}(y)}{\tau_d}\right)\right), \tag{23}$$

and

$$\bar{\ell}_i(y) = -\tau_d \log\left(\pi_c \exp\left(-\frac{\tilde{\ell}_{i,b}(y)}{\tau_d}\right) + (1-\pi_c)\exp\left(-\frac{\tilde{\ell}_{i,r}(y)}{\tau_d}\right)\right). \tag{24}$$

Hence, $\bar{\ell}_x(y)$ is a temperature-smoothed lower envelope of the two bias calibration energies, which will be used in the subsequent fusion.

Let

$$\bar{p}_x(y) \triangleq \frac{\exp(-\bar{\ell}_x(y))}{\sum_{y' \in \mathcal{Y}} \exp(-\bar{\ell}_x(y'))} \tag{25}$$

be the surrogate posterior induced by $\bar{\ell}_x(y)$. Substituting Eq. (25) into the factorization in Eq. (22) yields, up to a Naive-Bayesian (NB) label-independent constant,

$$\ell_{\mathrm{NB}}(y) \doteq \sum_{i=1}^{n} \bar{\ell}_i(y) - (n-1)\bar{\ell}_g(y) = \bar{\ell}_g(y) + \sum_{i=1}^{n} \big(\bar{\ell}_i(y) - \bar{\ell}_g(y)\big), \tag{26}$$

which shows that the local evidence of each cue contributes a residual correction relative to the shared global branch.

Label-Conditioned Normalized Fusion. Eq. (26) is not yet suitable as the final rule, because not every cue is semantically relevant to every label, and direct accumulation scales with both the number and the correlation of compatible cues. To account for this, let c_i denote the semantic type of cue z_i, and let $C(y)$ be a fixed label-conditioned compatibility set. The set of cue indices is defined based on anatomy:

$$\mathcal{I}(y) \triangleq \left\{ i \in \{1, \dots, n\} \;:\; c_i \in C(y) \text{ and } \bar{\ell}_i(y) < \infty \right\}. \tag{27}$$

For each cue branch, a non-negative family coefficient is assigned:

$$w_i(y) \triangleq \lambda_{\phi(c_i)}, \tag{28}$$

where $\phi(c_i)$ maps the cue type to its family, and $\lambda_{\phi(c_i)} \geq 0$ is the corresponding family coefficient.

With $\mathcal{I}(y)$ and $w_i(y)$ defined above, the accumulation implied by Eq. (26) is adapted to the label-dependent setting as

$$r(y) \triangleq \sum_{i \in \mathcal{I}(y)} w_i(y)\big(\bar{\ell}_i(y) - \bar{\ell}_g(y)\big). \tag{29}$$

Here, $r(y)$ collects the additional evidence contributed by the compatible cue branches relative to the global branch.

Since the number of compatible cues may vary across labels, directly using $r(y)$ would make the scale of the correction depend on the amount of local evidence. We therefore normalize $r(y)$ by the total active cue weight and define

$$\ell_{\mathrm{fuse}}(y) \triangleq \bar{\ell}_g(y) + \frac{r(y)}{1 + \sum_{i \in \mathcal{I}(y)} w_i(y)}. \tag{30}$$

Equivalently,

$$\ell_{\mathrm{fuse}}(y) = \frac{\bar{\ell}_g(y) + \sum_{i \in \mathcal{I}(y)} w_i(y)\bar{\ell}_i(y)}{1 + \sum_{i \in \mathcal{I}(y)} w_i(y)}. \tag{31}$$

Therefore, each candidate label $y \in \mathcal{Y}$ is assigned a fused scalar score $\ell_{\mathrm{fuse}}(y)$, where the fusion follows a shared normalized residual form but activates only the cues compatible with y. The final prediction is obtained by ranking the resulting label-wise fused energies over the full label set:

$$y_{\mathrm{final}} = \arg\min_{y \in \mathcal{Y}} \ell_{\mathrm{fuse}}(y). \tag{32}$$

Equivalently, the fused energies induce the surrogate posterior

$$\hat{p}_{\mathrm{fuse}}(y) = \frac{\exp(-\ell_{\mathrm{fuse}}(y))}{\sum_{y' \in \mathcal{Y}} \exp(-\ell_{\mathrm{fuse}}(y'))}, \tag{33}$$

so that

$$y_{\mathrm{final}} = \arg\max_{y \in \mathcal{Y}} \hat{p}_{\mathrm{fuse}}(y) = \arg\min_{y \in \mathcal{Y}} \ell_{\mathrm{fuse}}(y). \tag{34}$$

Thus, the final decision is made by a global ranking of the label-wise fused energies, while the local evidence entering each score remains label-conditioned through the compatibility set $\mathcal{I}(y)$.

Implementation of Verbalizer Marginalization. To better separate semantically close labels, such as *Scratching or touching neck* and *Scratching or touching chest*, we use two verbalizers for each candidate label y: the raw label string $v^{\text{lab}}(y)$ and a short description $v^{\text{desc}}(y)$. For any branch input x in Sec. 3.4, including the global branch g and a cue-conditioned branch $x_i = (g, z_i)$, we compute two teacher-forced scores of g and x_i by Eq. 11, denoted by $\ell_x^{\text{lab}}(y)$ and $\ell_x^{\text{desc}}(y)$, respectively.

We then define a single branch-wise label energy by marginalizing over the two verbalizers:

$$\ell_x^{\text{vm}}(y) = -\log\Big(\alpha_y \exp\Big(-\ell_x^{\text{lab}}(y)\Big) + (1-\alpha_y)\exp\Big(-\ell_x^{\text{desc}}(y)\Big)\Big), \tag{35}$$

where the coefficient $\alpha_y \in (0,1)$ controls the relative contribution of the raw label verbalizer and the description verbalizer.

This marginalized score is used in place of $\ell_x(y)$ in the calibration stage. Accordingly,

$$\tilde{\ell}_{x_b}^{\text{vm}}(y) = \ell_x^{\text{vm}}(y) - \ell_{x_b}^{\text{vm}}(y), \qquad \tilde{\ell}_{x_r}^{\text{vm}}(y) = \ell_x^{\text{vm}}(y) - \ell_{x_r}^{\text{vm}}(y), \tag{36}$$

and Eq. (9) in the main paper is applied in the same way to obtain the calibrated NLL $\bar{\ell}_x^{\text{vm}}(y)$.

Importantly, this implementation does not change the mathematical form of the multi-cue fusion in Sec. 3.5. The fusion derivation only requires each branch to provide a calibrated label-wise energy. Therefore, after replacing $\bar{\ell}_g(y)$ and $\bar{\ell}_i(y)$ by their marginalized counterparts $\bar{\ell}_g^{\text{vm}}(y)$ and $\bar{\ell}_i^{\text{vm}}(y)$, Eq. (14) in the main paper remains valid:

$$\ell_{\text{fuse}}^{\text{vm}}(y) = \frac{\bar{\ell}_g^{\text{vm}}(y) + \sum_{i\in\mathcal{I}(y)} w_i(y)\bar{\ell}_i^{\text{vm}}(y)}{1 + \sum_{i\in\mathcal{I}(y)} w_i(y)}. \tag{37}$$

Thus, verbalizer marginalization only refines how each label NLL is constructed before calibration and fusion, while leaving the overall calibration and multi-cue fusion framework mathematically unchanged.

E Experimental Setup

E.1 Experimental Configurations

This section provides the experimental configurations. The complete cue sets in iMiGUE and MA52 are listed below. We use cues based on the anatomy and the body association.

iMiGUE. The cues used in iMiGUE are the following:

$$C = \left\{\begin{matrix}\texttt{arms, body-hand, face, fingers, hands,}\\ \texttt{head-hand, mouth, neck, shoulders, torso}\end{matrix}\right\}, \tag{38}$$

where body-hand and head-hand denote the interactions between the body and hands, and between the head and hands, respectively. For each label y, the label-conditioned cue set $C(y)$ is fixed as shown in Table 5. There is no corresponding cue for *Illustrative BLs*, since this class is not defined as an MG in iMiGUE. For the remaining tree-search parameters, we set the patch size of each node to 224. The k_w and k_h are both set to 2, and the overlap ratio is defined as 0.25. The tree-search batch size of the tree search is fixed to be 8. The label descriptions were generated by GPT-5.4-thinking and are provided in Table 14. The parameters related to multi-cue fusion are listed in Table 3.

Table 3: Parameters of multi-cue fusion in iMiGUE.

Parameter	Qwen2.5-VL	InternVL3.5	MiMo-VL
τ_d	0.9	0.7	1.1
π_g	0.5	0.5	0.5
π_c	0.5	0.5	0.5
α_y	0	0.85	0.97
w_{head}	0.2	1.0	1.5
w_{hand}	1.0	0.8	1.12
w_{torso}	0.2	0.4	0.1
w_{other}	0.5	0.2	1.0

Table 4: Parameters of multi-cue fusion in MA52.

Parameter	Qwen2.5-VL
τ_d	0.5
π_g	0.5
π_c	0.5
α_y	0.25
w_{head}	0.7
$w_{upper-limbs}$	0.1
$w_{lower-limbs}$	0.25
w_{body}	0.1

MA52. The cues used in MA52 are the following:

$$C = \left\{\begin{matrix}\texttt{feet, hands, hands and arms, head, head-hand,}\\ \texttt{leg-hand, legs, mouth, neck, torso}\end{matrix}\right\}. \tag{39}$$

The label-conditioned cue set is listed in Table 12. For the tree search parameters, we set the patch size to 112 since the video resolution is lower than that of iMiGUE. The k_w and k_h are also set to 2, and the overlap ratio is defined as 0.25. The label descriptions for MA52 are taken from its official material, as shown in Table 14. The parameters related to multi-cue fusion are listed in Table 4.

E.2 Prompt Design

MGR Prompt. In this work, we did not invest much effort in prompt tuning. We simply use the two base prompts for iMiGUE and MA52 datasets in all free-form generation experiments. The prompt for iMiGUE is listed below. Here, `{labels}` denotes the full label set of the target dataset.

```
Which micro-gesture best describes the person
in this video clip? When deciding, pay attention
to visible cues from: hands, fingers, face,
mouth, eyes, head, neck, shoulders, arms, and
torso.
Now do TOP-5 instead of TOP-1:
Return exactly 5 UNIQUE labels from the provided
LABEL SET, ranked from most likely (#1) to less
likely (#5).
Each label must be an EXACT string from LABEL
SET. Do not output anything except the JSON
object below.
```

Table 5: Label-conditioned cue sets $C(y)$ for iMiGUE.

Label y	Label-Conditioned Cue Set $C(y)$	Label y	Label-Conditioned Cue Set $C(y)$
Turtle neck	{neck}	Folding arms	{body-hand, arms}
Bulging face, deep breath	{face}	Dusting off clothes	{body-hand, arms}
Touching hat	{head-hand}	Putting arms behind body	{arms}
Touching or scratching head	{head-hand}	Moving torso	{torso}
Touching or scratching forehead	{head-hand, face}	Sitting straightly	{torso}
Covering face	{head-hand, face}	Scratching or touching arms	{arms, hands}
Rubbing eyes	{head-hand, face}	Rubbing or holding hands	{hands}
Touching or scratching facial parts	{face, head-hand}	Crossing fingers	{fingers, hands}
Touching ears	{head-hand, face}	Minaret gesture	{fingers, hands}
Biting nails	{head-hand, mouth}	Playing or manipulating objects	{hands}
Touching jaw	{mouth, head-hand, face}	Hold back arms	{arms}
Touching or scratching neck	{neck, head-hand}	Head up	{neck, face}
Playing or adjusting hair	{head-hand, face}	Pressing lips	{mouth, face}
Buckle button, pulling shirt collar, adjusting tie	{body-hand}	Arms akimbo	{arms}
Touching or covering suprasternal notch	{body-hand, neck}	Shaking shoulders	{shoulders, torso}
Scratching back	{body-hand}	Illustrative BLs	∅

```
Before outputting, verify that "labels" has
exactly 5 unique non-empty strings. If not, fix
it.
LABEL SET:
{labels}
Output JSON only:
{
   "sequence": "video clip",
   "labels": ["<rank1>", "<rank2>", "<rank3>",
"<rank4>", "<rank5>"]
}
```

The prompt for MA52 is analogous, with minor dataset-specific modifications.

```
Which micro-movement best describes the person
in this video clip? When deciding, pay attention
to visible cues from: hands, fingers, face,
mouth, eyes, head, neck, shoulders, arms, torso,
legs, feet.
Now do TOP-5 instead of TOP-1:
Return exactly 5 UNIQUE labels from the provided
LABEL SET, ranked from most likely (#1) to less
likely (#5).
Each label must be an EXACT string from LABEL
SET. Do not output anything except the JSON
object below.
Before outputting, verify that "labels" has
exactly 5 unique non-empty strings. If not, fix
it.
LABEL SET:
{labels}
Output JSON only:
{
   "sequence": "video clip",
   "labels": ["<rank1>", "<rank2>", "<rank3>",
"<rank4>", "<rank5>"]
}
```

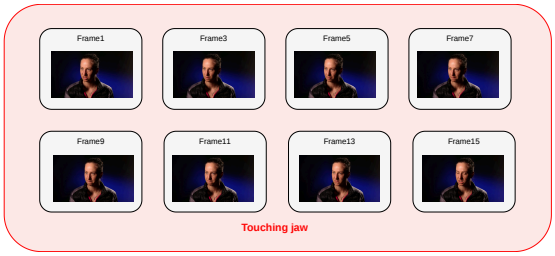
Notably,to encourage the MLLM to return exactly five unique labels from the predefined label set and to reduce misaligned outputs, we explicitly impose these constraints in the prompt.

For NLL-based inference, we use a semantically aligned prompt but remove these output-format requirements. The MGR prompt will be combined with a wrapper prompt before being fed to the MLLM.

```
Which micro-gesture best describes the person
in this video clip? When deciding, pay attention
to visible cues from: hands, fingers, face,
mouth, eyes, head, neck, shoulders, arms, and
torso.
Candidate labels:
{labels}
Respond with exact one label from the candidate
list.
```

However, we observe that if the label list is not explicitly added in the prompt, the results of NLL-based Qwen2.5-VL become substantially different, as shown in Table 6. If the label list is explicitly added, the prediction-collapse problem will be amplified. In this context, we choose the prompt variant that yields lower performance but remains semantically closer to the free-form generation setting.

Tree Search Prompt. In tree search, there are three question prompts and two wrapper prompts which are combined with the question prompts, as discussed in SM B. These prompts differ between Qwen2.5-VL and the other backbones, since the models exhibit different sensitivities to prompt wording. All of them are listed in Table 13.



(a)

(b)

Figure 1: A failure case of free-form generation, analogous to Fig. 3. The GT of this sample is *Pressing lips*. (a): When only the original video clip is provided, the vanilla MLLM predicts *Touching jaw*. (b): When additional local evidence of the mouth is provided, the vanilla MLLM will predict *Pressing lips*.

F Complete Analysis of MLLM Failures in MGR

F.1 Analysis of Free-Form Generation Results

This section provides a failure case of free-form generation, while the correct prediction is obtained by additionally providing the local evidence.

F.2 Analysis of NLL-Based Label Scoring Results

To further quantify the second observation in Sec. 4.2, we analyze the NLL-based label scoring results in the bias-diagnosis space defined in SM C. Specifically, $A_{\text{blank}}(x)$ and $A_{\text{lowrank}}(x)$ measure how strongly the blank and lowrank references favor the baseline wrong label y_w over the ground-truth label y_t, respectively. Therefore, a positive A_{blank} indicates language prior bias, while a positive A_{lowrank} indicates appearance bias beyond the blank reference. Since the calibration subtracts the corresponding reference (x_b/x_r) score, larger positive values imply that the corresponding calibration method removes stronger wrong-label support.

At the sample level, Fig. 3 shows that the two calibrations correct overlapping yet clearly distinct subsets of errors. Among the baseline-wrong samples, blank calibration solely corrects 6.1%, lowrank calibration corrects 5.9% on its own, and both calibrations correct 2.3%, while 85.8% remain wrong. Accordingly, blank and lowrank calibrations correct 8.4% and 8.2% of the baseline-wrong samples in total, respectively. However, only 16.1% of the corrected samples are shared by the two branches. This quantitatively supports the statement in Sec. 4.2 that the two calibrations are complementary rather than redundant. At the same time, the large unresolved portion shows that bias correction alone is insufficient to solve zero-shot MGR, which is consistent with the need for localized evidence acquisition.

Figs. 2 and 4 further show that this complementarity is strongly class-dependent under the long-tailed label distribution. Blank calibration is particularly effective on classes whose errors are more strongly driven by the language prior, such as *Crossing fingers* (85.1% of baseline-wrong samples corrected by blank calibration), *Folding arms* (58.0%), and *Touching or scratching facial parts* (41.9%). In contrast, lowrank calibration is relatively stronger on classes where static appearance induces a different bias pattern, such as *Rubbing eyes* (72.0%), *Touching jaw* (45.7%), *Touching ears* (62.2%), and even *Illustrative BLs* (8.0%) in the most error-dominant class. For each class, we compare blank and lowrank calibration by their class-wise correction effect, defined as the corrected fraction minus the harmed fraction within that class. By this criterion, lowrank calibration is better on 41.9% of the classes, blank calibration is better on 25.8%, and they are comparable on the remaining 32.3%. Fig. 2 also shows that over-correction is concentrated in a few rare categories with very small baseline-correct subsets, rather than being the dominant pattern. Therefore, the main signal is not a uniform advantage of one calibration over the other, but rather a class-dependent trade-off between the two.

The mechanism plots in Fig. 5a and Fig. 5b help explain why the two calibrations behave differently. For samples corrected only by blank calibration, A_{blank} is clearly positive and large (mean 1.21, median 1.37), whereas A_{lowrank} is slightly negative on average (mean -0.14, median -0.17). This indicates that these failures are mainly dominated by language prior bias, while the appearance-bias term does not provide additional wrong-label preference. By contrast, lowrank-only fixes have a much larger positive A_{lowrank} (mean 0.92, median 1.03) together with a smaller and more dispersed A_{blank} (mean 0.86, median 0.29), showing that these errors are better explained by appearance bias exposed by the lowrank reference. Samples corrected by both calibrations have the largest A_{blank} (mean 1.53) and still positive A_{lowrank} (mean 0.23), indicating that some failures simultaneously contain both language and appearance bias. Even the unresolved group remains positive on average in both quantities ($A_{\text{blank}} = 0.86$, $A_{\text{lowrank}} = 0.59$), suggesting that residual bias is still present, although bias correction alone is often insufficient to reverse the ranking.

Overall, the quantitative diagnosis in this section is consistent with the second paragraph of Sec. 4.2. Vanilla MLLMs suffer not only from insufficient localized evidence, but also from two partially separable prior components in label scoring: language-prior bias revealed by the blank calibration and appearance bias revealed by the lowrank calibration. Because these two components are only partially overlapping and strongly class-dependent, the calibration-aggregating formulation in Sec. 3.4 is better justified than choosing either reference alone. This also explains why lowrank-only calibration can be weaker than blank-only calibration in overall performance, yet still remains useful after aggregation, since it corrects a different subset of samples and classes.

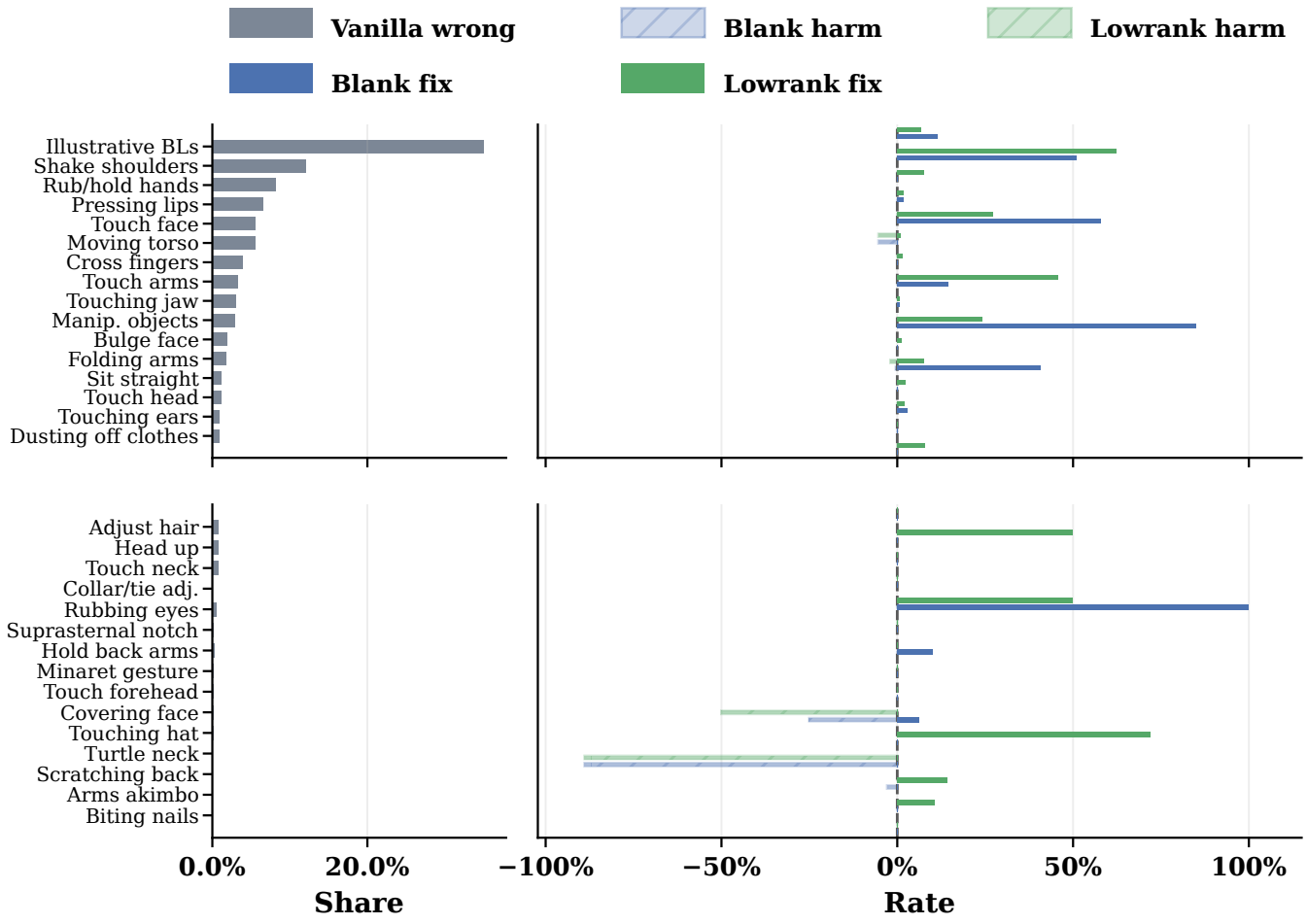


Figure 2: Class-wise effect of blank and lowrank calibration in NLL-based label scoring. Left: the ground-truth class share of samples that are misclassified by the vanilla model (*vanilla wrong*). Right: per-class calibration effect, where *fix* means that a sample is originally misclassified by the vanilla model but becomes correct after the corresponding calibration method, and *harm* means that a sample is originally correct under the vanilla model but becomes wrong after calibration. The rates are normalized by class size.

Table 6: Comparison of NLL-based Qwen2.5-VL with and without the label list in the prompt.

Method	mCA@1	mCA@5	MF1
Qwen2.5-VL	0.1667	0.3561	0.0563
Qwen2.5-VL w/ label list	0.0523	0.2524	0.0072

G NLL-Based Label Scoring vs. Prompt-Based Free-Form Generation

This section complements Sec. 4.3 in two aspects. First, we revisit why the raw NLL-based scoring results in Table 1 in the main paper are often weaker than prompt-based free-form generation under the current aligned prompt design. Our goal here is not to explain the internal mechanism of blank calibration. Instead, we use blank calibration as an intervention: if subtracting the blank reference substantially reduces prediction concentration and partially restores performance, then the weak NLL-based scoring results are more plausibly attributed to prior-driven score collapse than to a complete absence of useful visual evidence. Second, we revisit the dataset-related reversal observed in Table 1 in the main paper under raw inference.

For a label set Y, let $p_{\mathrm{Top1}}(y)$ denote the Top-1 prediction histogram over Y. To quantify prediction concentration beyond mCA@1, we additionally report the number of distinct Top-1 labels, the dominant-label ratio $\max_y p_{\mathrm{Top1}}(y)$, the normalized entropy

$$H_{\mathrm{norm}} = -\frac{1}{\log |Y|} \sum_{y \in Y} p_{\mathrm{Top1}}(y) \log p_{\mathrm{Top1}}(y), \tag{40}$$

and the Jensen–Shannon divergence (JSD) between the Top-1 prediction histogram and the ground-truth label histogram. Here, lower

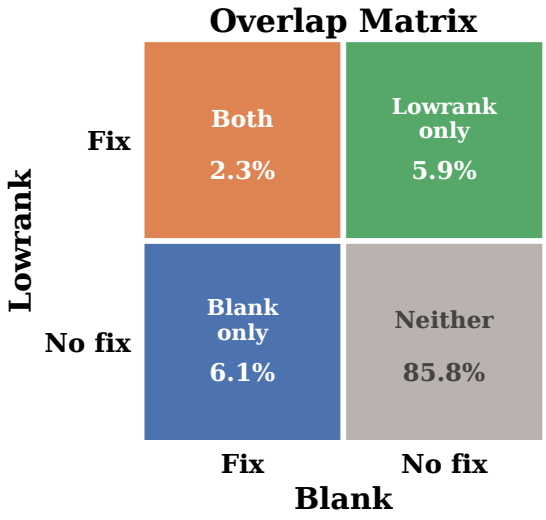


Figure 3: Sample-level overlap of the two calibrations on vanilla-wrong samples. *Both* means that the sample is corrected by both blank and lowrank calibration; *Blank only* and *Lowrank only* mean that only one calibration corrects it; *Neither* means that neither method corrects it. Each cell reports the corresponding percentage within the vanilla-wrong subset.

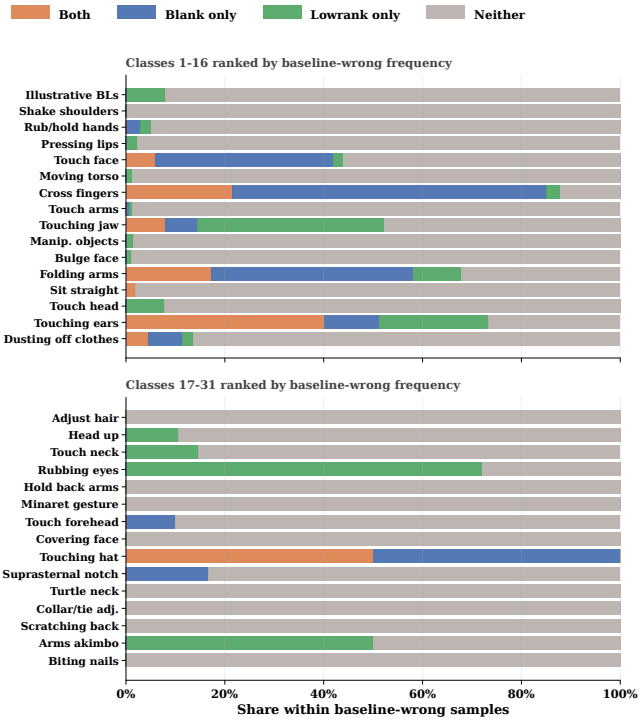


Figure 4: Per-class overlap composition on vanilla-wrong samples. For each class, the bar is decomposed into four groups: *Both*, *Blank only*, *Lowrank only*, and *Neither*, using the same definitions as in Fig. 3. Thus, each bar shows how the two calibrations overlap, or fail to overlap, when correcting errors in that class.

H_{norm} indicates a more collapsed Top-1 prediction distribution with poorer label-space coverage, whereas larger JSD indicates a larger global discrepancy between the model's aggregate Top-1 preference and the GT label histogram.

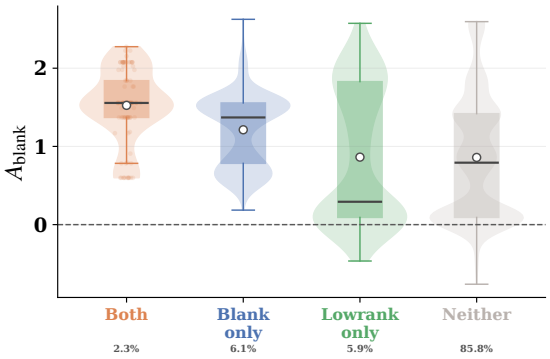
A More Severe Prediction-Collapse Problem in Raw NLL-Based Label Scoring. The main paper already shows that prompt-based free-form generation itself is not free from prediction collapse on iMiGUE. However, Table 7 shows that under the aligned label-list prompt, the raw NLL-based scoring amplifies this effect much more severely. Compared with prompt-based generation, raw NLL-based scoring produces far fewer distinct Top-1 labels, much larger dominant-label ratios, lower normalized entropy, and larger JSD values for all three backbones. Moreover, the dominant Top-1 label under raw NLL-based scoring collapses to the same sink label, *Buckle button, pulling shirt collar, adjusting tie*, for all three backbones. Therefore, the weaker NLL-based scoring performance in Table 1 in the main paper is better understood as a consequence of more severe prediction collapse, rather than simply as a generic inferiority of the NLL scoring path.

Blank Calibration Alleviates the Collapse. Blank calibration does not serve as the object of explanation in this section; instead, it acts as an intervention that tests whether the raw NLL weakness is prior-driven. As shown in Table 7, after blank calibration, the Top-1 distributions become substantially less concentrated for all three backbones: the number of distinct Top-1 labels increases, the dominant-label ratio drops sharply, the normalized entropy rises, and the JSD to the ground-truth distribution consistently decreases. Meanwhile, mCA@1 also recovers substantially. Since the blank reference suppresses visual content and mainly reflects language prior, this consistent reduction in prediction concentration supports the interpretation that a major part of the weak NLL-based scoring performance comes from prior-driven score collapse.

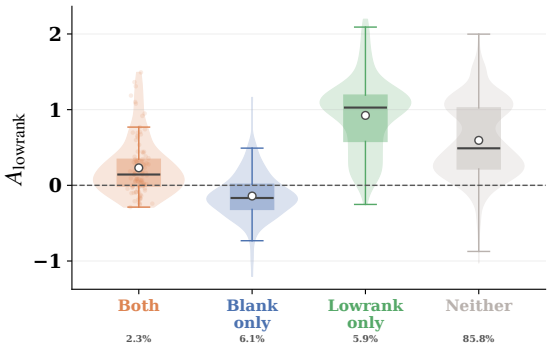
Dataset-Related Reversal under Raw Inference. The dataset-related reversal mentioned in Sec. 4.3 is better discussed under raw inference only. As shown in Table 1 in the main paper, prompt-based free-form generation is consistently stronger on iMiGUE than on MA52 for all three compared backbones, whereas NLL-based scoring shows the opposite cross-dataset preference, i.e., it is consistently stronger on MA52 than on iMiGUE. This counterintuitive reversal should not be interpreted as a simple ranking of visual difficulty between the two datasets. A more plausible explanation is that raw NLL ranking is strongly modulated by dataset-dependent prior structure in the label space. Concretely, on iMiGUE the three backbones collapse toward the same clothing-related sink label, whereas on MA52 the dominant labels shift to more posture- or body-configuration-like categories such as *Sitting straightly* and *Arms akimbo*. This interpretation is also consistent with the ground-truth label histograms, where iMiGUE is substantially more imbalanced than MA52. Therefore, the reversal is more appropriately understood as a change in collapse pattern and prior structure across datasets.

Table 7: iMiGUE prediction-distribution comparison among prompt-based free-form generation, raw NLL-based scoring, and blank-calibrated NLL-based scoring. For prompt-based and raw NLL-based, mCA@1 is aligned with Table 1 in the main paper; the remaining statistics are computed from the corresponding Top-1 histograms. Lower dominant ratio and JSD are better, while higher #Top-1 labels and H_{norm} indicate less concentrated predictions.

Dataset	Model	Mode	mCA@1	#Top-1 labels	Dominant ratio	H_{norm}	JSD	Dominant Top-1 label
iMiGUE	Qwen2.5-VL-7B	Prompt-based	0.1615	19	37.6%	0.503	0.478	Touching or scratching forehead
iMiGUE	Qwen2.5-VL-7B	Vanilla NLL	0.0523	6	89.9%	0.116	0.643	Buckle button, pulling shirt collar, adjusting tie
iMiGUE	Qwen2.5-VL-7B	Blank-cal. NLL	0.1363	23	38.6%	0.614	0.430	Crossing fingers
iMiGUE	MiMo-VL-7B	Prompt-based	0.2445	27	54.1%	0.491	0.339	Rubbing or holding hands
iMiGUE	MiMo-VL-7B	Vanilla NLL	0.0325	2	99.7%	0.006	0.673	Buckle button, pulling shirt collar, adjusting tie
iMiGUE	MiMo-VL-7B	Blank-cal. NLL	0.1573	26	28.9%	0.623	0.310	Touching or scratching facial parts
iMiGUE	InternVL3.5-8B	Prompt-based	0.2034	22	48.2%	0.531	0.388	Rubbing or holding hands
iMiGUE	InternVL3.5-8B	Vanilla NLL	0.0329	2	75.5%	0.161	0.666	Buckle button, pulling shirt collar, adjusting tie
iMiGUE	InternVL3.5-8B	Blank-cal. NLL	0.1768	21	25.1%	0.598	0.337	Touching jaw



(a) Distribution of A_{blank} across the four overlap groups.



(b) Distribution of A_{lowrank} across the four overlap groups.

Figure 5: Bias diagnosis by overlap group. The four groups (*Both*, *Blank only*, *Lowrank only*, and *Neither*) are defined in the same way as in Fig. 3. A_{blank} measures language-prior bias revealed by the blank reference, while A_{lowrank} measures appearance bias revealed by the lowrank reference relative to the blank reference.

Prompt-sensitivity note. For fairness, the NLL-based prompt is chosen to be semantically aligned with the prompt-based setting. However, the main paper already shows that NLL-based scoring is sensitive to this prompt choice when the candidate label list is included. Therefore, the weaker raw NLL results reported here should be interpreted as an observation under the current aligned prompt design, rather than as evidence that NLL-based scoring is intrinsically inferior in all settings.

Table 8: Full comparison between Zero-MELO and supervised discriminative models on the iMiGUE dataset.

Method	mCA@1	mCA@5	MF1
BlockerGCN(J)	0.2324	0.6080	0.2375
BlockerGCN(B)	0.2362	0.6040	0.2407
BlockerGCN(V)	0.2375	0.5820	0.2467
BlockerGCN(3s)	0.2658	0.6469	0.2809
PoseC3D	**0.3402**	**0.7159**	**0.3583**
MMN(J)	0.2991	0.6626	0.3062
MMN(B)	0.2335	0.5953	0.2428
MMN(2s)	0.2890	0.6565	0.3008
Zero-MELO (Ours)	0.2684	0.5775	0.1325

H Full Results of the Comparison between Zero-MELO and Supervised Discriminative Models

We further report the results of supervised discriminative models for reference. As shown in Table 2 in the main paper, Zero-MELO surpasses all variants of BlockerGCN and remains competitive in terms of mCA@1. The gaps to MMN(J) and MMN(2s) are relatively small, whereas PoseC3D remains clearly stronger. Although the MF1 of Zero-MELO remains weaker than that of supervised discriminative models, relatively low MF1 is also commonly observed among MLLM-based methods in Table 1 in the main paper.

I Experiments across Different Backbones

We further evaluate Zero-MELO across different backbones. Specifically, InternVL3.5 and MiMo-VL are implemented on iMiGUE. As shown in Tables 9 and 10, Zero-MELO consistently achieves better results on both backbones. In particular, mCA@1 improves from 20.34% to 24.25% on InternVL3.5 and from 24.45% to 29.77% on MiMo-VL, while mCA@5 and MF1 are also slightly improved.

Table 9: Comparison between Zero-MELO and the corresponding vanilla MLLM under the InternVL3.5 backbone on iMiGUE.

InternVL3.5	iMiGUE		
	mCA@1	mCA@5	MF1
Prompt-based zero-shot	0.2034	0.4358	0.0927
NLL-based	0.0329	0.2408	0.0007
Zero-MELO (Ours)	**0.2425**	**0.4375**	**0.0974**

Table 10: Comparison between Zero-MELO and the corresponding vanilla MLLM under the MiMo-VL backbone on iMiGUE.

MiMo-VL	iMiGUE		
	mCA@1	mCA@5	MF1
Prompt-based zero-shot	0.2445	0.5643	0.1506
NLL-based	0.0325	0.2480	0.0008
Zero-MELO (Ours)	**0.2977**	**0.5834**	**0.1534**

Table 11: Comparison experiments across different calibration strategies.

Method	mCA@1	mCA@5	MF1
Vanilla	0.0357	0.2084	0.0094
+ Cali (Lowrank)	0.1455	0.3717	0.0569
+ Cali (Blank)	0.2040	0.4943	0.0855
+ Cali (Aggregation)	**0.2363**	**0.5436**	**0.1097**

This suggests that the advantage of Zero-MELO is not limited to a single MLLM backbone.

J Experiments across Different Calibration Strategies

We ablate the proposed calibration strategies, namely blank calibration, lowrank calibration, and their aggregation. Notably, the verbalizer marginalization is introduced in these experiments, and thus the results in this section are not the same as them in SM G. As shown in Table 11, the vanilla model performs poorly, indicating that Qwen2.5-VL is overly affected by dominant biases. Using only the blank calibration already yields a large improvement (mCA@1=0.2040, mCA@5=0.4943, MF1=0.0855), while the lowrank calibration is weaker (mCA@1=0.1455, mCA@5=0.3717, MF1=0.0569). This is also consistent with SM F.2. The best results are obtained by aggregating the blank and lowrank calibrations through the aggregation (mCA@1=0.2363, mCA@5=0.5436, MF1=0.1097). This indicates that the two calibrations provide complementary information: the blank calibration offers stronger robustness to language-prior bias, whereas the lowrank calibration contributes additional discriminative structure. Their aggregation produces the most reliable label scores.

K Detailed Implementations of Component-Validity Ablations

In Sec. 4.4, the contributions of the major components of Zero-MELO are studied under a unified evaluation setup. All results use the same label set, the same tree-searched cue-guided views, the same label-conditioned cue compatibility setting introduced in Sec. 3.5, and the same verbalizer marginalization coefficient. The difference lies only in how the final label score is constructed.

Vanilla. This implementation uses the raw global NLL $\ell_g(y)$ directly, without calibration or local evidence. The difference between this implementation and the pure NLL-based vanilla baseline lies in the constrained input format. The pure NLL-based baseline uses a single global video input,

$$y^*_{NLL} = \arg\min_{y \in \mathcal{Y}} \ell(y \mid g, t_{NLL}), \tag{41}$$

where t_{NLL} indicates the normal prompt stated in SM E.2. By contrast, this vanilla implementation reformulates the original global clip into the same paired-input format used by each cue branch. Specifically, the prediction is given by

$$y^*_{vanilla} = \arg\min_{y \in \mathcal{Y}} \ell(y \mid (g, b), t_{zero-melo}) \tag{42}$$

where b is a blank view used as a placeholder, and $t_{zero-melo}$ denotes the combined prompt formed by the zoom wrapper prompt and t_{NLL}. This constrained global input is used to align the input format with that of each cue branch, so that no extra bias is introduced.

+ TS.. This implementation introduces tree-searched local evidence without applying test-time calibration. It uses the global branch as an anchor and refines it with the most supportive compatible local cue branch.

+ Cali (Blank). This implementation uses only the global score with blank calibration, in order to test whether suppressing the language prior alone is sufficient. Result of this calibration is not aligned with that in Table 7, since the verbalizer marginalization technique is leveraged.

+ Cali (Aggregate). This implementation uses the dual-reference calibrated global score in Sec. 3.4, which aggregates the results of blank and lowrank calibrations.

+ TS + Cali (Blank). This implementation combines tree-searched local evidence with blank-only calibrated global scores, but does not yet use the final multi-cue fusion rule. It is worth noting that the **+TS** implementation is not expected to perform strongly on its own since the visual modality is overwhelmed by the language-prior bias. Notably, tree search only produces a set of candidate local views, but in the zero-shot setting, there is no GT prior indicating which single cue branch should be selected as the decisive local evidence for the unknown target label. As a result, single-cue inference after tree search is inherently underdetermined. This is precisely the reason why Zero-MELO introduces a multi-cue fusion module: instead of committing to one potentially unreliable local branch, it fuses multiple compatible local evidence sources on top of the calibrated global anchor.

+ TS + Cali + F(Max). This implementation corresponds to a cue-expert variant, in which the final decision is determined by the most supportive calibrated cue branch over these compatible cue sets. The difference between **+ TS + Cali** and this ablation is that the former is still a single-cue refinement baseline, while the latter is already a multi-cue fusion baseline. Specifically, **+ TS + Cali (Blank)** uses the blank-calibrated global branch as an anchor and refines it with only the most supportive compatible blank-calibrated cue branch. In contrast, **+ TS + Cali + F(Max)** jointly considers the aggregated calibrated global branch and all compatible calibrated cue branches, and performs label-wise expert selection by taking the lowest calibrated score among them.

+ TS + Cali + F(Average). This implementation corresponds to a uniform fusion variant, in which the calibrated global branch and all compatible calibrated cue branches are averaged with equal weight.

+ TS + Cali + F(Ours). This implementation is the full method in Sec. 3.5, namely Label-Conditioned Normalized Fusion, which combines the calibrated global branch and the compatible calibrated cue branches using label-conditioned cue selection and family-dependent weighting.

In this way, the ablation progressively reveals the effects of local evidence acquisition, test-time calibration, and the final multi-cue fusion rule.

Table 12: Label-conditioned cue sets $C(y)$ for MA52.

Label y	Label-Conditioned Cue Set $C(y)$	Label y	Label-Conditioned Cue Set $C(y)$
Shaking body	{torso, legs}	Spread legs	{legs}
Sitting straightly	{torso}	Closing legs	{legs}
Shrugging	{torso}	Crossing legs	{legs}
Turning around	{torso, legs}	Stretching feet	{feet, legs}
Rising up	{torso, legs}	Retracting feet	{feet, legs}
Bowing head	{head, neck}	Tiptoe	{feet, legs}
Head up	{head, neck}	Scratching or touching neck	{hands, neck}
Tilting head	{head, neck}	Scratching or touching chest	{hands, torso}
Turning head	{head, neck}	Scratching or touching back	{hands, torso}
Nodding	{head, neck}	Scratching or touching shoulder	{hands}
Shaking head	{head, neck}	Arms akimbo	{torso}
Scratching arms	{hands, hands and arms}	Crossing arms	{torso}
Playing objects	{hands, hands and arms}	Playing or tidying hair	{hands, head, head-hand}
Putting hands together	{hands, hands and arms}	Scratching or touching hindbrain	{hands, head, head-hand}
Rubbing hands	{hands, hands and arms}	Scratching or touching forehead	{hands, head-hand}
Pointing oneself	{hands, torso, hands and arms}	Scratching or touching face	{hands, head-hand}
Clenching fist	{hands, hands and arms}	Rubbing eyes	{hands, head-hand}
Stretching arms	{hands, hands and arms}	Touching nose	{hands, head-hand}
Retracting arms	{hands, hands and arms}	Touching ears	{hands, head, head-hand}
Waving	{hands, hands and arms}	Covering face	{hands, head-hand}
Spreading hands	{hands, hands and arms}	Covering mouth	{hands, mouth, head-hand}
Hands touching fingers	{hands, hands and arms}	Pushing glasses	{hands, head-hand}
Other finger movements	{hands, hands and arms}	Patting legs	{hands, legs, leg-hand}
Illustrative gestures	{hands, hands and arms}	Touching legs	{hands, legs, leg-hand}
Shaking legs	{legs, feet}	Scratching legs	{hands, legs, leg-hand}
Curling legs	{legs, feet}	Scratching feet	{hands, feet, leg-hand}

Table 13: Prompt templates for region cues used by different backbones. Here, {cue} denotes the queried region cue.

Prompt type	Qwen2.5-VL	InternVL3.5 / MiMo-VL
	Wrapper Prompts	
Global wrapper	<video>	Main video frames: <video>
Zoom wrapper	<video> This is the main video, and the section enclosed by the red rectangle in each frame is the focus region. <video> This is the zoomed-in view of the focus region.	Main video frames (the section enclosed by the red rectangle in each frame is the focus region): <video> Zoomed-in video frames of the focus region: <video>
	Single Cue Prompts	
CQ (region)	In the current view, is the {cue} clearly visible and large enough to follow its motion over time? Answer Yes or No.	Based only on visible evidence, is the person's {cue} visible in this view? Answer Yes or No.
Existence (region)	Is there a {cue} in the current view? Answer Yes or No.	Based only on visible evidence, can you see the person's {cue} in this view? Answer Yes or No.
Latent (region)	In the current view, is the {cue} at least partially visible but too small or blurred, so that zooming in would likely make it clearer? Answer Yes or No.	Is the {cue} visible in the current view but not yet clear enough for precise inspection, so zooming in may help? Answer Yes or No.
	Interaction Cues Prompts	
CQ (interaction)	In the current view, are the {a} and the {b} both visible in the same local area, and is that local area large enough to follow possible contact or relative movement over time? Answer Yes or No.	Based only on visible evidence, are the person's {a} and {b} visible in the same local area of this view? Answer Yes or No.
Existence (interaction)	Are there the {a} and the {b} in the current view? Answer Yes or No.	Based only on visible evidence, can you see the person's {a} and {b} in the same local area of this view? Answer Yes or No.
Latent (interaction)	In the current view, are the {a} and the {b} at least partially visible in the same local area but too small or blurred, so that zooming in would likely make their contact or relative movement clearer? Answer Yes or No.	Are the {a} and the {b} visible in the same local area but not yet clear enough for precise inspection, so zooming in may help? Answer Yes or No.

Table 14: Descriptions used for verbalizer marginalization in iMiGUE and MA52.

Dataset	Label	Description
iMiGUE	Turtle neck	The movement of drawing the head and neck inward toward the shoulders, making the neck appear shortened.
iMiGUE	Bulging face, deep breath	The movement of taking a noticeable deep breath with visible expansion around the face and upper chest.
iMiGUE	Touching hat	The movement of bringing the hand close to and touching or adjusting the hat or its brim.
iMiGUE	Touching or scratching head	The movement of bringing the hand close to and touching or scratching the scalp or top of the head.
iMiGUE	Touching or scratching forehead	The movement of bringing the hand close to and touching or scratching the forehead.
iMiGUE	Covering face	The movement of covering a large portion of the face with one or both hands.
iMiGUE	Rubbing eyes	The movement of bringing the hand close to and rubbing the area around one or both eyes.
iMiGUE	Touching or scratching facial parts	The hand touches or scratches a local facial area such as the cheek or side of the face, not the forehead, ear, or jawline.
iMiGUE	Touching ears	The hand contacts or rubs the ear or earlobe at the side of the head.
iMiGUE	Biting nails	The movement of bringing the fingers close to the mouth and biting the nails.
iMiGUE	Touching jaw	The hand contacts the jawline or chin below the mouth.
iMiGUE	Touching or scratching neck	The hand contacts the front or side of the neck below the jaw.
iMiGUE	Playing or adjusting hair	The hand manipulates or rearranges visible hair strands instead of only touching the scalp or ear.
iMiGUE	Buckle button, pulling shirt collar, adjusting tie	The hand pinches or adjusts clothing near the collar, buttons, tie, or upper chest.
iMiGUE	Buckle button, pulling shirt collar, adjusting tie	The hand pinches or adjusts clothing near the collar, buttons, tie, or upper chest.
iMiGUE	Touching or covering suprasternal notch	The hand touches the small hollow at the base of the neck between the collarbones.
iMiGUE	Scratching back	The movement of reaching to and scratching the back or rear upper-body area.
iMiGUE	Folding arms	The movement of bringing the arms across the front of the torso in a folded or crossed posture.
iMiGUE	Dust offing clothes	The hand brushes or pats the clothing surface to remove dust or lint rather than adjusting the collar or tie.
iMiGUE	Dusting off clothes	The hand brushes or pats the clothing surface to remove dust or lint rather than adjusting the collar or tie.
iMiGUE	Putting arms behind body	The movement of moving one or both arms behind the torso and keeping them there.
iMiGUE	Moving torso	The movement of the upper body shifting, swaying, leaning, or rotating in place.
iMiGUE	Sitting straightly	The dynamic process of straightening the back and upper body into an upright seated posture.
iMiGUE	Sitting straightly	The dynamic process of straightening the back and upper body into an upright seated posture.
iMiGUE	Scratching or touching arms	One hand reaches across to touch or scratch the other arm or forearm.
iMiGUE	Rubbing or holding hands	Both hands stay in direct contact by rubbing, clasping, or holding each other, without clear finger crossing, steepling, or visible object use.
iMiGUE	Crossing fingers	The fingers of the two hands are visibly crossed or interlaced while the hands stay together.
iMiGUE	Minaret gesture	The fingertips of both hands meet in a steeple-like shape while the palms remain apart.
iMiGUE	Playing or manipulating objects	One or both hands visibly handle or adjust an external object held in the hands.
iMiGUE	Hold back arms	The movement of holding the arms back in a restrained posture close to or behind the torso.
iMiGUE	Head up	The movement of raising the head upward, typically as a single motion.
iMiGUE	Pressing lips	The movement of pressing the lips tightly together, usually without hand contact.
iMiGUE	Arms akimbo	The movement of placing one or both hands on the hips with the elbows angled outward.
iMiGUE	Shaking shoulders	The movement of the shoulders shaking or repeatedly moving up and down.
iMiGUE	Illustrative BLs	Communicative hand or arm gestures that accompany speech rather than self-touching or object-focused movements.
MA52	Shaking body	The movement of the upper body from side to side.
MA52	Sitting straightly	The dynamic process of straightening the back.
MA52	Shrugging	Shrugging or moving the shoulders back and forth, indicating movements involving only the shoulders.
MA52	Turning around	The movement of rotating the torso or whole body to change the facing direction.

Dataset	Label	Description
MA52	Rising up	The preparatory movements before standing up from a seated position.
MA52	Bowing head	Lower the head slowly or abruptly, typically as a single motion.
MA52	Head up	Raise the head slowly or abruptly, typically as a single motion.
MA52	Tilting head	Tilt the head toward the shoulder.
MA52	Turning head	Turn the head backward or sideways, typically as a single motion.
MA52	Nodding	The movement of continuous up-and-down head.
MA52	Shaking head	The movement of continuous turn of the head left and right.
MA52	Scratching arms	The movement of scratching the other arm with one hand.
MA52	Playing objects	The movement of playing with or adjusting objects in hand.
MA52	Putting hands together	The movement of bringing both hands closer together until palms together.
MA52	Rubbing hands	The movement of touching and rubbing the hands together.
MA52	Pointing oneself	The movement of the hand that indicates direction and self-reference.
MA52	Clenching fist	The movement of bending the fingers into a fist, including partial clenching of the fingers.
MA52	Stretching arms	The movement of raising or extending the arm forward or sideways.
MA52	Retracting arms	The movement of extending and retracting the arm.
MA52	Waving	The movement of waving one's hand or arm.
MA52	Spreading hands	The movement of holding one's hand open with the palm facing up.
MA52	Hands touching fingers	The movement of changing from having the fingers of both hands uncrossed to interlaced.
MA52	Other finger movements	Excluding finger movements as defined within the upper limb.
MA52	Illustrative gestures	The illustrative gestures made with either the left arm or the right arm.
MA52	Shaking legs	The movement of shaking the leg up and down, either continuously or briefly.
MA52	Curling legs	The movement of sitting with one leg crossed over the other.
MA52	Spread legs	The movement of increasing the distance between the knees.
MA52	Closing legs	The movement of moving the legs from apart to together.
MA52	Crossing legs	The movement of legs gradually turning from uncrossing to crossing.
MA52	Stretching feet	The movement of extending the foot forward or sideways.
MA52	Retracting feet	Retract the foot towards the body.
MA52	Tiptoe	Place the foot sideways or rest the heel on a stool with the toe touching the ground.
MA52	Scratching or touching neck	The movement of bringing the hand close to and touching the neck area.
MA52	Scratching or touching chest	The movement of bringing the hand close to and touching the front of the upper body.
MA52	Scratching or touching back	The movement of bringing the hand close to and touching the back of the upper body.
MA52	Scratching or touching shoulder	The movement of bringing the hand close to or touching the shoulder area.
MA52	Arms akimbo	The movement of placing one or both hands on the hips.
MA52	Crossing arms	The movement of crossing one's arms over one's chest.
MA52	Playing or tidying hair	Hand movements that adjust hair on the sides or top of the head, including women arranging their hanging hair.
MA52	Scratching or touching hindbrain	Touch the back of the head or the hair on the back of the head.
MA52	Scratching or touching forehead	The movement of scratching or touching the forehead.
MA52	Scratching or touching face	The movement of touching the cheek area.
MA52	Rubbing eyes	The movement of bringing the hand close to and touching the area around the eyes.
MA52	Touching nose	The movement of bringing the hand close to and touching the nose area.
MA52	Touching ears	The movement of bringing the hand close to and touching the ear, or touching and rubbing the ear.
MA52	Covering face	The movement of covering a large part of face region, with one or both hands.
MA52	Covering mouth	The movement of covering the mouth with the hand.
MA52	Pushing glasses	The movement of bringing the hand close to and touching the glasses.
MA52	Patting legs	Move a single hand close to and touch the leg in a dynamic process.
MA52	Touching legs	The movement of touching the leg with a single hand.
MA52	Scratching legs	The movement of scratching the lower leg with the hand.
MA52	Scratching feet	The movement of scratching the foot with the hand.