

Comparing Contrastive and Triplet Loss in Audio–Visual Embedding: Intra-Class Variance and Greediness Analysis

Donghuo Zeng^[0000–0002–6425–6270]

KDDI Research, Inc., Japan
do-zeng@kddi-research.jp

Abstract. Contrastive loss and triplet loss are widely used objectives in deep metric learning, yet their effects on representation quality remain insufficiently understood. We present a theoretical and empirical comparison of these losses, focusing on intra- and inter-class variance and optimization behavior (e.g., greedy updates). Through task-specific experiments with consistent settings on synthetic data and real datasets—MNIST, CIFAR-10—it is shown that triplet loss preserves greater variance within and across classes, supporting finer-grained distinctions in the learned representations. In contrast, contrastive loss tends to compact intra-class embeddings, which may obscure subtle semantic differences. To better understand their optimization dynamics, By examining loss-decay rate, active ratio, and gradient norm, we find that contrastive loss drives many small updates early on, while triplet loss produces fewer but stronger updates that sustain learning on hard examples. Finally, across both classification and retrieval tasks on MNIST, CIFAR-10, CUB-200, and CARS196 datasets, our results consistently show that triplet loss yields superior performance, which suggests using triplet loss for detail retention and hard-sample focus, and contrastive loss for smoother, broad-based embedding refinement.

Keywords: Contrastive loss · Triplet loss · Greedy Optimization · Variance analysis.

1 Introduction

Deep metric learning seeks to embed inputs into a space where geometric proximity reflects semantic similarity, enabling tasks such as image classification [6, 8] and image retrieval [5, 6]. Two of the most popular margin-based objectives are contrastive loss [2] and triplet loss [8]. While both aim to maximize inter-class separation, their different formulations yield distinct gradient patterns—and hence different “greediness” during training—that strongly influence embedding geometry and convergence [1, 7].

Why study gradient behavior? Understanding how each loss allocates gradient effort—whether via many small, diffuse updates or fewer large, targeted

steps—is crucial for tasks requiring fine-grained retrieval or robust classification. A “greedy” loss will continue to enforce margins on easy samples, potentially over-compacting clusters, whereas a more restrained update pattern may better preserve intra-class diversity.

Variance structure . We quantify how each loss is managed 1. *Intra-class variance*: dispersion of samples within a class, and 2. *Inter-class variance*: separation margins between classes. Using overall and per-class variance statistics, plus PCA projections of the original data vs. embeddings from each loss, we show that triplet loss maintains higher within-class spread and clearer between-class gaps on Synthetic data, MNIST, and CIFAR-10, whereas contrastive loss tends to collapse clusters and blur subtle distinctions.

Optimization greediness We define *greediness* via three metrics: *Loss-decay rate*: Epochs required to reduce loss by 90% from the initial value, *Active-sample ratio*: fraction of pairs/triplets with nonzero gradient, *Gradient norm*: average magnitude of parameter updates.

On MNIST and CIFAR-10, contrastive loss reaches 90% loss reduction by epoch 27, with a 65% active-sample ratio and an average gradient norm of approximately 0.12—resulting in many small, diffuse updates and early convergence. In contrast, triplet loss requires until epoch 43 to reach the same reduction, with only 38% active triplets but significantly larger gradient norms (≈ 0.27), enabling more focused updates on hard examples and better preservation of embedding diversity.

Finally, we validate both losses on classification and retrieval tasks across MNIST, CIFAR-10, CUB-200, and CARS196, consistently finding that triplet loss outperforms contrastive loss. By formalizing variance analysis and greediness metrics, our study clarifies how each objective sculpts embedding geometry and training dynamics, and offers guidance on loss selection: use triplet loss for detail retention and hard-sample emphasis, and contrastive loss for smoother, broad-based refinement.

2 Foundations of Contrastive and Triplet Loss

2.1 Contrastive and Triplet Loss

Contrastive and triplet losses form the foundation of deep metric learning, where a neural network $\mathbf{f}(\cdot)$ maps inputs into an embedding space so that semantically similar samples are close and dissimilar ones are separated. We simplify notation by using a single embedding function \mathbf{f} for all inputs.

Contrastive Loss: Originally proposed by Hadsell *et al.* [2], contrastive loss is

$$\mathcal{L}_{\text{con}} = \sum_{(x,y) \in P} \|\mathbf{f}(x) - \mathbf{f}(y)\|^2 + \sum_{(x,y) \in N} [m - \|\mathbf{f}(x) - \mathbf{f}(y)\|]_+^2, \quad (1)$$

where P and N are sets of positive and negative pairs, $m > 0$ is the margin enforcing a minimum separation that controls the trade-off between intra-class

compactness and inter-class separation. A larger m encourages greater inter-class distances but may permit more intra-class variance, while a smaller m enforces tighter clusters. $[z]_+ = \max(0, z)$. $\|\cdot\|$ denotes the L2 norm. By independently pulling every positive pair together and pushing every negative pair apart—even after the margin is met—contrastive loss exhibits a “greedy” optimization behavior, resulting in many small gradient updates across samples [7].

Triplet Loss: Introduced in FaceNet by Schroff *et al.* [8], triplet loss uses triplets (a, p, n) of anchor, positive, and negative:

$$\mathcal{L}_{\text{tri}} = \sum_{(a,p,n)} [\|\mathbf{f}(a) - \mathbf{f}(p)\|^2 - \|\mathbf{f}(a) - \mathbf{f}(n)\|^2 + m]_+. \quad (2)$$

Here, the same margin m ensures the anchor–positive distance is at least m smaller than the anchor–negative distance. Once a triplet satisfies this ranking constraint, it no longer contributes gradients, yielding fewer but larger updates focused on hard examples [3,9]. The key conceptual differences include: (1) *Intra-class dispersion control:* Contrastive loss can collapse within-class samples under a fixed margin; triplet loss permits richer spread [7]. (2) *Inter-class margin enforcement:* Contrastive loss enforces a hard absolute gap; triplet loss ensures only relative separation. (3) *Greedy optimization behavior:* Contrastive loss continues to update all pairs post-margin, resulting in frequent low-magnitude updates. Triplet loss applies gradients only to violating triplets, concentrating updates on harder examples (Section 4.2). (4) *Ranking vs. Absolute Distance:* Triplet loss’s ranking formulation makes it particularly suited to retrieval tasks (e.g., face or product retrieval [3]), where preserving relative similarities is paramount.

2.2 Variance and Optimization Greediness

Maintaining an appropriate structure in the embedding space—where samples of the same class are compact yet not collapsed, and different classes remain well-separated—is essential for both fine-grained retrieval and classification. To quantify this structure, we compute intra-class and inter-class variances as follows:

$$\begin{aligned} \sigma_{\text{intra}}^2 &= \frac{1}{C} \sum_{c=1}^C \frac{1}{N_c} \sum_{i \in I_c} \|z_i - \mu_c\|^2, \quad \text{where} \quad \mu_c = \frac{1}{N_c} \sum_{i \in I_c} z_i, \quad z_i = \mathbf{f}(x_i), \\ \sigma_{\text{inter}}^2 &= \frac{1}{C(C-1)} \sum_{c \neq c'} \|\mu_c - \mu_{c'}\|^2, \end{aligned} \quad (3)$$

where C is the number of classes, I_c is the set of indices for class c , and $N_c = |I_c|$. Here, σ_{intra}^2 measures the average spread of embeddings within each class, while σ_{inter}^2 quantifies the average separation between the centroids of the classes.

Optimization greediness We term *greediness* the propensity of a loss to continue optimizing already-satisfied constraints, potentially leading to excessive intra-class compaction—where embeddings within a class become overly concentrated—or even dimensional collapse, where the embedding space reduces to a lower-dimensional subspace [4]. When measured by (1) *Loss-decay rate* = $\min \{ e \mid \mathcal{L}^{(e)} \leq 0.1 \cdot \mathcal{L}^{(0)} \}$, where $\mathcal{L}^{(e)}$ denotes the average loss at epoch e . This measures how quickly the loss decreases to 10% of its initial value and captures coarse convergence speed. (2) *Active Ratio* = $\frac{|\{(x,y) \in P \cup N : \mathcal{L}_{\text{con/tri}}(x,y) > 0\}|}{|\text{Batch}|}$, the fraction of samples with nonzero loss per batch [6]. This quantifies how many samples continue to drive learning. (3) *gradient norm* computed as the \mathcal{L}_2 norm of the loss gradient: $(\|\nabla \mathcal{L}\|_2)$, measuring the overall magnitude of parameter updates.

Contrastive loss typically shows a high active ratio and low gradient norm, leading to widespread low-magnitude updates across the batch—even when constraints are already satisfied [6, 7]. Triplet loss tends to activate fewer samples but produces stronger gradients concentrated on difficult examples [3, 6]. These distinct behaviors reflect deeper trade-offs between convergence speed and structural preservation, explored further in Section 4.2.

3 Experimental Framework

3.1 Datasets

Synthetic data We generate synthetic data in a fixed 128-dimensional space with 10 clusters, each containing 200 samples, plus Gaussian outliers.

1. **Class centers:** For each class $c \in \{1, \dots, 10\}$, draw

$$g_c \sim \mathcal{N}(0, I_{128}), \quad \mu_c = 5 g_c$$

so that each center has covariance $25I_{128}$.

2. **Covariance and noise:** For each class c , sample a random matrix $A_c \sim \mathcal{N}(0, I_d)$ and set $\Sigma_c = A_c A_c^\top$. Compute the Cholesky factor L_c of $\Sigma_c + 10^{-3}I_d$. Then for each of the 200 points:

$$z_i \sim \mathcal{N}(0, I_{128}), \quad n_i = L_c z_i, \quad x_i = \mu_c + 1.4 n_i.$$

3. **Label overlap (probability $p = 0.1$):** Assign each point label c , but with probability 0.1 reassign it to a random class in $\{1, \dots, 10\}$.
4. **Gaussian outliers (fraction 0.05):** After sampling all cluster points,

$$n_{\text{outliers}} = \lfloor 10 \times 200 \times 0.05 \rfloor = 100$$

points are appended that drawn from $\mathcal{N}(0, 15^2 I_{128})$, each labeled as -1 .

Real dataset We evaluate on two classification and three retrieval datasets, with CIFAR-10, CARS196, and CUB-200 embeddings extracted via Vision Transformer (ViT)¹. (1) *MNIST*: 10 classes, grayscale 28×28 images; 60,000 training and 10,000 test samples ($\sim 6,000$ per class in training, $\sim 1,000$ per class in testing). (2) *CIFAR-10*: 10 classes, RGB 32×32 images; 50,000 training and 10,000 test samples (5,000 per class in training, 1,000 per class in testing). (3) *CARS196*: 196 classes fine-grained categories; 8,144 training and 8,041 test images (~ 42 images per class). (4) *CUB-200*: 200 fine-grained bird categories; 11,788 images (5,994 for training, 5,794 for testing).

3.2 Training Details

Model Architectures: Synthetic data leverages a simple MLP: two fully connected layers (128→64→32) with ReLU activations and L2 normalization on the 32-D output. Real data (MNIST, CIFAR-10) uses a CNN-like model: two Conv-ReLU-MaxPool blocks followed by linear layers (flatten→128→64) and L2-normalized embeddings. Retrieval tasks with CIFAR-10, CARS196, and CUB-200 use a frozen ViT-B/32 backbone with a 512-D embedding head and L2 normalization. *Optimization Setup*: All models are trained with Adam (learning rate=1e-3, weight decay=1e-5), batch size=64, for 50 epochs, and margin $m = 1.0$ for both losses. Euclidean distance is used for all pairwise comparisons. *Loss Sampling Strategies*: For both contrastive and triplet loss, we sample 50% positive and 50% negative pairs, excluding outliers (label -1) from positives. *Diagnostics and Visualization*: We track loss curves, active ratio (fraction of non-zero losses per batch), and gradient norms to analyze optimization dynamics. Embedding visualization is performed via PCA. Code availability ²

4 Results and Analysis

4.1 Variances analysis

Table 1: statistics of intra- and inter-class variance on Synthetic data and MNIST

Loss	Synthetic data				MNIST			
	Intra-class		Inter-class		Intra-class		Inter-class	
	μ	σ^2	μ	σ^2	μ	σ^2	μ	σ^2
Contrastive	0.031	6.4e-05	1.2149	0.0710	0.0030	0.0001	1.0347	0.0064
Triplet	0.074	0.0001	1.4399	0.0342	0.0059	0.0001	1.4840	0.0047

* paired t-test, $p < 0.001$

We can observe in Table 1 that on Synthetic data, the triplet loss preserves approximately 2.4

¹ https://huggingface.co/docs/transformers/v4.13.0/en/model_doc/vit

² <https://anonymous.4open.science/r/t-2025>

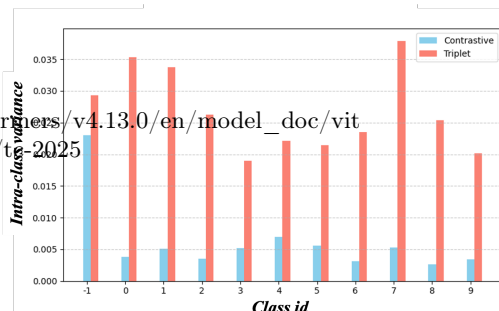


Fig. 1: Intra-class variance for each class

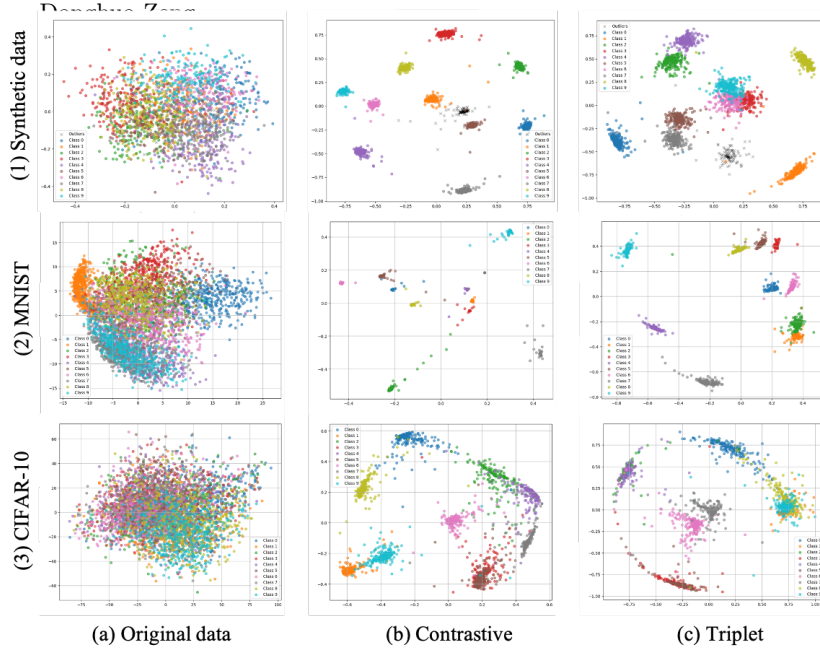


Fig. 2: PCA embedding visualizations of three distinct datasets

times more average intra-class variance than the contrastive loss ($0.074/0.031 \approx 2.4$). This indicates that embeddings trained with triplet loss exhibit greater within-class diversity. A paired t -test on per-class intra-class variances confirms this difference is statistically significant ($p < 0.001$). The average inter-class distance is slightly higher for triplet loss (1.4399 vs. 1.2149), with lower variability in inter-class distances ($\sigma^2=0.0342$ vs. 0.0710), suggesting more consistent separation between classes. As illustrated in Fig. 1, the intra-class variances under triplet loss are uniformly higher across all classes compared to contrastive loss. A similar trend is observed for the MNIST dataset. These traits—greater within-class diversity and clearer class separation—make triplet loss well-suited for downstream tasks needing robust embedding generalization [7, 8]. Fig. 2 shows PCA projections of the learned embeddings. In general, *contrastive loss* yields tightly clustered classes, while *triplet loss* allows for more natural, dispersed clusters that better reflect the underlying data structure.

4.2 Greedy Optimization Behavior of Loss Functions

Different metric-learning losses induce distinct update patterns—what we term “greediness”—measured by three metrics: Loss-decay rate from loss curve, active ratio, and gradient

Metric	Contrastive	Triplet
Active ratio	65%	38%
Gradient norm	0.12	0.27
Loss-decay rate	27	43

norm. Table 2, we briefly report Loss-decay rate (the epoch by which 90% of the initial loss is eliminated) and focus on comparing results: Contrastive loss reaches 90% loss reduction by epoch 27, engages a large share of samples (65% active ratio), and shows modest gradients (norm ≈ 0.12). This combination yields many small, diffuse updates and early plateauing of training. Triplet loss, however, achieves 90% Loss-decay only by epoch 43, with fewer active triplets (38%) but stronger updates (norm ≈ 0.27). These sharper, focused updates prolong learning and help preserve fine-grained distinctions in the embedding space. Fig. 3 illustrates that *contrastive loss* causes a sharper and earlier collapse in intra-class distances, converging faster to a low-variance state than *triplet loss*. This delayed decay in triplet loss, however, sustains learning on harder examples, which helps explain its superior early retrieval performance (see Section 4.3).

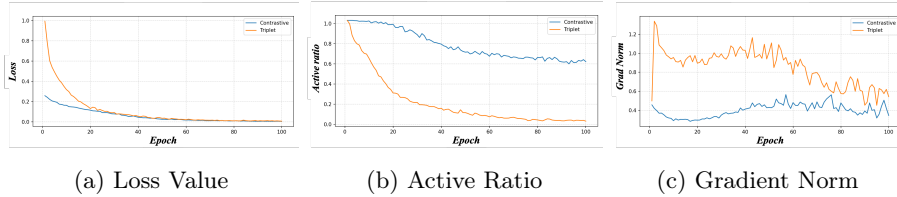


Fig. 3: The greediness metrics over training epochs.

4.3 Application to Classification and Retrieval

To demonstrate the practical impact of our variance-structure and greedy-optimization analyses, we evaluate both classification and retrieval tasks. Classification tests global separation—requiring clear inter-class margins and compact clusters—while retrieval measures fine-grained neighbor ranking—benefiting from preserved intra-class variance. Together, they link embedding geometry and optimization dynamics to real-world performance.

Table 3 shows that triplet loss achieves higher classification accuracy (MNIST: 0.9933 vs. 0.9869; CIFAR-10: 0.9371 vs. 0.8998), while Table 4 demonstrates its superior retrieval r@1 performance across CIFAR-10 (0.9192 vs. 0.8433), CARS196 (0.2982 vs. 0.2542), and CUB-200 (0.3421 vs. 0.3154), with smaller gaps at r@5 and r@10. These results confirm that triplet’s broader intra-class

Table 3: Classification accuracy on MNIST and CIFAR-10

Loss	MNIST	CIFAR-10
Contrastive	0.9869	0.8998
Triplet	0.9933	0.9371

variance preserves fine distinctions—boosting $r@1$ —while still enforcing inter-class margins for high accuracy. In contrast, contrastive’s many small, rapid updates over-compact clusters and hurt both retrieval and separability. Balancing intra-class spread with update intensity is therefore key to optimal classification and retrieval.

Table 4: Retrieval recall@k ($k=1,5,10$) on three datasets.

Loss	CIFAR-10			CARS196			CUB-200		
	r@1	r@5	r@10	r@1	r@5	r@10	r@1	r@5	r@10
Contrastive	0.8433	0.9701	0.9899	0.2542	0.5249	0.6596	0.3154	0.5489	0.6897
Triplet	0.9192	0.9694	0.9793	0.2982	0.5540	0.6667	0.3421	0.5876	0.7234

5 Conclusion

We presented a theoretical and empirical comparison of contrastive and triplet loss in deep metric learning, focusing on both embedding structure and optimization behavior. Our variance analysis shows that triplet loss preserves greater intra- and inter-class variance, supporting finer-grained distinctions, while contrastive loss tends to compact intra-class representations. Through metrics such as loss-decay rate, active ratio, and gradient norm, we also find that contrastive loss applies frequent small updates, whereas triplet loss produces fewer but stronger updates, concentrating learning on hard examples. Across classification and retrieval tasks, triplet loss consistently outperforms contrastive loss. These findings suggest that triplet loss is better suited for detail-preserving, discriminative embeddings, while contrastive loss favors smoother, broad-based representation learning. Future work includes exploring hybrid losses and adaptive margins that better balance precision and generalization.

References

1. Benyamin Ghojogh, Milad Sikaroudi, Sobhan Shafiei, Hamid R Tizhoosh, Fakhri Karray, and Mark Crowley. Fisher discriminant triplet and contrastive losses for training siamese networks. In *2020 international joint conference on neural networks (IJCNN)*, pages 1–7. IEEE, 2020.
2. 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)*, pages 1735–1742. IEEE, 2006.
3. Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 390–398, 2017.
4. Li Jing, Pascal Vincent, Yann LeCun, and Yuandong Tian. Understanding dimensional collapse in contrastive self-supervised learning. *ArXiv*, abs/2110.09348, 2021.
5. Wu-Jun Li, Sheng Wang, and Wang-Cheng Kang. Feature learning based deep supervised hashing with pairwise labels. *arXiv preprint arXiv:1511.03855*, 2015.

6. R. Manmatha, Chaoxia Wu, Alex Smola, and Philipp Krähenbühl. Sampling matters in deep embedding learning. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2859–2867, 2017.
7. Kevin Musgrave, Serge Belongie, and Ser-Nam Lim. A metric learning reality check. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXV 16*, pages 681–699. Springer, 2020.
8. 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*, pages 815–823, 2015.
9. Qian Wang, Jian Wang, Wen Liu, Yichao Gao, and Yongdong Xu. Multi-similarity loss with general pair weighting for deep metric learning. In *CVPR*, pages 5022–5030, 2019.