TEST-TIME MATCHING: UNLOCKING COMPOSITIONAL REASONING IN MULTIMODAL MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Multimodal models have achieved remarkable progress, yet recent studies suggest they struggle with *compositional reasoning*, often performing at or below random chance on established benchmarks. We revisit this problem and show that widely used evaluation metrics systematically *underestimate* model capabilities. To address this, we introduce a *group matching score* that better leverages group structure and uncovers substantial hidden competence in both contrastive vision–language models (VLMs) and multimodal large language models (MLLMs). Moreover, simply overfitting to the induced group matchings at test time transfers this hidden competence into higher scores under the original evaluation metric, closing much of the reported gap. With this adjustment, GPT-4.1 becomes the *first system to surpass estimated human performance* on Winoground. Building on this insight, we propose *Test-Time Matching* (TTM), an iterative self-training algorithm that bootstraps model performance without any external supervision. TTM delivers further non-trivial improvements: for example, **SigLIP-B16 with TTM surpasses GPT-4.1 on MMVP-VLM, establishing a new state of the art**. Importantly, TTM is broadly effective even on benchmarks without metric-induced effects or group structures, **achieving relative gains exceeding** 85.7% on challenging datasets such as Whatsup. Across 16 datasets and variants, our experiments consistently demonstrate that TTM unlocks hidden compositional reasoning ability and advances the frontier of multimodal evaluation.

1 INTRODUCTION

Compositional reasoning—correctly binding objects, attributes, and relations across modalities—is a stringent test of multimodal understanding. Recent benchmarks probe this ability by organizing examples into small *groups* of images and captions that differ in subtle but systematic ways (Thrush et al., 2022; Tong et al., 2024; Burapacheep et al., 2024; Hsieh et al., 2023; Kamath et al., 2023). For example, Winoground consists of 2×2 groups where both captions contain the same words but in different orders, so that each caption correctly describes only one of the two images.

Despite their impressive general utility, both contrastive vision-language models (VLMs) and multimodal large language models (MLLMs) have been reported to perform at or below random guessing on these tasks (Thrush et al., 2022; Diwan et al., 2022; Kamath et al., 2023; Tong et al., 2024; Burapacheep et al., 2024; Li et al., 2024). On Winoground, for instance, widely-used models trail far behind the estimated human performance of 85.5 (Thrush et al., 2022), with the previous state of the art reaching only 58.75 via GPT-4V scaffolding and prompt tuning (Wu et al., 2023; Vaishnav & Tammet, 2025).

We revisit this conclusion and show that commonly used metrics can *underestimate* model capability. We introduce a group matching score (GroupMatch) that better leverages group structure by selecting the *best overall matching* rather than requiring each pairwise comparison to succeed, as in the standard group score (GroupScore) (Thrush et al., 2022; Tong et al., 2024; Burapacheep et al., 2024); see Section 3.1 for details. This change alone reveals substantial hidden competence: with simple test-time overfitting to GroupMatch (*simple matching*), GPT-4.1 improves from 69.75 to 91.38 on Winoground and from 68.15 to 88.52 on MMVP-VLM; SigLIP-B16 jumps from 10.25 to 67 on

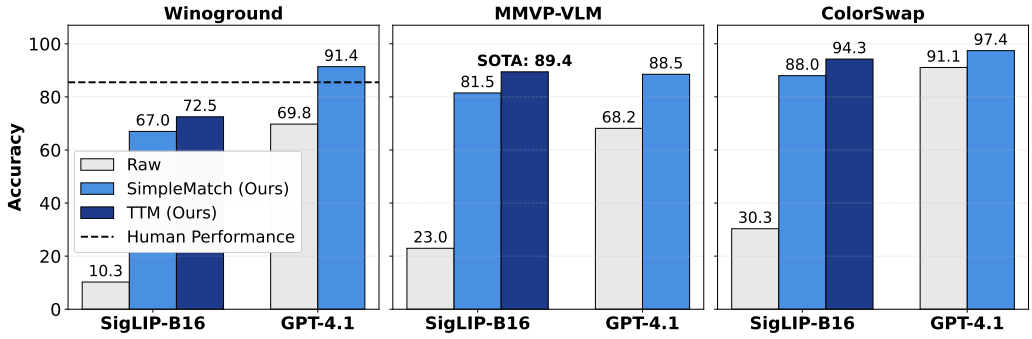

Figure 1: SimpleMatch and TTM significantly boost VLMs and MLLMs performance on compositional reasoning benchmarks Winoground, MMVP-VLM, and ColorSwap. *Left:* GPT-4.1 w/ SimpleMatch surpass human performance on Winoground. *Middle:* SigLIP-B16 w/ TTM outperforms GPT-4.1 on MMVP-VLM, establishing a new state of the art.

Winoground and from 22.96 to 81.48 on MMVP-VLM. Notably, GPT-4.1 becomes the first system to surpass the estimated human performance of 85.5 on Winoground (left plot of Fig. 1).[1]

Building on this insight, we introduce *Test-Time Matching* (TTM), an iterative self-learning algorithm that bootstraps model performance without any external supervision. TTM converts high-confidence model-induced matchings into pseudo-labels and progressively lowers selection thresholds to expand coverage over the test set. This yields *additional, non-trivial* gains on top of GroupMatch: SigLIP-B16 reaches 72.5 on Winoground and 89.44 on MMVP-VLM. Remarkably, TTM elevates SigLIP to match GPT-4.1 on ColorSwap and **surpass GPT-4.1 on MMVP-VLM, establishing a new state of the art.** See middle plot of Fig. 1 for details. Crucially, TTM is broadly effective even where metric changes cannot help—on $1 \times k$ benchmarks such as SugarCrepe (Hsieh et al., 2023) and Whatsup (Kamath et al., 2023), where GroupScore and GroupMatch coincide, TTM still delivers substantial test-time improvements, including **up to** 85.7% **relative gains** on Whatsup (Fig. 3).

Finally, we generalize beyond group-structured datasets by casting evaluation as a single global assignment between all images and captions. Even one-shot global matching outperforms raw GroupScore, and applying the global variant of TTM provides further improvements, demonstrating that the test-time matching principle extends robustly beyond local-group settings.

Contributions. We summarize our main contributions below:

1. **Revisiting evaluation.** We introduce a group matching score (GroupMatch) that better leverages group structures and reveals substantial hidden capability previously masked by GroupScore.
2. **Test-time self-bootstrapping.** We propose TTM, an iterative, supervision-free algorithm that converts confident model-induced matchings into pseudo-labels, yielding significant additional gains at test time across models and datasets.
3. **Broad applicability and new SOTAs.** We demonstrate improvements on 16 dataset variants spanning 2×2, $1 \times k$, and non-grouped settings, achieving new state-of-the-art results (e.g., SigLIP surpassing GPT-4.1 on MMVP-VLM) and the first Winoground result above human performance.

Paper organization. In Section 2, we review group-structured evaluation for compositional reasoning. Section 3 revisits evaluation metrics, introduces the group matching score (GroupMatch), presents our test-time matching (TTM) algorithm, and extends it to global (non-grouped) settings. Section 4 reports results across 2×2, $1 \times k$, and global variants, with ablations and analysis. We conclude in Section 5. We defer related work, formal proofs, and additional experimental details and results to the Appendix.

[1]We use GPT-4.1, the latest GPT model that still provides log probabilities, enabling more accurate computation of similarity scores (Lin et al., 2024). GPT-5 does not currently support log probability outputs.

2 PRELIMINARIES

We study the evaluation of compositional reasoning in multimodal models. Most benchmarks for this task are organized into *groups* of images and captions, typically of shape $k \times k$ or $1 \times k$. Within each group, the images and captions differ in subtle but systematic ways. For example, the widely used Winoground dataset consists of groups with two images and two captions; both captions contain the same set of words but in different orders, so that each caption correctly describes only one of the two images (Thrush et al., 2022).

To succeed on such benchmarks, a model must simultaneously align each image with its correct caption and each caption with its correct image. Formally, let $s_{ij} := s(I_i, C_j)$ denote the similarity score between image I_i and caption C_j. For contrastive vision-language models such as CLIP (Radford et al., 2021) and SigLIP (Zhai et al., 2023), s_{ij} is typically given by the inner product of image and text embeddings. For multimodal large language models, similarity can be derived from metrics such as VQAScore (Lin et al., 2024). We collect all scores into a similarity matrix s, which shares the same shape as the group.

Group Score for $k \times k$ Groups. Consider a group of k images and k captions with ground-truth pairings $\{(I_i, C_i)_i\}$ hidden from the learner. The most widely used evaluation metric is the GroupScore (Thrush et al., 2022; Tong et al., 2024; Burapacheep et al., 2024). This metric outputs 1 if the model finds a bijection that (i) assigns the correct caption to each image and (ii) assigns the correct image to each caption, and 0 otherwise. Mathematically, we have

$$\mathsf{GroupScore}(s) := \begin{cases} 1 & \forall i: \; s_{ii} > \max_{j \neq i} s_{ij} \quad \text{and} \quad s_{ii} > \max_{j \neq i} s_{ji}, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

Group Score for $1 \times k$ Groups. For benchmarks with group shape $1 \times k$ (Kamath et al., 2023; Hsieh et al., 2023), the GroupScore reduces to simpler metrics. If there is a single image and multiple captions, the GroupScore coincides with the TextScore, which equals 1 if the model selects the correct caption. Conversely, if there is a single caption and multiple images, it reduces to the ImageScore, which equals 1 if the model selects the correct image.

3 METHODS

Our approach begins with a re-examination of evaluation metrics for compositional reasoning. We introduce an alternative group matching score that reveals hidden model capability and enables improvements (Section 3.1). Building on this, we propose an iterative test-time matching (TTM) algorithm that bootstraps model performance without external supervision (Section 3.2). We then extend TTM beyond group-structured datasets to a global matching formulation applicable to general settings (Section 3.2.1).

3.1 REVISITING EVALUATION METRICS: FROM RANDOM GUESSING TO MATCHING

Most multimodal compositional reasoning benchmarks adopt the evaluation metrics described in Section 2. Despite the practical utility of multimodal models, existing results show that they perform poorly on these benchmarks—often even *worse than random guessing* (Thrush et al., 2022; Diwan et al., 2022; Kamath et al., 2023; Tong et al., 2024; Burapacheep et al., 2024; Li et al., 2024).[2]

Revisiting evaluation metrics. Such counter-intuitive outcomes motivate us to re-examine the evaluation metrics themselves. To this end, we first consider the performance of a *random guessing model*. Suppose we have a group of k images $\{I_i\}_{i=1}^k$ and k captions $\{C_i\}_{i=1}^k$, with ground-truth pairings $\{(I_i, C_i)\}_{i=1}^k$ hidden from the learner (Thrush et al., 2022; Tong et al., 2024; Burapacheep et al., 2024). For each pair (I_i, C_j), the random guessing model assigns a similarity score $\mathsf{sim}(I_i, C_j) \sim \mathrm{unif}([0, 1])$, producing a similarity matrix $s \in \mathbb{R}^{k \times k}$ with entries $s_{ij} := \mathsf{sim}(I_i, C_j)$.

[2]These benchmarks are widely adopted; for example, Winoground (Thrush et al., 2022) and MMVP (Tong et al., 2024) each have roughly 500 citations at the time of our paper submission.

Under the widely used GroupScore metric, achieving a score of 1 requires the similarity matrix s to satisfy $2k^2 - 2k$ constraints (see Eq. (1)). Equivalently, each diagonal entry s_{ii} must be the largest element in both its row and column—a highly restrictive condition. The probability of achieving a perfect group score under random guessing is given below (see Appendix A.2 for proofs).

Proposition 1. *For random similarity scores s,* $\mathbb{P}(\mathsf{GroupScore}(s) = 1) = \frac{(k-1)!}{(2k-1)!}$.

Group matching score: an alternative metric. We next propose an alternative evaluation metric that evaluates the *best overall matching* rather than individual pairwise comparisons. Let π denote a matching from images to captions, where $\pi(i)$ is the caption assigned to image i. We define the GroupMatch as

$$\mathsf{GroupMatch}(s) := \begin{cases} 1 & \text{if } \sum_{i=1}^{k} s_{i,\pi^\star(i)} > \sum_{i=1}^{k} s_{i,\pi(i)}, \quad \forall\, \pi \neq \pi^\star, \\ 0 & \text{otherwise,} \end{cases}$$

where $\pi^\star : i \mapsto i$ denotes the ground-truth matching. Intuitively, the match score is 1 if the *total similarity* of the correct matching exceeds that of all other possible matchings. For $k = 2$, this reduces to the simple condition $s_{11} + s_{22} > s_{12} + s_{21}$. Since there are $k!$ possible matchings and, under random guessing, each is equally likely to maximize the total score, we obtain the following result.

Proposition 2. *For random similarity scores s,* $\mathbb{P}(\mathsf{GroupMatch}(s) = 1) = \frac{1}{k!}$.

Simple test-time matching: exploiting evaluation gaps. While there is nothing wrong with evaluating models using the popular group score GroupScore, two key observations emerge:

- $\mathbb{P}(\mathsf{GroupMatch}(s) = 1) > \mathbb{P}(\mathsf{GroupScore}(s) = 1)$ for all integers $k > 1$.
- If the correct matching $\pi^\star$ is selected, overfitting to $\pi^\star$ at test time guarantees a group score of 1.

Together, these observations reveal an *arbitrage opportunity*: one can improve the group score by simply overfitting to the matching induced by the GroupMatch at the test-time. We call this method SimpleMatch with GroupMatch. In the commonly studied case with $k = 2$, the expected group score of a random guessing model increases from $1/6$ under GroupScore to $1/2$ under GroupMatch.

Empirical validation. We further evaluate this idea on SigLIP (Zhai et al., 2023) and GPT-4.1 across widely used compositionality benchmarks, including Winoground (Thrush et al., 2022), MMVP-VLM (Tong et al., 2024), and Colorswap (Burapacheep et al., 2024). Results are summarized in Fig. 1. Previously, the best reported Winoground group score was 58.75, achieved by GPT-4V with additional tuning (Wu et al., 2023; Vaishnav & Tammet, 2025). In contrast, SimpleMatch with our proposed GroupMatch allows SigLIP-B16 to reach 67, surpassing this prior SoTA. Even more strikingly, GPT-4.1 improves from 69.75 to 91.38 via SimpleMatch, *becoming the first system to surpass the estimated human performance of 85.5 on this benchmark.*

3.2 Test-Time Matching: Iterative Bootstrapping Model Performance

The alternative metric in Section 3.1 reveals hidden model capability. To push performance further, we introduce an *iterative* test-time matching algorithm that bootstraps model performance and achieves new state-of-the-art results. Our method applies to groups of various shapes (including $k \times k$ and $1 \times k$) and also extends to datasets without group structures (Section 3.2.1).

High-level idea. We present our test-time matching algorithm in Algorithm 1, which proceeds iteratively for T iterations. At each round $t \in [T]$, the current model f_{t-1} induces candidate matchings for all groups, which are treated as pseudo-labels. The algorithm then retains only those matchings it is most confident about, and finetunes on them to obtain the next model f_t. By repeating this process, the model gradually bootstraps itself at test time without any external supervision.

The core of Algorithm 1 lies in two design choices: (1) how pseudo-labels are induced within each group, and (2) how the confidence thresholds are scheduled across iterations. We discuss both below.

Algorithm 1 Test-Time Matching (TTM)

Input: Pretrained f_0; test set of groups $\mathcal{D} = \{G_i\}_{i=1}^n$; number of iterations T; thresholds $\{\tau_t\}_{t=1}^T$.

1: **for** iteration $t = 1$ to T **do**
2: Initialize pseudo-labeled set $\mathcal{S}_t \leftarrow \emptyset$.
3: **for** each group $G_i \in \mathcal{D}$ **do**
4: Induce matching $\pi_{f_{t-1}}(G_i) \leftarrow \arg\max_\pi s(\pi; G_i, f_{t-1})$.
5: Compute margin $\Delta(G_i; f_{t-1})$ as

$$\Delta(G_i; f_{t-1}) \leftarrow s(\pi_{f_{t-1}}(G_i); G_i, f_{t-1}) - \max_{\pi \neq \pi_{f_{t-1}}(G_i)} s(\pi; G_i, f_{t-1}).$$

6: **if** $\Delta(G_i; f_{t-1}) \geq \tau_t$ **then**
7: $\mathcal{S}_t \leftarrow \mathcal{S}_t \cup \{(G_i, \pi_{f_{t-1}}(G_i))\}$.
8: Finetune model on $\mathcal{S}_t$ to obtain f_t. `// Self-training with no external supervision.`

Output: Adapted model f_T.

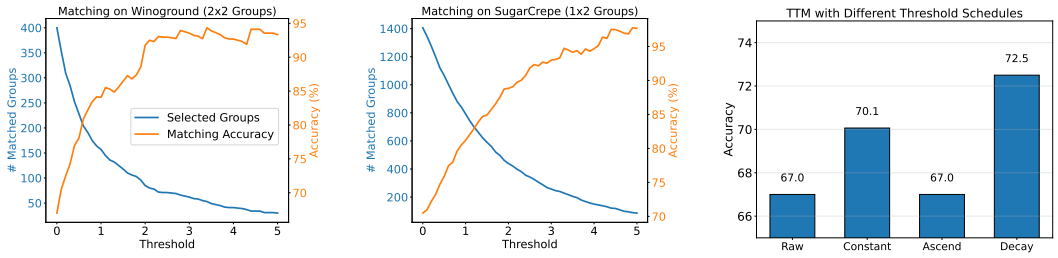

Figure 2: Matching results on Winoground (*left*, with 2×2 groups) and SugarCrepe (*middle*, the replace relation subset, with 1×2 groups) with SigLIP-B16. TTM with different threshold schedules on Winoground with SigLIP-B16 (*right*).

Group matching and pseudo-labeling. For a group G and model f_{t-1}, we define the induced matching

$$\pi_{f_{t-1}}(G) := \arg\max_\pi s(\pi; G, f_{t-1}),$$

where $s(\pi; G, f_{t-1}) := \sum_u s_{u,\pi(u)}(G; f_{t-1})$ denotes the total similarity of matching π on G under f_{t-1}. For example, in a 2×2 group, $\pi_{f_{t-1}}(G) = (1 \mapsto 1, 2 \mapsto 2)$ if $s_{11} + s_{22} > s_{12} + s_{21}$, and $(1 \mapsto 2, 2 \mapsto 1)$ otherwise. For a $1 \times k$ group (one image, k captions), the induced match is $(1 \mapsto \arg\max_{j \in [k]} s_{1j})$. We convert $\pi_{f_{t-1}}(G)$ into a pseudo-label $(G, \pi_{f_{t-1}}(G))$ and add it to the training set $\mathcal{S}_t$ only when its *margin*

$$\Delta(G; f_{t-1}) := s(\pi_{f_{t-1}}(G); G, f_{t-1}) - \max_{\pi \neq \pi_{f_{t-1}}(G)} s(\pi; G, f_{t-1})$$

exceeds a threshold τ_t. By controlling the threshold, we allow models to incorporate data based on their confidence in the induced matching.

Iterative threshold scheduling. Lower thresholds τ_t yield more pseudo-labels but at lower precision, while higher thresholds produce fewer but cleaner labels. This trade-off is illustrated in the left and middle plots of Fig. 2, which show the number of pseudo-matched groups (blue) and the accuracy among matched groups (orange) under varying thresholds. To balance quality and coverage, we adopt a decaying schedule $\tau_{t+1} < \tau_t$, allowing the model to first learn from high-precision pseudo-labels before gradually expanding to the full test set. The right plot of Fig. 2 confirms this intuition: decay schedules outperform fixed thresholds overall. In practice, we find it effective to set the initial threshold τ_1 so that lower than 30% of groups are pseudo-matched, and the final threshold τ_T so that more than 90% of the test set is covered. Both cosine and linear decay schedules perform well. Further analyses and ablations are provided in Section 4.5.

Our TTM algorithm can be viewed as a form of test-time training, a paradigm that has gained significant attention with the advent of powerful pre-trained models (Sun et al., 2020; Gandelsman et al., 2022; Hardt & Sun, 2023; Hübotter et al., 2024; Akyürek et al., 2024). Most prior approaches, however, treat each test instance in isolation, producing instance-specific finetuned models and

often relying on instance-specific in-context examples (Akyürek et al., 2024). In contrast, TTM leverages pseudo-labels across the entire test set to iteratively update a single model under an adaptive thresholding schedule. Crucially, our pseudo-labeling scheme exploits matching, either locally (Section 3.2) or globally (Section 3.2.1), to improve label quality and supervision.

3.2.1 Test-Time Matching without group structures

While Algorithm 1 is designed for datasets organized into local groups, the same principle extends naturally to settings without any predefined group structure. In this case, we treat the entire dataset as a single global matching problem between all images and all captions.

Let $\mathcal{S}_I$ denote the set of images and $\mathcal{S}_C$ the set of captions. Without loss of generality, assume $|\mathcal{S}_I| \leq |\mathcal{S}_C|$ so that each image can be assigned to one caption. Let $s \in \mathbb{R}^{|\mathcal{S}_I| \times |\mathcal{S}_C|}$ be the similarity matrix produced by a model f. The induced global matching is defined as

$$\pi_f := \arg\max_{\pi:\, \mathcal{S}_I \to \mathcal{S}_C} \sum_{i \in \mathcal{S}_I} s_{i,\pi(i)}, \tag{2}$$

which maximizes the total similarity over image-caption pairs. Eq. (2) corresponds to the classical *assignment problem*, which can be efficiently solved by strongly-polynomial time algorithms such as the Hungarian algorithm (Kuhn, 1955).

Analogous to Algorithm 1, we adopt an iterative schedule with pseudo-labeling. At iteration t, let $\pi_{f_{t-1}}$ be the global matching induced by model f_{t-1}. Because the entire dataset is treated as a single group, group-level margin thresholding loses granularity: the model would either accept all matches or none. To address this, we apply thresholding at the level of individual pairs. Specifically, the pseudo-label set at iteration t is

$$\mathcal{S}_t := \{(i, \pi_{f_{t-1}}(i)) : s_{i,\pi_{f_{t-1}}(i)} \geq \tau_t\},$$

where τ_t is the similarity threshold. This threshold can be set either as an absolute value or relative to the distribution of similarity scores (e.g., the p-th percentile). Following the same principle as in Algorithm 1, we begin with a relatively high threshold to ensure high-precision pseudo-labels and gradually decay it over iterations to expand coverage and bootstrap performance across the test set.

4 Experiments

We describe the experimental setups in Section 4.1, report our main results in Sections 4.2 to 4.4, and provide analyses and ablations in Section 4.5. Additional experimental details and results are deferred to Appendix A.3.

4.1 Experimental setups

Datasets. We evaluate on five challenging compositionality benchmarks for multimodal models: Winoground (Thrush et al., 2022), MMVP-VLM (Tong et al., 2024), Colorswap (Burapacheep et al., 2024), SugarCrepe (Hsieh et al., 2023), and Whatsup (Kamath et al., 2023). Winoground, MMVP-VLM, and Colorswap consist of 2×2 groups; we also construct non-grouped variants by discarding group structure (Section 3.2.1). SugarCrepe consists of 1×2 groups and Whatsup of 1×4 groups; we evaluate on 4 different subsets of SugarCrepe and all 2 subsets of Whatsup. Following Li et al. (2024), we further convert Whatsup into 4 different variants with 2×2 groups. In total, our evaluation spans 16 dataset variations.

Models. We test both contrastive vision–language models and multimodal large language models. For contrastive models, we use SigLIP (Zhai et al., 2023) and CLIP (Radford et al., 2021) at multiple scales, including SigLIP-B16, SigLIP-L16, CLIP-B16, and CLIP-B32. For multimodal large language models, we include GPT-4.1, where image-text similarity is derived from VQAScore (Lin et al., 2024).

Evaluation metrics. For GPT-4.1, we report raw GroupScore and GroupMatch-induced performance via SimpleMatch (Section 3.1). For contrastive models (CLIP and SigLIP), we additionally include results with TTM (Algorithm 1). Specifically: on 2×2 datasets we report (i) raw GroupScore,

Table 1: Performance on Winoground, MMVP-VLM, and ColorSwap. Raw model performance is reported with GroupScore, SimpleMatch corresponds to GroupMatch (Section 3.1), and TTM applies Algorithm 1. We report absolute gains (Δ), relative gains, and relative error reductions of TTM over SimpleMatch. Cells highlighted in indicate results with TTM, while cells in mark the **SOTA** performance for each dataset.

Dataset / Model	Raw	SimpleMatch	TTM	Δ	Error Red.
Winoground					
GPT-4.1	69.75 $\pm$ 0.56	91.38 $\pm$ 0.80	–	–	–
CLIP-B16	7.25	60.00	**65.44** $\pm$ **1.10**	**+5.4** (**9.1%** $\uparrow$)	**13.6%** $\downarrow$
SigLIP-B16	10.25	67.00	**72.50** $\pm$ **0.64**	**+5.5** (**8.2%** $\uparrow$)	**16.7%** $\downarrow$
SigLIP-L16	13.00	69.50	**72.75** $\pm$ **0.64**	**+3.3** (**4.7%** $\uparrow$)	**10.7%** $\downarrow$
MMVP-VLM					
GPT-4.1	68.15 $\pm$ 0.00	88.52 $\pm$ 0.83	–	–	–
CLIP-B16	5.19	72.59	**80.19** $\pm$ **0.81**	**+7.6** (**10.5%** $\uparrow$)	**27.7%** $\downarrow$
SigLIP-B16	22.96	81.48	**89.44** $\pm$ **0.96**	**+8.0** (**9.8%** $\uparrow$)	**43.0%** $\downarrow$
ColorSwap					
GPT-4.1	91.08 $\pm$ 0.28	97.42 $\pm$ 0.14	–	–	–
CLIP-B16	12.00	77.67	**85.75** $\pm$ **0.64**	**+8.1** (**10.4%** $\uparrow$)	**36.2%** $\downarrow$
SigLIP-B16	30.33	88.00	**94.25** $\pm$ **0.43**	**+6.3** (**7.1%** $\uparrow$)	**52.1%** $\downarrow$
SigLIP-L16	37.00	91.33	**96.08** $\pm$ **0.43**	**+4.8** (**5.2%** $\uparrow$)	**54.8%** $\downarrow$

(ii) GroupMatch-induced performance, and (iii) TTM-boosted performance; on $1 \times k$ datasets we report (i) raw GroupScore and (ii) TTM-boosted performance, since GroupScore and GroupMatch coincide in this case; and on datasets without group structures we report (i) raw GroupScore (with known groups), (ii) the percentage of correctly matched pairs under Eq. (2), and (iii) TTM-boosted performance (variants in Section 3.2.1). In all cases, we highlight performance gains from TTM—over GroupMatch for 2×2 datasets, over GroupScore for $1 \times k$ datasets, and over global matching (Eq. (2)) for datasets without group structures. All results are averaged over four random runs, with standard deviations reported.

4.2 TTM ACHIEVES NEW SOTAS

We evaluate on three popular compositionality benchmarks—Winoground, MMVP-VLM, and ColorSwap—all consisting of 2×2 groups and considered challenging for multimodal models. Raw group scores are typically at or below random guessing. Previous state-of-the-art results include 58.75 on Winoground (GPT-4V with prompt tuning (Wu et al., 2023; Vaishnav & Tammet, 2025)), 70.7 on MMVP (via a GPT-4o multi-agent system with tool use (Zhang et al., 2024c)),[3] and 87.33 on ColorSwap without training-set access (95.33 with finetuning on the training set (Burapacheep et al., 2024)).

Simple matching results. Applying SimpleMatch (Section 3.1) to CLIP, SigLIP, and GPT-4.1 already yields striking improvements (Table 1). SigLIP with SimpleMatch surpasses all prior state-of-the-art results (without access to ColorSwap's training set). GPT-4.1 with SimpleMatch sets new records on all three benchmarks. Most notably, GPT-4.1 improves from 69.75 to 91.38 on Winoground, *becoming the first system to surpass the estimated human performance of 85.5* (Thrush et al., 2022). These findings confirm that the GroupMatch metric can unlock substantial hidden compositional reasoning ability.

Test-time matching results. We next apply TTM (Algorithm 1) to CLIP and SigLIP, enabling further performance boosts at test time without external supervision. As shown in Table 1, TTM *delivers consistent gains over* SimpleMatch, with relative improvements of around 10% for CLIP-B16 and SigLIP-B16. Although the absolute boosts may appear modest, they are *highly significant*:

[3]This result is on MMVP, the variant designed for MLLMs. We instead evaluate on MMVP-VLM, a version suited for contrastive models from the same paper. Reported performance is similar across the two variants; for example, LLaMA-3-V-8B scores 50 on MMVP and 49.6 on MMVP-VLM (Li et al., 2024).

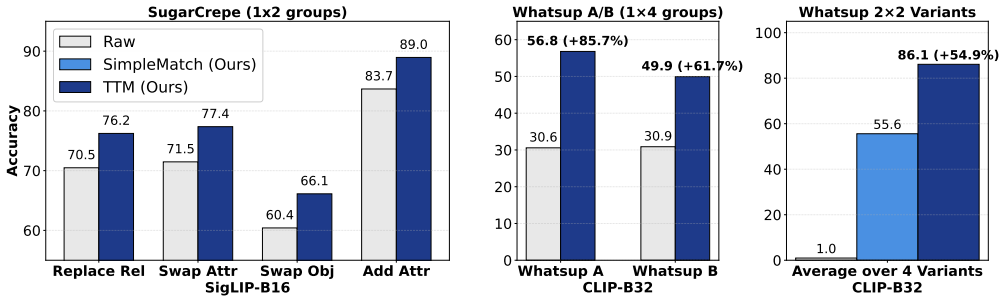

Figure 3: TTM results on datasets without metric-induced boosts. *Left:* results on SugarCrepe with 1×2 groups. *Middle:* results on Whatsup with 1×4 groups. *Right:* average results on Whatsup variants with 2×2 groups. SimpleMatch is the same as raw performance on $1 \times k$ groups.

Table 2: Performance on non-grouped variants of Winoground, MMVP-VLM, and ColorSwap. Raw model performance is reported with GroupScore, SimpleMatch corresponds to the performance of global assignment introduced in Section 3.2.1, and TTM applies the global variant of Algorithm 1. We show absolute gains (Δ), relative gains, and relative error reduction of TTM over SimpleMatch.

Datasets	SigLIP-B16	SimpleMatch	+TTM	Δ	Error Red.
Winoground	10.25	44.38	**46.78** $\pm$ 1.05	**+2.4** (**5.4%** $\uparrow$)	**4.3%** $\downarrow$
MMVP-VLM	22.96	39.63	**44.54** $\pm$ 2.02	**+4.9** (**12.4%** $\uparrow$)	**8.1%** $\downarrow$
ColorSwap	30.33	88.00	**92.00** $\pm$ 1.24	**+4.0** (**4.5%** $\uparrow$)	**33.3%** $\downarrow$

by comparison, methods scaffolding GPT-4V achieve only a 1.25-point absolute gain on Winoground (Vaishnav & Tammet, 2025; Zhang et al., 2024a). Crucially, TTM raises SigLIP to GPT-4.1's level on ColorSwap and **allows SigLIP to surpass GPT-4.1 on MMVP-VLM, establishing a new state of the art**. These results highlight that TTM provides a powerful mechanism to further enhance model performance directly at test time, with no external supervision.

4.3 TTM IMPROVES MODELS WITHOUT METRIC-INDUCED BOOSTS

To evaluate the effectiveness of Algorithm 1 beyond cases where performance can be inflated by alternative metrics, we consider benchmarks with groups of shape $1 \times k$. In this setting, the GroupScore and GroupMatch coincide, so no performance gain can be obtained by overfitting another evaluation metric.

We experiment on four SugarCrepe subsets with 1×2 groups and both Whatsup subsets with 1×4 groups, reporting results in Fig. 3 (left and middle plots). Even without metric-induced gains, Algorithm 1 provides substantial test-time improvements. The gains are particularly striking on the Whatsup datasets, where **performance increases by up to** 85.7%, turning these previously difficult tasks into tractable ones. Inspired by Li et al. (2024), we further convert the Whatsup datasets into four directional variants with 2×2 group structure. As shown in the right plot of Fig. 3, Algorithm 1 again yields large performance boosts (54.9% relative gains on top of SimpleMatch), demonstrating its robustness across both $1 \times k$ and 2×2 settings. These results demonstrate that TTM is broadly effective, even when evaluation metrics themselves cannot induce gains.

4.4 TTM IMPROVES MODELS WITHOUT GROUP STRUCTURES

To assess the generality of Algorithm 1, we evaluate its global variant introduced in Section 3.2.1 on datasets *without any predefined group structure.* Specifically, we flatten Winoground, MMVP-VLM, and ColorSwap by removing local $k \times k$ groups, resulting in a general dataset with an image set $\mathcal{S}_I$ and a caption set $\mathcal{S}_C$.

We report three metrics: (i) raw GroupScore (with known groups), (ii) SimpleMatch with global assignment accuracy under Eq. (2), and (iii) TTM-boosted performance with the global variant of Algorithm 1. Results show that even global assignment without groups structures substantially

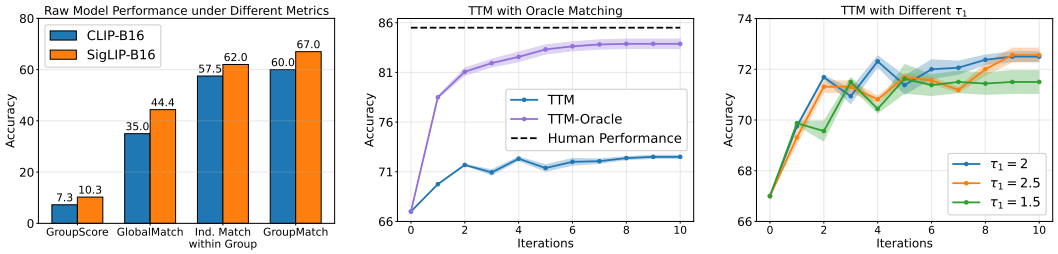

Figure 4: *Left:* Raw model performance on Winoground under different evaluation metrics. *Middle:* Skyline performance of TTM with oracle matching on Winoground using SigLIP-B16. *Right:* TTM performance under different initial thresholds τ_1, evaluated on Winoground with SigLIP-B16.

outperforms the vanilla GroupScore, highlighting the benefit of matching-based evaluation. More importantly, applying the iterative global TTM algorithm yields further gains over global assignment alone, with especially large relative error reductions on ColorSwap; see Table 2. This demonstrates that the test-time matching principle extends robustly beyond group-structured datasets.

4.5 Analyses and Ablations

Group structures provide stronger supervision. The key advantage of GroupMatch over GroupScore is its ability to better exploit group structures for pseudo-labeling. To assess this benefit, we examine raw performance of CLIP and SigLIP under different evaluation metrics (Fig. 4, left). In addition to GroupScore and GroupMatch, we consider GlobalMatch, which allows global matching but ignores group structure (Section 3.2.1), and *individual match*, which uses group structure but performs no joint matching: it individually match images to captions within the group. Overall, GroupMatch provides the strongest supervision signal, making it most effective for guiding pseudo-labeling.

Skyline performance with oracle matching. To study the full potential of TTM, we evaluate an oracle variant that incorporates pseudo-labels into $\mathcal{S}_t$ only when they are correct (i.e., with oracle access). As shown in Fig. 4 (middle), this oracle variant enables TTM to bootstrap more aggressively and approach human-level performance on Winoground. This suggests that improving pseudo-label quality—e.g., through limited external supervision—could further enhance TTM 's effectiveness.

Sensitivity to thresholds. As discussed in Section 3.2, we generally recommend a decaying threshold schedule that begins with high-quality pseudo-labels and gradually expands coverage. In our experiments, the final threshold τ_T is set to either 0 (full coverage) or 0.1 (labels covering $> 90\%$ of the data). The initial threshold τ_1 is more dataset- and model-dependent: we find it effective to set τ_1 such that lower than 30% of the test set is pseudo-labeled. Fig. 4 (right) shows results for $\tau_1 \in \{1.5, 2, 2.5\}$ on Winoground with SigLIP-B16, corresponding to $\{30\%, 21\%, 18\%\}$ coverage. Despite variations, all settings yield consistent gains, highlighting that TTM reliably improves performance without external supervision.

5 Discussion

In this paper, we revisited the puzzle of multimodal compositional reasoning, where popular models have long appeared to perform at or below random chance. We showed that part of this gap arises from rigid evaluation metrics that underestimate model capabilities. By introducing the group matching score and applying simple test-time overfitting, we uncovered substantial hidden competence in both contrastive VLMs and MLLMs—enough for GPT-4.1 to surpass estimated human performance on Winoground. Building on this insight, we proposed TTM, an iterative self-training algorithm that bootstraps performance without external supervision, enabling SigLIP to outperform GPT-4.1 on MMVP-VLM and establishing new state-of-the-art results across multiple benchmarks. These findings demonstrate that revisiting evaluation and employing strategic test-time adaptation can meaningfully advance the frontier of multimodal reasoning.

REFERENCES

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Ekin Akyürek, Mehul Damani, Adam Zweiger, Linlu Qiu, Han Guo, Jyothish Pari, Yoon Kim, and Jacob Andreas. The surprising effectiveness of test-time training for few-shot learning. *arXiv preprint arXiv:2411.07279*, 2024.

Christopher G Atkeson, Andrew W Moore, and Stefan Schaal. Locally weighted learning. *Artificial intelligence review*, 11(1):11–73, 1997.

Maria-Florina Balcan, Andrei Broder, and Tong Zhang. Margin based active learning. In *International Conference on Computational Learning Theory*, pp. 35–50. Springer, 2007.

Léon Bottou and Vladimir Vapnik. Local learning algorithms. *Neural computation*, 4(6):888–900, 1992.

Jirayu Burapacheep, Ishan Gaur, Agam Bhatia, and Tristan Thrush. Colorswap: A color and word order dataset for multimodal evaluation. *arXiv preprint arXiv:2402.04492*, 2024.

Rui M Castro and Robert D Nowak. Minimax bounds for active learning. In *International Conference on Computational Learning Theory*, pp. 5–19. Springer, 2007.

Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. *IEEE Transactions on Neural Networks*, 20(3):542–542, 2009.

François Chollet. On the measure of intelligence. *arXiv preprint arXiv:1911.01547*, 2019.

Francois Chollet, Mike Knoop, Gregory Kamradt, and Bryan Landers. Arc prize 2024: Technical report. *arXiv preprint arXiv:2412.04604*, 2024.

William S Cleveland. Robust locally weighted regression and smoothing scatterplots. *Journal of the American statistical association*, 74(368):829–836, 1979.

William S Cleveland and Susan J Devlin. Locally weighted regression: an approach to regression analysis by local fitting. *Journal of the American statistical association*, 83(403):596–610, 1988.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Sanjoy Dasgupta, Adam Tauman Kalai, and Adam Tauman. Analysis of perceptron-based active learning. *Journal of Machine Learning Research*, 10(2), 2009.

Anuj Diwan, Layne Berry, Eunsol Choi, David Harwath, and Kyle Mahowald. Why is winoground hard? investigating failures in visuolinguistic compositionality. *arXiv preprint arXiv:2211.00768*, 2022.

Yossi Gandelsman, Yu Sun, Xinlei Chen, and Alexei Efros. Test-time training with masked autoencoders. *Advances in Neural Information Processing Systems*, 35:29374–29385, 2022.

Steve Hanneke. Theory of active learning. *Foundations and Trends in Machine Learning*, 7(2-3), 2014.

Moritz Hardt and Yu Sun. Test-time training on nearest neighbors for large language models. *arXiv preprint arXiv:2305.18466*, 2023.

Cheng-Yu Hsieh, Jieyu Zhang, Zixian Ma, Aniruddha Kembhavi, and Ranjay Krishna. Sugarcrepe: Fixing hackable benchmarks for vision-language compositionality. *Advances in neural information processing systems*, 36:31096–31116, 2023.

Jonas Hübotter, Sascha Bongni, Ido Hakimi, and Andreas Krause. Efficiently learning at test-time: Active fine-tuning of llms. *arXiv preprint arXiv:2410.08020*, 2024.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Amita Kamath, Jack Hessel, and Kai-Wei Chang. What's" up" with vision-language models? investigating their struggle with spatial reasoning. *arXiv preprint arXiv:2310.19785*, 2023.

Akshay Krishnamurthy, Alekh Agarwal, Tzu-Kuo Huang, Hal Daumé III, and John Langford. Active learning for cost-sensitive classification. *Journal of Machine Learning Research*, 20(65):1–50, 2019.

Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

Ananya Kumar, Tengyu Ma, and Percy Liang. Understanding self-training for gradual domain adaptation. In *International conference on machine learning*, pp. 5468–5479. PMLR, 2020.

Siting Li, Pang Wei Koh, and Simon Shaolei Du. Exploring how generative mllms perceive more than clip with the same vision encoder. *arXiv preprint arXiv:2411.05195*, 2024.

Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. Evaluating text-to-visual generation with image-to-text generation. In *European Conference on Computer Vision*, pp. 366–384. Springer, 2024.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Nikita Puchkin and Nikita Zhivotovskiy. Exponential savings in agnostic active learning through abstention. In *Conference on learning theory*, pp. 3806–3832. PMLR, 2021.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Rylan Schaeffer, Brando Miranda, and Sanmi Koyejo. Are emergent abilities of large language models a mirage? *Advances in neural information processing systems*, 36:55565–55581, 2023.

Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in neural information processing systems*, 33:596–608, 2020.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *International conference on machine learning*, pp. 9229–9248. PMLR, 2020.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Tristan Thrush, Ryan Jiang, Max Bartolo, Amanpreet Singh, Adina Williams, Douwe Kiela, and Candace Ross. Winoground: Probing vision and language models for visio-linguistic compositionality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5238–5248, 2022.

Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9568–9578, 2024.

Mohit Vaishnav and Tanel Tammet. A cognitive paradigm approach to probe the perception-reasoning interface in vlms. *arXiv preprint arXiv:2501.13620*, 2025.

Yifan Wu, Pengchuan Zhang, Wenhan Xiong, Barlas Oguz, James C Gee, and Yixin Nie. The role of chain-of-thought in complex vision-language reasoning task. *arXiv preprint arXiv:2311.09193*, 2023.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 11975–11986, 2023.

Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *Advances in neural information processing systems*, 34:18408–18419, 2021.

Daoan Zhang, Junming Yang, Hanjia Lyu, Zijian Jin, Yuan Yao, Mingkai Chen, and Jiebo Luo. Cocot: Contrastive chain-of-thought prompting for large multimodal models with multiple image inputs. *arXiv preprint arXiv:2401.02582*, 2024a.

Jifan Zhang, Yifang Chen, Gregory Canal, Arnav Mohanty Das, Gantavya Bhatt, Stephen Mussmann, Yinglun Zhu, Jeff Bilmes, Simon Shaolei Du, Kevin Jamieson, and Robert Nowak. Labelbench: A comprehensive framework for benchmarking adaptive label-efficient learning. *Journal of Data-centric Machine Learning Research*, 2024b.

Zhehao Zhang, Ryan Rossi, Tong Yu, Franck Dernoncourt, Ruiyi Zhang, Jiuxiang Gu, Sungchul Kim, Xiang Chen, Zichao Wang, and Nedim Lipka. Vipact: Visual-perception enhancement via specialized vlm agent collaboration and tool-use. *arXiv preprint arXiv:2410.16400*, 2024c.

Xiaojin Jerry Zhu. Semi-supervised learning literature survey. 2005.

Yinglun Zhu and Robert Nowak. Active learning with neural networks: Insights from nonparametric statistics. *Advances in Neural Information Processing Systems*, 35:142–155, 2022a.

Yinglun Zhu and Robert Nowak. Efficient active learning with abstention. *Advances in Neural Information Processing Systems*, 35:35379–35391, 2022b.

A Appendix

A.1 Related Work

Multimodal models, compositional reasoning, and evaluation metrics. Contrastive vision-language models (VLMs) such as CLIP and SigLIP (Radford et al., 2021; Zhai et al., 2023) and multimodal LLMs (MLLMs) such as GPT (Achiam et al., 2023; Hurst et al., 2024) and Gemini (Team et al., 2023; Comanici et al., 2025) series have achieved impressive progress across a wide range of tasks. Yet both VLMs and MLLMs have been shown to struggle on benchmarks specifically designed to test *compositional reasoning*—the ability to correctly bind objects, attributes, and relations across modalities (Thrush et al., 2022; Tong et al., 2024; Burapacheep et al., 2024; Hsieh et al., 2023; Kamath et al., 2023; Diwan et al., 2022; Li et al., 2024). These benchmarks are typically organized into small groups of images and captions that differ in subtle but systematic ways (e.g., captions with identical words but different orderings). The prevailing evaluation metric is the GroupScore, which requires models to simultaneously assign each image to its correct caption and each caption to its correct image. While rigorous, this metric is also unforgiving: raw model performance often falls at or below random guessing (Diwan et al., 2022; Li et al., 2024). Despite recent efforts to improve compositional reasoning for multimodal models (Wu et al., 2023; Zhang et al., 2024c; Vaishnav & Tammet, 2025), progress remain modest. For instance, the previous state of the art on Winoground was only 58.75—achieved by scaffolding GPT-4V (Wu et al., 2023; Vaishnav & Tammet, 2025)—still well below the estimated human performance of 85.5 (Thrush et al., 2022). Our work takes a different perspective: instead of modifying the models, we revisit the evaluation itself. We introduce a group matching score (GroupMatch) that evaluates the best overall matching rather than isolated pairwise comparisons, revealing substantial hidden competence in both VLMs and MLLMs. Crucially, by simply overfitting to the induced matchings at test time, this hidden ability transfers back into higher scores under the original group metric, closing much of the reported gap. With this adjustment, GPT-4.1 becomes the first system to *surpass estimated human performance* on Winoground. This finding echoes broader observations that measured ability can be highly sensitive to metric design (Schaeffer et al., 2023), underscoring the need to further research into multimodal evaluation protocols.

Test-time training, self-training, and pseudo-labeling. Test-time training adapts models at inference to improve performance, with roots in early work on local learning and instance-specific adaptation (Cleveland, 1979; Cleveland & Devlin, 1988; Bottou & Vapnik, 1992; Atkeson et al., 1997). The idea has regained attention in the era of large pretrained models, where test-time self-supervision can be exploited without additional labeled data (Sun et al., 2020; Gandelsman et al., 2022). Recent studies show that finetuning on retrieved data based on test prompts can significantly improve large language models (Hardt & Sun, 2023; Hübotter et al., 2024), and test-time training has become a key component in tackling reasoning-heavy benchmarks such as ARC (Chollet, 2019; Chollet et al., 2024; Akyürek et al., 2024). Our test-time matching algorithm (TTM) shares this motivation but differs in key ways. Most prior work treats each test instance in isolation, producing instance-specific finetuned models, sometimes relying on instance-specific in-context examples (Akyürek et al., 2024). In contrast, we leverage GroupMatch-induced pseudo-labels across the *entire test set*, updating a single model iteratively with an adaptive thresholding schedule. This connects naturally to the rich literature on self-training (Kumar et al., 2020) and semi-supervised learning (Zhu, 2005; Chapelle et al., 2009; Sohn et al., 2020; Zhang et al., 2021, 2024b), where pseudo-labels drive improvements. Our contribution lies in exploiting group structure—both locally and globally—for pseudo-labeling. Finally, our adaptive thresholding schedule resonates with classical ideas in active learning (Castro & Nowak, 2007; Balcan et al., 2007; Dasgupta et al., 2009; Hanneke, 2014; Krishnamurthy et al., 2019; Puchkin & Zhivotovskiy, 2021; Zhu & Nowak, 2022a,b). Whereas active learning typically queries the most uncertain data for human annotation, our approach inverts this logic: we begin with the most confident pseudo-labels, then gradually relax thresholds to expand coverage. This reversed perspective is central to TTM 's effectiveness, allowing it to reliably improve model performance even in the absence of external supervision.

A.2 Supporting results from Section 3

Suppose we have a group of k images $\{I_i\}_{i=1}^k$ and k captions $\{C_i\}_{i=1}^k$, with ground-truth pairings $\{(I_i, C_i)\}_{i=1}^k$ hidden from the learner (Thrush et al., 2022; Tong et al., 2024; Burapacheep et al.,

2024). For each pair (I_i, C_j), the random guessing model assigns a similarity score $\mathsf{sim}(I_i, C_j) \sim \mathrm{unif}([0,1])$, producing a similarity matrix $s \in \mathbb{R}^{k\times k}$ with entries $s_{ij} := \mathsf{sim}(I_i, C_j)$.

Recall the GroupScore is calculated as

$$\mathsf{GroupScore}(s) := \begin{cases} 1 & \forall i: \ s_{ii} > \max_{j\neq i} s_{ij} \quad \text{and} \quad s_{ii} > \max_{j\neq i} s_{ji}, \\ 0 & \text{otherwise.} \end{cases}$$

We next provide the proof for Proposition 1.

Proposition 1. *For random similarity scores s, $\mathbb{P}(\mathsf{GroupScore}(s) = 1) = \frac{(k-1)!}{(2k-1)!}$.*

Proof. Because the entries of s are i.i.d. sampled from a continuous distribution (here $\mathrm{unif}([0,1])$), ties occur with probability 0, so we may use strict inequalities throughout.

Denote $d_i := s_{ii}$ and, for $i \neq j$, set $m_{ij} := \min\{d_i, d_j\}$. By the definition of the GroupScore, the event $\{\mathsf{GroupScore}(s) = 1\}$ is equivalent to requiring $s_{ij} < m_{ij}$ and $s_{ji} < m_{ij}$ for every $i \neq j$. Conditioning on the diagonal $d = (d_1, \ldots, d_k)$ and using independence of the off-diagonal entries,

$$\mathbb{P}\big(\mathsf{GroupScore}(s) = 1 \mid d\big) = \prod_{i<j} \mathbb{P}(s_{ij} < m_{ij})\, \mathbb{P}(s_{ji} < m_{ij}) = \prod_{i<j} m_{ij}^2.$$

Let $0 \leq x_1 \leq \cdots \leq x_k \leq 1$ be the order statistics of $(d_1, \ldots, d_k)$. We then have $m_{ij} = x_{\min\{r(i), r(j)\}}$, where $r(\cdot)$ is the rank, hence

$$\prod_{i<j} m_{ij}^2 = \prod_{a=1}^{k} x_a^{2(k-a)}.$$

Since $(x_1, \ldots, x_k)$ are the order statistics of i.i.d. $\mathrm{unif}([0,1])$ samples, their joint density is $k!$ on the ordered simplex $\{0 \leq x_1 \leq \cdots \leq x_k \leq 1\}$ (and 0 elsewhere). Therefore,

$$\mathbb{P}\big(\mathsf{GroupScore}(s) = 1\big) = k! \int_{0\leq x_1\leq\cdots\leq x_k\leq 1} \prod_{a=1}^{k} x_a^{2(k-a)}\, dx_1 \cdots dx_k.$$

For $1 \leq \ell \leq k$ and $y \in [0,1]$, define

$$I_\ell(y) := \int_{0<x_1<\cdots<x_\ell<y} \prod_{a=1}^{\ell} x_a^{2(k-a)}\, dx_1 \cdots dx_\ell.$$

We claim that, for $\ell = 1, \ldots, k$,

$$I_\ell(y) = \frac{y^{\,\ell(2k-\ell)}}{\prod_{r=1}^{\ell} r(2k-r)}.$$

This is proved by induction on ℓ. For $\ell = 1$,

$$I_1(y) = \int_0^y x^{2(k-1)}\, dx = \frac{y^{2k-1}}{2k-1}.$$

Assume it holds for $\ell - 1$. Then

$$\begin{aligned} I_\ell(y) &= \int_0^y x_\ell^{2(k-\ell)}\, I_{\ell-1}(x_\ell)\, dx_\ell \\ &= \frac{1}{\prod_{r=1}^{\ell-1} r(2k-r)} \int_0^y x_\ell^{2(k-\ell)+(\ell-1)(2k-(\ell-1))}\, dx_\ell \\ &= \frac{1}{\prod_{r=1}^{\ell-1} r(2k-r)} \cdot \frac{y^{\,\ell(2k-\ell)}}{\ell(2k-\ell)}, \end{aligned}$$

since $2(k-\ell)+(\ell-1)(2k-(\ell-1)) = \ell(2k-\ell)-1$. Thus the claim holds. Taking $\ell = k$ and $y = 1$ gives

$$\int_{0\le x_1\le\cdots\le x_k\le 1} \prod_{a=1}^{k} x_a^{2(k-a)}\, dx_1\cdots dx_k = I_k(1) = \frac{1}{\prod_{r=1}^{k} r(2k-r)}.$$

Therefore,

$$\mathbb{P}\big(\mathsf{GroupScore}(s)=1\big) = k!\prod_{r=1}^{k}\frac{1}{r(2k-r)} = \frac{(k-1)!}{(2k-1)!}.$$

□

A.3 Other details for experiments

A.4 Additional details and hyperparameters

We provide additional experimental details and hyperparameter settings below. For TTM, we set the number of iterations to $T = 10$ and train the model for 20 epochs per iteration (30 epochs on Winoground). Across all experiments, we use AdamW (Loshchilov & Hutter, 2017) with weight decay 0.05 and $(\beta_1, \beta_2) = (0.9, 0.999)$. The learning rate follows a cosine decay schedule and is restarted at each iteration with a multiplicative factor of 0.95. We use a batch size of 50 for 2×2 datasets and 100 for $1 \times k$ datasets; the batch size is defined at the group level (e.g., 50 groups of size 2×2 per batch).[4]

Tables 3 to 5, report, for each dataset–model pair, the initial threshold τ_1, the final threshold τ_T, the threshold decay schedule (linear or cosine), and the learning rate (lr).

Table 3: Hyperparameters used for experiments in Section 4.2.

Dataset	Model	τ_1	τ_T	Schedule	lr
Winoground	CLIP-B16	0.9	0	linear	2.0×10^{-5}
	SigLIP-B16	2.0	0	linear	1.0×10^{-5}
	SigLIP-L16	2.0	0.1	cosine	4.0×10^{-5}
ColorSwap	CLIP-B16	2.3	0	cosine	4.0×10^{-5}
	SigLIP-B16	1.0	0	cosine	4.0×10^{-5}
	SigLIP-L16	2.5	0	cosine	4.0×10^{-5}
MMVP-VLM	CLIP-B16	2.0	0	linear	1.0×10^{-5}
	SigLIP-B16	2.0	0.1	cosine	2.0×10^{-5}

A.4.1 Complete results from Section 4.3

We present complete empirical results for Fig. 3 below in Tables 6 to 8.

A.5 The Use of Large Language Models (LLMs)

LLMs were used to polish the writing of this paper.

[4] We slightly increase the batch size when the total number of groups is just above a multiple of the default size. For instance, if the dataset contains 102 groups, we set the batch size to 51.

Table 4: Hyperparameters used for experiments in Section 4.3.

Variant	**Model**	τ_1	τ_T	Schedule	lr
Replace Relation	SigLIP-B16	2.1	0	cosine	1.0×10^{-5}
Swap Attribute	SigLIP-B16	1.8	0	cosine	1.0×10^{-5}
Swap Object	SigLIP-B16	2.0	0	cosine	1.0×10^{-5}
Add Attribute	SigLIP-B16	2.5	0	cosine	1.0×10^{-5}
Whatsup A (1×4)	CLIP-B32	0.55	0	linear	1.0×10^{-5}
Whatsup B (1×4)	CLIP-B32	0.80	0	linear	1.0×10^{-5}
A–Left–Right	CLIP-B32	0.25	0	linear	1.0×10^{-5}
A–On–Under	CLIP-B32	0.85	0	linear	1.0×10^{-5}
B–Left–Right	CLIP-B32	0.50	0	cosine	2.0×10^{-5}
B–Front–Behind	CLIP-B32	1.30	0	cosine	2.0×10^{-5}

Table 5: Hyperparameters used for experiments in Section 4.4. τ_1 is selected based on the percentile criterion described in Section 3.2.1. We find that global matching performs better with a slightly smaller τ_1 than the corresponding percentile selected from the grouped case.

Dataset	Model	τ_1	τ_T	Schedule	lr
Winoground	SigLIP-B16	0.50	0	linear	1.0×10^{-5}
ColorSwap	SigLIP-B16	0.50	0	linear	4.0×10^{-5}
MMVP-VLM	SigLIP-B16	0.55	0	linear	2.0×10^{-5}

Table 6: Performance on SugarCrepe datasets (1×2 groups). TTM consistently improves model performance without metric-induced boosts. We report absolute gains (Δ), relative gains, and relative error reductions.

Datasets	SigLIP-B16	+ TTM	Δ	Error Reduction
Replace Relation	70.48	**76.23** $\pm$ 0.51	**+5.8** (**8.2%** $\uparrow$)	**19.5%** $\downarrow$
Swap Attribute	71.47	**77.36** $\pm$ 0.71	**+5.9** (**8.2%** $\uparrow$)	**20.6%** $\downarrow$
Swap Object	60.41	**66.12** $\pm$ 2.06	**+5.7** (**9.5%** $\uparrow$)	**14.4%** $\downarrow$
Add Attribute	83.67	**88.95** $\pm$ 0.83	**+5.3** (**6.3%** $\uparrow$)	**32.3%** $\downarrow$

Table 7: Performance on Whatsup A/B datasets (1×4 groups). TTM consistently improves model performance without metric-induced boosts. We report absolute gains (Δ), relative gains , and relative error reductions.

Datasets	CLIP-B32	+ TTM	Δ	Error Reduction
Whatsup A	30.58	**56.8** $\pm$ 1.84	**+26.2** (**85.7%** $\uparrow$)	**37.7%** $\downarrow$
Whatsup B	30.88	**49.94** $\pm$ 2.58	**+19.1** (**61.7%** $\uparrow$)	**27.6%** $\downarrow$

Table 8: Performance on Whatsup 2×2 directional subsets: LR: left-right, OU: on-under; FB: front-behind. We report baseline (CLIP-B32), SimpleMatch, and TTM. Δ and error reduction are computed relative to SimpleMatch.

Datasets	CLIP-B32	SimpleMatch	+ TTM	Δ	Error Reduction
A-LR	0	40.78	**95.87** $\pm$ 4.42	**+55.1** (**135.1%** $\uparrow$)	**93.0%** $\downarrow$
A-OU	3.88	78.64	**99.03** $\pm$ 0	**+20.4** (**25.9%** $\uparrow$)	**95.5%** $\downarrow$
B-LR	0	55.88	**82.84** $\pm$ 0.49	**+27.0** (**48.2%** $\uparrow$)	**61.1%** $\downarrow$
B-FB	0	47.06	**66.67** $\pm$ 1.30	**+19.6** (**41.7%** $\uparrow$)	**37.0%** $\downarrow$