Batch Decorrelation for Active Metric Learning

Priyadarshini Kumari , **Ritesh Goru** , **Siddhartha Chaudhuri** [2] and **Subhasis Chaudhuri**
IIT Bombay
[2]Adobe Research
priyadarshini.k@iitb.ac.in, riteshgoru@iitb.ac.in, sidch@cse.iitb.ac.in, sc@ee.iitb.ac.in

Abstract

We present an active learning strategy for training parametric models of distance metrics, given triplet-based similarity assessments: object x_i is more similar to object x_j than to x_k. In contrast to prior work on class-based learning, where the fundamental goal is classi cation and any implicit or explicit metric is binary, we focus on *perceptual* metrics that express the *degree* of (dis)similarity between objects. We nd that standard active learning approaches degrade when annotations are requested for *batches* of triplets at a time: our studies suggest that correlation among triplets is responsible. In this work, we propose a novel method to *decorrelate* batches of triplets, that jointly balances informativeness and diversity while decoupling the choice of heuristic for each criterion. Experiments indicate our method is general, adaptable, and outperforms the state-of-the-art.

1 Introduction

De ning a good distance metric over a space of objects is important for many machine learning tasks, such as classi cation, clustering and object segmentation. Often, such a metric represents binary relationships among objects: for instance, it may consider two objects similar if they belong to the same class and dissimilar otherwise. However, domains like cognitive science and human-computer interaction (HCI) often require capturing the more granular and complex notion of *perceptual* distance: a *continuous* measure of the perceived dissimilarity between two objects. This allows *relative* assessments such as: object x is more similar to y than to z, even when all three objects are from the same class. Indeed, it is even possible for two objects from different classes (say, a dolphin and a sh) to be perceptually more similar than two objects from the same class (the dolphin and any land mammal). Learning a robust predictor of the perceptual distance between any two objects from some input domain, given training data over a subset of that domain, is therefore an important and challenging problem. In recent research, this predictor is commonly a deep neural network.

While regressing a metric from numerical measurements of the distances between pairs of objects is straightforward, it is usually very dif cult for human annotators to accurately and consistently estimate such distances. Instead, it is easier to answer *ranking* questions: is object x more similar to object y or z? Such "triplet" comparisons are the backbone of metric learning, in both binary (categorization) and continuous (perceptual) contexts. The number of possible triplet comparisons grows cubically – $O(n^3)$ for n objects – and the cost of annotating them all is prohibitive. However, for reasonably well-behaved metrics the triplets contain much redundancy, since the underlying measure is assumed to at least approximately follow the triangle inequality. This suggests that good models can be trained on much smaller triplet sets.

The main challenge, then, is selecting *which* small subset of triplets should be annotated. This task is addressed by *active learning*, a standard technique to minimize effort by iteratively requesting annotations for the most informative samples (here, triplets) from a large pool of unlabeled data. In each iteration, a batch of samples is labeled and added to the training set, and the model is updated. The process repeats until a desired accuracy is achieved. Active learning has been effective for many applications [Adomavicius and Tuzhilin, 2005; Calinon *et al.*, 2007; Settles, 2012; Yang *et al.*, 2017; Zhang *et al.*, 2017], but only a few studies [Tamuz *et al.*, 2011; Heim *et al.*, 2015; Lohaus *et al.*, 2019] employ it for triplet-based perceptual metric learning. These methods are non-parametric (the majority use multidimensional scaling (MDS)) and cannot predict the distance between novel objects at test time. But they also have another signi cant shortcoming. To reduce the incremental training cost and enable ef cient use of human resources, annotations are typically requested in *batches* of, say, a few hundred triplets at a time, after which the model is updated taking the new training batch into account. Prior active learning strategies often perform worse than baseline random sampling when operating in batch mode (Figure 1). We nd that *correlation* among samples in a batch is responsible: individually informative samples are likely to form strongly correlated clusters, lowering the expected bene t. There have been very recent efforts to alleviate this for classi cation [Kirsch *et al.*, 2019; Ash *et al.*, 2020], but none for metric learning.

In this paper, we propose the **rst batch decorrelation strategy developed speci cally for triplet-based active metric learning**. Our method jointly balances informativeness and diversity of a batch of triplets while decoupling the

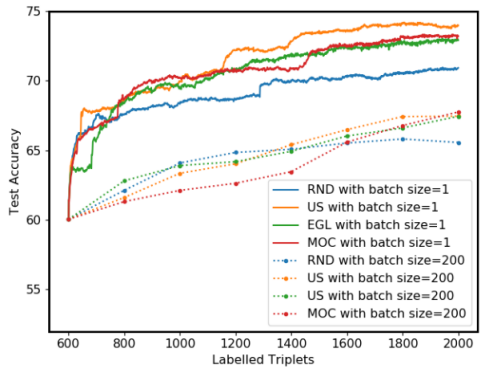

Figure 1: Negative impact of correlated triplets on active learning. Deep metric learning with 3 standard AL heuristics (uncertainty (US), expected gradient length (EGL), model output change (MOC)) all outperform random sampling (RND) when triplets are added one at a time from the Yummly food dataset. With 200-triplet batches, they are comparable to or worse than random. (Note: batch size also affects performance with random sampling because we mimic real-world computational budgets by allotting a fixed (high) number of training epochs after each batch. When training with single triplets, the network gets to train more often, hence longer overall.)

choice of heuristic for each criterion (the approach of Ash et al. [2020], developed for classification, represents both using the same unnormalized gradients). We select an overcomplete set of informative triplets, and find well-separated yet informative representatives from them using farthest-point sampling, a technique borrowed from visual computing [Eldar *et al.*, 1997]. We develop several alternative measures of triplet separation, and compare their effectiveness over a range of different input modalities, tasks, and hyperparameter settings. Our framework is robust and outperforms the current state-of-the-art in all evaluated scenarios. We demonstrate the method for neural network-based learners, but it can be easily adapted to any parametric learning scenario.

2 Related Work

We overview related work on active learning in three categories. For a general survey of deep metric learning, we recommend recent papers such as [Priyadarshini K *et al.*, 2019].

2.1 Active learning for class-based learning.

There is significant work on active learning for *classification* [Settles, 2012], exploring heuristics like uncertainty sampling, query-by-committee, expected model change, expected error/variance minimization, and information gain [Settles, 2012; Krishnamurthy *et al.*, 2017; Bachman *et al.*, 2017; Konyushkova *et al.*, 2017]. Among these, uncertainty sampling is an effective and widely used strategy. Two other interesting works [Hoi *et al.*, 2006; Wei *et al.*, 2015] select a batch of diverse and informative samples by posing active learning as a submodular subset selection problem. In contrast to our approach, all these methods focus on object classification.

2.2 Active learning for deep learning.

Several recent works apply active learning heuristics to neural networks. Sener and Savarese [2018] define active learning as a core-set problem to select diverse samples. Sinha et al. [2019] adversarially select outliers to the distribution of labeled samples. Gal et al. [2017] propose Bayesian active learning (BALD) for image classification, maximizing mutual information between model parameters and model prediction. It works well when selecting samples one at a time but fails for larger batches. To address this, Kirsch et al. [2019] introduce a modified BALD for batches. They optimize informativeness and diversity by maximizing mutual information between selected points and model parameters.

Ash et al. [2020] present another method (BADGE) for diversification where samples with diverse gradient vectors are picked for annotation. However, they use the gradient w.r.t. the most probable label instead of the expected gradient, and select a batch using k-means++ [Arthur and Vassilvitskii, 2006] over all unlabeled samples in the gradient space. This has shortcomings: (1) the gradient is used for *both* informativeness and diversity, whereas our method decouples these choices to leverage the best individual heuristics; (2) k-means++ will pick sample centroids around the origin (small gradient vectors) if there are dense clusters there, hence the bias towards more informative samples is reduced; and (3) k-means++ requires an explicit triplet embedding and cannot work with just a distance measure, unlike our farthest-point sampling. Experiments indicate that our approach consistently outperforms BADGE adapted to metric learning.

All these methods share our goal of efficient learning from fewer annotations in a batch setting, but they focus on classification problems. In contrast, we aim to learn a (perceptual, parametric) distance metric from triplet constraints. This requires *triplet-specific* informativeness and diversity measures, that can also be used to compare unseen data samples.

2.3 Active triplet-based metric learning.

There are only a few papers on active metric learning from triplet-based relative constraints. Tamuz et al. [2011] sample triplets which reduce the uncertainty in object location in the embedding space. Heim et al. [2015] propose a similar approach and show that perceptual similarity can be modeled with even fewer triplets if auxiliary features are used. Again, both methods work well in single-triplet mode but fail with neural networks in a batch setting. Moreover, both methods are non-parametric and designed to achieve data efficiency for MDS: they cannot predict the distance between novel objects. Recently, Lohaus et al. [2019] proposed a triplet uncertainty estimate by modeling triplet ordering as a Gaussian process, a natural choice for active learning as it measures prediction confidence. However, similar to MDS-based approaches, it is non-parametric and the learning complexity is $O(n^3)$. Finally, none of these methods optimize diversity in addition to informativeness during batch selection.

3 Method

In this section, we describe our framework for batch-mode active metric learning. We first briefly define the problem of perceptual metric learning from triplet comparisons, using a deep network as the regressor. Next, we discuss an active data sampling approach for learning the metric efficiently. Finally,

as our main technical contribution, we introduce a method for decorrelating batches of triplets, to address the known limitations of active learning with large incremental batches.

3.1 Triplet-Based Deep Metric Learning

We are given a set of objects $X = \{x_i\}_1^n \in \ ^d$, described by their d-dimensional features. We also have relative comparisons of their pairwise distances as a set of triplets $C = \{(x_i, x_j, x_k)\}$. Here, each triplet indicates that the anchor object x_i is more similar to x_j than to x_k. Our goal is to learn a distance metric $d : \ ^d \times \ ^d \rightarrow$ which preserves the triplet orderings – $d(x_i, x_j) < d(x_i, x_k)$. As is standard, we assume the comparisons can be noisy and inconsistent, hence the solution is approximate and we only try to maximize the number of preserved orderings. Although this framework applies to any kind of metric learning, including binary metrics for classi cation, our focus is to learn perceptual metrics reflecting continuously varying degrees of (dis)similarity.

Typically, perceptual (dis)similarity between objects is modeled by projecting the data using a learned embedding kernel $\phi : \ ^d \rightarrow \ ^{\hat{d}}$ such that the distance between two objects $d\ (x, y) = \|\phi(x) \quad \phi(y)\|$ reflects the perceptual distance. Following prior work [Lu *et al.*, 2017; Zhang *et al.*, 2018; Priyadarshini K *et al.*, 2019], we use a deep network to represent the embedding kernel ϕ. For each triplet, we pass its three objects through three copies of the network with shared weights. We train the network using an exponential triplet loss that tries to maximize the squared distance margin $d^2(x_i, x_k) \quad d^2(x_i, x_j)$ for each triplet (x_i, x_j, x_k):

$$\mathcal{L}(x_i, x_j, x_k, \phi) = \sum_C e^{\ d\ \ x_i,x_k)\ \ d\ \ x_i,x_j))} \qquad (1)$$

The model is trained using standard back-propagation. In all our experiments, the network comprises a series of fully connected layers with ReLU nonlinearities. The hyperparameters are manually tuned for each dataset, as detailed in Section 4.

3.2 Deep Active Metric Learning

The number of possible triplet comparisons of n objects grows cubically – $O(n^3)$ – and gathering annotations for all of them is prohibitively expensive. However, for a reasonably smooth metric that at least approximately respects triangle inequality, a much smaller subset of "informative" triplets is usually suf cient to capture its salient features. Active learning is a standard technique to discover such a subset. It is an iterative method that greedily repeats the following two steps. First, given an incremental labeling budget b, and the set of previously labeled triplets L, it selects a batch T of b new triplets (from some potentially large set of unlabeled triplets U) and requests annotations (i.e. ground-truth orderings) for them. Second, the batch is added to the set of labeled triplets and the metric model is retrained. The process repeats till some desired accuracy is reached. The main technical challenge is to choose the most informative new triplets in each batch, such that the model reaches its asymptotic accuracy in as few iterations as possible.

The informativeness of a triplet can be measured by various heuristics, such as model uncertainty, prediction variance, or mutual information. In this paper, we adopt model uncertainty as a representative measure which has proved to be very effective in practice [Settles, 2012], though we also show experiments with other measures. We consider a triplet "uncertain", and hence informative, if the model is highly uncertain in predicting its ordering: $d\ (x_i, x_k) \approx d\ (x_i, x_j)$. A probability distribution over the triplet response is a natural way to measure uncertainty [Tamuz *et al.*, 2011]:

$$p_{ijk} = p(t|\phi) = \frac{\mu + d^2(x_i, x_k)}{2\mu + d^2(x_i, x_k) + d^2(x_i, x_j)} \qquad (2)$$

Here, p_{ijk} denotes the probability of object i being closer to j than to k in the embedding space, $\mu >= 0$ is a hyperparameter, and $p_{ijk} + p_{ikj} = 1$. The model's uncertainty about the unlabeled triplet can then be expressed as the expected Shannon entropy over all possible orderings: $f\ (t) = \quad p_{ijk} \log p_{ijk} \quad p_{ikj} \log p_{ikj}$. The score of a selected batch T, naïvely ignoring correlations between triplets, can be de ned by summing up the individual uncertainties:

$$f\ (T) = \sum_{t \in T} f\ (t) \qquad (3)$$

However, simply using this score to select each incremental batch, as $T^* = \arg\max_{T \subset U, |T|=b} f\ (T)$, is fast (it reduces to picking the b top-scoring triplets) but suboptimal. The top b triplets are individually informative, but may not be so jointly. For instance, they may all be tightly clustered in the single most uncertain region in the embedding space, having high redundancy and giving the model no wider information. To address this shortcoming, we choose a *diverse yet informative* batch by applying decorrelation measures, discussed next.

3.3 Triplet Batch Decorrelation

Selecting a single triplet in each iteration ($b = 1$) is highly inef cient since it requires infeasibly frequent retraining. To reduce this overhead, the standard approach is to pick larger batches ($b \gg 1$), but developing an optimal batch selection policy is dif cult because of correlation between the most informative triplets as discussed above (in other words, informativeness is not compositional). In the general case, the problem is NP-hard. Some prior works develop effective greedy approximations by de ning batch utility as a monotone submodular function [Golovin and Krause, 2011], but making the utility function submodular is not always natural.

Instead, we propose a two-step approach, each step involving greedy sampling. The rst step emphasizes **informativeness**: we pick an *overcomplete* set S of the most informative individual triplets, of much larger size than b, from the unlabeled set U. The second step emphasizes **diversity**: we *subsample* S to select a nal batch T^* that effectively covers S, given a measure ρ of the distance (non-correlation) between triplets conditioned on the current model ϕ. Picking the batch that minimizes the distance of each triplet in S to its nearest triplet in T^* is also NP-hard. However, it has been shown that farthest-point-sampling (FPS), a greedy approach originally developed for image processing [Eldar *et al.*, 1997], gives a near-optimal solution. In our adaptation of FPS, we begin with the most separated pair of triplets, and then recursively select the unselected triplet that is farthest from the selected set. Formally, the recursion is de-

Algorithm 1 Batch-Mode Active Learning

Input: Neural network architecture ϕ, object features $X \subset$ ^{d}, labeled triplets L, unlabeled triplets U, initial pool size l, batch size b, oversampling size $k > b$

Initialize: Train initial model ϕ_0 on l randomly selected labeled triplets L_0.

```
for m=1,2,...,M do
    S_m = arg max_{S⊂U,|S|=k} f (S) using Eq. 3
    T*_m = arg max_{{t_i,t_j}⊂S_m} ρ (t_i, t_j)
    for n=3,...,b do
        T*_m ← T*_m ∪ {arg max_{t∈S_m\T_m} min_{t'∈T_m} ρ (t, t')}
    end for
    L ← L ∪ T*_m
    Train starting from model φ_{m 1}, on labeled triplets L
    with loss in Eq. 1, to obtain updated model φ_m.
end for
return Final model φ_M
```

ned as $T \leftarrow T \cup \arg\max_{t \in S \setminus T} \min_{t' \in T} \rho\ (t, t')$. Combining the two steps, as formalized in Algorithm 1, we obtain a diverse yet informative batch.

The remaining challenge is to de ne ρ . To incorporate triplet informativeness in this measure as well, in case S is not uniformly informative, we de ne $\rho\ (t, t') := f\ (t) \times f\ (t') \times$ (t, t'), discouraging uninformative triplets from being selected even if they are far apart. Now, we propose several intuitive new ways of de ning the purely "geometric" term (t, t'), since sample-wise (dis)similarity measures like L_p or cosine metrics do not directly apply to unordered triplets.

1. **Gradient distance:** Intuitively, two triplets can be considered dissimilar if they change the model differently. Since the neural network is optimized using gradient descent, we represent each triplet by the expected gradient (since the ordering is unknown) of its loss w.r.t. the last layer ϕ_{out} of the network:

$$g(t) = p_{ijk} \frac{\partial \mathcal{L}(x_i, x_j, x_k, \phi)}{\partial \phi_{\text{out}}} + p_{ikj} \frac{\partial \mathcal{L}(x_i, x_k, x_j, \phi)}{\partial \phi_{\text{out}}} \tag{4}$$

   The distance is then $(t, t') = 1 \quad \frac{g\ t)}{|g\ t)|}, \frac{g\ t')}{|g\ t')|} \Big\rangle$. Note the contrast with the related gradient-based decorrelation of Ash et al. [2020], as detailed in Section 2.

2. **Euclidean distance:** We concatenate the object embeddings of a triplet into a 3-dimensional vector: $\phi(x_i, x_j, x_k) = \phi(x_i) \oplus \phi(x_j) \oplus \phi(x_k)$. The expected L2 distance over possible orderings gives:

$$(t, t') = \sum_{y \in \{ijk, ikj\}} \frac{1}{2} \|\phi(t^y) \quad \phi(t')\|, \tag{5}$$

   where t^y denotes one of the two orderings of the triplet.

3. **Centroidal distance:** Instead of representing a triplet by its concatenated object embeddings, we represent it by the (ordering-independent) centroid of the embeddings of its three objects: $c(t) = \frac{1}{3}(\phi(x_i) + \phi(x_j) + \phi(x_k))$. Then, the distance between triplets is simply the L2 distance between their centroids: $(t, t') = \|c(t) \quad c(t')\|$. Note that unlike the preceding methods, this is agnostic to triplet annotation and robust to model bias.

4. **Oriented distance:** The centroid of a triplet is a single point, throwing away information about the *orientation* of the triplet in the embedding space. We incorporate such information into the oriented distance, de ned as the sum of the distances between the anchors of two triplets, and between their "orientation vectors" $\hat{r}(x_i, x_j, x_k) = \text{unit}\,(\phi(x_k) + \phi(x_j) \quad 2\phi(x_i))$:

$$(t, t') = d\ (x_i, x_i') + (1 \quad \langle \hat{r}(t), \hat{r}(t') \rangle), \tag{6}$$

   where x_i, x_i' are the anchor objects of t, t' respectively.

4 Experiments

To evaluate the performance of our batch decorrelation method, we perform several experiments, addressing three questions: (Q1) How effective is our method for different batch (and initial pool) sizes, varying levels of noise rates in the training set, and different triplet diversity measures, across different datasets and modalities? (Q2) How important is decoupling the choices of informativeness and diversity heuristics, and how effective is decorrelation with different triplet informativeness measures? (Q3) How effective are individual components of our framework? We start by introducing our datasets and evaluation setup before discussing each question in subsequent sections.

4.1 Datasets and Evaluation Setup

We evaluate our method on two challenging real-world datasets, as well as on synthetic data.

Synthetic Data. We generate 100 samples X from a 10-D standard normal distribution. From X, we randomly sample 20K training and 20K test triplets, each with a ground truth ordering from a prede ned (random) Mahalanobis metric. To check if our framework is robust to noisy data, we randomly flip the order of 20 of the training triplets.

Yummly Food Data. This dataset has 72148 triplets de ned over 73 images of food items [Wilber *et al.*, 2014]. The triplet constraints are based on taste similarity. Each item is represented by an L1-normalized 6-D feature vector indicating the proportions of different taste properties: salty, savory, sour, bitter, sweet, and spicy. To generate a train/test split we randomly subsample 20K training and 20K test triplets.

Haptic Texture Data. This dataset has 108 surface materials (metals, paper, fabrics, etc), each represented by 10 texture signals [Strese *et al.*, 2017] produced by tracing a haptic stylus over the physical surface. The triplets are generated from the ground truth perceptual dissimilarity metric gathered in user experiments [Priyadarshini K *et al.*, 2019]. As before, we randomly subsample 20K training and 20K test triplets. Each texture signal is represented by a 32-D CQFB spectral feature vector [Priyadarshini K *et al.*, 2019].

Evaluation setup. The performance of the learned perceptual metric is evaluated after each training round, consisting of selecting one new batch of triplets followed by updating the model. We measure model perfomance by the *triplet generalization accuracy* (TGA): the fraction of test triplets

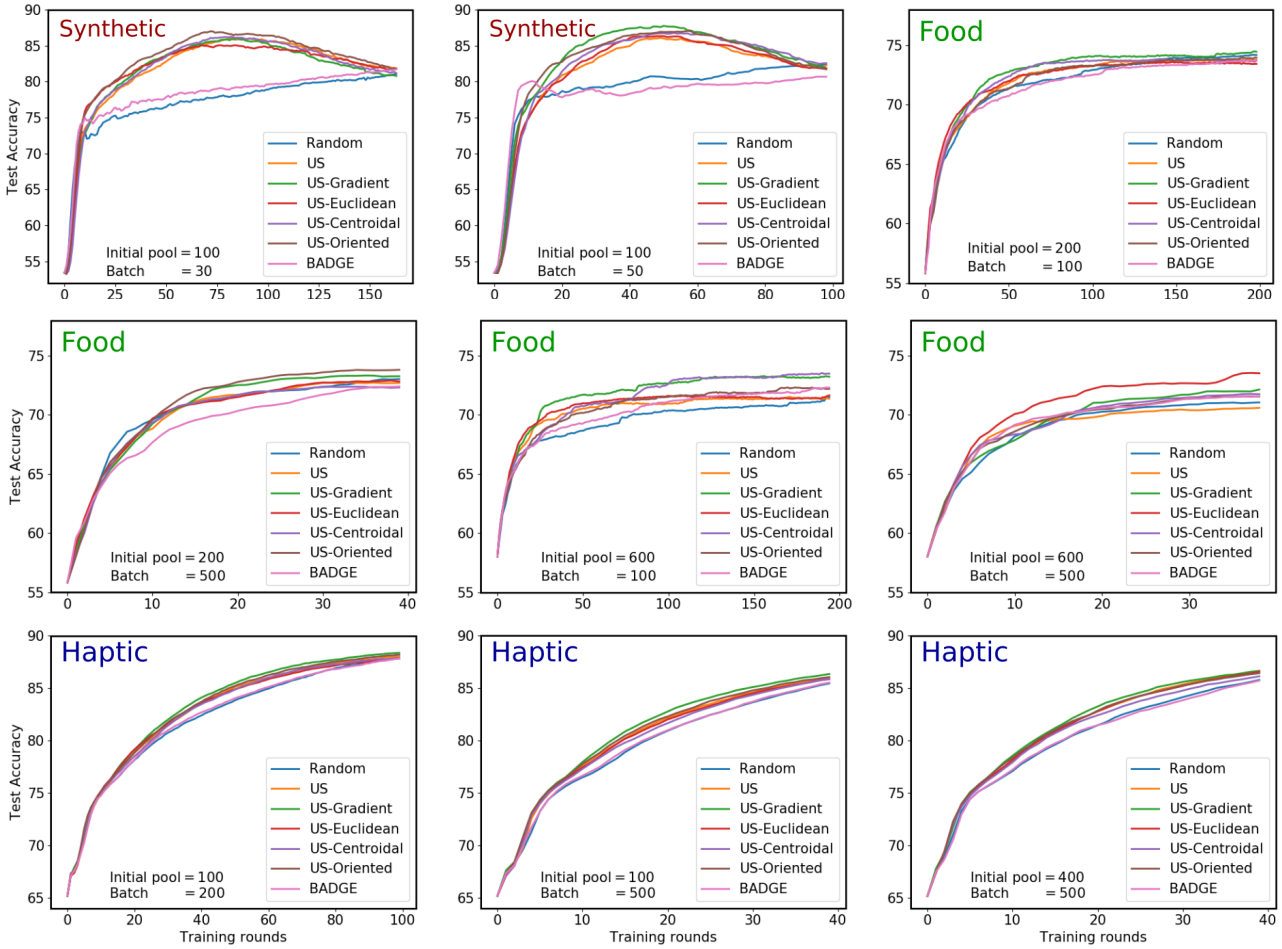

Figure 2: Performance of active learning for different batch and initial pool sizes, on real and synthetic datasets. Different variants of our batch decorrelation method (US- *Dist*⟩) are evaluated and compared to three baselines: Random sampling (Random), basic uncertainty sampling without decorrelation (US), and adapted BADGE [Ash et al., 2020]. Our method consistently outperforms the baselines across all three datasets. The standard deviation ranges across datasets are: Synthetic: min 0 7 (US-Gradient), max 2 55 (BADGE); Food: min 1 5 (US-Gradient), max 3 4 (US-Gradient); Haptic: min 0 266 (US-Euclidean), max 0 673 (Random).

whose ordering is consistent with the learned metric. On each dataset, we repeat the experiment over 5 random train/test splits and report the mean accuracy after each round.

Variants and Baselines. We report the performance of each of the 4 triplet distance measures in Section 3.3 as **US-**⟨ ***ist***⟩, denoting the variant of our method where we rst select the most informative triplets via uncertainty sampling, and then pick a diverse subset using triplet distance measure ⟨*Dist*⟩ (e.g. gradient distance). We compare to three baselines: (1) **Random**: Triplets are uniformly sampled at random; (2) **US**: Uncertainty-based approach *without* decorrelation, which selects the top b triplets with highest entropy as per Eq. 3 [Tamuz *et al.*, 2011; Heim *et al.*, 2015]; (3) **BADGE**: Gradient-based approach that selects a set of triplets with diverse loss gradients using k-means (the gradient for a triplet is computed for the most probable label, as opposed to our expected gradient) [Ash *et al.*, 2020].

Hyperparameters. We manually tuned network hyperparameters in each dataset: *Synthetic:* 3 FC layers with 10, 20, 10 neurons resp.; *Food:* 3 FC layers with 6, 12, 12 neurons resp.; *Haptic:* 4 FC layers with 32, 32, 64, 32 neurons resp. All layers have ReLU activation. We train using Adam [Kingma and Ba, 2015] with learning rate 10 ^{4}. Each training round is budgeted 200 epochs for synthetic data and 1000 for food and haptic data. For all our experiments, the size of the overcomplete set (S) of informative triplets is twice the budget (b). We found that our method did not bene- t from a larger S. This suggests it can be well suited to large datasets, though more testing is needed to con rm this.

4.2 Batch and Initial Pool Sizes, Noise Impact, and Triplet Distance Measures (Q1)

In general, as shown in Figure 2, our method achieves consistent improvements over the baselines. For all settings and datasets, multiple variants of our method outperform all baselines, often by large margins, though the same triplet distance measure is not always the most accurate. Relevant to practical settings, no variant of our method ever fails signi cantly: performance is maintained across the board.

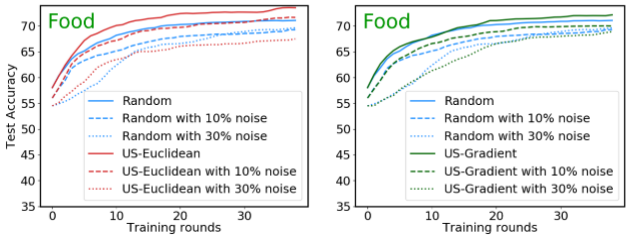

Figure 3: Performance of our method in the presence of additional labeling noise, for two different triplet decorrelation measures. Note that since this is a real-world dataset, some noise is likely present in the initial triplets as well.

Effect of batch size. Intuitively, the bene ts of decorrelation should be more signi cant for larger batches. Our experimental results con rm this hypothesis: as batch size increases, the performance gain of our method over random and uncertainty sampling is larger (Figure 2). Moreover, with increasing batch size, the performance of US also decreases, and sometimes becomes even worse than random. This further con rms that active learning based on just informativeness, without decorrelation, is ineffective.

Effect of initial pool. Figure 2 illustrates that an initial model trained on a very small pool of labeled data may not represent the underlying distribution. As a result, such a biased model may fail to sample informative triplets. As expected, our method outperforms baselines by larger margins when the initial pool is larger and more informative.

Effect of noisy labels. To evaluate the noise robustness of our method, we induce additional noise in the data by randomly flipping the order of some triplets. In Figure 3, we observe that the performance of all methods suffers from the increasing noise rate. However, our method outperforms random sampling, even in the presence of a moderate level of extra noise, and succumbs only to very high noise rates.

Effect of triplet distance measures. Among different decorrelation measures, we found triplet gradient distance usually outperforms or matches variants and baselines. However, for very large batches, all variants of batch-decorrelated active learning perform better than standard baselines.

4.3 Other Informativeness Measures (Q2)

Since we decouple the choices for informativeness and diversity heuristics, we can easily examine other informativeness measures. Figure 4 shows the effect of replacing uncertainty (US) with (a) model output change (MOC) [Freytag *et al.*, 2014], and (b) expected gradient length (EGL) [Huang *et al.*, 2016] on the Food dataset. We see that decorrelation improves both MOC and EGL, especially for larger batches, but does not always outperform random sampling, especially for EGL [Baldridge and Palmer, 2009]. This highlights the importance of an appropriate informativeness measure, and of decoupling the choice of this measure from triplet diversity.

4.4 Ablation Studies (Q3)

We compared triplet batch selection with the following variants: (1) just uncertainty-based informativeness (**US**),

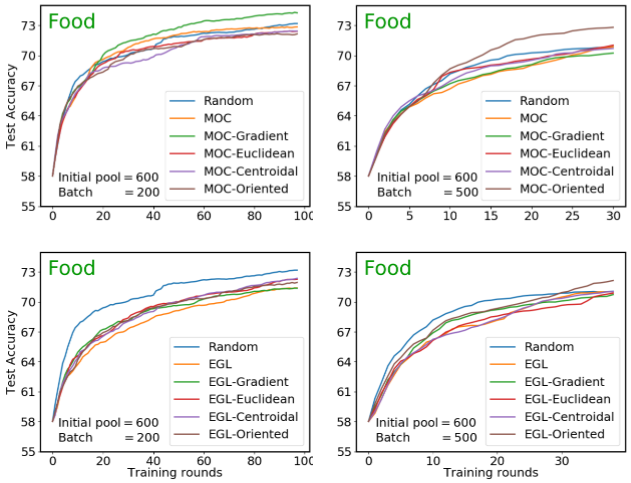

Figure 4: Performance of batch decorrelated AL with different triplet informativeness measures. EGL: Expected Gradient Length, MOC: Model Output Change.

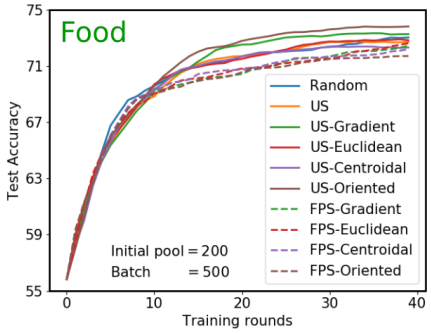

Figure 5: Ablation studies showing the performance of individual components of our method.

(2) just diversity measure in different embedding spaces (**FPS-**$\langle$ ***ist***$\rangle$), and (3) combining both informativeness and diversity (**US-**$\langle$ ***ist***$\rangle$, i.e. our full method). As shown in Figure 5, active learning with combined measures performs much better than having informativeness or diversity alone. This again con rms the importance of diversifying a batch of informative objects for an effective active learning policy.

5 Conclusion

In this work, we presented a novel batch-mode active learning approach for triplet-based deep metric learning, that jointly balances the informativeness and diversity of a batch of triplets while decoupling the exact choice of heuristic for each criterion. We developed different measures for triplet decorrelation and found them effective for improving the performance of existing active learning approaches, which generally fail when labels are acquired in large batches. We further demonstrated that our decorrelation strategy generalizes easily to different informativeness measures. We extensively evaluated our method on three datasets and showed that our method is robust to different batch sizes, initial pools, and architectures, and performs consistently better than baselines.

References

[Adomavicius and Tuzhilin, 2005] Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *TKDE*, 2005.

[Arthur and Vassilvitskii, 2006] David Arthur and Sergei Vassilvitskii. k-means++: The advantages of careful seeding. Technical report, 2006.

[Ash *et al.*, 2020] Jordan T Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. In *ICLR*, 2020.

[Bachman *et al.*, 2017] Philip Bachman, Alessandro Sordoni, and Adam Trischler. Learning algorithms for active learning. In *ICML*, 2017.

[Baldridge and Palmer, 2009] Jason Baldridge and Alexis Palmer. How well does active learning actually work?: Time-based evaluation of cost-reduction strategies for language documentation. In *EMNLP*, 2009.

[Calinon *et al.*, 2007] Sylvain Calinon, Florent Guenter, and Aude Billard. On learning, representing, and generalizing a task in a humanoid robot. *TSMCC*, 2007.

[Eldar *et al.*, 1997] Yuval Eldar, Michael Lindenbaum, Moshe Porat, and Yehoshua Y Zeevi. The farthest point strategy for progressive image sampling. *TIP*, 1997.

[Freytag *et al.*, 2014] Alexander Freytag, Erik Rodner, and Joachim Denzler. Selecting influential examples: Active learning with expected model output changes. In *ECCV*, 2014.

[Gal *et al.*, 2017] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep Bayesian active learning with image data. In *ICML*, 2017.

[Golovin and Krause, 2011] Daniel Golovin and Andreas Krause. Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *JAIR*, 2011.

[Heim *et al.*, 2015] Eric Heim, Matthew Berger, Lee Seversky, and Milos Hauskrecht. Active perceptual similarity modeling with auxiliary information. In *AAAI*, 2015.

[Hoi *et al.*, 2006] Steven CH Hoi, Rong Jin, Jianke Zhu, and Michael R Lyu. Batch mode active learning and its application to medical image classi cation. In *ICML*, 2006.

[Huang *et al.*, 2016] Jiaji Huang, Rewon Child, Vinay Rao, Hairong Liu, Sanjeev Satheesh, and Adam Coates. Active learning for speech recognition: the power of gradients. In *NeurIPSW*, 2016.

[Kingma and Ba, 2015] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[Kirsch *et al.*, 2019] Andreas Kirsch, Joost van Amersfoort, and Yarin Gal. BatchBALD: Ef cient and diverse batch acquisition for deep Bayesian active learning. In *NeurIPS*, 2019.

[Konyushkova *et al.*, 2017] Ksenia Konyushkova, Raphael Sznitman, and Pascal Fua. Learning active learning from data. In *NeurIPS*, 2017.

[Krishnamurthy *et al.*, 2017] Akshay Krishnamurthy, Alekh Agarwal, Tzu-Kuo Huang, Hal Daume III, and John Langford. Active learning for cost-sensitive classi cation. In *ICML*, 2017.

[Lohaus *et al.*, 2019] Michael Lohaus, Philipp Hennig, and Ulrike von Luxburg. Uncertainty estimates for ordinal embeddings. *arXiv preprint arXiv:1906.11655*, 2019.

[Lu *et al.*, 2017] Rui Lu, Kailun Wu, Zhiyao Duan, and Changshui Zhang. Deep ranking: Triplet MatchNet for music metric learning. In *ICASSP*, 2017.

[Priyadarshini K *et al.*, 2019] Priyadarshini K, Siddhartha Chaudhuri, and Subhasis Chaudhuri. PerceptNet: Learning perceptual similarity of haptic textures in presence of unorderable triplets. In *WHC*, 2019.

[Sener and Savarese, 2018] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *ICLR*, 2018.

[Settles, 2012] Burr Settles. Active learning. *SLAIML*, 2012.

[Sinha *et al.*, 2019] Samarth Sinha, Sayna Ebrahimi, and Trevor Darrell. Variational adversarial active learning. In *ICCV*, 2019.

[Strese *et al.*, 2017] M. Strese, Y. Boeck, and E. Steinbach. Content-based surface material retrieval. In *WHC*, 2017.

[Tamuz *et al.*, 2011] Omer Tamuz, Ce Liu, Serge Belongie, Ohad Shamir, and Adam Tauman Kalai. Adaptively learning the crowd kernel. In *ICML*, 2011.

[Wei *et al.*, 2015] Kai Wei, Rishabh Iyer, and Jeff Bilmes. Submodularity in data subset selection and active learning. In *ICML*, 2015.

[Wilber *et al.*, 2014] Michael J Wilber, Iljung S Kwak, and Serge J Belongie. Cost-effective hits for relative similarity comparisons. In *AAAI*, 2014.

[Yang *et al.*, 2017] Lin Yang, Yizhe Zhang, Jianxu Chen, Siyuan Zhang, and Danny Z Chen. Suggestive annotation: A deep active learning framework for biomedical image segmentation. In *MICCAI*, 2017.

[Zhang *et al.*, 2017] Ye Zhang, Matthew Lease, and Byron C Wallace. Active discriminative text representation learning. In *AAAI*, 2017.

[Zhang *et al.*, 2018] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.