OBSERVER-SPECIFIC METRIC LEARNING VIA SEMANTIC FACTORIZATION AND GATED PROJECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Similarity judgments can vary across observers: observers may emphasize different attributes of the same data and thereby induce distinct similarity metrics. Yet conventional metric learning does not explicitly distinguish observer-specific structure from similarity shared across a population. We address two aspects of this problem. First, we formulate *observer-specific metric learning* (OSML) as recovering a target observer's similarity metric from sparse relational judgments. To distinguish observer-specific learning from population-shared predictability, we introduce a population-aware evaluation framework and Observer-Specific Gain (OSG), which compares agreement with the target observer against agreement with non-target observers. Across datasets and neural-network observer populations, population diversity relates to cross-observer predictability, supporting the need for population-aware evaluation. Second, we propose *Semantic Factorization and Gated Projection* (SFGP), which constructs a factorized semantic representation and selectively reweights its components for each target observer, constraining metric learning under limited triplet supervision. SFGP consistently improves target agreement and achieves larger observer-specific gains than conventional end-to-end triplet learning. Together, our results provide a framework for characterizing and evaluating similarity as a population of related yet individually distinct metrics, while showing that leveraging a shared factorized semantic representation can improve individual metric learning under limited observations.

1 INTRODUCTION

Metric learning aims to learn a representation space in which distances reflect semantic similarity (Schroff et al., 2015). Most approaches implicitly assume a single notion of similarity. However, different observers may assign different importance to semantic attributes and thereby induce distinct similarity spaces over the same data (Malloy et al., 2024; Hoang et al., 2022; Sundaram et al., 2025; Simmons & Estes, 2008). Such variation can arise across both humans and models, which may develop different representations and similarity structures even under the same architecture and learning objective.

We formulate the recovery of such individual similarity structures as *Observer-Specific Metric Learning* (OSML). Given sparse relational judgments from a target observer, human or model, OSML learns a metric that reproduces that observer's similarity structure without access to other observers. We view an observer's metric as containing both population-shared and observer-specific structure, motivating population-relative evaluation (Fig. 1a). Target accuracy alone cannot distinguish these components: high accuracy may reflect structure shared across observers rather than recovery of what distinguishes the target. We therefore introduce the *Population Baseline*, which measures agreement with non-target observers, and *Observer-Specific Gain* (OSG), which measures target agreement beyond this baseline. Recovering this observer-specific component is challenging under sparse relational supervision, where many metrics may satisfy the observed judgments. To constrain this problem, we propose *Semantic Factorization and Gated Projection* (SFGP), which factorizes a label-independent semantic representation and selectively combines observer-relevant units through gated projection (Fig. 1b). This constrains metric learning to a structured semantic space. We study OSML using independently trained models as controlled observers, with access to their similarity structures restricted to sparse relational judgments. Across benchmark datasets and model populations, SFGP improves target agreement and achieves larger observer-specific gains

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

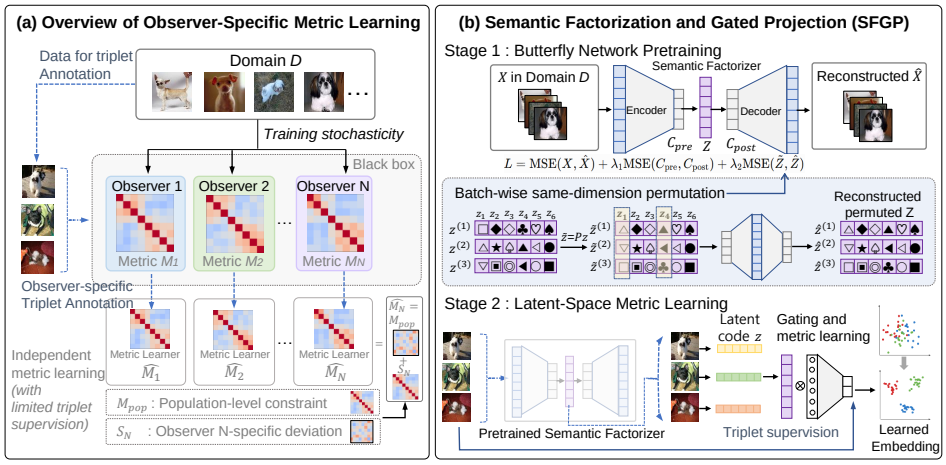

Figure 1: **Overview of OSML and SFGP.** (a) Observer-specific similarity metrics are viewed in relation to shared population structure. (b) SFGP factorizes a general representation into semantic units and selectively combines them through gated projection for observer-specific metric learning.

than conventional end-to-end triplet learning, particularly under limited supervision. These results provide a basis for studying individual similarity structures beyond population-shared regularities and, ultimately, for extending OSML and SFGP to heterogeneous human observers. Our contributions are threefold:

- We formulate OSML as the problem of recovering a target observer's similarity metric from sparse relational judgments.
- We introduce a population-relative evaluation framework based on the Population Baseline and OSG to distinguish target-specific recovery from population-level predictability.
- We propose Semantic Factorization and Gated Projection (SFGP), a structured approach for recovering observer-specific metrics from sparse triplet supervision.

2 Related Work

2.1 Observer-Specific and Personalized Metric Learning

Learning similarity structure from relational information has been widely studied in metric learning, contrastive learning, and ordinal embedding (Weinberger & Saul, 2009; Hadsell et al., 2006; Chen et al., 2020; Terada & Luxburg, 2014). Metric and contrastive learning use pairwise or triplet relations to learn task-relevant representations, whereas ordinal embedding recovers latent geometry from relative comparisons and has been used to reconstruct perceptual spaces from human similarity judgments (Anand et al., 2026). Personalized metric learning and perceptual embedding further model individual variation, whereas crowd-level approaches aggregate judgments to recover shared similarity structure (Canal et al., 2022; Roads & Mozer, 2019; Roads & Love, 2021). Prior work has rarely considered individual metric recovery relative to the observer population. The same target accuracy can reflect different degrees of observer-specific learning depending on shared structure. Our framework addresses this distinction by using population-level predictability to separate shared similarity from observer-specific recovery.

2.2 Factorized Representations and Constrained Metric Learning

Metric learning uses pairwise or triplet supervision to learn relational structure among observations. Under sparse supervision, multiple metrics may satisfy the observed relations, limiting generalization to unseen relations (Wang et al., 2024; Vankadara et al., 2023; Wilber et al., 2014). Prior work addresses this underdetermination through regularization and low-dimensional projections (Qi et al.,

2009; Davis et al., 2007). Factorized representation learning provides another structural constraint by organizing representations into latent factors capturing distinct sources of variation (Kim & Mnih, 2018; Mathieu et al., 2019). Rather than promoting fully disentangled or statistically independent factors, we seek a less redundant and more differentiated semantic space whose components can be selectively reweighted for observer-specific metric learning. We therefore constrain metric learning to a factorized semantic space, where gated projection selects observer-relevant components instead of directly learning an unconstrained metric from sparse relational supervision.

3 METHODS

3.1 PROBLEM FORMULATION

Independently trained observers can develop different similarity metrics even when trained on the same domain under the same learning objective (Kornblith et al., 2019; Ding et al., 2021). At the same time, their similarity judgments often exhibit agreement, suggesting that individual metrics contain structure shared across the observer population (Izydorczyk & Bröder, 2026). Consequently, in a population with low metric diversity, a learner trained on a single observer may achieve high triplet agreement without fully recovering what distinguishes that observer, simply because the target is highly predictable from the population. Target agreement alone is therefore insufficient to determine how much observer-specific information has been learned. We define an **observer** as an entity possessing an individual similarity metric over a common domain. An observer may be an independently trained neural network or a human subject. A collection of M independently formed observers constitutes a **population**, $\mathcal{P} = \{o_1, o_2, \ldots, o_M\}$. Each observer o_i is associated with a similarity metric $d_i : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. Rather than observing d_i directly, we assume access only to relative similarity judgments. Given a triplet $t = (a, b, c)$, observer o_i provides a binary judgment $y_i(t)$ indicating whether b or c is more similar to the anchor a. For a target observer o_t, only a subset of its triplet judgments, $\mathcal{D}_t = \{(t_k, y_t(t_k))\}_{k=1}^{N}$, is available for training, with no observations from other observers. We define **Observer-Specific Metric Learning (OSML)** as learning a target-specific metric f_t from $\mathcal{D}_t$ that approximates the target metric d_t. OSML learns the target metric independently of the observer population, while our evaluation interprets the learned metric relative to population-shared structure. Conceptually, a target observer's similarity judgments may contain both structure predictable from other observers and structure specific to the target. The objective is therefore not only to measure agreement with the target, but also to quantify how much target-specific information is recovered beyond what is predictable from the observer population.

3.2 POPULATION-AWARE EVALUATION OF OBSERVER-SPECIFIC METRIC LEARNING

To quantify observer-specific recovery relative to the population, we introduce three population-aware measures:

$$D_m = 1 - \frac{\sum_{j \neq m} S(o_m, o_j)}{M - 1}, \quad A_{\mathrm{pop},m} = \frac{\sum_{j \neq m} A(f_m, o_j)}{M - 1}, \quad A_{\mathrm{sp},m} = A(f_m, o_m). \tag{1}$$

Here, D_m denotes the diversity of observer o_m, where $S(o_m, o_j)$ is the agreement between the triplet judgments of observers o_m and o_j; a larger D_m indicates greater deviation from the population. For each target observer o_m, a metric learner f_m is trained exclusively on its triplet observations. $A_{\mathrm{pop},m}$ and $A_{\mathrm{sp},m}$ denote its agreement with the remaining population and the target observer, respectively, where $A(f_m, o_j)$ denotes triplet agreement between f_m and observer o_j. Because f_m is trained only on o_m, $A_{\mathrm{pop},m}$ reflects population-level structure captured from the target observer. Finally, we define the **Observer-Specific Gain (OSG)** as

$$\mathrm{OSG}_m = \frac{A_{\mathrm{sp},m} - A_{\mathrm{pop},m}}{1 - A_{\mathrm{pop},m}}. \tag{2}$$

The numerator measures the additional agreement with the target observer beyond population agreement, while the denominator represents the maximum possible improvement above this baseline. OSG therefore measures the normalized observer-specific gain beyond the agreement explained by population-level structure. In summary, conventional evaluation measures agreement with the target observer, whereas our population-aware evaluation measures how much additional agreement is achieved for the target beyond what is shared with the population (See Appendix A.1).

3.3 Semantic Factorizer Architecture

We assume that observer-specific metrics arise from different semantic weightings over a shared semantic representation. Based on this assumption, we propose **Semantic Factorization and Gated Projection (SFGP)**, which consists of a domain-level Semantic Factorizer followed by observer-specific Gated Projection. The Semantic Factorizer is a self-supervised autoencoder with a butterfly-shaped latent structure, referred to as the *Butterfly Network*. Let x denote an input sample, $E(x)$ its encoder feature, and E and D the encoder and decoder, respectively. The network consists of a PreCompressor, an Expander, and a PostCompressor,

$$x \xrightarrow{E} E(x) \xrightarrow{\text{PreCompressor}} c_{\text{pre}} \xrightarrow{\text{Expander}} z \xrightarrow{\text{PostCompressor}} c_{\text{post}} \xrightarrow{D} \hat{x}, \tag{3}$$

where c_{pre} and c_{post} are compact representations before and after semantic expansion, respectively, and z denotes the expanded semantic representation. The PreCompressor removes variations unnecessary for reconstruction, the Expander increases representational capacity in the semantic space, and the PostCompressor suppresses redundant degrees of freedom introduced during expansion. The Semantic Factorizer maintains z as a spatial feature map (Jaegle et al., 2021), preserving both global semantic information and local spatial structure for subsequent observer-specific metric learning. It is trained using reconstruction together with self-supervised consistency constraints, yielding a shared semantic representation from which observer-specific metrics can subsequently be learned. The training objectives are described below (Fig. 1b).

3.4 Semantic Compression and Expansion

A conventional autoencoder compresses its input through a single latent bottleneck (Alemi et al., 2016; Hinton & Salakhutdinov, 2006), which can entangle multiple semantic concepts within a compact representation. To better separate and reuse these concepts for observer-specific metric learning, the Semantic Factorizer employs a semantic compression–expansion process. For notational convenience, we use uppercase symbols to denote the corresponding batch representations in the remainder of this section. Given the encoder output $E(X)$, the PreCompressor produces a representation C_{pre}. The Expander reconstructs this representation through a semantic space rather than simply increasing its feature dimension. Specifically, C_{pre} is converted into spatial tokens, from which keys K and values V are obtained. A set of learnable semantic queries Q aggregates information across spatial locations using cross-attention (Vaswani et al., 2017), with $A = \mathrm{Softmax}(QK^\top/\sqrt{d})$ and $U = AV$, where d is the key dimension. Each query therefore captures a different combination of spatial information in the semantic space. The semantic representation U is projected back to spatial tokens, reshaped into a feature map, and refined by local convolutions to produce the expanded representation Z. The PostCompressor maps Z to C_{post}, suppressing redundant degrees of freedom introduced during expansion. This compression–semantic reconstruction–recompression process provides a structured latent space that combines global semantic aggregation with local spatial structure for subsequent observer-specific metric learning.

3.5 Self-supervised Semantic Factorization

Reconstruction alone may allow the Expander to learn a trivial mapping without forming meaningful semantic structure. We therefore introduce compression and permutation-based cycle consistency losses to encourage a structured latent representation. The overall objective is

$$\mathcal{L} = \mathcal{L}_{\text{rec}} + \lambda_1 \mathcal{L}_{\text{comp}} + \lambda_2 \mathcal{L}_{\text{perm}}. \tag{4}$$

The reconstruction loss is defined as $\mathcal{L}_{\text{rec}} = \mathrm{MSE}(X, \hat{X})$, where X and $\hat{X}$ denote the input and reconstructed samples, respectively. The compression consistency loss, $\mathcal{L}_{\text{comp}} = \mathrm{MSE}(C_{\text{pre}}, C_{\text{post}})$, encourages consistency between the compressed representations before and after semantic expansion. We further impose cycle consistency on the semantic representation Z. Let P_r denote a random channel-wise permutation that independently permutes sample assignments within each semantic channel, and let $Z'_r = P_r(Z)$. Both the original and permuted representations are passed through the reconstruction and re-encoding path,

$$Z \xrightarrow{\text{PostCompressor}} C_{\text{post}} \xrightarrow{D} \hat{X} \xrightarrow{E} \hat{C}_{\text{pre}} \xrightarrow{\text{Expander}} \hat{Z}, \tag{5}$$

Table 1: **ImageNet-100 class pairs used to construct the 12 binary subdomains.**

Domain	Category pair	Domain	Category pair	Domain	Category pair
D1	Saluki – Toy Terrier	D5	Komondor – Chime	D9	Throne – Pickup Truck
D2	Window Screen – Sarong	D6	Chihuahua – Shih-Tzu	D10	Lorikeet – Mortarboard
D3	Carbonara – Gasmask	D7	Robin – Kuvasz	D11	Red Fox – Stinkhorn
D4	Computer Keyboard – Gibbon	D8	Milk Can – African Hunting Dog	D12	Vizsla – Great Dane

with the same path applied to Z'_r to obtain $\hat{Z}'_r$. The permutation-based cycle consistency loss is defined as $\mathcal{L}_{\text{perm}} = \text{MSE}(\hat{Z}, Z) + \mathbb{E}_{P_r}[\text{MSE}(\hat{Z}'_r, Z'_r)]$, where the expectation is approximated by Monte Carlo sampling during training. Together, $\mathcal{L}_{\text{rec}}$ preserves input information, $\mathcal{L}_{\text{comp}}$ maintains consistency across semantic expansion, and $\mathcal{L}_{\text{perm}}$ encourages the semantic representation to remain recoverable under channel-wise recombination, providing a structured basis for subsequent observer-specific component selection and reweighting (See Appendix A.2 and A.3).

3.6 Spatial Architecture and Dimension Selection

For the spatial implementation used in our experiments, the PreCompressor, Expander, and PostCompressor representations are $C_{\text{pre}} \in \mathbb{R}^{d_c \times 7 \times 7}$, $Z \in \mathbb{R}^{d_z \times 7 \times 7}$, and $C_{\text{post}} \in \mathbb{R}^{d_c \times 7 \times 7}$, respectively, where d_c and d_z denote the numbers of compressed and expanded semantic channels, respectively, with $d_z > d_c$. The decoder architecture is implementation-dependent; our image experiments use transposed-convolution upsampling followed by a three-channel CNN output layer. More generally, the Semantic Factorizer retains the bottleneck–expansion–bottleneck structure, while the backbone, spatial resolution, and decoder can be adapted to the data modality and architecture. We determine d_c and d_z using a two-stage elbow-point procedure (Satopaa et al., 2011). We first select the smallest d_c before reconstruction degrades sharply without expansion, then increase d_z with d_c fixed until further expansion yields little improvement. The dimensions are fixed before OSML.

3.7 Observer-specific Metric Learning via Gated Projection

After pretraining the Semantic Factorizer, the decoder is discarded and the semantic representation is used for observer-specific metric learning. We denote the pretrained mapping from input to semantic representation as $Z = F(X)$. We assume that observer differences can be modeled by assigning different importance to shared semantic components rather than learning entirely new representations. Let $Z = [z_1, \ldots, z_{d_z}]$ denote the semantic representation, where z_k represents the kth semantic channel, and let $g = [g_1, \ldots, g_{d_z}]$ denote the corresponding importance gate. We set $g_k = 1$ for all $k = 1, \ldots, d_z$ and train the metric head to preserve the target observer's triplet relations among the resulting embeddings. Afterward, each semantic channel is independently permuted across the validation samples, and the resulting change in validation performance is used as its importance score. A larger performance drop indicates greater importance of that channel to the target observer. The importance scores are normalized to a range $[a, b]$ to initialize g. The metric learner is then retrained using the gated representation $\tilde{Z} = g \odot Z$, where $\odot$ denotes channel-wise multiplication, while g is jointly optimized with the metric head. This gated projection uses permutation sensitivity to estimate observer-specific importance and reformulates observer-specific metric learning as a weighting problem over a shared semantic space. The overall procedure is summarized in Algorithm 1.

4 Experiments

4.1 Experimental Setup

We use two dataset settings. To assess OSML, we construct 12 binary subdomains by randomly selecting class pairs from ImageNet-100, denoted D1–D12. All subdomains use the same data source and binary-class setup, allowing comparisons across semantic domains under matched conditions. The class pairs are listed in Table 1. To evaluate SFGP for OSML, we use three benchmarks: **CUB-200-2011 (CUB)** (Wah et al., 2011), **Stanford Cars (CAR)** (Krause et al., 2013), and **ImageNet-100 (IMG)** (Deng et al., 2009). For comparability, we construct a class-balanced training pool of 5,864 samples for each dataset by random subsampling to match the smallest dataset. Training samples are drawn from these fixed pools. We use **ResNet-18** and **ViT-Tiny** for binary-subdomain experiments,

Algorithm 1 Metric Learning with SFGP

Require: Domain data $\mathcal{X}$, target-observer training triplets $\mathcal{D}_t$, validation triplets $\mathcal{V}_t$, gate range $[a, b]$
Ensure: Observer-specific metric (embedding) function f_t

Stage 1: Semantic factorization
Pretrain the Semantic Factorizer F on $\mathcal{X}$ using $\mathcal{L}_{\text{rec}} + \lambda_1 \mathcal{L}_{\text{comp}} + \lambda_2 \mathcal{L}_{\text{perm}}$
Stage 2: Observer-specific gated projection
Set $g_k \leftarrow 1$ for $k = 1, \ldots, d_z$ and define $f_t(x) = H(g \odot F(x))$
Train H on $\mathcal{D}_t$ to preserve target triplet relations among $\{f_t(x_a), f_t(x_b), f_t(x_c)\}$, with $\mathcal{V}_t$
Evaluate baseline validation performance A_{val}
for $k = 1, \ldots, d_z$ **do**
    Permute channel k across $\mathcal{V}_t$, evaluate $A_{\text{val}}^{(k)}$, and set $s_k \leftarrow A_{\text{val}} - A_{\text{val}}^{(k)}$
end for
Normalize $\{s_k\}_{k=1}^{d_z}$ to $[a, b]$ to initialize g
Reinitialize H and jointly optimize H and g on $\mathcal{D}_t$ by enforcing the target triplet relations
return $f_t(x) = H(g \odot F(x))$

Table 2: **Population-aware performance across 12 domains (D1–D12).** D1–D6 are shown individually; Avg. is the mean over all 12 domains. Values are averaged across target observers and trials, with $A_{\text{pop}} = \langle A_{\text{pop},m} \rangle_m$, $A_{\text{sp}} = \langle A_{\text{sp},m} \rangle_m$, and $\text{OSG} = \langle \text{OSG}_m \rangle_m$. $\overline{D}_m$ denotes mean observer diversity. The highest OSG for each backbone and reported domain is bolded.

		D1			D2			D3			D4			D5			D6			**Avg.**		
Backbone	Alg.	A_{pop}	A_{sp}	OSG	A_{pop}	A_{sp}	OSG	A_{pop}	A_{sp}	OSG	A_{pop}	A_{sp}	OSG	A_{pop}	A_{sp}	OSG	A_{pop}	A_{sp}	OSG	A_{pop}	A_{sp}	OSG
ResNet-18	TM	.631	.673	**.109**	.712	.747	**.119**	.794	.812	**.091**	.585	.618	**.078**	.522	.541	.041	.562	.600	**.088**	.669	.692	**.068**
	SL	.591	.604	.032	.660	.657	-.013	.783	.790	.022	.621	.634	.034	.502	.513	.021	.560	.586	.055	.653	.663	.029
	CC	.572	.605	.086	.688	.703	.047	.769	.777	.030	.629	.628	-.003	.532	.546	.033	.567	.585	.045	.656	.669	.034
	AG	.579	.576	-.013	.633	.637	.011	.812	.817	.022	.576	.598	.055	.514	.539	**.051**	.531	.544	.031	.646	.655	.022
		$\overline{D}_m = .370$			$\overline{D}_m = .240$			$\overline{D}_m = .162$			$\overline{D}_m = .342$			$\overline{D}_m = .342$			$\overline{D}_m = .417$			Avg. $\overline{D}_m = .258$		
ViT-Tiny	TM	.894	.914	**.145**	.678	.850	**.553**	.801	.834	**.168**	.727	.857	**.556**	.823	.895	**.471**	.897	.925	**.241**	.752	.846	**.387**
	SL	.865	.878	.077	.647	.767	.369	.757	.781	.090	.726	.830	.455	.798	.861	.334	.819	.838	.074	.730	.805	.284
	CC	.878	.888	.072	.659	.744	.290	.797	.827	.150	.731	.822	.413	.818	.878	.336	.861	.890	.171	.748	.812	.266
	AG	.910	.924	.143	.677	.762	.349	.719	.717	-.013	.752	.825	.438	.845	.890	.455	.920	.928	.104	.757	.803	.253
		$\overline{D}_m = .055$			$\overline{D}_m = .306$			$\overline{D}_m = .185$			$\overline{D}_m = .274$			$\overline{D}_m = .151$			$\overline{D}_m = .063$			Avg. $\overline{D}_m = .215$		

and **ResNet-50** and **ViT-Small** for multi-class experiments (He et al., 2016; Touvron et al., 2021; Dosovitskiy et al., 2020).

4.2 Population-Aware Evaluation of Observer-Specific Metric Learning

To evaluate OSML, we constructed independent populations of 20 model observers for each D1–D12 domain. Each observer was instantiated as a classifier trained on 1,000 class-balanced samples using either ResNet-18 or ViT-Tiny. Training hyperparameters were fixed within each backbone, while random seeds varied initialization and data sampling, yielding observers with distinct learned representations and similarity metrics. Observer representations were extracted immediately before the classification head, using the 512-dimensional post-pooling representation for ResNet-18 and the 192-dimensional pre-logits representation for ViT-Tiny. For each domain, data unused by any observer were sampled to construct 4,140 common triplets. Each observer labeled these triplets by comparing distances between samples in its representation space, yielding an observer-specific triplet judgment set divided into 3,420 training, 360 validation, and 360 test judgments. A separate learner was trained for each observer using the same backbone family with a 64-dimensional metric head. All learners within a domain received judgments on the same triplets. We trained the learners under four metric-learning objectives: triplet margin (TM) (Schroff et al., 2015), soft-margin triplet (SL) (Hermans et al., 2017), Circle (CC) (Sun et al., 2020), and angular (AG) (Wang et al., 2017) (See Appendix A.4). Table 2 reports A_{pop}, A_{sp}, OSG, and D_m, averaged across target observers and trials. Because D_m depends only on the observer population, it is independent of the learner objective. Under otherwise matched conditions, TM generally produced larger OSG than the other objectives, showing that population-aware evaluation can distinguish learning objectives by the amount of target-specific information recovered beyond population-level predictability. Fig. 2 further relates observer diversity to population predictability. Within domains, D_m was associated with

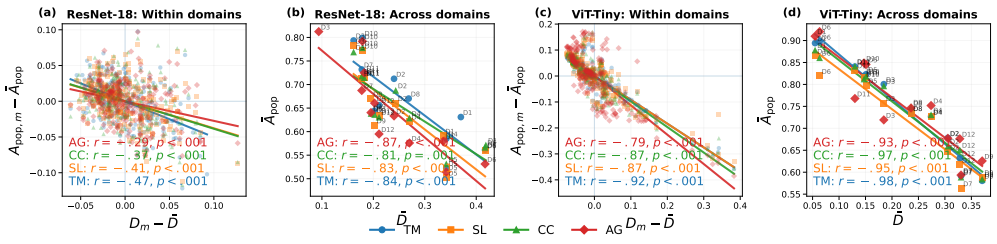

Figure 2: **Observer diversity and population predictability within and across domains.** Within-domain associations use population-centered values, $D_m - \bar{D}$ and $A_{\text{pop},m} - \bar{A}_{\text{pop}}$, whereas across-domain associations use the corresponding domain means, $\bar{D}$ and $\bar{A}_{\text{pop}}$. Points represent individual observers within domains and domain means across domains. Lines show algorithm-specific least-squares regressions for TM, SL, CC, and AG; Pearson's r and p-values are reported in each panel.

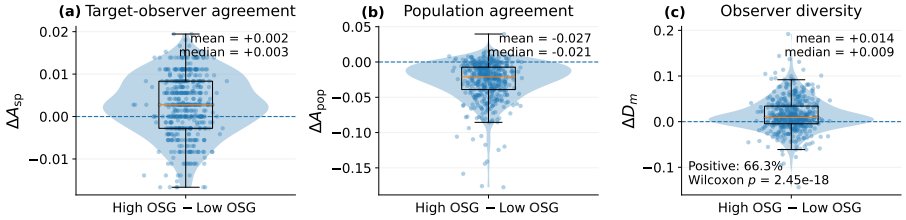

Figure 3: **Paired comparison of high- and low-OSG learners matched by experimental condition and target agreement.** Differences are computed as High OSG minus Low OSG for (a) A_{sp}, (b) A_{pop}, and (c) D_m. Each point represents one of 492 matched strata (A_{sp} bin width $= .02$).

$A_{\text{pop},m}$ after removing domain-level mean differences, while the relationship was more pronounced across domains between $\bar{D}$ and $\bar{A}_{\text{pop}}$. Populations with greater metric diversity tended to show lower population predictability, consistent with less similarity information being shared across observers. These results do not establish one metric-learning objective as generally superior, as each objective depends on its hyperparameters and training configuration. Rather, they show that population-aware evaluation can distinguish metric-learning methods by how much observer-specific information they recover beyond shared population structure.

4.3 Validating OSG as a Measure of Observer-Specific Learning

We next examined whether OSG reflects observer-specific learning rather than simply higher agreement with the target observer. Learners were matched within the same backbone, learning objective, domain, and trial, and further grouped into A_{sp} bins of width .02 to control for target-observer agreement. Within each matched stratum, learners were divided into high- and low-OSG groups. As shown in Fig. 3, target agreement was closely matched between the two groups ($\Delta A_{\text{sp}} = .002$), whereas high-OSG learners showed lower population agreement ($\Delta A_{\text{pop}} = -.027$). More importantly, they were associated with more distinctive target observers ($\Delta D_m = .014$; median $= .009$). The difference was positive in 66.3% of matched strata and significant in a paired one-sided Wilcoxon test ($p = 2.45 \times 10^{-18}$) (Wilcoxon, 1945). The decrease in A_{pop} is partly expected because OSG increases as A_{pop} decreases when A_{sp} is matched. In contrast, D_m is defined independently of learner performance. Its systematic increase with OSG therefore provides independent evidence that larger OSG is associated with targets that are more distinct from their population. Together, these results support OSG as a measure of observer-specific agreement recovered beyond what is predictable from population-level structure.

4.4 Observer-Specific Metric Learning with SFGP

We evaluated SFGP for metric learning and OSML on the three M-datasets. For each of CUB, CAR, and IMG, we independently sampled 3,000 class-balanced images from the fixed training pool ten

Table 3: **Observer-specific metric learning performance across representation spaces.** Results are reported for 1,000 and 2,000 training triplets on CAR, CUB, and IMG using ResNet-50 and ViT-Small. A_{sp} and OSG denote the averages of $A_{\mathrm{sp},m}$ and OSG_m across target observers.

	ResNet-50								ViT-Small							
	CAR				CUB				IMG				CAR				CUB				IMG			
	1,000		2,000		1,000		2,000		1,000		2,000		1,000		2,000		1,000		2,000		1,000		2,000	
Method	A_{sp}	OSG	A_{sp}	OSG	A_{sp}	OSG	A_{sp}	OSG	A_{sp}	OSG	A_{sp}	OSG	A_{sp}	OSG	A_{sp}	OSG	A_{sp}	OSG	A_{sp}	OSG	A_{sp}	OSG	A_{sp}	OSG
Direct AG	.519	-.014	.494	-.020	.534	-.012	.512	-.026	.495	**-.010**	.506	**.001**	.532	**.000**	.529	**.014**	.529	**-.028**	.540	**-.001**	.515	**-.010**	.536	**-.013**
SFGP+AG	**.614**	**.001**	**.618**	**-.012**	**.608**	**.005**	**.594**	**.002**	**.647**	-.025	**.626**	-.008	**.566**	-.007	**.573**	-.030	**.563**	-.030	**.575**	-.029	**.568**	-.017	**.589**	-.072
Direct AMST	.568	.005	.587	.012	.604	.008	.631	**.047**	.577	.010	.613	.022	.575	.012	.601	.015	.600	.020	**.632**	.036	.579	.002	.631	.032
SFGP+AMST	**.645**	**.037**	**.652**	**.042**	**.643**	**.029**	**.635**	.034	**.677**	**.045**	**.694**	**.052**	**.634**	**.013**	**.645**	**.022**	**.616**	**.030**	.625	**.041**	**.644**	**.012**	**.660**	**.042**
Direct CC	.565	.015	.583	.007	.595	.016	.628	**.045**	.572	.020	.606	.012	.583	.011	.588	.014	.597	.022	.618	**.034**	.570	.021	.607	.028
SFGP+CC	**.632**	**.015**	**.636**	**.014**	**.641**	**.019**	**.639**	.029	**.674**	**.022**	**.687**	**.049**	**.631**	**.020**	**.642**	**.019**	**.612**	**.033**	**.627**	.021	**.647**	**.022**	**.656**	**.030**
Direct SL	.571	.005	.591	.019	.616	.023	.614	.023	.579	.014	.609	.016	.582	.006	.586	.022	.601	.018	**.635**	**.045**	.570	-.002	.628	.023
SFGP+SL	**.644**	**.029**	**.669**	**.055**	**.645**	**.026**	**.646**	**.047**	**.699**	**.081**	**.706**	**.069**	**.621**	**.020**	**.629**	**.032**	**.607**	**.022**	.625	.038	**.637**	**.020**	**.641**	**.028**
Direct TM	.590	.007	.596	.032	.621	**.032**	.641	.051	.584	.018	.622	.023	.593	.015	.603	.032	.600	**.037**	.621	**.060**	.574	.022	.631	.037
SFGP+TM	**.652**	**.036**	**.669**	**.042**	**.634**	.031	**.647**	**.059**	**.690**	**.057**	**.711**	**.060**	**.636**	**.022**	**.654**	**.033**	**.623**	.025	**.645**	.050	**.647**	**.038**	**.659**	**.042**

Table 4: **Ablation study of SFGP.** Each component of SFGP was removed individually while keeping the remaining components unchanged. Results are reported using 1,000 training triplets and the triplet margin (TM) loss. A_{sp} and OSG denote the averages of $A_{\mathrm{sp},m}$ and OSG_m, respectively.

	Removed Component					ResNet-50						ViT-Small					
Method	Gate	L_{comp}	L_{perm}	Comp.	Expand.	CAR		CUB		IMG		CAR		CUB		IMG	
						A_{sp}	OSG	A_{sp}	OSG	A_{sp}	OSG	A_{sp}	OSG	A_{sp}	OSG	A_{sp}	OSG
SFGP						**0.652**	**0.036**	**0.634**	**0.031**	**0.690**	**0.057**	**0.636**	0.022	**0.623**	**0.025**	**0.660**	**0.038**
w/o Gate	✓					0.626	0.034	0.624	0.017	0.683	0.045	0.630	0.018	0.619	0.019	0.642	0.037
w/o L_{comp}		✓				0.639	0.001	0.633	0.023	0.668	0.043	0.626	0.038	0.615	0.022	0.645	0.017
w/o L_{perm}			✓			0.638	0.015	0.615	0.019	0.683	0.048	0.593	-0.018	0.614	0.005	0.633	0.013
w/o Compression				✓		0.635	0.014	0.622	0.016	0.681	0.052	0.620	**0.024**	0.618	0.014	0.649	0.042
w/o Expansion					✓	0.633	-0.003	0.629	-0.005	0.682	0.046	0.629	0.022	0.620	0.021	0.635	0.017
Conventional AE	✓	✓	✓		✓	0.638	0.028	0.622	-0.004	0.656	0.028	0.630	0.022	0.605	0.004	0.641	0.029
β-VAE						0.580	0.007	0.552	0.008	0.672	0.035	0.598	0.009	0.565	-0.004	0.652	0.028
Pretrained-z						0.554	-0.034	0.557	0.009	0.568	-0.010	0.537	-0.015	0.561	-0.017	0.552	-0.005

times to train 10 ResNet-18 observer classifiers, with 200, 196, and 100 output classes, respectively. Triplets constructed from each dataset's test pool were labeled by each observer. Separately, we pretrained ResNet-50- and ViT-Small-based Semantic Factorizers on unlabeled images from each dataset, using $\lambda_1 = 10^{-8}$ and $\lambda_2 = 10^{-4}$. The compression and expansion dimensions (d_c, d_z) were $(160, 192)$ for CAR and CUB and $(192, 240)$ for IMG. The resulting factorized representations Z were paired with the ResNet-18 observer judgments for metric learning. For each observer and factorizer backbone, we trained a metric head consisting of a 2D residual block, global average pooling, a 512-dimensional nonlinear projection, and a linear projection to a 64-dimensional metric embedding. Channel-wise gates were initialized from observer-specific channel importance within $(0.95, 1.05)$ and jointly optimized with the metric head. We used 1,000 or 2,000 training triplets, 424 validation triplets, and 500 test triplets.

Table 3 compares SFGP with direct metric learning, where a 64-dimensional metric head was attached to ResNet-50 or ViT-Small and optimized using the same observer-specific judgments without semantic factorization or gated projection. We evaluated TM, SL, CC, AG, and adaptive-margin soft triplet (AMST) losses. Results are reported as the median of three independent runs. SFGP achieved higher A_{sp} and OSG in most settings, with particularly consistent gains in A_{sp} and generally larger gains under the smaller triplet budget. In some CUB settings, however, higher A_{sp} did not yield higher OSG, showing that successful recovery of population-shared similarity structure does not necessarily imply greater observer-specific recovery. Thus, population-aware evaluation distinguishes overall metric learning from recovery of observer-specific variation (See Appendix A.5).

Table 4 further evaluates the SFGP components. Removing the gate, compression, expansion, or either factorization loss generally reduced performance, with varying effects across datasets and backbones. We also compared SFGP with a conventional autoencoder, a β-VAE (Higgins et al., 2017), and Pretrained-z, which directly feeds the pretrained backbone representation to the metric head. These results support the combined semantic-factorization and gated-projection design over dimensionality reduction or pretrained representations alone (See Appendix A.7).

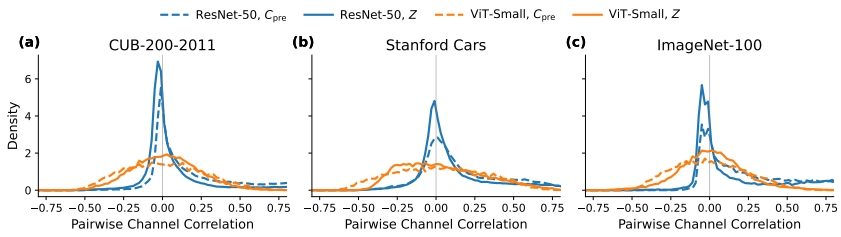

Figure 4: **Inter-channel correlation before and after latent expansion.** Pairwise Pearson correlation distributions for C_{pre} and Z using two backbones on (a) CUB, (b) CAR, and (c) IMG.

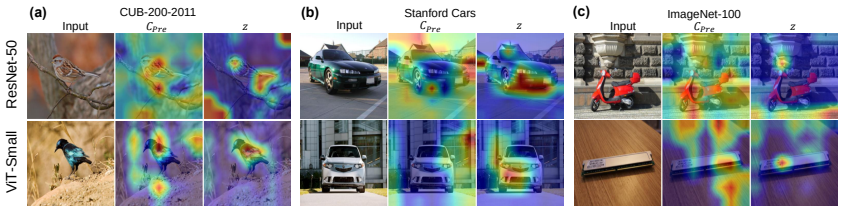

Figure 5: **Spatial activation patterns before and after semantic factorization.** Representative channel activation maps for C_{pre} and Z using two backbones on (a) CUB, (b) CAR, and (c) IMG.

4.5 Validating Semantic Factorization in the Butterfly Network

Having evaluated SFGP for OSML, we next examined the factorized representation Z in terms of inter-channel redundancy and spatial activation patterns. Fig. 4 compares pairwise channel correlations before and after latent expansion. Across all three datasets and both backbones, correlations in Z were more concentrated around zero than in C_{pre}, indicating reduced linear redundancy among the expanded channels. Further, Fig. 5 shows that representative channels in Z exhibited more spatially localized and differentiated responses to salient image regions than those in C_{pre}. Although these observations do not establish complete semantic disentanglement, reduced inter-channel redundancy and differentiated spatial responses indicate that expansion reorganizes C_{pre} into a more factorized representation, supporting the semantic factorization underlying SFGP's observer-specific gated projection (See Appendix A.8).

5 Conclusions

We introduced OSML as the problem of recovering an individual observer's similarity metric from sparse relational judgments. Our population-aware evaluation distinguishes target agreement from similarity structure predictable at the population level, with OSG quantifying target-specific agreement beyond this population predictability. The observed relationship between observer diversity and cross-observer predictability further shows why target agreement alone is insufficient to characterize observer-specific metric recovery. We further proposed SFGP to constrain OSML under sparse supervision. Across datasets, backbones, and metric-learning objectives, SFGP improved target agreement and achieved larger observer-specific gains than direct metric learning, particularly under limited triplet supervision. Representation and ablation analyses further support the complementary roles of semantic factorization and gated projection. While our experiments use neural networks as controlled observers, the framework is intended to extend to settings where individual metrics are more heterogeneous and observations are limited. Human observers represent an important case, as exhaustive sampling of individual judgments is often impractical. Future work should examine the reliability and stability of human judgments, population composition, efficient judgment sampling, and improved semantic factorization. More broadly, OSML provides a framework for characterizing and evaluating similarity as a population of related yet individually distinct metrics, and for determining how well individual similarity structure can be recovered beyond what is predictable from the population.

AI USE STATEMENT

We used large language models (LLMs) to assist with grammatical editing, text condensation, and LaTeX formatting. LLM assistance was applied at the paragraph level rather than to the manuscript as a whole. We also used LLMs to search for relevant literature, and all retrieved references and their relevance to this work were independently verified by the authors. LLMs were further used to assist with Python code implementation for the experiments. Their use was limited to implementing specific, author-defined algorithms as individual modules rather than developing or modifying the project as a whole, and all AI-assisted code was reviewed and validated by the authors. To minimize the risk of unintended changes or errors, we did not use agentic AI tools operating from high-level instructions; instead, LLMs were used interactively to provide specific outputs in response to explicit, narrowly scoped instructions. The authors take full responsibility for the final manuscript, code, and experimental results.

REPRODUCIBILITY STATEMENT

We provide the code used in our experiments as supplementary material, along with documentation describing the data preprocessing steps needed to run the experiments. The benchmark datasets are publicly available and are therefore not included. Instead, we provide detailed instructions for preparing the datasets in the same way as in our experiments. The Appendix also provides the main experimental settings, parameter values, and baseline methods used in our comparisons.

REFERENCES

Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. Deep variational information bottleneck. *arXiv preprint arXiv:1612.00410*, 2016.

Vivek Anand, Alec Helbling, Mark Davenport, Gordon Berman, Sankaraleengam Alagapan, and Christopher Rozell. Lore: Jointly learning the intrinsic dimensionality and relative similarity structure from ordinal data. In *International Conference on Learning Representations*, volume 2026, pp. 3622–3655, 2026.

Gregory Canal, Blake Mason, Ramya Korlakai Vinayak, and Robert Nowak. One for all: Simultaneous metric and preference learning over multiple users. *arXiv preprint arXiv:2207.03609*, 2022.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1597–1607. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/chen20j.html.

Jason V Davis, Brian Kulis, Prateek Jain, Suvrit Sra, and Inderjit S Dhillon. Information-theoretic metric learning. In *Proceedings of the 24th international conference on Machine learning*, pp. 209–216, 2007.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Frances Ding, Jean-Stanislas Denain, and Jacob Steinhardt. Grounding representation similarity through statistical testing. *Advances in neural information processing systems*, 34:1556–1568, 2021.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, volume 2, pp. 1735–1742. IEEE, 2006.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*, 2017.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. β-vae: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations (ICLR)*, 2017. Poster.

Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.

Nghia Hoang, Anoop Deoras, Tong Zhao, Jin Li, and George Karypis. Learning personalized item-to-item recommendation metric via implicit feedback. In Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera (eds.), *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 1062–1077. PMLR, 28–30 Mar 2022. URL https://proceedings.mlr.press/v151/hoang22a.html.

David Izydorczyk and Arndt Bröder. Food, mammals, & countries: Stimuli, their pairwise similarities, and mds dimension values for modeling judgment, categorization, and memory in three real-world domains. *Behavior Research Methods*, 58(9):265, 2026.

Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *International conference on machine learning*, pp. 4651–4664. PMLR, 2021.

Hyunjik Kim and Andriy Mnih. Disentangling by factorising. In *International conference on machine learning*, pp. 2649–2658. PMLR, 2018.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International conference on machine learning*, pp. 3519–3529. PMlR, 2019.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pp. 554–561, 2013.

Tyler Malloy, Maria José Ferreira, Fei Fang, and Cleotilde Gonzalez. Leveraging a cognitive model to measure subjective similarity of human and GPT-4 written content. In Libby Barak and Malihe Alikhani (eds.), *Proceedings of the 28th Conference on Computational Natural Language Learning*, pp. 517–527, Miami, FL, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.conll-1.40. URL https://aclanthology.org/2024.conll-1.40/.

Emile Mathieu, Tom Rainforth, Nana Siddharth, and Yee Whye Teh. Disentangling disentanglement in variational autoencoders. In *International conference on machine learning*, pp. 4402–4412. PMLR, 2019.

Guo-Jun Qi, Jinhui Tang, Zheng-Jun Zha, Tat-Seng Chua, and Hong-Jiang Zhang. An efficient sparse metric learning in high-dimensional space via l 1-penalized log-determinant regularization. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pp. 841–848, 2009.

Brett D Roads and Bradley C Love. Enriching imagenet with human similarity judgments and psychological embeddings. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3546–3556. IEEE, 2021.

Brett D Roads and Michael C Mozer. Obtaining psychological embeddings through joint kernel and metric learning. *Behavior research methods*, 51(5):2180–2193, 2019.

Ville Satopaa, Jeannie Albrecht, David Irwin, and Barath Raghavan. Finding a" kneedle" in a haystack: Detecting knee points in system behavior. In *2011 31st international conference on distributed computing systems workshops*, pp. 166–171. IEEE, 2011.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 815–823, 2015.

Sabrina Simmons and Zachary Estes. Individual differences in the perception of similarity and difference. *Cognition*, 108(3):781–795, 2008.

Yifan Sun, Changmao Cheng, Yuhan Zhang, Chi Zhang, Liang Zheng, Zhongdao Wang, and Yichen Wei. Circle loss: A unified perspective of pair similarity optimization. In *2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 6397–6406. IEEE, 2020.

Shobhita Sundaram, Julia Chae, Yonglong Tian, Sara Beery, and Phillip Isola. Personalized representation from personalized generation. In *International Conference on Learning Representations*, volume 2025, pp. 92572–92606, 2025.

Yoshikazu Terada and Ulrike Luxburg. Local ordinal embedding. In Eric P. Xing and Tony Jebara (eds.), *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pp. 847–855, Beijing, China, 22–24 Jun 2014. PMLR. URL https://proceedings.mlr.press/v32/terada14.html.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pp. 10347–10357. PMLR, 2021.

Leena Chennuru Vankadara, Michael Lohaus, Siavash Haghiri, Faiz Ul Wahab, and Ulrike Von Luxburg. Insights into ordinal embedding algorithms: A systematic evaluation. *Journal of Machine Learning Research*, 24(191):1–83, 2023.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

Jian Wang, Feng Zhou, Shilei Wen, Xiao Liu, and Yuanqing Lin. Deep metric learning with angular loss. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2612–2620. Ieee, 2017.

Zhi Wang, Geelon So, and Ramya Korlakai Vinayak. Metric learning from limited pairwise preference comparisons. *arXiv preprint arXiv:2403.19629*, 2024.

Kilian Q Weinberger and Lawrence K Saul. Distance metric learning for large margin nearest neighbor classification. *Journal of machine learning research*, 10(2), 2009.

Michael Wilber, Iljung Kwak, and Serge Belongie. Cost-effective hits for relative similarity comparisons. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 2, pp. 227–233, 2014.

Frank Wilcoxon. Individual comparisons by ranking methods. *Biometrics bulletin*, 1(6):80–83, 1945.

A Appendix

A.1 Mathematical Interpretation of Observer-Specific Gain

We provide a mathematical interpretation of the Observer-Specific Gain (OSG) used in our population-aware evaluation. For a learner f_m trained on the judgments of target observer o_m, let

$$A_{\mathrm{sp},m} = A(f_m, o_m) \tag{6}$$

denote agreement with the target observer, and

$$A_{\mathrm{pop},m} = \frac{1}{M-1} \sum_{j \neq m} A(f_m, o_j) \tag{7}$$

denote mean agreement with the remaining observers. OSG is defined as

$$\mathrm{OSG}_m = \frac{A_{\mathrm{sp},m} - A_{\mathrm{pop},m}}{1 - A_{\mathrm{pop},m}}, \qquad A_{\mathrm{pop},m} < 1. \tag{8}$$

Population agreement as a baseline. The use of $A_{\mathrm{pop},m}$ as a baseline follows from evaluating the same learner f_m against the target and non-target observers. Because f_m is trained only on judgments from o_m, its agreement with observers o_j, $j \neq m$, reflects the extent to which what it learned from the target transfers to observers whose judgments were not used for training. Structure in the target judgments that is broadly shared across observers should support agreement not only with o_m, but also with non-target observers, and therefore contribute to $A_{\mathrm{pop},m}$. By contrast, structure specific to o_m should preferentially increase agreement with the target rather than agreement with the remaining population.

Accordingly, $A_{\mathrm{pop},m}$ provides an empirical reference for the cross-observer generalizability of the learned metric. This distinction is important because high target agreement alone does not establish observer-specific learning. For example, if observers make highly similar judgments, a learner may achieve high $A_{\mathrm{sp},m}$ simply by learning structure that is broadly shared across the population. Comparing $A_{\mathrm{sp},m}$ with the agreement achieved by the same learner on non-target observers distinguishes such population-generalizable agreement from agreement that is preferentially directed toward the target.

Thus, the difference

$$A_{\mathrm{sp},m} - A_{\mathrm{pop},m} \tag{9}$$

measures excess target agreement relative to the learner's own cross-observer generalization level. Importantly, this does not require $A_{\mathrm{pop},m}$ to be an exact estimate of a latent population-shared component. Rather, it serves as an operational baseline obtained by evaluating the same learned metric on observers not used for its training.

Normalized excess agreement. For a fixed $A_{\mathrm{pop},m}$, the quantity $1 - A_{\mathrm{pop},m}$ represents the maximum possible improvement in agreement above the population baseline, since agreement cannot exceed one. The numerator, $A_{\mathrm{sp},m} - A_{\mathrm{pop},m}$, is the observed excess agreement with the target observer relative to this baseline. Therefore,

$$A_{\mathrm{sp},m} - A_{\mathrm{pop},m} = \mathrm{OSG}_m \left(1 - A_{\mathrm{pop},m}\right), \tag{10}$$

or equivalently,

$$A_{\mathrm{sp},m} = A_{\mathrm{pop},m} + \mathrm{OSG}_m \left(1 - A_{\mathrm{pop},m}\right). \tag{11}$$

Thus, OSG measures the fraction of the available improvement beyond population agreement that is realized as additional agreement with the target observer.

Boundary properties. The definition has several direct consequences. First,

$$\mathrm{OSG}_m = 0 \quad \Longleftrightarrow \quad A_{\mathrm{sp},m} = A_{\mathrm{pop},m}, \tag{12}$$

such that no excess agreement with the target is observed relative to the population baseline. Second,

$$\mathrm{OSG}_m = 1 \quad \Longleftrightarrow \quad A_{\mathrm{sp},m} = 1, \tag{13}$$

corresponding to perfect target agreement. Consequently, whenever $A_{\mathrm{sp},m} \in [A_{\mathrm{pop},m}, 1]$,

$$0 \leq \mathrm{OSG}_m \leq 1. \tag{14}$$

If $A_{\mathrm{sp},m} < A_{\mathrm{pop},m}$, OSG becomes negative, indicating that the learner agrees less with its target observer than, on average, with the non-target observer population.

Monotonicity with target agreement. For a fixed population baseline,

$$\frac{\partial\, \mathrm{OSG}_m}{\partial A_{\mathrm{sp},m}} = \frac{1}{1 - A_{\mathrm{pop},m}} > 0. \tag{15}$$

Hence, among learners with the same population agreement, greater agreement with the target observer necessarily corresponds to greater OSG.

Normalization for the available improvement. The raw difference $A_{\mathrm{sp},m} - A_{\mathrm{pop},m}$ depends on the amount of improvement that remains possible above $A_{\mathrm{pop},m}$. A learner with high population agreement has less room for additional target agreement than one with low population agreement. Dividing by $1 - A_{\mathrm{pop},m}$ accounts for this ceiling and expresses excess target agreement relative to the available improvement. OSG therefore places learners with different population baselines on a common relative scale.

Interpretation as observer specificity. Together, these properties provide an operational interpretation of OSG as observer-specific gain. $A_{\mathrm{pop},m}$ characterizes how well the target-trained learner generalizes across observers, while $A_{\mathrm{sp},m}$ measures its agreement with the observer whose judgments were used for learning. OSG quantifies the extent to which target agreement exceeds this cross-observer reference, normalized by the maximum possible excess above that reference. It therefore distinguishes high target agreement that is broadly shared across observers from high target agreement that is preferentially aligned with the target observer.

This interpretation should not be understood as an exact decomposition of the learned representation into statistically independent population-shared and observer-specific latent components. In particular, $A_{\mathrm{pop},m}$ is an empirical population reference rather than a direct measurement of a latent shared component. OSG instead provides an operational measure of observer specificity: it quantifies preferential agreement with the target relative to the learner's own level of cross-observer generalization.

The empirical analyses in the main text provide an additional test of this interpretation. After controlling for target agreement A_{sp}, learners with higher OSG show lower population agreement and are associated with more distinctive target observers, as measured by D_m. Thus, the mathematical interpretation of OSG as normalized excess target agreement is consistent with its empirical association with observer distinctiveness.

A.2 Semantic Factorizer Architecture and Parameters

The Semantic Factorizer follows the PreCompressor–Expander–PostCompressor architecture described in the main text. Given a $224 \times 224 \times 3$ image, a pretrained backbone first extracts an

intermediate representation, which is compressed into C_{pre}, expanded into the semantic representation Z, and recompressed into C_{post} before image reconstruction. We use ResNet-50 and ViT-Small as backbones and adapt the factorization modules to their respective feature-map and token representations.

PreCompressor. For ResNet-50, features are extracted through the third residual stage, yielding a $1024 \times 14 \times 14$ feature map. The PreCompressor applies a 3×3 convolution with stride 2, BatchNorm, ReLU, and a residual block, producing $C_{\text{pre}} \in \mathbb{R}^{d_c \times 7 \times 7}$. For ViT-Small/16, the CLS token is removed from the backbone output, leaving 196 patch tokens with embedding dimension 384. These tokens are aggregated into d_c compressed semantic units using learnable queries and multi-head cross-attention.

Semantic Expander. The Expander maps the compressed representation into a higher-capacity semantic space with $d_z > d_c$. For ResNet-50, C_{pre} is first projected from d_c to d_z channels while retaining its 7×7 spatial structure. The resulting 49 spatial tokens are aggregated using 64 learnable semantic queries and eight-head cross-attention. The semantic units are refined by a feed-forward network, projected back to the 49 spatial locations, and further refined by residual convolutional blocks to produce $Z \in \mathbb{R}^{d_z \times 7 \times 7}$.

For ViT-Small, d_z learnable semantic queries attend to the compressed semantic units through multi-head cross-attention to produce the expanded representation Z. Feed-forward refinement is applied before and after the attention operation. Thus, although the backbone representations differ in form, both implementations use learnable semantic queries to reorganize the compressed representation into an expanded semantic space.

PostCompressor and reconstruction. The expanded representation Z is recompressed to the original compression dimension d_c by the PostCompressor, producing C_{post}. Thus, both backbone-specific implementations follow the same abstract compression–expansion–recompression structure,

$$C_{\text{pre}} \in \mathbb{R}^{d_c} \;\longrightarrow\; Z \in \mathbb{R}^{d_z} \;\longrightarrow\; C_{\text{post}} \in \mathbb{R}^{d_c}, \qquad d_z > d_c,$$

where spatial or token dimensions are omitted for clarity.

For image reconstruction, C_{post} is mapped back to a spatial feature map and decoded through transposed-convolution upsampling blocks. Each upsampling block uses a 4×4 transposed convolution with stride 2, followed by BatchNorm, ReLU, and residual refinement. A final 3×3 convolution maps the decoded feature map to three channels, followed by a sigmoid activation to reconstruct the 224×224 input image.

Factorizer pretraining. The Semantic Factorizer is pretrained at the dataset level without class labels or observer-specific triplet judgments. The compression dimension d_c and expansion dimension d_z are selected separately for each dataset–backbone configuration using the two-stage elbow-point procedure described in the main text. Beyond reconstructing the input, pretraining constrains the representations before and after expansion and requires the semantic representation to remain recoverable after reconstruction and channel-wise recombination. The corresponding self-supervised objective and permutation procedure are detailed in Appendix A.3. After pretraining, the decoder is discarded and the factorized representation Z is used for subsequent observer-specific gated projection.

A.3 Detailed Procedure for Self-Supervised Semantic Factorization

We next detail the self-supervised objective used to train the Semantic Factorizer described above. It combines reconstruction, compression-consistency, and permutation-based cycle-consistency terms, all implemented using mean squared error (MSE). The overall objective is

$$\mathcal{L} = \mathcal{L}_{\text{rec}} + \lambda_1 \mathcal{L}_{\text{comp}} + \lambda_2 \mathcal{L}_{\text{perm}}. \tag{16}$$

Reconstruction and compression consistency. For an input x, the backbone and factorization modules produce C_{pre}, Z, and C_{post}, from which the decoder reconstructs $\hat{x}$. The reconstruction loss is defined as $\mathcal{L}_{\text{rec}} = \text{MSE}(x, \hat{x})$ and preserves information required to reconstruct the input.

The compression-consistency loss is $\mathcal{L}_{\text{comp}} = \text{MSE}(C_{\text{pre}}, C_{\text{post}})$. It constrains the compressed representations on the two sides of the Expander to remain consistent while allowing the intermediate representation Z to reorganize the information into semantic components.

Original latent cycle. Reconstruction of the input alone does not require the expanded semantic representation itself to be recoverable. We therefore additionally impose cycle consistency in the semantic space. Given Z, the PostCompressor and decoder first reconstruct an input, which is subsequently passed through the encoder, PreCompressor, and Expander to obtain a recovered semantic representation $\hat{Z}$:

$$Z \xrightarrow{\text{PostCompressor}} C_{\text{post}} \xrightarrow{D} \hat{x} \xrightarrow{E} \hat{C}_{\text{pre}} \xrightarrow{\text{Expander}} \hat{Z}. \tag{17}$$

The original latent-cycle loss is $\mathcal{L}_{\text{org}} = \text{MSE}(\hat{Z}, Z)$. When generating the reconstructed input for this cycle, the decoding path is evaluated without gradient propagation. Consequently, the cycle objective encourages the encoding and factorization path to recover the supplied semantic representation rather than adapting the decoder specifically to the cycle target.

Channel-wise semantic permutation. We extend this constraint by requiring recovery after recombination of semantic channels across samples. Let $Z \in \mathbb{R}^{B\times C\times H\times W}$, where B denotes the batch size, C the number of semantic channels, and $H \times W$ the spatial dimensions. For each semantic channel c, an independent permutation π_c of the samples in the mini-batch is drawn. A permuted representation $Z'_r = P_r(Z)$ is then constructed as $Z'_{r,b,c,:,:} = Z_{\pi_c(b),c,:,:}$.

This operation preserves the complete $H \times W$ spatial map within each semantic channel while independently changing its assignment across samples. Consequently, different channels of a permuted representation may originate from different samples, providing a channel-wise recombination of the semantic representation rather than perturbing the values within individual feature maps.

The recombined representation Z'_r is passed through the same reconstruction and re-encoding cycle,

$$Z'_r \xrightarrow{\text{PostCompressor}} C'_{\text{post}} \xrightarrow{D} \hat{x}'_r \xrightarrow{E} \hat{C}'_{\text{pre}} \xrightarrow{\text{Expander}} \hat{Z}'_r, \tag{18}$$

and its cycle consistency is measured by $\text{MSE}(\hat{Z}'_r, Z'_r)$. As in the original latent cycle, the reconstruction from the permuted representation is generated without gradient propagation through the decoding path.

Monte Carlo permutation consistency. Because the channel-wise permutation P_r is stochastic, consistency under a single permutation could depend strongly on a particular recombination. We therefore formulate the permutation component as an expectation over random permutations and approximate it during training using R independently sampled permutations. The resulting permutation-based cycle-consistency loss is

$$\mathcal{L}_{\text{perm}} = \mathcal{L}_{\text{org}} + \frac{1}{R}\sum_{r=1}^{R} \text{MSE}(\hat{Z}'_r, Z'_r). \tag{19}$$

The first term requires recovery of the original semantic representation, whereas the second requires recovery after channel-wise recombination. Averaging over multiple random permutations approximates the expected consistency of the factorized representation under such recombinations.

Overall training procedure. The reconstruction, compression-consistency, and permutation-based cycle-consistency terms are jointly optimized during factorizer pretraining. The reconstruction term preserves information about the input, the compression term constrains the representations surrounding the Expander, and the permutation term encourages the expanded semantic representation to remain recoverable when semantic channels are independently recombined across samples. This training is performed before observer-specific metric learning; the resulting Z is subsequently fixed as the factorized representation on which observer-specific gated projection is learned.

Factorizer pretraining hyperparameters. The Semantic Factorizer was optimized with AdamW using an initial learning rate of 5×10^{-4}, weight decay of 10^{-4}, and batch size 72. Training was performed for up to 300 epochs with cosine-annealing learning-rate scheduling and early stopping

with a patience of 20 epochs based on validation reconstruction loss. The permutation-based cycle-consistency term was estimated using 10 independently sampled channel-wise permutations per mini-batch.

A.4 Metric-Learning Objectives and Hyperparameters

We evaluated five metric-learning objectives under the same training protocol. For a triplet (a, p, n), let $\mathbf{e}_a$, $\mathbf{e}_p$, and $\mathbf{e}_n$ denote the learned embeddings of the anchor, positive, and negative samples, respectively. Unless otherwise specified, distances are squared Euclidean distances, $d_{ap} = \|\mathbf{e}_a - \mathbf{e}_p\|_2^2$ and $d_{an} = \|\mathbf{e}_a - \mathbf{e}_n\|_2^2$.

Triplet Margin Loss (TM). We used the standard triplet margin objective (Schroff et al., 2015),

$$\mathcal{L}_{\mathrm{TM}} = \left[\|\mathbf{e}_a - \mathbf{e}_p\|_2 - \|\mathbf{e}_a - \mathbf{e}_n\|_2 + m\right]_+, \tag{20}$$

with margin $m = 0.2$.

Soft-Margin Triplet Loss (SL). We replaced the hard hinge margin with a softplus objective (Hermans et al., 2017),

$$\mathcal{L}_{\mathrm{SL}} = \log\left[1 + \exp\left(d_{ap} - d_{an}\right)\right]. \tag{21}$$

No explicit margin or additional scaling parameter was used.

Circle Loss (CC). For Circle loss (Sun et al., 2020), embeddings were first ℓ_2-normalized and positive and negative cosine similarities were defined as $s_p = \cos(\mathbf{e}_a, \mathbf{e}_p)$ and $s_n = \cos(\mathbf{e}_a, \mathbf{e}_n)$. We used

$$\alpha_p = [1 + m - s_p]_+, \qquad \alpha_n = [s_n + m]_+, \tag{22}$$

$$\Delta_p = 1 - m, \qquad \Delta_n = m, \tag{23}$$

followed by

$$\mathcal{L}_{\mathrm{CC}} = \mathrm{softplus}\left[\gamma\alpha_n(s_n - \Delta_n) - \gamma\alpha_p(s_p - \Delta_p)\right]. \tag{24}$$

We used margin $m = 0.25$ and scale $\gamma = 32$.

Angular Loss (AG). For angular loss (Wang et al., 2017), the three embeddings were ℓ_2-normalized. Let $\mathbf{c} = (\mathbf{e}_a + \mathbf{e}_p)/2$. The implemented objective was

$$\mathcal{L}_{\mathrm{AG}} = \mathrm{softplus}\left[4\tan^2(\alpha)(\mathbf{e}_a - \mathbf{e}_p)^\top(\mathbf{c} - \mathbf{e}_n)\right], \tag{25}$$

with angular parameter $\alpha = 45°$.

Adaptive-Margin Soft Triplet Loss (AMST). AMST uses normalized embeddings and cosine distances $d_{ap}^c = 1 - \cos(\mathbf{e}_a, \mathbf{e}_p)$ and $d_{an}^c = 1 - \cos(\mathbf{e}_a, \mathbf{e}_n)$. For each triplet, the adaptive margin is

$$m_{\mathrm{ad}} = m_0 + \beta\left|d_{ap}^c - d_{an}^c\right|, \tag{26}$$

where the distance-dependent term is detached from gradient computation. The loss is then

$$\mathcal{L}_{\mathrm{AMST}} = \mathrm{softplus}\left[\gamma\left(d_{ap}^c - d_{an}^c + m_{\mathrm{ad}}\right)\right]. \tag{27}$$

We used base margin $m_0 = 0.20$, adaptive scale $\beta = 0.20$, and $\gamma = 20$.

Common training settings. All objectives used the same metric-learning protocol. The metric head was optimized with Adam for at most 100 epochs using a learning rate of 10^{-4}, weight decay of 10^{-5}, and batch size 128. The learning rate was scheduled with cosine annealing over the maximum training duration. Validation accuracy (agreement) was used for checkpoint selection, with early stopping after 15 epochs without improvement.

A.5 Implementation Details for SFGP and Direct Metric Learning

For the M-dataset experiments, the pretrained Semantic Factorizer was frozen during observer-specific metric learning. For each target observer, the corresponding triplet judgments were divided into training, validation, and test sets. We considered 1,000 and 2,000 training triplets, while using 424 validation and 500 test triplets in both settings. Metric learners were optimized with Adam and a cosine-annealing learning-rate schedule for up to 100 epochs, with validation-based early stopping. The initial learning rate was 10^{-4} and the batch size was 128.

For SFGP, the factorized representation Z was extracted without updating the Semantic Factorizer. A separate metric head was trained for each target observer. For spatial latent representations, the head applies a residual convolutional block followed by global average pooling, a 512-dimensional nonlinear projection, and a final projection to the metric embedding. For each triplet, the learner predicted the closer item to the anchor by comparing squared Euclidean distances in the metric space. Triplet agreement was defined as the fraction of these binary predictions matching the observer's judgments.

Gated projection was performed after an initial metric learner had been fitted without channel reweighting. For each semantic channel k, we independently permuted that channel across validation samples while preserving the remaining channels and measured the resulting decrease in triplet agreement,

$$s_k = A_{\mathrm{val}} - A_{\mathrm{val}}^{(k)}.$$

The importance scores were standardized and transformed as

$$g_k = 0.95 + 0.1\,\sigma(z_k/0.5),$$

yielding gate values initialized within $(0.95, 1.05)$. A new metric head was then initialized and trained using the gated representation $\tilde{Z} = g \odot Z$. In the implementation used for these experiments, the gate parameters were jointly optimized with the metric head.

The direct-learning baseline used the same observer-specific triplet judgments and metric-learning objectives but omitted semantic factorization and gated projection. A metric head was attached directly to the ResNet-50 or ViT-Small representation and optimized under the corresponding loss. We evaluated triplet margin (TM), soft-margin triplet (SL), Circle (CC), angular (AG), and adaptive-margin soft triplet (AMST) objectives. The same train/validation/test triplet partitions were used across methods to isolate the effect of the representation and gated projection.

For population-aware evaluation, each target-specific learner was evaluated not only against its target observer but also against the test judgments of every other observer. These cross-observer agreements were used to compute A_{pop}, while agreement with the target yielded A_{sp}; OSG was then computed from the two measures as defined in the main text. This evaluation allows improvements in overall target agreement to be distinguished from improvements specifically attributable to recovery of observer-specific structure.

A.6 Population-aware performance across 12 domains

Summary statistics for the individual datasets not reported in Table 2 or Section 4.2 are provided in Table 5.

A.7 Ablation Study

Table 4 presents an ablation study of the proposed Semantic Factorization and Gated Projection (SFGP) under the 1,000-training-triplet setting using the triplet margin (TM) loss. Each experiment was repeated three times, and the median performance is reported.

To evaluate the contribution of each component, we removed one component at a time while keeping all remaining components unchanged. Removing the gate disables observer-specific semantic

Table 5: Population-aware performance for domains D7–D12. For each target observer, D_m, A_{pop}, A_{sp}, and OSG denote observer diversity, population agreement, target agreement, and observer-specific gain, respectively. $\overline{D}_m$ denotes D_m averaged across target observers and trials. The highest OSG for each backbone and domain is shown in bold.

		D7			D8			D9			D10			D11			D12		
Backbone	Alg.	A_{pop}	A_{sp}	OSG	A_{pop}	A_{sp}	OSG	A_{pop}	A_{sp}	OSG	A_{pop}	A_{sp}	OSG	A_{pop}	A_{sp}	OSG	A_{pop}	A_{sp}	OSG
ResNet-18	TM	.732	.750	**.060**	.670	.694	**.068**	.644	.659	.035	.799	.808	**.039**	.719	.735	**.058**	.656	.669	.032
	SL	.713	.718	.031	.670	.689	.054	.613	.618	.006	.772	.778	.029	.715	.728	.048	.634	.643	.024
	CC	.718	.729	.044	.638	.652	.033	.636	.657	**.056**	.778	.772	-.041	.714	.729	.054	.631	.641	.029
	AG	.687	.691	.003	.640	.666	.067	.661	.670	.028	.792	.785	-.037	.725	.728	.011	.595	.609	**.034**
		$\overline{D}_m = .178$			$\overline{D}_m = .195$			$\overline{D}_m = .202$			$\overline{D}_m = .179$			$\overline{D}_m = .182$			$\overline{D}_m = .211$		
ViT-Tiny	TM	.591	.722	**.309**	.734	.846	.503	.580	.696	**.352**	.819	.900	**.471**	.841	.893	**.353**	.632	.825	**.527**
	SL	.563	.679	.267	.735	.832	.462	.588	.690	**.306**	.816	.872	.367	.832	.863	**.205**	.617	.768	**.403**
	CC	.589	.688	.239	.749	.864	**.510**	.584	.667	.272	.812	.841	.177	.841	.868	.201	.651	.769	.364
	AG	.593	.655	.203	.746	.836	.485	.626	.702	.296	.849	.876	.254	.768	.788	.127	.676	.732	.192
		$\overline{D}_m = .330$			$\overline{D}_m = .236$			$\overline{D}_m = .370$			$\overline{D}_m = .147$			$\overline{D}_m = .130$			$\overline{D}_m = .328$		

reweighting. Removing L_{comp} or L_{perm} eliminates the corresponding self-supervised constraint during Semantic Factorizer pretraining.

The Compression and Expansion ablations modify the Butterfly Network architecture itself. In the *w/o Compression* setting, the PreCompressor no longer compresses the encoder representation, and its output dimensionality is set equal to that of the Expander. Conversely, in the *w/o Expansion* setting, the Expander no longer increases the representation dimensionality, and its output dimensionality is fixed to that of the PreCompressor. These settings remove the asymmetric compression–expansion structure while preserving the remaining network architecture.

We further compare SFGP with three alternative representation-learning strategies. **Conventional AE** denotes a standard autoencoder trained solely with the reconstruction objective. **β-VAE** replaces the proposed self-supervised objectives with a variational bottleneck and is trained using the reconstruction loss together with a KL-divergence regularization weighted by $\beta = 10^{-3}$, following Higgins et al. (2017). The encoder, decoder, compression dimension, and semantic dimension are kept identical to those of SFGP, while the proposed compression-consistency and permutation-cycle losses are omitted. Finally, **Pretrained-z** directly feeds the pretrained backbone representation to the observer-specific metric learner without Semantic Factorizer pretraining, semantic expansion, or gated projection.

A.8 Visualization of Factorized Representations

To qualitatively examine how semantic factorization reorganizes spatial representations, we visualized channel-wise activation maps before and after the factorization stage. For an input image x, the factorizer produces the pre-factorized representation $C_{\text{pre}} \in \mathbb{R}^{d_c \times h \times w}$ and the expanded latent representation $Z \in \mathbb{R}^{d_z \times h \times w}$. For a selected channel k, we extracted its spatial response map,

$$H_k^{(r)} = r_k(x), \qquad r \in \{C_{\text{pre}}, Z\},$$

and resized it to the input resolution using bilinear interpolation. Each map was independently normalized to $[0, 1]$,

$$\widetilde{H}_k^{(r)} = \frac{H_k^{(r)} - \min H_k^{(r)}}{\max H_k^{(r)} - \min H_k^{(r)} + \epsilon},$$

and overlaid on the corresponding input image to visualize the spatial regions associated with that channel.

Figures 6–8 show representative examples from CUB-200-2011, Stanford Cars, and ImageNet-100 for both ResNet-50 and ViT-Small backbones. For each input, the visualization compares selected channels of C_{pre} with their corresponding channels in Z, allowing the spatial response patterns before and after factorization to be directly inspected.

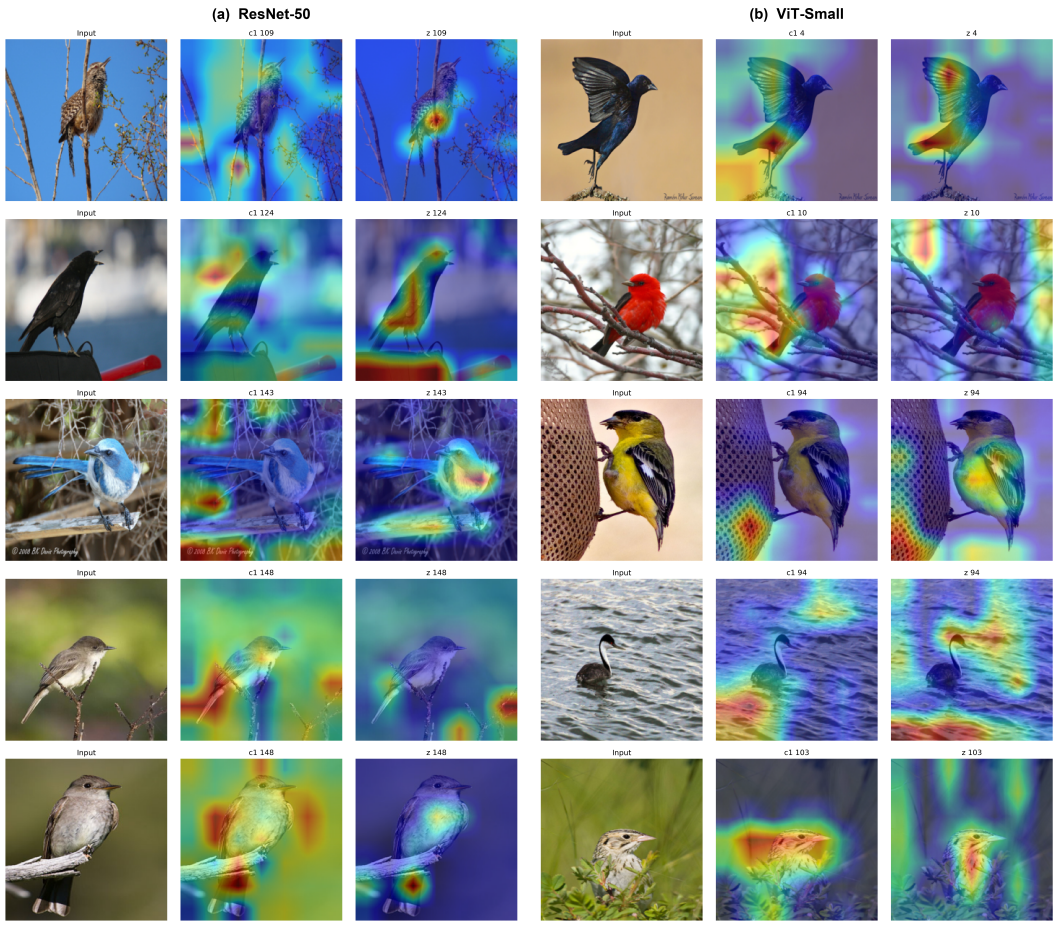

Figure 6: **Spatial activation patterns of factorized representations on CUB-200-2011.** Representative activation maps are shown for (a) ResNet-50 and (b) ViT-Small. For each example, the input image is followed by the activation maps of the pre-factorized representation C_{pre} and the expanded latent representation Z.

The visualizations show that the transformation from C_{pre} to Z modifies the spatial organization of channel responses rather than simply preserving the original activation patterns. Across examples, individual latent channels exhibit localized and distinct response patterns over different image regions. This behavior is consistent with the intended role of the factorization stage: to reorganize the backbone representation into latent components that can subsequently be selected and combined by the observer-specific gated projection. These visualizations are qualitative and are intended to complement, rather than establish, the quantitative evidence for representation factorization reported in the main text.

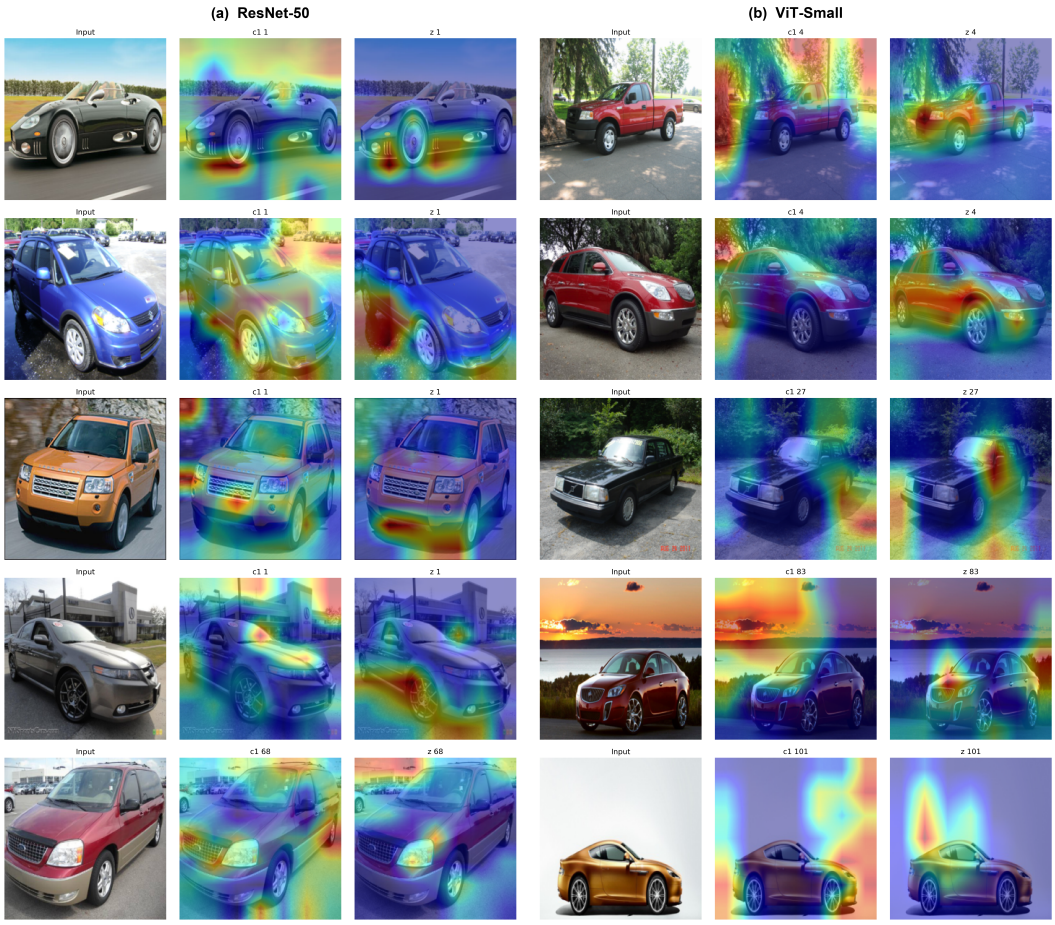

Figure 7: **Spatial activation patterns of factorized representations on Stanford Cars.** Representative activation maps are shown for (a) ResNet-50 and (b) ViT-Small. For each example, the input image is followed by the activation maps of the pre-factorized representation C_{pre} and the expanded latent representation Z.

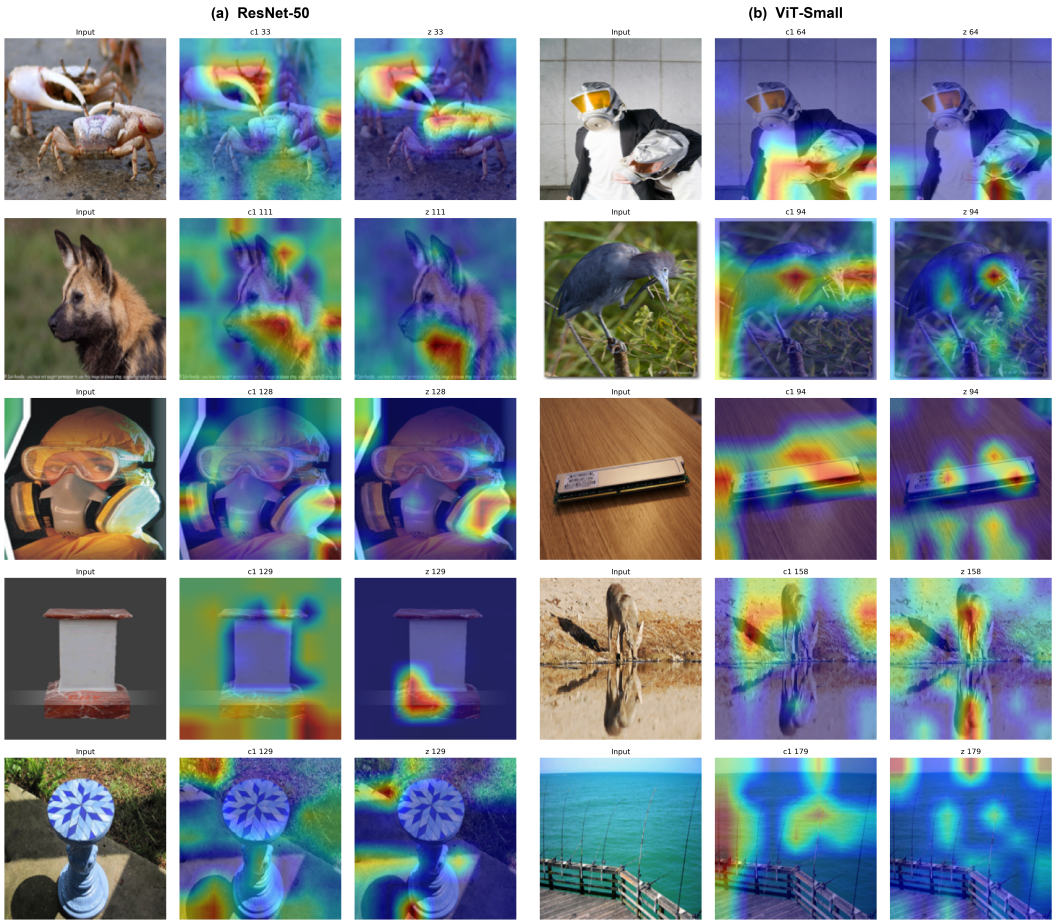

Figure 8: **Spatial activation patterns of factorized representations on ImageNet-100.** Representative activation maps are shown for (a) ResNet-50 and (b) ViT-Small. For each example, the input image is followed by the activation maps of the pre-factorized representation C_{pre} and the expanded latent representation Z.