



The Hebrew University of Jerusalem
The Rachel and Selim Benin School of Computer Science and Engineering

Geometrical Active Sampling

דגימה פעילה גיאומטרית

Elias Wakile

Thesis submitted in partial fulfillment of the requirements
for the Master of Sciences degree
in Computer Science

Under the supervision of **Prof. Daphna Weinshall**

March 2025



האוניברסיטה העברית בירושלים
בית הספר להנדסה ולמדעי המחשב על שם רחל וסלים בנין
החוג למדעי המחשב

Geometrical Active Sampling

דגימה פעילה גיאומטרית

מוגש על ידי
אליאס וכילה

עבודת גמר לתואר מוסמך במדעי המחשב

עבודה זו הונחתה על ידי
פרופ' דפנה וינשל

אדר התשפ"ה

Abstract

Traditional Active Learning (AL) methods often struggle in low-budget scenarios, where only a limited number of data points can be annotated. Recent advances in representation and self-supervised learning have significantly improved the geometric structure of data embeddings, paving the way for novel approaches to sample selection. In this work, we introduce *Geometrical Active Sampling (GAS)*, an innovative AL algorithm tailored for the low-budget regime. GAS utilizes a spherical coding framework to effectively cover the latent space of image embeddings, with cosine similarity as the core metric. Grounded in Metric Geometry, Metric Embedding Theory, and Spherical Codes, our approach offers a robust and principled methodology. By leveraging coding the subspaces of the latent space, GAS addresses key challenges of achieving optimal coding of the space and maintaining separation between points. Empirical results demonstrate that GAS is less affected by an increase in *Sample Complexity*, *Label Complexity*, or both. Moreover, it outperforms existing SOTA AL methods in the low-to-mid-budget regime.

Acknowledgements

I would like to express my deepest gratitude to my advisor, Prof. Daphna Weinshall, for her invaluable guidance, support, and encouragement throughout this research. She has been attentive not only to my questions regarding research but also to matters of personal life, offering me valuable advice and life lessons. Her mentorship has taught me both discipline, diligence, and humility, qualities that extend beyond academia and into life itself. I am truly grateful for the opportunity to learn from her.

To my dearest sister, Mary, I cannot thank you enough. Your unwavering support, encouragement, and belief in me have been a source of strength throughout this journey. Your love and guidance have made this achievement possible, always there to listen and advise, and to push me forward. To my sister, Rana, I am also deeply thankful for her support, kindness, encouragement and belief in me. To my sister, Reem, I am grateful for her presence and support. Likewise, I extend my heartfelt gratitude to my parents Johnny and Amal, whose love, sacrifices, and wisdom have shaped me into the person I am today. I am grateful to have you by my side.

I owe special thanks to my spiritual father, Elder Ephraim of the Monastery of Vatopedi to whom I am eternally grateful. Through his prayers and counsels, I was inspired to pursue Computer Science. His prayers and his spiritual guidance has given me the resilience to succeed in my studies.

I also want to remember my grandmother, Mary, who was always attentive to me as a child. Though she passed away in 2018, I know she would have been overjoyed and proud to see me graduate. Alongside her, I thank my aunt, Eva, who always listened to me and never doubted my potential, often insisting I was a genius long before I had achieved anything.

Finally, I extend my appreciation to all those—friends, colleagues, and mentors—who have supported and inspired me throughout this journey, and especially to my best friend, Uri. This work is as much a result of their encouragement as it is of my own efforts.

Contents

1	Introduction	1
1.1	Cold Start	1
1.1.1	The Problem	2
1.1.2	Current Solutions	2
1.1.3	The Proposed Solution	2
1.2	Thesis Structure	3
2	Background	5
2.1	Learning Without Labels	5
2.1.1	Unsupervised Learning	6
2.1.2	Self-Supervised Learning	6
2.1.3	Representation Learning	6
2.2	Active Learning	7
2.3	Theoretical Preliminaries	9
2.3.1	From Euclidian Distance to Cosine Distance	9
2.3.2	From Maximum Coverage to Spherical Codes	10
3	Previous Work	12
3.1	Uncertainty Sampling Approaches	12
3.2	Diversity Sampling Approaches	15
3.2.1	Purely Diversity Sampling Approaches	15

3.2.2	Diversity with Uncertainty Sampling Approaches . . .	17
4	Methods	20
4.1	Our Approach	20
4.1.1	Spherical Code	20
4.1.2	Metric Embedding Theory	21
4.1.3	Self-supervised learning	21
4.1.4	Geometrical Sampling	21
4.2	The Algorithm	22
4.2.1	Computational Complexity Analysis	24
5	Theoretical Framework	26
5.1	Metric Geometry	26
5.2	Metric Embedding Theory	27
5.2.1	Definitions & Assumptions	27
5.2.2	Representation Learning	29
5.2.3	Embedding Subspace Distortion	30
5.3	Spherical Codes	34
6	Results	36
6.1	Methodology	36
6.1.1	Datasets	36
6.1.2	Learning Model Architecture	37
6.1.3	Self-Supervised Representation Learning	38
6.1.4	Representation and Dataset-Specific Delta Selection	38
6.2	Main Results	39
6.3	Ablation Study	47
6.3.1	Shrink Rate Hyperparameter	47
6.3.2	Cosine Similarity Threshold Hyperparameter	50

<i>CONTENTS</i>	v
7 Conclusion	53
A List of Subsets of the Imagenet classes	59

List of Figures

6.1	Comparison of AL strategies on CIFAR-10 across low-, mid-, and high-budget settings. The x-axis represents the number of labeled samples acquired, while the y-axis denotes the mean difference (from random) in test accuracy. The shaded region around each plot represents the error bars, where we conducted each experiment five times. The value on the plot is the mean, and the shaded area around it indicates the standard deviation.	40
6.2	Comparison of AL strategies on CIFAR-100 across low-, mid- and high-budget settings. The x-axis represents the number of labeled samples acquired, while the y-axis denotes the mean difference (from random) in test accuracy. The shaded region around each plot represents the error bars, where we ran each experiment five times.	42
6.3	Comparison of AL strategies on ImageNet-100 across low-, mid- and high-budget settings. The x-axis represents the number of labeled samples acquired, while the y-axis denotes the mean difference (from random) in test accuracy. The shaded region around each plot represents the error bars, where we conducted each experiment five times.	44
6.4	Comparison of AL strategies on ImageNet-200 across low-, and mid-budget settings. The x-axis represents the number of labeled samples acquired, while the y-axis denotes the mean difference (from random) in test accuracy. The shaded region around each plot represents the error bars, where we conducted each experiment five times.	46

6.5 Ablation study on the effect of ζ on test accuracy while sampling with GAS with a budget size of 3000. Each point represents the mean accuracy across five runs, with standard deviation error bars. 48

6.6 Ablation study on the effect of ζ on test accuracy while sampling with GAS with a budget size of 5000. Each point represents the mean accuracy across five runs, with standard deviation error bars. 49

6.7 Ablation study on the effect of the parameter δ on test accuracy while sampling with GAS a set of points with a budget of 3000. Each point represents the mean accuracy across 5 runs, with error bars indicating the standard deviation. . . . 51

6.8 Ablation study on the effect of the parameter δ on test accuracy while sampling with GAS a set of points with a budget of 5000. Each point represents the mean accuracy across 5 runs, with error bars indicating the standard deviation. . . . 52

List of Tables

6.1	Values of δ chosen for each dataset	38
6.2	Comparison of AL strategies results on CIFAR-10 across low-, mid-, and high-budget settings. Mean \pm SE of test accuracy where we conducted each experiment five times.	39
6.3	Comparison of AL strategies results on CIFAR-100 across low-, mid-, and high-budget settings. Mean \pm SE of test accuracy where we conducted each experiment five times.	41
6.4	Comparison of AL strategies results on ImageNet-100 across low-, mid-, and high-budget settings. Mean \pm SE of test accuracy where we conducted each experiment five times.	43
6.5	Comparison of AL strategies results on ImageNet-200 across low-, mid-, and high-budget settings. Mean \pm SE of test accuracy where we conducted each experiment five times.	45

1 Introduction

In the era of Big Data, we are confronted with vast amounts of unannotated information, particularly in domains such as image recognition. Deep learning models, despite their impressive performance within the high annotation budget framework, tend to struggle when the annotation budget is limited. **Active Learning (AL)** is a framework designed to address this challenge by addressing the following question: how to effectively pick a small subset of data points to label, which will maximize the learning potential for a **Deep Learning (DL)** model. We propose an algorithm for the low-budget AL problem which is necessary for the expansion of DL to fields where professional image annotation can be costly.

For the most part, deep AL strategies rely on a combination of two main strategies. *Uncertainty Sampling* [10, 15, 21, 28] prioritizes examples for which the learning model is least certain about, aiming to maximize the learning potential of each new annotation. *Diversity Sampling* [1, 15, 16, 21, 31, 36] seeks examples that are chosen from diverse regions of the data distribution, in order to represent it wholly and reduce redundancy in the annotation.

1.1 Cold Start

Research in active learning has predominantly focused on mid to high-budget scenarios [1, 10, 15, 21, 26, 31, 32]. Consequently, most existing algorithms are ill-suited for low-budget active learning. In the following subsection, we will analyze the shortcomings of these high-budget algorithms in low-budget settings and propose novel strategies to address these challenges, ensuring effective and efficient active learning under resource

constraints.

1.1.1 The Problem

Recent research indicates that both strategies face significant limitations in the low-budget domain. Accordingly, *Uncertainty Sampling* often fails to provide meaningful improvements due to insufficient model calibration during the early stages of training [16, 20]. Likewise, *Diversity Sampling* struggles in low budgets because the inherently small sample size limits its ability to capture the full data distribution effectively, often leading to suboptimal representation [21]. As a result, most AL methods fail to outperform random selection, a phenomenon that is termed “cold start”.

1.1.2 Current Solutions

Despite these challenges, some methods have demonstrated better performance in low-budget scenarios by sampling points aiming for spatial coverage [2, 16, 28, 36], which seeks to balance representation and uncertainty in a more efficient way. For instance, coverage-based approaches aim to strategically select samples that are both informative and distributed across critical regions of the data space. These methods mitigate the “cold start” problem that plagues most traditional AL strategies, where improvements over random selection are marginal. This motivates the introduction of our new approach of Geometrical Sampling, which prioritizes the annotation of geometrically strategic points. We will detail regarding existing approaches for the cold-start problem as well as other existing Active Learning Algorithms in chapter 3.

1.1.3 The Proposed Solution

Building on the idea of spatial coverage, we introduce **Geometrical Active Sampling (GAS)**, a novel low-budget AL algorithm designed to efficiently spherically code the latent space while maintaining diversity in sample selection. Unlike traditional uncertainty and diversity sampling approaches, which struggle in low-budget settings due to unreliable model confidence estimates and limited representation capacity, our method is rooted in a

geometrical perspective, leveraging insights from metric geometry, metric embedding theory, and spherical codes. Our approach is motivated by the observation that modern self-supervised learning (SSL) techniques produce structured latent spaces, where distances between points correspond to meaningful semantic relationships. GAS exploits this structure by employing a spherical coding framework to optimize the placement of selected samples within the latent space. Specifically, our method iteratively refines the selection process by:

1. **Spherical Code-Based Sampling:** Modeling sample selection as an optimal coding problem in a high-dimensional space, ensuring that labeled points effectively represent the dataset’s intrinsic geometry.
2. **Metric Embedding Theory:** Leveraging distance-preserving transformations to analyze and optimize sample selection strategies.
3. **Progressive Subspace Projection:** Iteratively shrinking the subspace in which sampling occurs, refining sample selection and ensuring both global exploration and local refinement.

1.2 Thesis Structure

The remainder of this thesis is structured as follows:

- **Chapter 2: Background** – Introduces the core concepts underlying this work, including self-supervised and unsupervised learning, Active Learning, and relevant geometric foundations such as cosine similarity and spherical codes. This chapter establishes the theoretical context in which our method operates.
- **Chapter 3: Previous Work** – Reviews related literature on Active Learning methods, with a focus on uncertainty-based, diversity-based, and hybrid approaches. We position our method in contrast to prior work and highlight the limitations our approach aims to address.
- **Chapter 4: Method** – Details our proposed approach, Geometrical Active Sampling (GAS). We present the algorithmic formulation, discuss the role of metric geometry and spherical codes, and describe the selection mechanism based on cosine similarity thresholds.

- **Chapter 5: Theoretical Framework** – Provides a deeper mathematical treatment of the geometric concepts used in GAS, including metric embedding theory and angular separation, and analyzes the theoretical motivations behind our selection strategy.
- **Chapter 6: Results** – Describes the experimental setup, including datasets and model architectures. We benchmark GAS against existing Active Learning methods, report performance across various budgets, and analyze the effect of hyperparameter tuning through ablation studies.
- **Chapter 7: Conclusion** – Summarizes the key contributions of this work, discusses its limitations, and outlines potential directions for future research in geometry-driven Active Learning.

2 Background

To motivate and contextualize our proposed method, Geometrical Active Sampling (GAS), this chapter provides the necessary background across three key areas: learning without labels, active learning, and the theoretical underpinnings of our geometric approach. Together, these topics form the conceptual and technical foundation upon which GAS is built.

2.1 Learning Without Labels

A fundamental challenge in modern machine learning is the reliance on large amounts of labeled data, which can be expensive and time-consuming to obtain. To address this, a wide range of methods have been developed to learn directly from unlabeled data—unlocking the potential of raw data at scale. This section explores the key paradigms that enable learning without supervision, focusing on techniques that allow models to extract structure, patterns, and semantic understanding without the need for manual annotations.

We begin with unsupervised learning, which aims to uncover inherent patterns or groupings in data without any labels. We then turn to self-supervised learning, a powerful subclass of unsupervised learning that creates auxiliary tasks using pseudo-labels generated from the data itself. Finally, we discuss the broader concept of representation learning, which underpins both unsupervised and self-supervised methods, and plays a central role in enabling models to generalize across tasks by learning meaningful data embeddings.

2.1.1 Unsupervised Learning

Unsupervised learning refers to learning patterns or structures from data without any ground-truth labels. Common tasks in this domain include clustering and dimensionality reduction. These techniques aim to capture the underlying structure of the data—such as grouping similar examples or identifying latent factors—without explicit supervision.

2.1.2 Self-Supervised Learning

Self-supervised learning (SSL) is a subclass of unsupervised learning that constructs pretext tasks by generating pseudo-labels directly from the data itself. These tasks are designed so that, in the process of solving them, the model is encouraged to extract meaningful and generalizable features—without access to any human-provided annotations. Although the pretext task may seem artificial or unrelated to a downstream goal (e.g., predicting image rotations, patch positions, or augmented view similarity), success on these tasks requires the model to understand the semantic structure of the data behind the scenes. This process lies at the heart of representation learning, where the objective is to map raw input data into a latent space in which important properties of the data are preserved and easily accessible to downstream models.

2.1.3 Representation Learning

Self-supervised learning can thus be viewed as a powerful framework for learning representations without supervision, enabling the training of models that generalize well across tasks—even when labeled data is scarce. As we are working in the framework of image classification where data is high dimensional, we must consider the curse of dimensionality. It refers to the exponential increase in complexity associated with high-dimensional data spaces [18]. As the dimensionality of data increases, the volume of the space grows so rapidly that data points become sparse, making it difficult for models to generalize effectively [5]. In order to avoid this problem, some AL methods took advantage of the recent progress in self-supervised methods, and employed the embedding of images into a latent space using representation learning [1, 15, 16, 21, 26, 28, 36].

In our context, we leverage two prominent self-supervised representation learning methods: SimCLR [9] and DINOv2 [29], both of which have demonstrated strong performance in computer vision tasks.

SimCLR is a contrastive learning method that learns representations by maximizing agreement between different augmented views of the same image while pushing apart representations of views from different images. It relies on a contrastive loss applied to features produced by a deep encoder, where each training batch includes multiple positive and negative pairs created through random augmentations. The core idea is that in order to bring together positive pairs and repel negatives, the model must learn features that are invariant to the chosen augmentations and discriminative enough to differentiate between different images.

DINOv2, on the other hand, is based on self-distillation without labels. It trains a student network to match the output distributions of a teacher network, using a set of global and local image crops. While both networks process augmented views of the same image, only the teacher receives the full global context. The teacher's weights are not learned directly but are instead updated as an exponential moving average of the student's weights. This bootstrapping mechanism helps stabilize learning and avoids the need for negative samples or contrastive loss. DINOv2 also includes techniques like output sharpening and batch centering to prevent collapse, where all representations become identical.

In the next section, we turn our attention to Active Learning, and explain why it can naturally also be framed as a SSL problem.

2.2 Active Learning

Active Learning is an interactive learning paradigm that aims to optimize the use of labeled data by allowing the model to selectively query annotations for unlabeled samples. The central idea is that, rather than passively consuming a fixed labeled dataset, the learner actively identifies which data points are most informative and requests their labels. This process is analogous to a student posing questions to a teacher in order to focus on areas of uncertainty and accelerate their understanding.

The Active Learning cycle proceeds iteratively: the learner is initially provided with a large pool of unlabeled data and a small labeled subset (which may be empty). At each iteration, the model is trained on the current

labeled set and then used to evaluate the unlabeled pool. Based on a predefined selection strategy—often guided by uncertainty, diversity, or representativeness—the learner selects a batch of unlabeled samples to be labeled by an oracle (e.g., a human annotator). This process is repeated until a specified labeling budget is exhausted. The final predictive model is then trained on the accumulated labeled data.

Active Learning can be viewed as a specialized form of semi-supervised learning, as it operates in a setting where only a small subset of the available data is labeled, while the remainder remains unlabeled. Similar to traditional semi-supervised learning, Active Learning seeks to exploit the underlying structure of the unlabeled data to improve model performance. However, it extends this framework by introducing an interactive component, wherein the learning algorithm selectively queries labels for data points it deems most informative. This iterative querying process enables the learner to actively influence the labeling process, thereby optimizing the use of limited annotation resources. Consequently, Active Learning embodies the core principles of semi-supervised learning while adding a decision-making mechanism that prioritizes sample selection for efficient supervision.

A closely related idea is knowledge distillation, which offers another perspective on how learning can be guided with indirect supervision. Knowledge distillation is a technique originally developed for transferring knowledge from a large, pretrained model (the teacher) to a smaller, more efficient model (the student). In the context of self-supervised learning, this idea has been adapted to enable training without labels by using one network to generate learning signals for another.

In Active Learning, we similarly aim to distill knowledge—not from a larger model, but from a representation learning agent trained in a self-supervised manner. The goal is to extract meaningful structural information from the unlabeled data distribution and use it to guide sample selection. By leveraging this learned representation, we can identify the most informative and diverse samples to label. In this way, we transfer knowledge from the representation model into both the active learning strategy—which selects samples—and the supervised model—which ultimately learns from the labeled data. This distillation enables the system to make efficient use of limited labels while maintaining strong performance.

2.3 Theoretical Preliminaries

This section outlines the core geometric assumptions that underpin our approach. In the first part, we motivate the use of cosine distance as the natural similarity measure in self-supervised representation spaces, replacing the traditional reliance on Euclidean distance. In the second part, we present two classical problems—maximum coverage and spherical codes—which serve as conceptual tools for understanding sample diversity. While maximum coverage underlies several prior works, we adopt the spherical codes perspective to better reflect the structure of normalized latent spaces.

2.3.1 From Euclidian Distance to Cosine Distance

In RGB space, the high dimensionality of images introduces the “curse of dimensionality”, which often renders Euclidian Distance measures unreliable or misleading [5, 18]. Small variations in lighting, viewpoint, or noise can substantially change pixel values without altering the underlying semantics. By contrast, representation learning methods embed images into a space where key semantic or structural properties are more explicitly captured, mitigating this problem. Within these learned embeddings, cosine distance (or equivalently cosine similarity) focuses on the angular relationships among feature vectors rather than their magnitudes. This orientation-based property is particularly effective in contrastive learning frameworks, where vectors are normalized to the unit sphere, preserving semantic similarities while reducing the influence of scale [7, 9].

Definition 2.1. Cosine Distance: Let $x, y \in \mathbb{R}^d$ two vectors, and define *cosine – similarity* : $\mathbb{R}^d \times \mathbb{R}^d \rightarrow [0, 2]$ as:

$$\text{cosine – similarity}(x, y) = 1 - \frac{\langle x, y \rangle}{\|x\| \cdot \|y\|}$$

This metric respects the relative orientation between vectors while being invariant to their scale, making it particularly suitable for analyzing and comparing representations learned by contrastive or self-distillation-based

methods such as SimCLR and DINOv2. These methods optimize feature similarity through angular relationships, often using dot-product or cosine-based objectives [9, 29].

In this work, we adopt cosine distance as the core similarity measure in our active learning framework. This choice reflects the geometry of the latent space induced by self-supervised learning, and aligns naturally with the normalized embeddings we use. Importantly, it also enables the use of well-established tools from spherical geometry and metric embedding theory, which we leverage in the development of our selection strategy. In the following subsection, we introduce two classical problems—maximum coverage and spherical codes—to ground our geometric formulation of sample diversity and coverage.

2.3.2 From Maximum Coverage to Spherical Codes

In this section, we explore several classical problems from discrete geometry that provide the theoretical foundation for our approach. We begin with maximum coverage problem, which seeks to select a subset of elements that collectively cover as much of the space as possible—a central challenge in many Active Learning strategies [2, 16, 28, 31, 36].

Definition 2.2. Max Coverage: [36] Let $b \in \mathbb{N}$ denote an integer, \mathcal{U} denote a set of elements, and let $\mathcal{S} = \{\mathcal{S}_i\}_{i=1}^m$ denote a collection of subsets of \mathcal{U} . In the problem of Max Coverage we wish to find b subsets in \mathcal{S} with union of maximum cardinality:

$$\operatorname{argmax}_{\mathcal{S}' \subseteq \mathcal{S}; |\mathcal{S}'|=b} \left| \bigcup_{\mathcal{S}_i \in \mathcal{S}'} \mathcal{S}_i \right|$$

In our work, we consider data embedded in a latent space equipped with the cosine similarity as the underlying distance function. As such, traditional formulations of coverage based on Euclidean distance or ball coverings are not directly applicable. Instead, we turn to the concept of Spherical Coding, a well-established concept concerned with the optimal arrangement of points on the unit sphere to maximize angular separation. Spherical Codes naturally generalize the idea of space coverage when points lie on the unit sphere. In the following, we present the spherical codes problem,

which serves as a geometric analogue of maximum coverage under cosine distance constraints.

Definition 2.3. Spherical Coding: [35] For some space Ω and some angle $\theta \in [0, 2\pi]$, define *spherical coding* as

$$\mathcal{A}(\Omega, \theta) = \underset{A \subseteq \Omega: \forall x, y \in A: 1 - \frac{\langle x, y \rangle}{\|x\| \cdot \|y\|} < \theta}{\operatorname{argmax}} |A|$$

The spherical coding problem formalizes the notion of selecting a set of points on the unit sphere such that all pairwise cosine distances exceed a given angular threshold, while maximizing the cardinality of that set. This aligns closely with our goal in Active Learning: to choose a batch of samples that are both diverse and well-separated in the latent space, under cosine distance.

While the maximum coverage problem has served as the foundation for many existing AL strategies—often based on Euclidean assumptions—we depart from this formulation and instead adopt spherical coding as our guiding geometric framework. This shift is motivated not only by the underlying metric properties of modern self-supervised embeddings but also by the rich theoretical tools that spherical geometry provides. In our approach, this perspective enables a more faithful and theoretically grounded treatment of coverage and diversity in normalized latent spaces.

In the next chapter, we review a range of existing Active Learning methods—many of which are built on maximum coverage-like principles—and highlight how our formulation based on cosine similarity and spherical codes offers a principled alternative in the low-budget regime.

3 Previous Work

Over the years, numerous AL algorithms have been proposed, each aiming to maximize model performance while minimizing the number of labeled samples required. As we have seen in the introduction of Chapter 1, these methods can generally be categorized based on their selection criteria into two main paradigms: uncertainty-based sampling and diversity-based sampling. This chapter provides an overview of existing AL methods, structured into two main sections: Uncertainty Sampling Approaches and Diversity Sampling Approaches. Through this discussion, we analyze the strengths and limitations of existing AL strategies, laying the groundwork for the motivation and development of our proposed method.

3.1 Uncertainty Sampling Approaches

Uncertainty-based Active Learning methods focus on selecting samples where the model exhibits the highest uncertainty based on several criteria, under the assumption that labeling these will yield the greatest improvement in model performance. The underlying intuition is that uncertain predictions correspond to regions where the model lacks confidence, and acquiring labels for these samples should help refine decision boundaries. Various uncertainty measures have been proposed derived from the model’s predictions, including entropy-based confidence scores, margin-based methods, and Bayesian uncertainty estimation. These methods quantify uncertainty based on how spread out the probability distribution over the possible classes is, selecting the least confident samples for annotation. Below, we outline four fundamental uncertainty-based selection criteria, analyzing their strengths and limitations:

Uncertainty: Selects the points for which the model is least confident about their classification, typically defined as the sample with the highest predicted probability for a single class being the lowest among all unannotated samples. The selection rule is given by:

$$\operatorname{argmin}_{x \in \mathcal{X}} \left[\max_{y \in \mathcal{Y}} [\mathbb{P}_\theta(y|x)] \right]$$

While this method is simple to implement, it requires for its effectiveness that the model's probabilities be well calibrated, which is a strong assumption. This method ignores how the uncertainty is distributed across classes, which may lead to suboptimal queries and can be overly biased toward outlier samples, which might not be representative of the overall data distribution. Added to that, this method does not necessarily select samples near the decision boundary, which are often the most informative.

Margin: [30] Selects points that maximize the difference between the highest and second-highest predicted probabilities of the classes predicted by the learning model. The selection rule is given by:

$$\operatorname{argmax}_{x \in \mathcal{X}} \left[\max_{y \in \mathcal{Y}} [\mathbb{P}_\theta(y|x)] - \max_{y \in \mathcal{Y} \setminus \left[\operatorname{argmax}_{y \in \mathcal{Y}} [\mathbb{P}_\theta(y|x)] \right]} [\mathbb{P}_\theta(y|x)] \right]$$

This method is more robust than Least Confidence, as it focuses on decision boundary samples and works well for classifiers with softmax outputs, where the margin is a natural measure of confidence. Nonetheless, this method ignores the full class probability distribution, only considering the top two class probabilities. Hence, if the model is overconfident, the margin may not accurately reflect uncertainty. This method can be sensitive to imbalanced class distributions, where small margins may be a natural property rather than an indicator of high uncertainty.

Max Entropy: [32] Selects points that maximize Shannon entropy. This method uses information from all class probabilities, leading to more comprehensive uncertainty estimation. The selection rule is given by:

$$\operatorname{argmax}_{x \in \mathcal{X}} \left[\sum_{y \in \mathcal{Y}} [\mathbb{P}_\theta(y|x) \cdot \log(\mathbb{P}_\theta(y|x))] \right]$$

Wherefrom, it is more effective in multi-class classification, where uncertainty is not just about two competing classes but distributed across many

possibilities making it less sensitive to extreme probability values compared to Uncertainty. However, this method is the most computationally expensive so far, as it requires calculating the entropy of the full probability distribution. This method can still suffer from redundancy, selecting multiple similar uncertain points. Another major limitation is that the method gives priority to samples where probabilities are evenly spread across many classes, which is not always an indicator of useful uncertainty.

While the previously seen methods rely solely on model confidence scores, **Bayesian Active Learning By Disagreement (BALD)**: [21] takes a Bayesian approach by selecting points that maximize the mutual information between predictions and model parameters. Instead of selecting points where the model is merely uncertain, BALD prioritizes samples where the model is both uncertain and likely to reduce uncertainty arising from a lack of knowledge in the model rather than inherent noise in the data. This is achieved by measuring how much a sample contributes to reducing the overall uncertainty in the model parameters. The selection rule is given by:

$$\operatorname{argmax}_{x \in \mathcal{X}} \left[\sum_{y \in \mathcal{Y}} [\mathbb{P}_\theta(y|x) \cdot \log(\mathbb{P}_\theta(y|x))] \right] - \mathbb{E}_\theta \left[\sum_{y \in \mathcal{Y}} [\mathbb{P}_\theta(y|x) \cdot \log(\mathbb{P}_\theta(y|x))] \right]$$

Thence, this approach selects data points for labeling which will yield a model with an enhanced generalization and robustness. Nonetheless, it is noteworthy that this approach is computationally expensive and requires a probabilistic framework, which may not always be available in standard DL models.

While these approaches are effective at reducing classification errors, they often suffer from redundancy, as multiple uncertain samples may be clustered together in the latent space. This can lead to inefficient query selection, particularly in high-dimensional settings where uncertainty alone does not guarantee optimal data coverage. To address this limitation, recent methods aim to incorporate diversity constraints into uncertainty-based selection, ensuring that the labeled set remains both informative and well-distributed. In the next section, we explore key diversity-driven strategies and their role in improving sample selection beyond uncertainty-based approaches.

3.2 Diversity Sampling Approaches

Now, we will delve into diversity sampling, which seeks to improve generalization by ensuring that labeled samples are well-distributed across a latent space. Diversity sampling methods often operate in a latent space, where the high-dimensional input data (e.g., images) is embedded using a representation learning model $\phi : \mathcal{X} \rightarrow \mathbb{R}^d$ as we have seen in section 2.1.3. These embeddings aim to capture the semantic structure of the data while reducing dimensionality, allowing distances between points to reflect meaningful similarity. By working in this latent space, diversity-based methods can more effectively identify representative or well-distributed samples, enabling them to cover the data distribution with fewer queries. Next, we discuss diversity sampling methods, which explicitly aim to enhance space coverage by selecting samples that maximize representation. Finally, we explore hybrid approaches that integrate both uncertainty and diversity to achieve a balance between informativeness and coverage.

3.2.1 Purely Diversity Sampling Approaches

Purely diversity-based sampling methods aim to select a subset of points that best cover the data distribution, ensuring that the labeled set reflects the structure of the full dataset. Two notable approaches in this category are CoreSet, TypiClust and MaxHerding.

CoreSet: [31] Selects points minimizing the distance of any data point from its nearest cover labeled point. The selection rule is given by:

$$\operatorname{argmin}_{\mathcal{S} \subseteq \mathcal{X}, |\mathcal{S}|=b} \left[\min_{x \in \mathcal{X}} \left[\left[\max_{s \in \mathcal{S}} [\|\phi(x) - \phi(s)\|_2] \right] \right] \right]$$

While this approach has strong theoretical guarantees on space coverage, ensures selected samples are well-distributed and diverse, and is robust to label noise due to focus on structure rather than labels. It ignores uncertainty, potentially missing points near decision boundaries that are more informative and assumes that Euclidian distances are meaningful in the latent space which might not always be the case.

TypiClust: [16] Measures an example's typicality by the inverse of the average Euclidean distance to its k nearest neighbors (k is a hyperparameter). It

then selects the most typical examples from the largest uncovered clusters. The typicality is given by:

$$\text{Typicality}(x) = \left(\frac{1}{k} \cdot \sum_{\phi(x_i) \in k\text{-NN}(\phi(x))} \|\phi(x) - \phi(x_i)\|_2 \right)^{-1}$$

The approach focuses on representative and typical examples, which can lead to fast generalization, encourages selection from dense regions, avoiding outliers. Nonetheless, it might miss rare but informative samples near the decision boundary or in low-density areas. That is due to the fact that like CoreSet, it relies heavily on the clustering, hence the euclidian distance and does not rely on uncertainty.

MaxHerding: [2] a diversity-based Active Learning algorithm designed to select batches of samples that are maximally spread out in the latent space. Unlike uncertainty-based methods, MaxHerding does not explicitly consider model confidence or predictive entropy. Instead, it focuses solely on minimizing redundancy by encouraging geometric separation among selected samples. This is achieved through a kernel-based selection rule that promotes diversity within the batch by penalizing similarity. The selection rule is given by:

$$\operatorname{argmax}_{S \subseteq \mathcal{X}, |S|=b} \left[\mathbb{E}_{s \in S} \left[\max_{s' \in S} [k(s, s')] \right] \right]$$

Where $k(s, s')$ is a similarity kernel (e.g., based on Euclidean or cosine similarity) that quantifies how close two samples are in the latent space.

MaxHerding is particularly effective in promoting diverse batch selection, making it well-suited for scenarios where it is important to cover the latent space evenly. Its reliance on pairwise similarity enables it to capture the global structure of the data and reduce the selection of redundant samples, which is especially beneficial in high-dimensional representation spaces learned through self-supervised methods.

However, its exclusion of uncertainty from the selection rule is a significant limitation. In situations where the most informative samples are not necessarily the most spatially diverse—such as near decision boundaries—MaxHerding may fail to prioritize valuable queries. Moreover, the method’s performance is sensitive to the choice of similarity kernel, and if the kernel is not well-aligned with the semantic structure of the latent space, the selected samples may be diverse but not truly informative. As

its predecessors in the section, it relies heavily on the clustering, hence the Euclidian distance and does not rely on uncertainty. Finally, computing the kernel matrix over large candidate pools can be computationally intensive, posing challenges for scalability in large-scale settings.

The algorithms presented above embody the principle of diversity through spatial coverage, but their lack of attention to informative uncertainty limits their standalone effectiveness—especially in scenarios where fine decision boundary refinement is crucial. These limitations motivate hybrid approaches that incorporate both uncertainty and diversity, discussed in the next section.

3.2.2 Diversity with Uncertainty Sampling Approaches

While purely uncertainty-based methods aim to select the most ambiguous samples and purely diversity-based methods focus on well-distributed representative samples, diversity with uncertainty approaches aim to combine the strengths of both. These methods select informative samples that are not only uncertain but also diverse in the latent space, promoting label efficiency, generalization, and robustness. Below, we describe several influential algorithms in this class.

BADGE: [1] Batch Active learning by Diverse Gradient Embeddings selects points with a probability proportional to the squared distance to the nearest covered point on a gradient embedding of the cross entropy loss function with respect to the weights of the last layer of the neural network on which the model was trained. Let us assume that the embedding used is $\phi(x) = \nabla_{\theta_{last}}(\ell_{CE}(NN(x), \hat{y}_x))$ for \hat{y}_x the label predicted by the neural network model. The selection rule is given by:

$$\min_{\mathcal{S} \subseteq \mathcal{X}, |\mathcal{S}|=b} \left[\min_{x \in \mathcal{X}} \left[\left[\max_{s \in \mathcal{S}} [\|\phi(x) - \phi(s)\|_2] \right] \right] \right]$$

BADGE offers a compelling approach to Active Learning by combining uncertainty and diversity through gradient-based representations. By selecting points whose gradient embeddings are diverse—specifically, points that are far from previously selected ones in the gradient space—BADGE effectively targets uncertain samples that also promise to contribute new information. This makes it particularly effective in settings where the goal is to refine decision boundaries while maintaining broad coverage of the

data distribution. However, BADGE has several limitations. It relies heavily on the quality of pseudo-labels used to compute the gradients, which can be unreliable in early training stages when the model’s predictions are still unstable. In addition, it operates in a model-specific gradient space, rather than directly within the learned data representations, and computing these high-dimensional gradients for all unlabeled points introduces significant computational and memory overhead. In the following, we analyze ProbCover in detail, highlighting both its advantages and limitations.

To address these challenges, ProbCover replaces the gradient space with a latent space learned via self-supervised representation learning methods. **ProbCover:** [36] Selects points that maximize probabilistic coverage of the data distribution that jointly accounts for uncertainty and diversity, offering a principled and scalable approach to sample selection. The selection rule is given by:

$$\operatorname{argmin}_{\mathcal{S} \subseteq \mathcal{X}, |\mathcal{S}|=b} \left[\mathbb{P} \left[\bigcup_{s \in \mathcal{S}} B_\delta(s) \right] \right]$$

ProbCover is a hybrid Active Learning method that combines uncertainty sampling with a probabilistic variant of the set cover problem to promote geometric diversity. Its core strength lies in its ability to maintain a principled balance between uncertainty and diversity, selecting samples that are both informative and well-distributed across the representation space. This makes it particularly effective in the early stages of the Active Learning process, where coverage is critical.

However, ProbCover also has notable limitations. Its primary Achilles’ heel is its reliance on the Euclidean metric in the latent space. While Euclidean distance is convenient computationally, it may fail to capture meaningful relationships in learned representation spaces—especially those optimized with respect to cosine similarity or angular separation. As a result, once a certain level of coverage is reached, the method may begin selecting uninformative or redundant samples, in some cases performing worse than random sampling in later rounds.

Despite these drawbacks, ProbCover represents an important step in the right direction by attempting to unify uncertainty and diversity under a single objective. This intuition is refined by the approaches we will present followingly. Our proposed method, Geometrical Active Sampling (GAS), builds upon this same insight but goes further—choosing cosine similarity as the metric, grounding the δ selection process in a spherical coding framework that is both theoretically motivated and computationally

practical. GAS resolves the key limitations of ProbCover by aligning the geometry of the method with that of the embedding space, and by avoiding hard combinatorial reductions in favor of direct, geometry-aware selection.

DCoM: [28] DCoM (Dynamic Coverage and Margin mix) extends the ideas of ProbCover by addressing one of its key limitations—its poor performance in high-budget scenarios. While ProbCover focuses more on geometric diversity, DCoM incorporates a margin-based regularization term to account for model uncertainty, enabling more effective selection as the labeled pool grows. Conceptually, DCoM frames its selection rule as a regularized maximum coverage objective:

$$\operatorname{argmin}_{\mathcal{S} \subseteq \mathcal{X}, |\mathcal{S}|=b} \left[\mathbb{P} \left[\bigcup_{s \in \mathcal{S}} B_\delta(s) \right] + \alpha \cdot \operatorname{Margin}(\mathcal{S}) \right]$$

Where the margin term penalizes confident predictions, and α is a temperature hyperparameter tuned by DCoM to balance diversity and uncertainty. Despite these improvements, DCoM introduces several practical limitations. Like ProbCover, it relies on Euclidean distance in the latent space, which may not accurately capture semantic relationships in modern representation learning, where cosine similarity is often more appropriate. Furthermore, the method incurs significant computational overhead: it requires computing a δ -expansion around each sample to estimate coverage and evaluating margin scores for all unlabeled points. In addition, DCoM introduces several hyperparameters that must be carefully tuned, including the temperature parameter for the coverage-uncertainty trade-off, the initial δ radius—similar to ProbCover—which controls coverage sensitivity, and the overall configuration of the δ -expansion process. These factors contribute to increased tuning complexity and runtime.

These limitations have motivated the development of subsequent methods that aim to further refine the balance between uncertainty and diversity, while addressing the geometric and computational drawbacks of earlier approaches. In our work, we introduce a novel paradigm; geometric-based sampling. Geometric-based approach takes a more structured perspective, leveraging the underlying metric space of the data representation to guide selection. These methods utilize tools from metric geometry, clustering, and embedding theory to ensure that queried samples both explore the space efficiently and maintain separation between labeled points. Geometric-based methods provide a promising alternative by explicitly modeling the structure of the latent space, balancing uncertainty, diversity, and space coverage more effectively.

4 Methods

In this thesis, we introduce a novel AL method – Geometrical Active Sampling (GAS), where we sample the images with the most δ -cosine-similar neighbors. In this chapter, we will present our proposed method.

4.1 Our Approach

The underlying objective is to obtain a subset of points that constitute an effective spherical code while using an embedding of the images within a given latent space. In *spherical code* we aim to choose a set of points with the most cosine-similar neighbors up to a certain threshold. In the context of AL, the objective is to select a subset of the unlabeled data-points that its projection onto the unit sphere is evenly distributed. We establish in Chapter 5 the theoretical preliminaries that justify the proposed strategy.

4.1.1 Spherical Code

As in most AL work, our method is greedy, populating the select subset of points on the sphere by continually adding points in a greedy manner. More specifically, we follow the greedy heuristic of sampling images with the highest number of δ -cosine-similar neighbors in the embedding space. The estimation of δ is done through spherical codes theory, as we elaborate in Section 5.3, in order to ensure that the representation space is neither overly dense (which can lead to redundancy) nor too sparse (which might omit relevant image variations), thus striking an effective balance.

4.1.2 Metric Embedding Theory

The problem we face is that When reaching proximity to full spherical coding for the computed δ , the score used for greedy selection can no longer differentiate between the remaining points to be selected from. Furthermore, it may deliver an imbalanced distribution of images covered by each sampled point [11, 17]. Our solution is to reduce the dimension of the latent space by projecting the embedding onto a subspace, at which point we continue sampling with respect to the projected latent space. This is one of the main novelties in the approach, and the way we do it is based on principled mathematical foundations. The effectiveness of these subspace projections is analyzed within Metric Embedding Theory, as discussed in Section 5.2 that guarantees the approximate preservation of the original geometry. The algorithm is described in Section 4.2.

4.1.3 Self-supervised learning

It is noteworthy that the image embeddings are generated using unsupervised representation learning methods [9, 29]. Importantly, our method does not rely on labeled data, either for defining neighborhoods or for generating the latent space and its subspaces. As such, the entire approach is fully self-supervised.

4.1.4 Geometrical Sampling

We refine our sampling by focusing on the embedding of data-points that have the highest number of neighbors within a chosen δ -cosine-similar neighborhood, thereby capturing a wide spectrum of semantic content while preserving meaningful angular separations. In this process, δ is specifically tuned to match the subspace of interest while preserving meaningful angular separations. Our approach assumes that the pairwise cosine similarities among all image representations are sampled from a particular distribution, and we determine δ based on this assumption. The detailed explanation of this distribution-based tuning procedure is presented in section 5.1.

4.2 The Algorithm

We introduce an algorithm for adaptively querying for annotation projected data-point embeddings from a high-dimensional latent space, where projections are formed by selecting a subset of coordinates from the original space. The design addresses two key challenges: ensuring diverse semantic coverage while maintaining meaningful angular separations among projected points. The parameter γ controls the shrinking in dimension of the projection, balancing complexity with the preservation of essential geometric relationships, while δ regulates separation on the unit sphere to ensure the effectiveness of the spherical code. This procedure serves as the practical implementation of our framework, with later sections establishing its theoretical guarantees, including distortion bounds and distribution-based threshold tuning.

This procedure greedily and incrementally selects *budget* unlabeled points with the most δ -cosine-similar neighbors without a queried sample in the embedding space under that same within that cosine similarity threshold. First, the algorithm computes embeddings for all data points and initializes the subspace proportion γ as 1. It then constructs a list of graphs via algorithm 2, where each graph connects vertices (embeddings) if their cosine distance is below δ . During each iteration, the algorithm identifies the point with the highest total degree across all sub-graphs, signifying it covers many others—and adds it to Q . If the maximum degree implies there is at most about 1 edge per graph in the list, it means the current graphs are not sufficiently covering the space. Thus, γ and δ are updated and appends to the list more graphs that are generated with algorithm 2. The formula for the update of δ is derived from theorem 5.2.

Algorithm 1 Geometrical Active Sampling

Input: labeled set $\mathcal{L} \subset \mathcal{X}$, unlabeled set $\mathcal{U} \subset \mathcal{X}$, shrink rate $\xi \in (0, 1)$, delta $\delta \in (0, 1)$, budget $\in \mathbb{N}$, Representation Learning Alg. $\phi : \mathcal{X} \rightarrow \mathbb{R}^d$

Initialize $\gamma = 1, \Delta \leftarrow \delta, \mathcal{Q} = [], \text{MinDegree} = 1$

$V \leftarrow \phi(\mathcal{L} \cup \mathcal{U})$

$\text{GraphList} \leftarrow \text{ConstructGraph}(V, \mathcal{L}, \mathcal{Q}, \gamma, \Delta)$

for $b = 1$ **to** budget **do**

$\text{MaxDegree} \leftarrow \max_{v \in V} \left\{ \sum_{G \in \text{GraphList}} \text{deg}_G(v) \right\}$

if $\text{MaxDegree} \leq 2 + \text{len}(\text{GraphList})$ **then**

$\gamma \leftarrow \gamma \cdot \xi$

if $\lfloor \gamma \cdot d \rfloor \leq 1$ **then**

$\Delta \leftarrow \delta * \sqrt{1/d}$

$\gamma \leftarrow 1$

end if

$\Delta \leftarrow \delta \cdot \sqrt{\gamma}$ (By theorem 5.2)

$\text{GraphList.concatenate}(\text{ConstructGraph}(V, \mathcal{L}, \mathcal{Q}, \gamma, \Delta))$

end if

$\text{Query} \leftarrow \underset{v \in V}{\text{argmax}} \left\{ \sum_{G \in \text{GraphList}} \text{deg}_G(v) \right\}$

$\mathcal{U}.remove(\text{Query})$

for $i = 1$ **to** $\text{len}(\text{GraphList})$ **do**

$E \leftarrow \text{GraphList}[i].E$

$\text{Covered} \leftarrow \{v \in V : (\text{Query}, v) \in E\}$

$\text{GraphList}[i].E \leftarrow \{(u, v) \in E : v \notin \text{Covered}\}$

end for

$\mathcal{Q}.append(\text{Query})$

end for

return \mathcal{Q}

Algorithm 2 ConstructGraph

Input: Dataset Embeddings V , Labeled Set $\mathcal{L} \subseteq \mathbb{X}$, Unlabeled Set $\mathcal{U} \subseteq \mathbb{X}$,
Query set \mathcal{Q} , Subspace factor $\gamma \in (0, 1)$

Initialize $d = V.shape[1]$, $IndexSet \leftarrow set(range(d))$, $GraphList \leftarrow []$

for $i = 1$ **to** $\lfloor 1/\gamma \rfloor$ **do**

$S \leftarrow IndexSet.choice(\lfloor \gamma \cdot d \rfloor)$

$IndexSet.remove(S)$

$GraphList.append(G = (V, \{(u, v) \in V \times V : 1 - \cos(u[S], v[S]) < \delta\}))$

end for

for $i = 1$ **to** $len(GraphList)$ **do**

$Covered \leftarrow \{v \in V : \exists u \in \mathcal{Q} \cup \mathcal{L} \text{ such that } (u, v) \in GraphList[i].E\}$

$GraphList[i].E \leftarrow \{(u, v) \in E : v \notin Covered\}$

end for

return GraphList

This algorithm creates multiple subspace-based graphs over the dataset embeddings V . First, it determines the original dimension d and initializes an index set to all coordinate positions. It then repeatedly samples a subset of these coordinates of size $\lfloor \gamma \cdot d \rfloor$, removing those coordinates from the index set to avoid reuse. For each chosen subset S , it constructs a graph by connecting two embeddings u and v with an edge if their cosine distance (i.e., $1 - \cos(u[S], v[S])$) is below a threshold δ . After generating all such graphs, each is pruned by removing edges pointing to any vertex already covered by either a labeled point in \mathcal{L} or a queried point in \mathcal{Q} . The final list of pruned graphs is then returned as *GraphList*.

4.2.1 Computational Complexity Analysis

We analyze here the time and space complexity of the proposed GEOMETRICAL ACTIVE SAMPLING (GAS) algorithm. Let $n = |\mathcal{X}|$ denote the number of samples in the dataset, d the dimension of the latent space onto which data is embedded, and ϕ the representation learning function used to em-

bed the data, i.e., $\phi : \mathcal{X} \rightarrow \mathbb{R}^d$. Let $b \in \mathbb{N}$ denote the query budget, and let $|E|$ represent the number of edges in a constructed graph, which is upper bounded by n^2 .

The initial embedding step involves applying the representation model ϕ to all n data points, which requires $\mathcal{O}(|\phi| \cdot n)$ time, where $|\phi|$ is the cost of computing a single embedding. This step is typically performed once at the beginning.

The main computational cost of the algorithm lies in the repeated invocation of the `CONSTRUCTGRAPH` routine. Each call to this routine constructs a set of graphs based on projections of the latent space and computes pairwise cosine distances between the n embedded points in each projected subspace. Since each projection has dimension proportional to d , the construction of a single graph takes $\mathcal{O}(n^2d)$ time. In the worst case, the algorithm constructs up to b such graphs—one per query—leading to a total worst-case cost of $\mathcal{O}(bn^2d)$. However, empirically, the number of graphs constructed is logarithmic in d , resulting in an average-case complexity of $\mathcal{O}(n^2d \log d)$.

During each query iteration, the algorithm identifies the sample with the highest aggregated degree across all graphs. This operation requires summing degrees, which takes $\mathcal{O}(|E| \cdot g)$ time per iteration, where g is the number of graphs constructed so far. Given that $|E| \leq n^2$ and $g \leq b$, the total cost over all b iterations is $\mathcal{O}(bn^2)$. Similarly, after each query, graph pruning involves removing all edges covered by the newly selected sample, which again requires $\mathcal{O}(|E|)$ operations per graph, or $\mathcal{O}(bn^2)$ overall. In summary, the overall **time complexity** is $\mathcal{O}(bn^2d)$ in the worst case and $\mathcal{O}(n^2d \log d)$ on average.

Regarding space complexity, the dominant factor is the storage of the graphs constructed during the process. Each graph requires up to $\mathcal{O}(n^2)$ space to store its edge set. With g graphs constructed, this yields a worst-case space complexity of $\mathcal{O}(bn^2)$ and $\mathcal{O}(n^2 \log d)$ in the average-case. In addition, the algorithm must store the embeddings of all data points, which requires $\mathcal{O}(nd)$ space. Hence, the total **space complexity** is $\mathcal{O}(nd + bn^2)$ in the worst case, and $\mathcal{O}(nd + n^2 \log d)$ on average.

This analysis highlights that the algorithm is particularly well-suited for scenarios where the query budget b is relatively small, as is typical in low-label Active Learning regimes. Moreover, the logarithmic average case for the number of graphs ensures practical scalability, even in high-dimensional latent spaces.

5 Theoretical Framework

In this chapter, we present the theoretical framework that underlies our algorithm and ensures its explainability, by grounding our method in established results from topological data analysis, Metric Geometry, Metric Embedding Theory, and Spherical codes. We offer a transparent account of how each step is motivated and validated. This foundation not only clarifies the algorithm’s underlying assumptions and design choices, but also provides a principled lens through which its behavior can be rigorously explained.

5.1 Metric Geometry

Assumption 5.1. Throughout the thesis, we assume that the latent space was optimized so that vectors corresponding to images in the original data are distributed approximated uniformly on the \mathbb{R}^d unit sphere. Accordingly, for two vectors $x, y \in \mathbb{R}^d$ representing two images in the latent space where $\theta = \text{cosine-similarity}(x, y)$, we can assume that $\theta \sim \frac{1}{2} + \text{Beta}\left(\frac{d-1}{2}, \frac{d-1}{2}\right)$ as discussed in [33]. Subsequently, by definition $\mathbb{E}[\theta] = 1$ as well as:

$$\mathbb{P}[\theta \leq \delta] = \frac{B\left(\delta - \frac{1}{2}, \frac{n-1}{2}, \frac{n-1}{2}\right)}{B\left(\frac{n-1}{2}, \frac{n-1}{2}\right)} \cdot \left(\frac{\delta^2 - 1}{4}\right)^{\frac{d-3}{2}} \quad (5.1)$$

Given this assumption, we aim to derive an optimal update rule for tuning the threshold δ when generating vertices in the projected subspace.

Lemma 5.2. *Let us assume that $\theta \sim \frac{1}{2} + \text{Beta}\left(\frac{d-1}{2}, \frac{d-1}{2}\right)$ and $\theta' \sim \frac{1}{2} + \text{Beta}\left(\frac{\gamma \cdot d-1}{2}, \frac{\gamma \cdot d-1}{2}\right)$ it holds that:*

$$\mathbb{P}[\theta \leq \delta] = \mathbb{P}[\theta' \leq \mathcal{O}(\sqrt{\gamma} \cdot \delta)]$$

Proof. By simple substitution into the Beta distribution CDF tail probability at eq. (5.1) and we confirm that once we set $\delta' = \mathcal{O}(\sqrt{\gamma} \delta)$, we match the tail probability $\mathbb{P}[\theta \leq \delta]$ for the distribution with parameter d to the tail probability $\mathbb{P}[\theta' \leq \delta']$ for the distribution with parameter $\sqrt{\gamma} \cdot d$. This completes the proof. \square

5.2 Metric Embedding Theory

We start by showing that focusing on a subset of coordinates in the learned embedding does not compromise the core topological information. This property enables us to systematically select points based on their cosine similarity—initially picking points that are at least δ apart—and then refining our sampling at a smaller δ while preserving crucial structural features. Rather than focusing on reducing complexity, the objective is to cover a broader range of examples, thereby capturing more local variations and nuances in the embedding. By reducing the number of coordinates by projection and increasing the sampling resolution at each stage, we maintain the fidelity of the topological information while expanding the diversity of sampled points.

5.2.1 Definitions & Assumptions

Below, We begin by introducing the fundamental distortion concepts from Metric Embedding Theory, along with the assumptions underlying the metric embedding framework used throughout our analysis.

Definition 5.3. Metric Space: A Metric Space is a pair (\mathcal{X}, d) where \mathcal{X} is a set and $d : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}^+$ is a metric, hence a distance function such that it

realizes for all $x, y, z \in \mathcal{X}$:

1. Positivity: $d(x, y) \geq 0$
2. Symmetry: $d(x, y) = d(y, x)$
3. Reflexivity: $d(x, y) = 0 \iff x = y$
4. Triangle Inequality: $d(x, z) \leq d(x, y) + d(y, z)$

Assumption 5.4. Notice that naturally, *cosine – similarity* is not a metric on \mathbb{R}^d as it does not realize the Triangle Inequality for all $x, y, z \in \mathbb{R}^d$. However, it does satisfy the triangle inequality when restricted to the unit sphere $\mathbb{S}^d \subset \mathbb{R}^d$. Therefore, for the purposes of the distortion analysis, we assume that all embeddings are normalized to unit length with respect to the ℓ_2 norm. This assumption has no effect on the distortion results presented in section 5.2.

Assumption 5.5. We propose to project the high-dimensional latent space into a lower-dimensional subspace by retaining only a γ -fraction of its coordinates, where $\gamma \in (\frac{1}{d}, 1)$ (with d being the original dimension of the latent space). As part of our proof for bounding the distortion under this projection, we assume that the cosine similarity of any two distinct embedded images is strictly less than 1. In section 5.1 we will show that the assumption we are making is not overly restrictive.

Definition 5.6. Embedding: Given two metric spaces (X, d_X) and (Y, d_Y) , an injection $f : X \rightarrow Y$ is called an embedding of X into Y .

Definition 5.7 (Stretch and Expected Stretch). Let (X, d_X) and (Y, d_Y) be metric spaces, and let $f : X \rightarrow Y$ be a (possibly randomized) mapping. The *stretch* of a pair $(u, v) \in X \times X$ under f is defined as

$$\text{stretch}_f(u, v) = \max \left\{ \frac{d_X(u, v)}{d_Y(f(u), f(v))}, \frac{d_Y(f(u), f(v))}{d_X(u, v)} \right\}.$$

If f is a randomized embedding, the *expected stretch* of the pair (u, v) is

$$\mathbb{E}_f[\text{stretch}_f(u, v)] = \mathbb{E}_f \left[\max \left\{ \frac{d_X(u, v)}{d_Y(f(u), f(v))}, \frac{d_Y(f(u), f(v))}{d_X(u, v)} \right\} \right].$$

The *distortion* of f is then defined as the maximum stretch across all pairs, or in the randomized case, the maximum expected stretch:

$$\text{distortion}(f) = \sup_{u \neq v \in X} \mathbb{E}_f [\text{stretch}_f(u, v)].$$

This formulation follows standard conventions used in works such as Bartal [3], Fakcharoenphol–Rao–Talwar [14], and Linial, London, and Rabinovich [25].

5.2.2 Representation Learning

The use of latent spaces in Active Learning inherently relies on the assumption that the underlying data manifold can be meaningfully represented as a metric space, where distances between points reflect meaningful notions of similarity or informativeness. This assumption is supported by foundational results in metric embedding theory, most notably Bourgain’s Theorem [27].

which guarantees that any finite metric space with n points can be embedded into a space of dimension $\mathcal{O}(\log(n))$ with only logarithmic distortion $\mathcal{O}(\log^2(n))$.

This implies that, under mild assumptions, the structure of the data can be faithfully preserved in a lower-dimensional latent space—justifying the practice of applying geometric reasoning (such as coverage, diversity, and separation) to learned representations.

Nonetheless, some methods maintain an underlying assumption that the Euclidean distance is meaningful in the learned latent space. This assumption is not always valid in modern representation learning methods of images, as they often optimize embeddings with respect to the cosine similarity [9, 29]. This discrepancy motivates our approach, which considers cosine similarity as the distance function of the representation learning latent space onto which images are embedded. In this manner, for examples, the covering problem can be reformulated as a spherical coding problem rather than a ball-covering problem. Thus, permitting the use of the well-developed theoretical framework of spherical coding for analysis. This perspective aligns more closely with the geometric properties of learned representations.

5.2.3 Embedding Subspace Distortion

we evaluate this method's effectiveness by bounding the distortion introduced when limiting our analysis to only a subset of the learned coordinates.

Lemma 5.8. *Let $\mathcal{R}_d = (\mathbb{R}^d, 1 - \text{cosine-similarity})$. For $d \geq 2$, let $u, v \in \mathcal{R}_d$, where each coordinate of u and v is sampled independently from a distribution \mathcal{X}_i , i.e., $u, v \sim \{\mathcal{X}_i\}_{i=1}^d$ with independent coordinates. Let $\gamma \in \left[\frac{1}{d}, 1\right)$, and let $S_\gamma \sim \mathcal{U}(\{S \subseteq [d] \mid |S| = \lfloor \gamma d \rfloor\})$ be a random subset of coordinates. Define the embedding $f_S(u) = (u_i)_{i \in S}$, which projects onto the selected coordinates. Then the expected distortion of the mapping f_{S_γ} satisfies:*

$$\text{distortion}(f_{S_\gamma}) \triangleq \sup_{u \neq v} \mathbb{E}_{S_\gamma} \left[\text{stretch}_{f_{S_\gamma}}(u, v) \right] \leq \frac{1}{\gamma}$$

Proof. We analyze:

$$\begin{aligned} & \sup_{u \neq v} \mathbb{E}_{S_\gamma} \left[\text{stretch}_{f_{S_\gamma}}(u, v) \right] = \\ & \mathbb{E} \left[\max \left\{ \frac{\langle u, v \rangle}{\|u\| \cdot \|v\|}, \frac{\langle f(u), f(v) \rangle}{\|f(u)\| \cdot \|f(v)\|} \right\} \right]. \end{aligned}$$

(1) Absolute value: Since cosine similarity is in $(-1, 1)$, we can take absolute values safely:

$$\leq \mathbb{E} \left[\max \left\{ \frac{\left| \frac{\langle u, v \rangle}{\|u\| \cdot \|v\|} \right|}{\left| \frac{\langle f(u), f(v) \rangle}{\|f(u)\| \cdot \|f(v)\|} \right|}, \frac{\left| \frac{\langle f(u), f(v) \rangle}{\|f(u)\| \cdot \|f(v)\|} \right|}{\left| \frac{\langle u, v \rangle}{\|u\| \cdot \|v\|} \right|} \right\} \right].$$

(2) Norm inequality: Since $f(u)$ is a projection onto a subset of coordinates

then $\|u\| = \sqrt{\sum_{i=1}^d u_i^2} \geq \sqrt{\sum_{u_i^2 \geq 0 \wedge S \subseteq d} u_i^2} = \|f_S(u)\|$ whereof:

$$\|f(u)\| \leq \|u\| \quad \text{and} \quad \|f(v)\| \leq \|v\|,$$

we get:

$$\left| \frac{\langle u, v \rangle}{\langle f(u), f(v) \rangle} \right| \geq \left| \frac{\langle u, v \rangle / (\|u\| \cdot \|v\|)}{\langle f(u), f(v) \rangle / (\|f(u)\| \cdot \|f(v)\|)} \right|,$$

and similarly for the inverse.

So:

$$\mathbb{E} \left[\text{stretch}_{f_{S_\gamma}}(u, v) \right] \leq \mathbb{E} \left[\max \left\{ \left| \frac{\langle u, v \rangle}{\langle f(u), f(v) \rangle} \right|, \left| \frac{\langle f(u), f(v) \rangle}{\langle u, v \rangle} \right| \right\} \right].$$

(3) Independence and coordinate sampling: Since $u, v \sim \{\mathcal{X}_i\}_{i=1}^d$ with i.i.d. coordinates therefore $\mathbb{E}_{u_1, v_1, u_2, \dots, v_d} [\dots] = \mathbb{E}_{u_1} [\mathbb{E}_{v_1} [\mathbb{E}_{u_2} [\dots \mathbb{E}_{v_d} [\dots]]]]$, and since S is sampled uniformly at random:

$$\mathbb{E}_{S_\gamma} [\langle f(u), f(v) \rangle] = \gamma \cdot \langle u, v \rangle.$$

So the ratio $\langle u, v \rangle / \langle f(u), f(v) \rangle$ has expected value approximately $1/\gamma$, and likewise for the inverse.

Hence, by Jensen's inequality and symmetry:

$$\mathbb{E} \left[\max \left\{ \left| \frac{\langle u, v \rangle}{\langle f(u), f(v) \rangle} \right|, \left| \frac{\langle f(u), f(v) \rangle}{\langle u, v \rangle} \right| \right\} \right] \leq \max \left\{ \frac{1}{\gamma}, \gamma \right\} = \frac{1}{\gamma}$$

since $\gamma \in [1/d, 1)$. Therefore:

$$\mathbb{E}_{S_\gamma} \left[\text{stretch}_{f_{S_\gamma}}(u, v) \right] \leq \frac{1}{\gamma} \quad \Rightarrow \quad \text{distortion}(f_{S_\gamma}) \leq \frac{1}{\gamma}. \quad \square$$

Corollary 5.9. *Let $\mathcal{R}_d = (\mathbb{R}^d, \text{cosine-similarity})$. Under the conditions of Lemma 5.8 it holds that:*

$$\mathbb{E} \left[\text{dist}_{f_{S_\gamma}}(u, v) \right] \leq \frac{1}{\gamma}$$

To rigorously analyze the distortion introduced by the HUGS algorithm, we consider not only the effect of a single recursive step, but the total distortion accumulated across the entire recursion tree. At each level of recursion, a bounded distortion arises due to the limited separation of clusters, which depends on a parameter ξ controlling the shrinkage of scale. Importantly, since the algorithm continues partitioning until a condition involving $\gamma = \xi^k$ is met, the number of recursive levels grows roughly as $1/(1 - \xi)$. As a result, the overall distortion combines the per-level contribution, which scales as $1/\xi^k$, with the number of levels, yielding a total bound of the form $O\left(\frac{\xi^{1-t} - \xi}{1 - \xi}\right)$. In the following analysis, we formalize this expression and show that the value $\xi = \frac{k}{k+1}$ minimizes this upper bound—revealing a precise balance between local separation quality and global recursion depth.

Theorem 5.10 (Total Distortion via Expected Stretch). *Let $\xi \in (0, 1)$ be the shrinkage factor used in a hierarchical embedding algorithm such as HUGS. Suppose that at each recursion level t , the separation parameter is given by $\gamma_t = \xi^{k_t}$, where $k_t > 0$ is a level-dependent constant. Assume the embedding is randomized, and at each level t , the expected stretch between any pair (u, v) introduced by that level is at most $1/\gamma_t = 1/\xi^{k_t}$. If the recursion at level t proceeds as long as $d \cdot \gamma_t \geq 1$, then the total expected distortion across all levels is bounded by*

$$\mathbb{E}[\text{stretch}(u, v)] \leq \frac{\xi^{1-t} - \xi}{1 - \xi}$$

Proof. Let f denote the randomized hierarchical embedding constructed recursively over T levels. The expected stretch of a pair (u, v) under f is defined as

$$\mathbb{E}_f[\text{stretch}_f(u, v)] = \mathbb{E}_f \left[\frac{d_Y(f(u), f(v))}{d_X(u, v)} \right].$$

In recursive embedding algorithms such as those of Bartal [3, 4] and Fakcharoenphol–Rao–Talwar (FRT) [14], the global embedding is constructed as the composition (or union) of random decisions made at multiple scales or levels. At each level t , the space is partitioned into clusters of bounded diameter (typically scaling as ζ^t), and the pair (u, v) may be separated — in which case the distance assigned between them is proportional to the cluster scale at that level.

Let $\mathcal{E}_t(u, v)$ denote the event that u and v are separated at level t . Then the expected distance between $f(u)$ and $f(v)$ can be written as:

$$\mathbb{E}[d_Y(f(u), f(v))] = \sum_{t=0}^{T-1} \mathbb{E} \left[d_Y^{(t)}(f_t(u), f_t(v)) \cdot \mathbf{1}_{\mathcal{E}_t(u, v)} \right],$$

where f_t denotes the embedding contribution at level t , and $d_Y^{(t)}$ is the distance function induced at that level. Since the contributions at different levels are independent and additive in the construction of the full embedding (as in the FRT tree or HUGS coordinate embedding), and since indicator functions are disjoint (a pair is assigned distance at only one level), we can apply linearity of expectation to obtain:

$$\mathbb{E}[d_Y(f(u), f(v))] = \sum_{t=0}^{T-1} \mathbb{E}[d_Y^{(t)}(f_t(u), f_t(v))] \leq \sum_{t=0}^{T-1} \frac{1}{\zeta^{k_t}} \cdot d_X(u, v).$$

Hence, the expected stretch is:

$$\mathbb{E}[\text{stretch}_f(u, v)] = \frac{\mathbb{E}[d_Y(f(u), f(v))]}{d_X(u, v)} \leq \sum_{t=0}^{T-1} \frac{1}{\zeta^{k_t}}.$$

By the harmonic series sum it holds that:

$$\mathbb{E}[\text{stretch}_f(u, v)] \leq \sum_{t=0}^{T-1} \frac{1}{\zeta^{k_t}} = \frac{\zeta^{1-T} - \zeta}{1 - \zeta}$$

This additive decomposition arises because each recursive level contributes independently to the embedding, and for any given pair (u, v) , the separation (and hence nonzero distance) is charged to exactly one level (or at most a small constant number of levels), with each such contribution bounded in expectation. \square

5.3 Spherical Codes

In higher-dimensional spaces, spherical coding becomes more complex due to the exponential growth of the unit sphere's surface area and volume. A crucial factor in this complexity is the threshold for generating new vertices. A low threshold may confine sampling to densely populated regions of the latent space, while a high threshold can rapidly deplete the edges of uncoded images. To mitigate this, we derive the initial δ for our algorithm using Spherical Codes Theory, as outlined next.

Lemma 5.11. *Wyner's Lower Bound [35]:*

$$\left| \mathcal{A}(\mathbb{R}^d, \theta) \right| \geq \frac{\frac{d}{d-1} \cdot \frac{\sqrt{\pi} \cdot \Gamma\left(\frac{d+1}{2}\right)}{\Gamma\left(\frac{d+2}{2}\right)}}{\int_0^{\cos^{-1}(1-\theta)} \sin^{d-2}(\phi) \cdot d\phi} \quad (5.2)$$

This bound provides a theoretical lower limit on the number of spherical caps of angular radius θ required to cover the unit sphere in \mathbb{R}^d . In our setting, this corresponds to the number of regions needed to encode all embedded points in a dataset of size N , under a maximum allowed cosine dissimilarity δ . Letting $\theta = \cos^{-1}(1 - \delta)$, we interpret the right-hand side of eq. (5.2) as a function $f_d(\delta)$ that estimates the minimal number of spherical caps required to cover the space at distortion δ .

Proposed heuristic: As a practical method to determine the initial δ for our algorithm, we propose a heuristic based on Wyner's lower bound. Specifically, we search for the smallest $\delta > 0$ such that the number of spherical caps of angular radius $\theta = \cos^{-1}(1 - \delta)$ needed to cover the unit

sphere is at least equal to the number of dataset points N . That is, we select the smallest δ satisfying:

$$\left| \mathcal{A} \left(\mathbb{R}^d, \cos^{-1}(1 - \delta) \right) \right| \geq N. \quad (5.3)$$

Importantly, the dimension d used in this expression is not the ambient space dimension but an *intrinsic dimension* of the data manifold, estimated using prior research on dimensionality reduction and latent geometry [6, 8, 13, 24]. This ensures that the bound is applied within the effective geometry of the dataset rather than an unnecessarily high-dimensional ambient space.

Implementation details: We numerically evaluate the integral in eq. (5.2) for a range of δ values and compute the corresponding covering lower bound. The smallest δ for which the bound exceeds N is selected and fixed. This ensures that the angular separation induced by δ is small enough to allow for a representation of all N points in distinct spherical regions.

Justification: This approach avoids arbitrary selection of the distortion threshold by linking it to fundamental limits from information theory and spherical geometry. In high dimensions, this becomes particularly meaningful, as the geometry of the sphere leads to concentration of measure and strong constraints on angular resolution. The bound becomes sharpest in the range $\theta \in [\Theta(1/\sqrt{d}), \pi/3]$, where covering and packing arguments provide tight asymptotics.

Summary: By leveraging Wyner’s lower bound and iterating over candidate angular distortions, we obtain a principled method for choosing the smallest δ that guarantees full angular coverage of the dataset. The use of intrinsic dimension further ensures that the bound reflects the true complexity of the data, resulting in a distortion value that is both theoretically grounded and empirically relevant.

6 Results

In this chapter, we present empirical evaluations of our proposed algorithm in comparison with several existing active learning (AL) strategies across a range of scenarios. Our experiments specifically target the challenging low-budget regime, where only a small number of labels are initially available. Because the data is drawn from an unlabeled pool, the early labeled set is often imbalanced and may not include all classes, creating gaps that can hamper training. Nonetheless, our results demonstrate that the proposed approach remains robust under these constraints, consistently performing well despite the class imbalance and missing-class hurdle in the early stages.

6.1 Methodology

In this section, we describe the experimental setup used to evaluate our proposed active learning method. Our methodology is designed to systematically assess the model’s performance across datasets of increasing complexity while maintaining a controlled learning framework. Given our focus on low-budget active learning, where the number of labeled samples is constrained, it is essential to ensure that our experimental design allows for a fair comparison with existing active learning strategies.

6.1.1 Datasets

To evaluate our active learning method, we conduct experiments on CIFAR-10/100 [22] and subsets of ImageNet [12], leveraging their distinct properties to assess model performance across different dataset complexities.

CIFAR-10 and CIFAR-100 contain low-resolution (32×32) images with 10 and 100 classes, respectively. While CIFAR-10 includes broad, well-separated categories, CIFAR-100 introduces finer-grained distinctions, making the task more challenging. These datasets provide a controlled setting for evaluating sample selection, though their limited intra-class variability restricts their realism.

To address this, we also evaluate on ImageNet-100 and ImageNet-200, subsets of the high-resolution (224×224) ImageNet dataset. Training on the full ImageNet dataset, which includes 1000 classes and over 1.2 million images, requires prohibitively large computational resources. Instead, ImageNet-100 and ImageNet-200 provide a computationally feasible alternative that retains the high intra-class variability and complex decision boundaries characteristic of ImageNet. These subsets enable a more realistic assessment of active learning strategies in a setting where feature representations are more structured, and optimal sample selection becomes critical. ImageNet-100 enables direct comparison with CIFAR-100, isolating the effect of *sample complexity*, while ImageNet-200 facilitates the study of *label complexity*, offering more labels and data per class. Together, these datasets allow us to assess scalability and generalization under increasing dataset complexity. A full list of ImageNet subset classes appears in appendix A.

6.1.2 Learning Model Architecture

In the following experiments, we train a fully supervised ResNet-18 [19] on the labeled sets that we query from the previously chosen datasets. For training on CIFAR-10 and CIFAR-100, we employed a ResNet-18 architecture trained for 200 epochs. Optimization was performed using an SGD optimizer with Nesterov momentum 0.9, a weight decay of 0.0003, and a cosine learning rate schedule starting at 0.025. We used a batch size of 100 and applied random cropping and horizontal flipping for data augmentation. We used a validation set ratio of 0.1. For ImageNet-100/200, we only adjusted the base learning rate to 0.01. While ResNet-18 is no longer a SOTA architecture for these datasets, it remains a robust choice for comparing and evaluating active learning (AL) strategies on a fair footing.

6.1.3 Self-Supervised Representation Learning

Throughout the CIFAR-10 and CIFAR-100 experiments, we use a SimCLR [9] embedding to represent each image. We trained SimCLR using the implementation from [34] on CIFAR-10 and CIFAR-100 to extract semantically meaningful features. Specifically, we employed a ResNet-18 model with an MLP projection layer mapping to a 128-dimensional vector, trained for 512 epochs. The training hyperparameters were identical to those used in SCAN. After training, we used the 512-dimensional penultimate layer as the representation space.

Within the ImageNet-100 and ImageNet-200 experiments, we employed the official DINOv2 implementation [29] to extract features from images. Specifically, we used the ViT-S/14 model (Vision Transformer with a small architecture and 14×14 patch size), which was pretrained in a self-supervised manner on ImageNet. We followed the standard feature extraction protocol by utilizing the 384-dimensional L2-normalized penultimate layer as the representation space.

6.1.4 Representation and Dataset-Specific Delta Selection

In the adjacent table (table 6.1), we report the δ values chosen for each dataset, as computed by the spherical code heuristic introduced in section 5.3.

Table 6.1: Values of δ chosen for each dataset

Dataset Name	δ
CIFAR-10	0.24
CIFAR-100	0.18
ImageNet-100	0.24
ImageNet-200	0.22

6.2 Main Results

For CIFAR-10 and CIFAR-100, we compare a wide range of AL methods or optimization criteria to select the query set: (1) Random sampling, (2) Min margin [30], (3) Max entropy [32], (4) Uncertainty [23], (5) CoreSet [31], (6) BALD [21], (7) BADGE [1], (8) ProbCover [36], (9) MaxHerding [2], (10) DCOM [28], (11) Typiclust [16], and (12) our proposed algorithm GAS. A simple explanation regarding the objective of each of the methods is provided in chapter 3.

Table 6.2: Comparison of AL strategies results on CIFAR-10 across low-, mid-, and high-budget settings. Mean \pm SE of test accuracy where we conducted each experiment five times.

Budget	Random	Badge	Bald	Coreset	Dcom	Entropy	GAS
10	15.11 \pm 0.09	14.82 \pm 0.81	12.69 \pm 0.24	17.72 \pm 0.21	20.76 \pm 0.23	15.99 \pm 0.23	19.81 \pm 0.06
20	17.99 \pm 0.09	17.13 \pm 0.22	11.13 \pm 0.16	17.95 \pm 0.29	26.16 \pm 0.21	16.38 \pm 0.10	23.97 \pm 0.30
30	21.43 \pm 0.15	18.73 \pm 0.61	11.90 \pm 0.19	19.82 \pm 0.10	28.12 \pm 0.10	14.59 \pm 0.57	28.00 \pm 0.33
40	19.78 \pm 0.29	20.90 \pm 0.72	13.92 \pm 0.27	22.05 \pm 0.43	28.80 \pm 0.18	15.95 \pm 0.15	30.12 \pm 0.20
50	20.75 \pm 0.12	21.35 \pm 0.65	14.90 \pm 0.51	24.42 \pm 0.24	31.54 \pm 0.14	14.90 \pm 0.50	30.57 \pm 0.17
100	27.70 \pm 0.28	26.16 \pm 0.56	21.17 \pm 0.29	29.72 \pm 0.36	34.28 \pm 0.36	15.02 \pm 0.55	34.56 \pm 0.20
250	34.33 \pm 0.18	34.73 \pm 0.31	24.49 \pm 0.19	35.70 \pm 0.29	40.49 \pm 0.25	19.81 \pm 0.56	40.16 \pm 0.16
500	42.29 \pm 0.28	41.42 \pm 0.16	28.34 \pm 0.73	43.83 \pm 0.14	47.84 \pm 0.27	22.27 \pm 0.74	49.02 \pm 0.27
750	46.65 \pm 0.13	46.02 \pm 0.27	33.53 \pm 0.26	48.91 \pm 0.17	50.88 \pm 0.48	27.04 \pm 0.44	53.33 \pm 0.16
1000	49.24 \pm 0.22	49.61 \pm 0.16	36.53 \pm 0.56	52.96 \pm 0.37	54.02 \pm 0.22	32.56 \pm 0.10	56.42 \pm 0.18
2000	58.44 \pm 0.10	59.95 \pm 0.24	45.59 \pm 0.59	61.29 \pm 0.26	60.31 \pm 0.34	44.48 \pm 0.17	62.85 \pm 0.28
3000	64.03 \pm 0.21	64.49 \pm 0.38	55.58 \pm 0.40	66.62 \pm 0.14	64.80 \pm 0.44	53.04 \pm 0.32	67.87 \pm 0.15
4000	69.50 \pm 0.11	70.54 \pm 0.18	59.10 \pm 0.42	70.91 \pm 0.30	68.16 \pm 0.13	59.66 \pm 0.42	71.59 \pm 0.14
5000	73.69 \pm 0.22	73.70 \pm 0.11	63.97 \pm 0.35	73.20 \pm 0.17	71.56 \pm 0.28	64.79 \pm 0.32	74.42 \pm 0.32

Budget	Random	Margin	Maxherding	Probcover	Typiclust	Uncertainty	GAS
10	15.11 \pm 0.09	20.24 \pm 0.10	16.22 \pm 0.13	19.39 \pm 0.17	15.26 \pm 0.13	13.86 \pm 0.24	19.81 \pm 0.06
20	17.99 \pm 0.09	22.78 \pm 0.21	21.98 \pm 0.22	25.66 \pm 0.20	15.79 \pm 0.11	15.53 \pm 0.11	23.97 \pm 0.30
30	21.43 \pm 0.15	21.16 \pm 0.18	23.41 \pm 0.14	27.73 \pm 0.16	22.37 \pm 0.18	16.16 \pm 0.25	28.00 \pm 0.33
40	19.78 \pm 0.29	20.89 \pm 0.12	25.26 \pm 0.16	29.05 \pm 0.25	23.76 \pm 0.23	16.41 \pm 0.52	30.12 \pm 0.20
50	20.75 \pm 0.12	22.29 \pm 0.24	26.20 \pm 0.17	31.66 \pm 0.15	24.16 \pm 0.18	13.10 \pm 0.13	30.57 \pm 0.17
100	27.70 \pm 0.28	25.95 \pm 0.26	29.43 \pm 0.43	34.45 \pm 0.32	30.55 \pm 0.17	16.48 \pm 0.31	34.56 \pm 0.20
250	34.33 \pm 0.18	34.75 \pm 0.28	39.76 \pm 0.41	39.95 \pm 0.50	37.69 \pm 0.24	16.73 \pm 0.35	40.16 \pm 0.16
500	42.29 \pm 0.28	41.91 \pm 0.17	46.37 \pm 0.22	47.77 \pm 0.25	43.68 \pm 0.21	27.18 \pm 0.21	49.02 \pm 0.27
750	46.65 \pm 0.13	46.01 \pm 0.21	50.53 \pm 0.34	50.49 \pm 0.24	49.40 \pm 0.18	30.40 \pm 0.43	53.33 \pm 0.16
1000	49.24 \pm 0.22	50.30 \pm 0.20	54.29 \pm 0.10	54.03 \pm 0.21	52.79 \pm 0.22	31.67 \pm 0.35	56.42 \pm 0.18
2000	58.44 \pm 0.10	59.70 \pm 0.44	63.32 \pm 0.24	60.78 \pm 0.08	62.17 \pm 0.22	48.51 \pm 0.27	62.85 \pm 0.28
3000	64.03 \pm 0.21	65.52 \pm 0.27	67.76 \pm 0.14	64.38 \pm 0.32	66.47 \pm 0.27	57.13 \pm 0.10	67.87 \pm 0.15
4000	69.50 \pm 0.11	69.91 \pm 0.17	70.94 \pm 0.27	67.87 \pm 0.33	69.89 \pm 0.32	62.57 \pm 0.19	71.59 \pm 0.14
5000	73.69 \pm 0.22	73.44 \pm 0.18	73.21 \pm 0.33	70.74 \pm 0.23	74.05 \pm 0.19	68.18 \pm 0.29	74.42 \pm 0.32

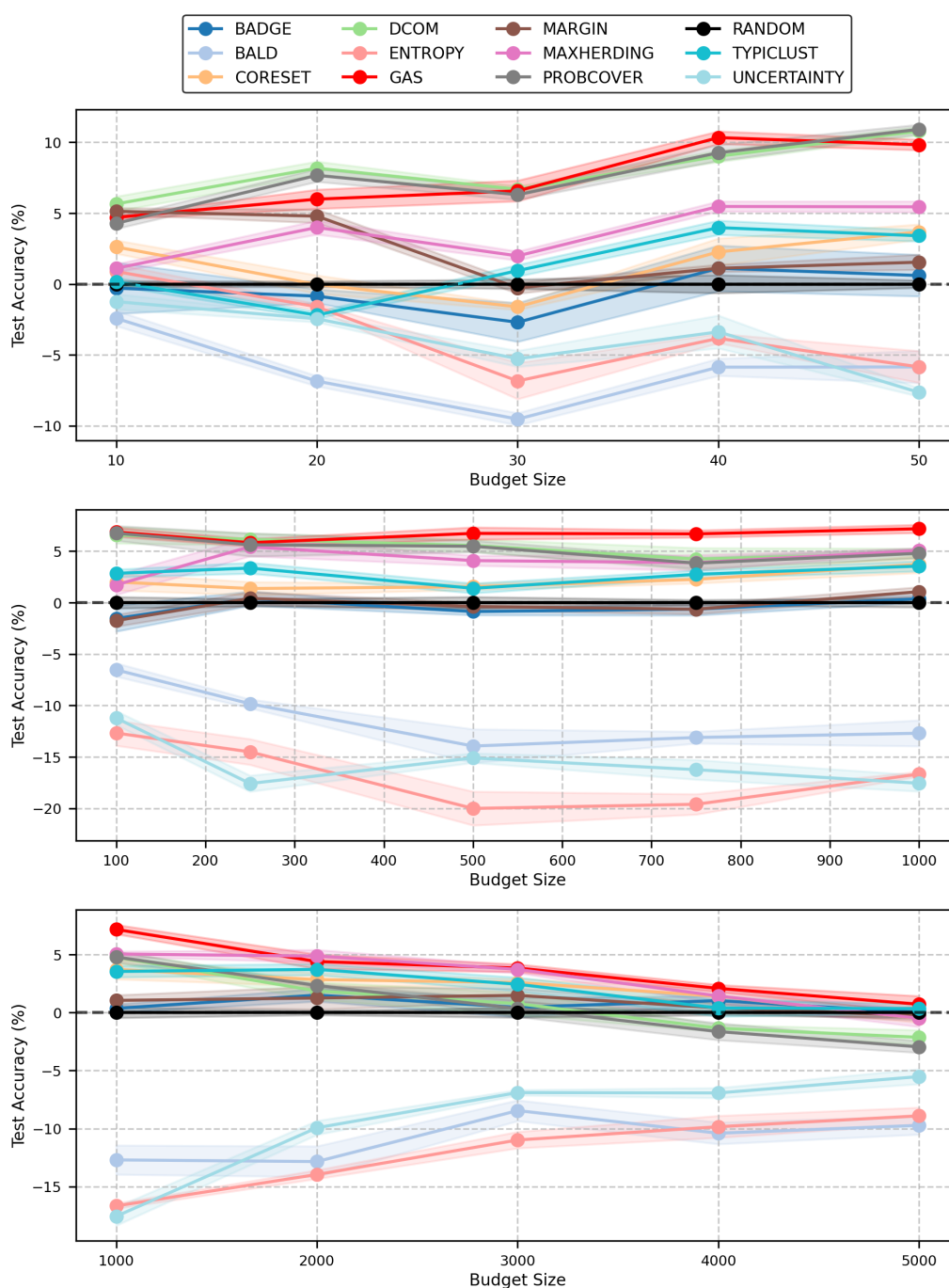


Figure 6.1: Comparison of AL strategies on CIFAR-10 across low-, mid-, and high-budget settings. The x-axis represents the number of labeled samples acquired, while the y-axis denotes the mean difference (from random) in test accuracy. The shaded region around each plot represents the error bars, where we conducted each experiment five times. The value on the plot is the mean, and the shaded area around it indicates the standard deviation.

As shown in fig. 6.1, our GAS algorithm matches the performance of SOTA methods with an improvement in the mid budget region. This result is expected given that CIFAR-10 is a relatively simple dataset, where existing strategies already perform well in the low-budget regime. However, to further analyze our algorithm’s strengths, we extend our comparison to CIFAR-100, a more challenging dataset with a larger number of classes and increased complexity. This allows for a better assessment of our method’s ability to handle more difficult classification tasks.

Table 6.3: Comparison of AL strategies results on CIFAR-100 across low-, mid-, and high-budget settings. Mean \pm SE of test accuracy where we conducted each experiment five times.

Budget	Random	Badge	Bald	Coreset	Dcom	Entropy	GAS
100	3.95 \pm 0.04	5.20 \pm 0.14	3.42 \pm 0.12	7.64 \pm 0.08	8.09 \pm 0.11	1.99 \pm 0.03	8.33 \pm 0.06
200	6.31 \pm 0.11	6.24 \pm 0.17	3.44 \pm 0.04	9.38 \pm 0.06	10.82 \pm 0.09	2.12 \pm 0.06	11.30 \pm 0.09
300	7.77 \pm 0.07	7.69 \pm 0.07	3.74 \pm 0.14	11.16 \pm 0.04	12.33 \pm 0.09	2.98 \pm 0.06	13.17 \pm 0.08
400	8.63 \pm 0.14	8.66 \pm 0.14	3.57 \pm 0.08	12.15 \pm 0.07	13.82 \pm 0.18	3.91 \pm 0.09	14.58 \pm 0.13
500	9.23 \pm 0.16	9.98 \pm 0.17	4.03 \pm 0.10	13.43 \pm 0.06	14.98 \pm 0.09	4.17 \pm 0.08	15.83 \pm 0.15
1000	13.43 \pm 0.06	13.95 \pm 0.13	5.93 \pm 0.04	17.27 \pm 0.15	19.65 \pm 0.13	6.30 \pm 0.08	19.97 \pm 0.11
2000	20.73 \pm 0.09	20.00 \pm 0.22	9.39 \pm 0.15	23.61 \pm 0.17	25.36 \pm 0.28	9.79 \pm 0.17	27.16 \pm 0.08
3000	24.84 \pm 0.12	24.04 \pm 0.16	12.03 \pm 0.11	27.84 \pm 0.16	29.96 \pm 0.07	13.58 \pm 0.25	30.81 \pm 0.08
4000	29.73 \pm 0.22	28.28 \pm 0.29	16.69 \pm 0.10	31.35 \pm 0.14	32.99 \pm 0.21	17.62 \pm 0.21	34.50 \pm 0.13
5000	32.89 \pm 0.12	32.09 \pm 0.15	20.03 \pm 0.08	33.53 \pm 0.17	35.62 \pm 0.27	20.95 \pm 0.08	37.32 \pm 0.22
6000	35.65 \pm 0.14	35.00 \pm 0.37	24.47 \pm 0.11	37.01 \pm 0.21	37.30 \pm 0.34	24.83 \pm 0.21	39.36 \pm 0.23
7000	38.01 \pm 0.21	37.89 \pm 0.21	28.21 \pm 0.13	39.24 \pm 0.20	38.48 \pm 0.15	27.52 \pm 0.08	41.21 \pm 0.29
8000	40.71 \pm 0.08	40.08 \pm 0.12	31.14 \pm 0.10	41.70 \pm 0.13	40.51 \pm 0.15	29.88 \pm 0.28	42.70 \pm 0.14
9000	42.66 \pm 0.20	41.95 \pm 0.19	33.62 \pm 0.15	43.25 \pm 0.09	41.83 \pm 0.25	32.08 \pm 0.21	44.28 \pm 0.18

Budget	Random	Margin	Maxherding	Probcov	Typiclust	Uncertainty	GAS
100	3.95 \pm 0.04	4.42 \pm 0.08	7.07 \pm 0.03	8.20 \pm 0.08	6.22 \pm 0.05	2.00 \pm 0.04	8.33 \pm 0.06
200	6.31 \pm 0.11	5.89 \pm 0.08	9.06 \pm 0.09	11.06 \pm 0.09	8.83 \pm 0.03	2.55 \pm 0.04	11.30 \pm 0.09
300	7.77 \pm 0.07	6.72 \pm 0.06	11.51 \pm 0.11	12.47 \pm 0.07	10.47 \pm 0.05	3.46 \pm 0.13	13.17 \pm 0.08
400	8.63 \pm 0.14	7.82 \pm 0.10	12.32 \pm 0.06	13.88 \pm 0.14	12.08 \pm 0.14	3.67 \pm 0.10	14.58 \pm 0.13
500	9.23 \pm 0.16	8.71 \pm 0.12	13.20 \pm 0.10	15.02 \pm 0.12	14.06 \pm 0.05	3.86 \pm 0.07	15.83 \pm 0.15
1000	13.43 \pm 0.06	12.30 \pm 0.19	18.14 \pm 0.06	19.50 \pm 0.25	17.82 \pm 0.08	5.96 \pm 0.15	19.97 \pm 0.11
2000	20.73 \pm 0.09	17.36 \pm 0.10	23.43 \pm 0.17	25.22 \pm 0.11	24.86 \pm 0.19	10.18 \pm 0.12	27.16 \pm 0.08
3000	24.84 \pm 0.12	22.97 \pm 0.19	27.92 \pm 0.09	29.02 \pm 0.10	29.04 \pm 0.17	13.63 \pm 0.13	30.81 \pm 0.08
4000	29.73 \pm 0.22	26.97 \pm 0.07	31.47 \pm 0.20	32.58 \pm 0.19	32.78 \pm 0.20	17.70 \pm 0.18	34.50 \pm 0.13
5000	32.89 \pm 0.12	30.71 \pm 0.21	33.56 \pm 0.22	34.50 \pm 0.12	36.02 \pm 0.16	21.41 \pm 0.19	37.32 \pm 0.22
6000	35.65 \pm 0.14	33.83 \pm 0.24	35.47 \pm 0.21	36.85 \pm 0.23	38.11 \pm 0.11	24.38 \pm 0.11	39.36 \pm 0.23
7000	38.01 \pm 0.21	36.74 \pm 0.30	36.53 \pm 0.09	37.79 \pm 0.09	40.55 \pm 0.04	27.31 \pm 0.27	41.21 \pm 0.29
8000	40.71 \pm 0.08	38.77 \pm 0.25	38.57 \pm 0.08	40.26 \pm 0.06	42.30 \pm 0.15	30.15 \pm 0.22	42.70 \pm 0.14
9000	42.66 \pm 0.20	41.46 \pm 0.11	39.90 \pm 0.15	41.73 \pm 0.11	44.60 \pm 0.13	33.12 \pm 0.33	44.28 \pm 0.18

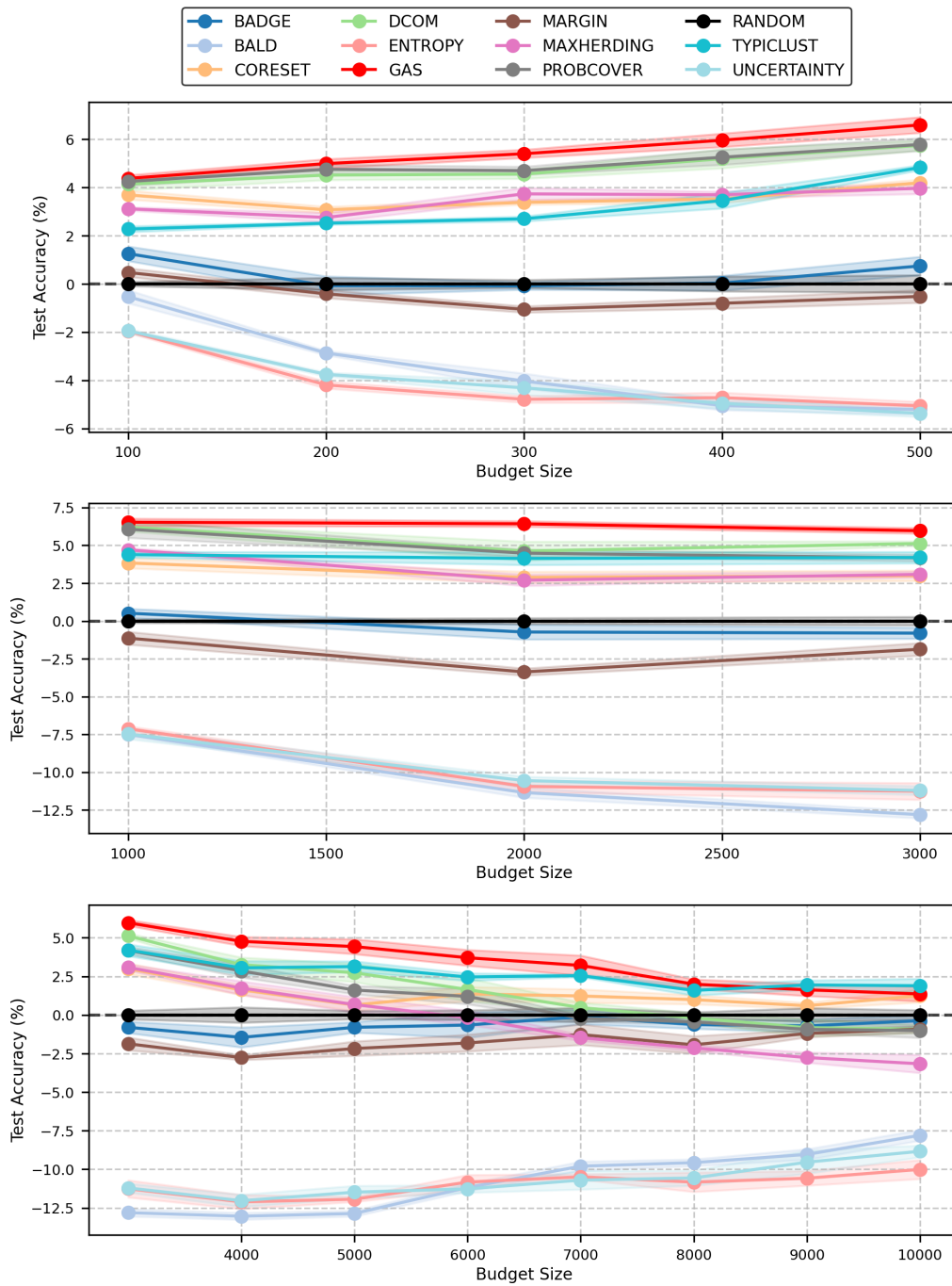


Figure 6.2: Comparison of AL strategies on CIFAR-100 across low-, mid- and high-budget settings. The x-axis represents the number of labeled samples acquired, while the y-axis denotes the mean difference (from random) in test accuracy. The shaded region around each plot represents the error bars, where we ran each experiment five times.

Our method demonstrates superiority against all other algorithms until the high-budget region, where performance differences among approaches diminish. When comparing CIFAR-10 to CIFAR-100 we notice that the only change is label complexity. Given that our algorithm outperforms SOTA algorithms when transitioning to a higher label complexity setting, we conclude that GAS excels at addressing a higher *label complexity*.

Having observed strong performance in the low- and mid-budget ranges for CIFAR-100 in fig. 6.2, we extend our analysis to even more complex datasets such as ImageNet-100/200. We first compare against ImageNet-100, as it permits us to increase the sample complexity while maintaining a controlled class diversity. When evaluating the ImageNet-100/200 subsets, we focus on the best-performing low-budget methods from the methods we have previously compared against Random sampling: (1) Probcover, (2) DCOM, (3) MaxHerding, and (4) our algorithm.

Table 6.4: Comparison of AL strategies results on ImageNet-100 across low-, mid-, and high-budget settings. Mean \pm SE of test accuracy where we conducted each experiment five times.

Budget	Dcom	MaxHerding	Probcover	Random	GAS
500	5.06 \pm 0.17	5.11 \pm 0.07	5.17 \pm 0.05	4.86 \pm 0.10	5.80 \pm 0.04
1000	8.63 \pm 0.12	9.36 \pm 0.11	8.76 \pm 0.13	8.64 \pm 0.08	9.44 \pm 0.15
1500	12.13 \pm 0.15	13.25 \pm 0.17	11.98 \pm 0.12	12.10 \pm 0.04	12.99 \pm 0.17
2000	16.32 \pm 0.17	16.67 \pm 0.23	15.07 \pm 0.07	16.04 \pm 0.18	17.45 \pm 0.13
2500	20.39 \pm 0.15	20.74 \pm 0.29	20.20 \pm 0.21	20.08 \pm 0.24	21.46 \pm 0.10
3000	24.44 \pm 0.24	24.46 \pm 0.12	23.59 \pm 0.13	24.38 \pm 0.24	25.68 \pm 0.15
4000	31.57 \pm 0.25	31.98 \pm 0.08	31.23 \pm 0.37	31.81 \pm 0.29	32.38 \pm 0.38
5000	37.84 \pm 0.40	38.28 \pm 0.19	36.86 \pm 0.19	37.58 \pm 0.22	38.69 \pm 0.32
6000	43.36 \pm 0.46	43.52 \pm 0.12	42.22 \pm 0.32	44.37 \pm 0.32	43.43 \pm 0.18
7000	46.96 \pm 0.22	47.92 \pm 0.36	47.16 \pm 0.29	48.52 \pm 0.26	48.21 \pm 0.24
8000	50.94 \pm 0.20	51.36 \pm 0.18	51.24 \pm 0.15	51.10 \pm 0.30	51.41 \pm 0.14
10000	57.12 \pm 0.13	57.23 \pm 0.26	57.20 \pm 0.06	57.03 \pm 0.24	56.69 \pm 0.04
12000	60.99 \pm 0.27	60.40 \pm 0.18	61.19 \pm 0.22	60.48 \pm 0.12	61.01 \pm 0.16

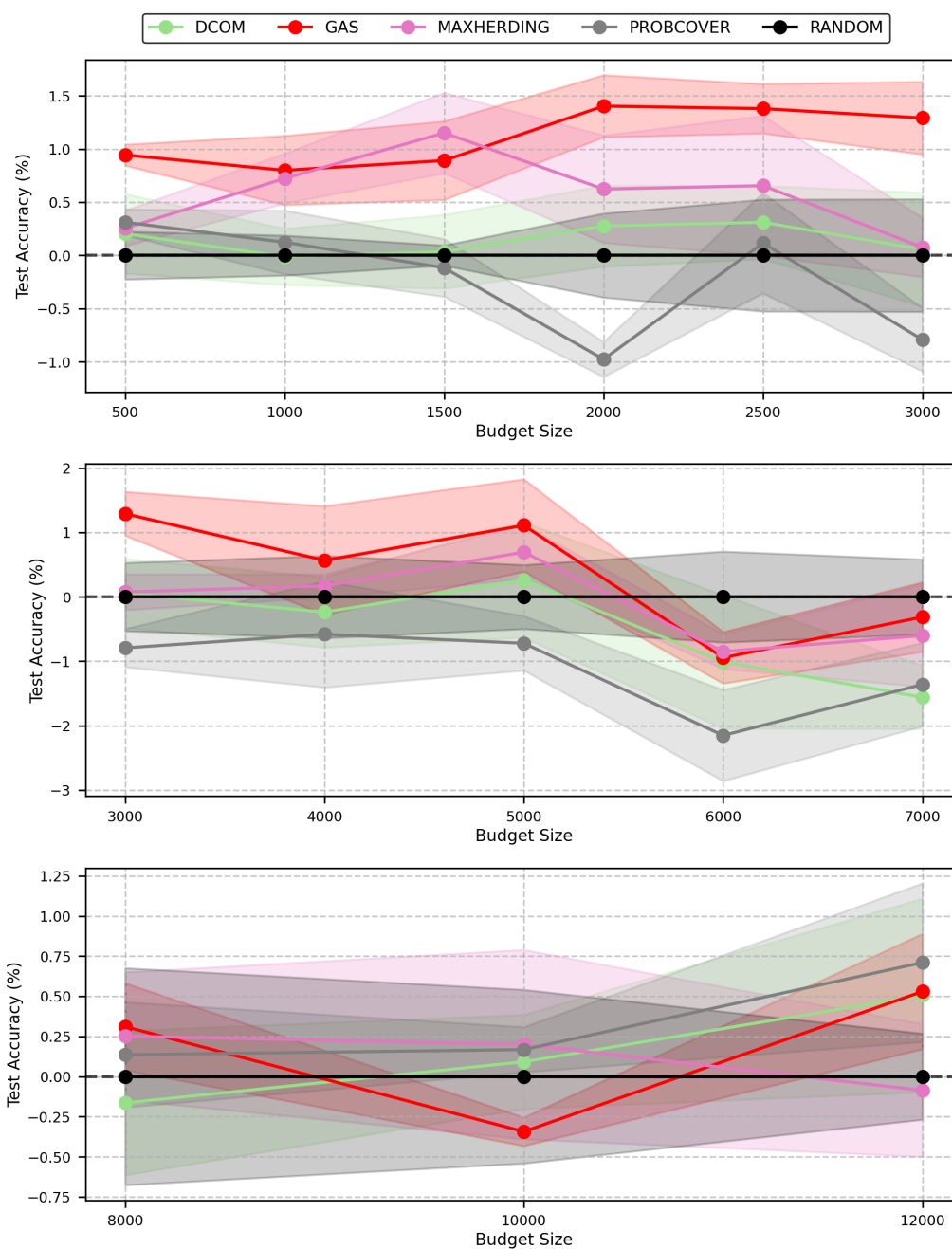


Figure 6.3: Comparison of AL strategies on ImageNet-100 across low-, mid- and high-budget settings. The x-axis represents the number of labeled samples acquired, while the y-axis denotes the mean difference (from random) in test accuracy. The shaded region around each plot represents the error bars, where we conducted each experiment five times.

In the low- and mid-budget settings, our method outperformed all competing algorithms, demonstrating superior generalization to higher-complexity datasets. However, as the budget increased—particularly toward the end of the mid-budget regime and into the high-budget setting—our algorithm, like all others, converged to results comparable to random selection. This suggests that at high budgets, most methods, including ours, reach a saturation point where active selection provides diminishing returns once a substantial portion of the dataset has been labeled. These findings underscore the strength of our approach against an increasing sample complexity while reaffirming the inherent limitations of active learning when abundant labeled data is available.

To further assess the scalability of our method, we extend our analysis to ImageNet-200, examining whether a simultaneous increase in both sample complexity and label complexity impacts performance. However, we did not evaluate the high-budget regime in ImageNet-200 due to computational constraints and the observation from ImageNet-100 that all methods, including ours, performed no better than random selection at high budgets. This ensures that our analysis remains focused on the most relevant budget ranges while maintaining computational feasibility.

Table 6.5: Comparison of AL strategies results on ImageNet-200 across low-, mid-, and high-budget settings. Mean \pm SE of test accuracy where we conducted each experiment five times.

Budget	Dcom	Herding	Probcov	Random	GAS
1000	3.69 \pm 0.04	3.32 \pm 0.06	3.43 \pm 0.04	3.91 \pm 0.05	3.81 \pm 0.08
2000	6.81 \pm 0.11	6.52 \pm 0.07	6.70 \pm 0.11	6.82 \pm 0.09	7.15 \pm 0.08
3000	10.04 \pm 0.09	10.59 \pm 0.08	10.34 \pm 0.05	10.59 \pm 0.06	10.84 \pm 0.09
4000	14.13 \pm 0.08	14.16 \pm 0.15	13.79 \pm 0.15	14.71 \pm 0.21	15.23 \pm 0.17
5000	18.94 \pm 0.10	18.90 \pm 0.19	18.75 \pm 0.17	18.99 \pm 0.13	19.00 \pm 0.30
6000	23.36 \pm 0.04	23.78 \pm 0.13	23.47 \pm 0.24	22.65 \pm 0.14	23.48 \pm 0.15
8000	31.89 \pm 0.11	32.46 \pm 0.22	32.51 \pm 0.41	31.26 \pm 0.19	32.19 \pm 0.15
10000	38.20 \pm 0.10	38.21 \pm 0.28	39.00 \pm 0.28	38.26 \pm 0.01	39.25 \pm 0.19
12000	43.37 \pm 0.13	43.54 \pm 0.17	44.26 \pm 0.12	43.44 \pm 0.08	44.08 \pm 0.11

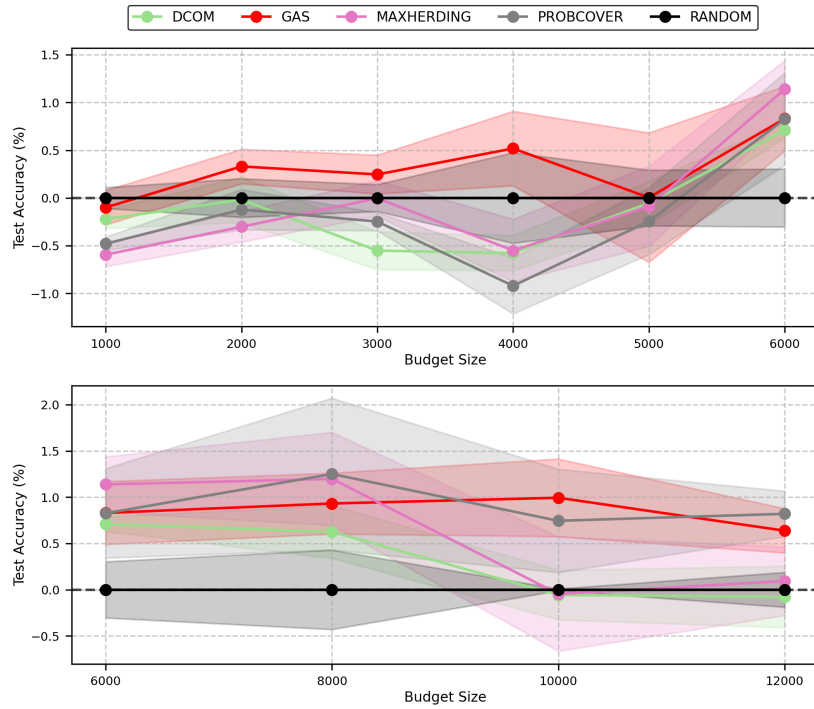


Figure 6.4: Comparison of AL strategies on ImageNet-200 across low-, and mid-budget settings. The x-axis represents the number of labeled samples acquired, while the y-axis denotes the mean difference (from random) in test accuracy. The shaded region around each plot represents the error bars, where we conducted each experiment five times.

Our results from fig. 6.4 show that our algorithm performs better than the other algorithms in the low and mid-budget settings on ImageNet-200. This proves the algorithm’s effectiveness against an increase in *Sample* and *Label Complexity*. Where other methods struggle against while GAS continues to enhance model performance. However, we note that the high standard error in the results is due to the selection $\zeta = 0.75$ (because of memory resource constraints), which induces a higher distortion than $\zeta = 0.95$. In the ablation study at section 6.3, we will show that a lower standard error is achievable with a greater shrink rate parameter ζ .

6.3 Ablation Study

To validate the design choices of Geometrical Active Sampling (GAS), we conduct an ablation study focusing on our two key hyperparameters: the shrink rate of the subspace to which embeddings are projected at each iteration and the cosine similarity threshold used for sample selection. These parameters play a crucial role in controlling the structure and diversity of the selected samples, ultimately influencing GAS’s effectiveness in the low-to-mid-budget regime.

6.3.1 Shrink Rate Hyperparameter

ζ Trade-of: As we have seen in theorem 5.9 we have that when we project our latent space into a subspace of proportion γ the expected distortion of two random points in that subspace is $\frac{1}{\gamma}$. Given that high distortion might generate disruptive edges or even worse, cause the omission of meaningful edges, one is motivated to pick a minimal γ . Nonetheless, the γ will generate a highly correlative subspace because both of them will share a large proportion of coordinates. Thus, the new subspace will not yield a considerable amount of new edges. Wherefrom, the algorithm will need to construct a substantial amount of graphs from subspaces which comes at a memory cost. As γ is initialized to 1 and the next iteration’s γ is obtained by multiplying it by ζ , careful consideration must be taken when tuning ζ . We conduct a few experiments below in order to study the trade-of.

To empirically analyze the impact of ζ on GAS’s performance, we conduct a series of experiments varying ζ from 0.4 to 1. Since tuning this parameter requires multiple runs across different values, we chose CIFAR-100 as our benchmark dataset, balancing computational efficiency and dataset complexity. ImageNet is computationally prohibitive for an extensive hyperparameter sweep, while CIFAR-10 is too simple to reflect meaningful differences in performance.

We evaluated GAS on budget sizes of 3,000 and 5,000, strategically chosen to cover critical points in the budget spectrum. 3,000 marks the upper edge of the mid-budget regime, where sample selection is still highly constrained, making the effects of ζ more pronounced. However, results in this

regime may be somewhat noisy, as a high number of projections and sampled points are required before the selection process stabilizes. Meanwhile, 5,000 represents the lower edge of the high-budget regime, where GAS approaches the saturation phase but still allows us to observe meaningful variations in performance. We do not examine the low-budget regime, as GAS does not perform a significant number of projections in this setting, meaning the influence of ζ would be minimal and results might be dominated by noise. This setup ensures that we focus on the most informative budget ranges, where the balance between distortion and redundancy plays a crucial role in performance.

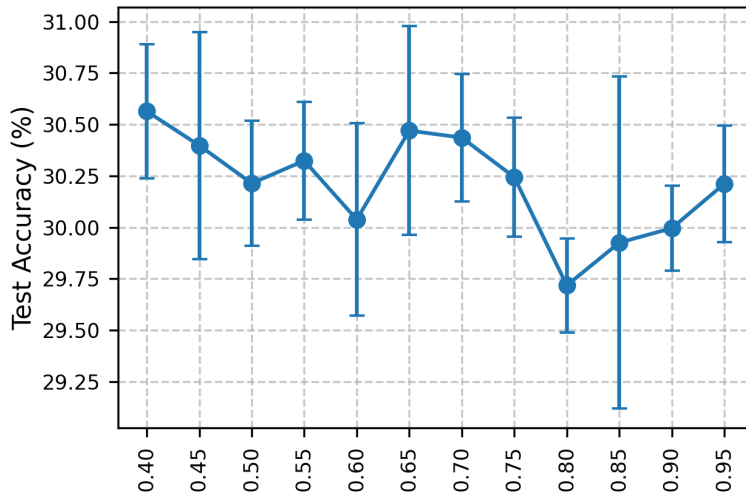


Figure 6.5: Ablation study on the effect of ζ on test accuracy while sampling with GAS with a budget size of 3000. Each point represents the mean accuracy across five runs, with standard deviation error bars.

For the budget size of 3,000, the experiment results indicate that accuracy remains largely consistent across all ζ values below 0.8 across multiple runs. In this range, differences in performance are minimal, suggesting that the method is relatively robust to the choice of ζ . To further evaluate the impact of ζ in a less constrained setting, we conduct an additional experiment at a budget size of 5,000. This budget represents the lower edge of the high-budget regime, where the selection process begins to approach saturation but still allows for meaningful differentiation between parameter choices. At this stage, the influence of ζ is expected to be less pronounced than in the mid-budget regime, as the active learning process has already accumulated a substantial number of labeled samples. However, since ζ

controls the rate at which subspaces shrink, its tuning remains crucial for balancing computational efficiency and selection diversity. By analyzing how accuracy and variance behave across different values of ζ , we aim to assess whether the previously identified optimal range remains valid in this higher-budget setting.

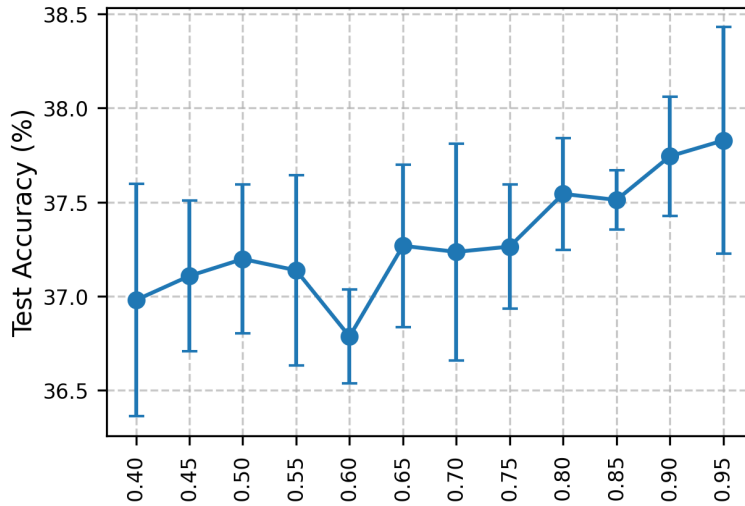


Figure 6.6: Ablation study on the effect of ζ on test accuracy while sampling with GAS with a budget size of 5000. Each point represents the mean accuracy across five runs, with standard deviation error bars.

The results of the experiment at a budget size of 5,000 suggest that accuracy tends to increase for ζ values above 0.8. However, this trend must be considered alongside the results from the 3,000-budget experiment, where $\zeta > 0.8$ yielded lower mean accuracy and significantly larger standard deviations, indicating instability. In contrast, $\zeta = 0.75$ delivered consistently strong performance across both settings, with solid accuracy and narrow error bars. This supports our selection of $\zeta = 0.75$ as a balanced choice between the two regimes—providing robustness, competitive accuracy, and stable behavior across varying budget sizes. Furthermore, from a computational standpoint, lower ζ values are preferable: smaller ζ reduces the overlap between consecutive subspaces, limiting the number of projections and thereby decreasing runtime and memory consumption. This advantage becomes especially relevant in the high-budget regime, where the overhead introduced by $\zeta > 0.8$ can be substantial. As discussed in theorem 5.10, lower ζ values also entail higher distortion due to reduced correlation between consecutive subspaces. Nonetheless, our findings suggest that $\zeta = 0.75$ strikes a favorable trade-off—achieving efficient and

stable performance without compromising accuracy.

6.3.2 Cosine Similarity Threshold Hyperparameter

In section 5.3, we discussed the theoretical motivation behind the selection of δ and the methodology used to determine its optimal value across different datasets. Here, we complement that analysis with an empirical ablation study to further validate our choice. Since δ governs the selection process by controlling the cosine similarity threshold, its tuning directly affects the diversity and informativeness of the selected samples. To assess the impact of different values of δ , we conduct a set of experiments, systematically varying δ in the range of 0.02 to 0.30 while keeping ζ fixed at 0.75. As in the previous ablation study (section 6.3.1), we test on CIFAR-100 with budget sizes of 3,000 and 5,000, ensuring that we evaluate the effect of δ at both the upper edge of the mid-budget regime and the lower edge of the high-budget regime. These budgets provide a meaningful contrast: in the mid-budget regime, sample selection remains constrained, making the impact of δ more pronounced, whereas in the high-budget regime, we assess whether tuning remains relevant as selection approaches saturation. This study aims to determine whether our chosen δ provides near-optimal performance in practice. In doing so, we demonstrate the robustness of our method and ensure that the selected δ value generalizes well across different budget regimes.

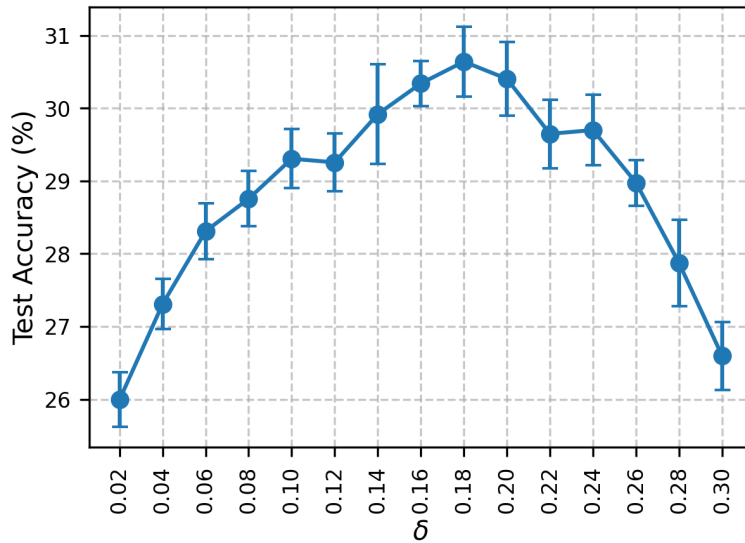


Figure 6.7: Ablation study on the effect of the parameter δ on test accuracy while sampling with GAS a set of points with a budget of 3000. Each point represents the mean accuracy across 5 runs, with error bars indicating the standard deviation.

The results of the experiment at a budget size of 3,000 indicate that the highest accuracy was achieved at $\delta = 0.18$. Notably, this aligns with the δ value we previously selected for CIFAR-100 based on section 5.3 further validating our choice. The fact that the empirical results confirm the theoretical selection reinforces the robustness of our tuning methodology.

Both outcomes suggest that $\delta = 0.18$ effectively balances the trade-off between maintaining diversity in the selected samples and ensuring sufficient coverage of the latent space ensuring that the queried set is neither overly dense (which can lead to redundancy) nor too sparse (which might omit relevant image variations) as previously mentioned in section 4.1.1.

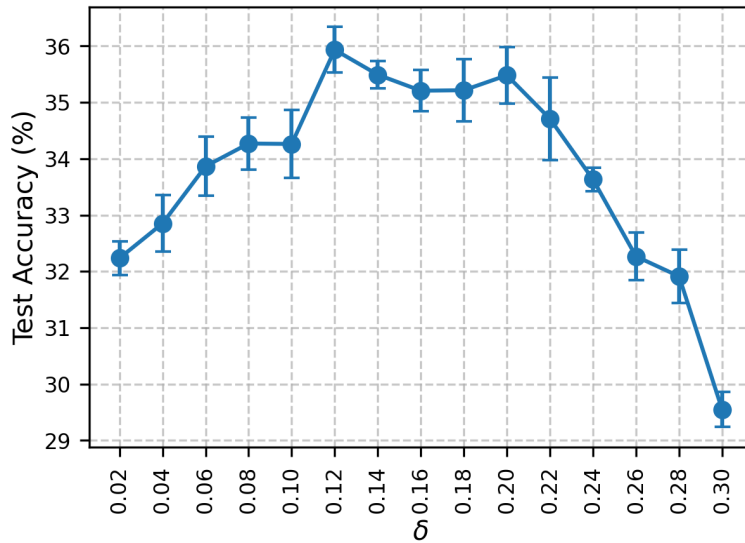


Figure 6.8: Ablation study on the effect of the parameter δ on test accuracy while sampling with GAS a set of points with a budget of 5000. Each point represents the mean accuracy across 5 runs, with error bars indicating the standard deviation.

The results of the experiment at a budget size of 5,000 show that accuracy reaches a plateau for δ values in the range of 0.12 to 0.20, all yielding comparable high performance. This suggests that, in the high-budget regime, the exact choice of δ within this range has a reduced impact on performance, as the selection process begins to saturate and additional labeled samples diminish the influence of the cosine similarity threshold. The observed plateau further supports the robustness of our previously selected value of $\delta = 0.18$, which falls within the optimal range identified in this experiment. Given that our method is designed for the low- and mid-budget regimes, where selection constraints are more pronounced, the fact that $\delta = 0.18$ remains within the high-performing range at a larger budget reinforces its generalization ability. These findings indicate that while tuning δ is crucial in the low- and mid-budget regimes, it becomes less sensitive in the high-budget regime, where sample selection is inherently more forgiving.

7 Conclusion

We study the problem of Active Learning (AL) in the low-budget regime, where annotation resources are extremely limited and coverage of the representation space is critical. Our approach is motivated by theoretical frameworks from metric geometry, metric embedding theory, and spherical coding. Specifically, we model the AL problem as a variant of the spherical code problem, under assumptions on the geometry of self-supervised latent spaces—assumptions that are well-justified in modern representation learning.

We propose Geometrical Active Sampling (GAS), an AL strategy designed to approximate spherical codes in the latent space by iteratively selecting the most “influential” points—those which best cover their neighborhoods across multiple local graphs. To maintain both semantic diversity and computational tractability, we introduce a dynamic expansion mechanism: the neighborhood radius δ is adaptively shrunk using a decay rate ζ , and additional graph layers are constructed only when the current structure fails to differentiate among candidate points. Importantly, the design of the algorithm, including the tuning of δ and the engineering of the decay rate ζ , was guided by the theoretical guarantees developed in our framework and empirically validated through an extensive ablation study, ensuring that our approach remains both principled and effective.

We evaluate GAS empirically in a supervised setting across several benchmark datasets. Results show that GAS consistently outperforms existing state-of-the-art AL methods in the low- and mid-budget regimes. Notably, GAS demonstrates robustness to increases in both Sample Complexity and Label Complexity, maintaining stable performance where other methods deteriorate. This confirms the value of grounding AL strategies in the geometric structure of the representation space, rather than relying solely on model uncertainty or Euclidean heuristics.

Our work suggests that combining representation learning with spherical geometry offers a powerful paradigm for AL, especially in data-efficient settings. While GAS is designed for the low-budget regime, the geometric principles underlying it may extend to broader sampling and learning scenarios. In future work, we aim to explore connections between GAS and other geometric design problems, such as optimal transport, and to investigate its integration with adaptive uncertainty-based criteria in real-time learning systems.

While GAS provides a principled and effective strategy for Active Learning in the low-budget regime, several promising directions remain open for future research:

- **Kernel-based subspace projection:** Instead of linear projections in the latent space, future work could explore the use of kernel methods to induce non-linear similarity measures, potentially capturing more complex structures in the data manifold.
- **Incorporating uncertainty and diversity regularization:** Introducing explicit regularization terms into the selection rule could provide finer control over the balance between uncertainty-driven exploration and geometric diversity, particularly in higher-budget regimes.
- **Dominating set-based selection strategies:** Replacing the current greedy heuristic with algorithms inspired by dominating set theory may yield more globally optimal or theoretically grounded approaches for selecting query points from the coverage graph.
- **Multi-embedding integration:** Leveraging multiple embeddings—e.g., from different self-supervised models—and aligning them via geometric correspondences could enhance robustness, semantic coverage, and generalization, especially in multi-modal or heterogeneous data settings.

Bibliography

- [1] Ash, J. T., Zhang, C., Krishnamurthy, A., Langford, J., and Agarwal, A. (2019). Deep batch active learning by diverse, uncertain gradient lower bounds. *arXiv preprint arXiv:1906.03671*.
- [2] Bae, W., Noh, J., and Sutherland, D. J. (2024). Generalized coverage for more robust low-budget active learning. In *European Conference on Computer Vision*, pages 318–334. Springer.
- [3] Bartal, Y. (1996). Probabilistic approximations of metric spaces and its algorithmic applications. In *37th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 184–193.
- [4] Bartal, Y. (1998). On approximating arbitrary metrics by tree metrics. In *Proceedings of the 30th Annual ACM Symposium on Theory of Computing (STOC)*, pages 161–168.
- [5] Bengio, Y., Lamblin, P., Popovici, D., and Larochelle, H. (2006). Greedy layer-wise training of deep networks. *Advances in neural information processing systems*, 19.
- [6] Campadelli, P., Casiraghi, E., Ceruti, C., and Rozza, A. (2015). Intrinsic dimension estimation: Relevant techniques and a benchmark framework. *Mathematical Problems in Engineering*, 2015(1):759567.
- [7] Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A. (2021). Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660.
- [8] Carter, K. M., Raich, R., and Hero III, A. O. (2009). On local intrinsic dimension estimation and its applications. *IEEE Transactions on Signal Processing*, 58(2):650–663.

- [9] Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.
- [10] Cho, S. J., Kim, G., Lee, J., Shin, J., and Yoo, C. D. (2024). Querying easily flip-flopped samples for deep active learning. *arXiv preprint arXiv:2401.09787*.
- [11] Cohn, H. and Zhao, Y. (2014). Sphere packing bounds via spherical codes. *Duke Mathematical Journal*, 163(10):1965 – 2002.
- [12] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.
- [13] Denti, F., Doimo, D., Laio, A., and Mira, A. (2022). The generalized ratios intrinsic dimension estimator. *Scientific Reports*, 12(1):20005.
- [14] Fakcharoenphol, J., Rao, S., and Talwar, K. (2004). A tight bound on approximating arbitrary metrics by tree metrics. *J. Comput. Syst. Sci.*, 69(3):485–497.
- [15] Gal, Y., Islam, R., and Ghahramani, Z. (2017). Deep bayesian active learning with image data. In *International conference on machine learning*, pages 1183–1192. PMLR.
- [16] Hachohen, G., Dekel, A., and Weinshall, D. (2022). Active learning on a budget: Opposite strategies suit high and low budgets. *arXiv preprint arXiv:2202.02794*.
- [17] Hamkins, J. (1996). *Design and analysis of spherical codes*. University of Illinois at Urbana-Champaign.
- [18] Hanka, R. and Harte, T. P. (1997). Curse of dimensionality: classifying large multi-dimensional images with neural networks.
- [19] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.
- [20] Huang, S.-J., Jin, R., and Zhou, Z.-H. (2010). Active learning by querying informative and representative examples. *Advances in neural information processing systems*, 23.

- [21] Kirsch, A., Van Amersfoort, J., and Gal, Y. (2019). Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning. *Advances in neural information processing systems*, 32.
- [22] Krizhevsky, A., Hinton, G., et al. (2009). Learning multiple layers of features from tiny images.(2009).
- [23] Lewis, D. D. (1995). A sequential algorithm for training text classifiers: Corrigendum and additional data. In *Acm Sigir Forum*, volume 29, pages 13–19. ACM New York, NY, USA.
- [24] Li, C., Farkhoor, H., Liu, R., and Yosinski, J. (2018). Measuring the intrinsic dimension of objective landscapes. *arXiv preprint arXiv:1804.08838*.
- [25] Linial, N., London, E., and Rabinovich, Y. (1995). The geometry of graphs and some of its algorithmic applications. *Combinatorica*, 15(2):215–245.
- [26] Mahmood, R., Fidler, S., and Law, M. T. (2021). Low budget active learning via wasserstein distance: An integer programming approach. *arXiv preprint arXiv:2106.02968*.
- [27] Matoušek, J. (1996). On the distortion required for embedding finite metric spaces into normed spaces. *Israel Journal of Mathematics*, 93(1):333–344.
- [28] Mishal, I. and Weinshall, D. (2024). Dcom: Active learning for all learners. *arXiv preprint arXiv:2407.01804*.
- [29] Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al. (2023). Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*.
- [30] Scheffer, T., Decomain, C., and Wrobel, S. (2001). Active hidden markov models for information extraction. In *International symposium on intelligent data analysis*, pages 309–318. Springer.
- [31] Sener, O. and Savarese, S. (2017). Active learning for convolutional neural networks: A core-set approach. *arXiv preprint arXiv:1708.00489*.
- [32] Shannon, C. E. (1948). A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423.

- [33] Smith, I., Ortmann, J., Abbas-Aghababazadeh, F., Smirnov, P., and Haibe-Kains, B. (2023). On the distribution of cosine similarity with application to biology. *arXiv preprint arXiv:2310.13994*.
- [34] Van Gansbeke, W., Vandenhende, S., Georgoulis, S., Proesmans, M., and Van Gool, L. (2020). Scan: Learning to classify images without labels. In *European conference on computer vision*, pages 268–285. Springer.
- [35] Wyner, A. D. (1965). Capabilities of bounded discrepancy decoding. *Bell System Technical Journal*, 44(6):1061–1122.
- [36] Yehuda, O., Dekel, A., Hacoheh, G., and Weinshall, D. (2022). Active learning through a covering lens. *Advances in Neural Information Processing Systems*, 35:22354–22367.

A List of Subsets of the Imagenet classes

For our experiments, we constructed ImageNet-100 and ImageNet-200 subsets based on a predefined list of ImageNet categories [12]. These subsets were used to evaluate our models across different levels of dataset complexity, enabling a structured comparison of representation learning performance.

Below, we provide the label selections for both ImageNet-100 and the additional 100 labels that complete ImageNet-200.

The selected labels for ImageNet-100 are as follows:

n02119789 (kit fox), n02100735 (English setter), n02110185 (Siberian husky), n02096294 (Australian terrier), n02102040 (English springer), n02066245 (grey whale), n02509815 (lesser panda), n02124075 (Egyptian cat), n02417914 (ibex), n02123394 (Persian cat), n02125311 (cougar), n02423022 (gazelle), n02346627 (porcupine), n02077923 (sea lion), n02110063 (malamute), n02447366 (badger), n02109047 (Great Dane), n02089867 (Walker hound), n02102177 (Welsh springer spaniel), n02091134 (whippet), n02092002 (Scottish deerhound), n02071294 (killer whale), n02442845 (mink), n02504458 (African elephant), n02092339 (Weimaraner), n02098105 (soft-coated wheaten terrier), n02096437 (Dandie Dinmont), n02114712 (red wolf), n02105641 (Old English sheepdog), n02128925 (jaguar), n02091635 (otterhound), n02088466 (bloodhound), n02096051 (Airedale), n02117135 (hyena), n02138441 (meerkat), n02097130 (giant schnauzer), n02493509 (titi), n02457408 (three-toed sloth), n02389026 (sorrel), n02443484 (black-footed ferret), n02110341 (dalmatian), n02089078 (black-and-tan coonhound), n02086910 (papillon), n02445715 (skunk), n02093256 (Staffordshire bullterrier), n02113978 (Mexican hairless), n02106382 (Bouvier des Flandres), n02441942 (weasel), n02113712

(miniature poodle), n02113186 (Cardigan), n02105162 (malinois), n02415577 (bighorn), n02356798 (fox squirrel), n02488702 (colobus), n02123159 (tiger cat), n02098413 (Lhasa), n02422699 (impala), n02114855 (coyote), n02094433 (Yorkshire terrier), n02111277 (Newfoundland), n02132136 (brown bear), n02119022 (red fox), n02091467 (Norwegian elkhound), n02106550 (Rotweiler), n02422106 (hartebeest), n02091831 (Saluki), n02120505 (grey fox), n02104365 (schipperke), n02086079 (Pekinese), n02112706 (Brabancon griffon), n02098286 (West Highland white terrier), n02095889 (Sealyham terrier), n02484975 (guenon), n02137549 (mongoose), n02500267 (indri), n02129604 (tiger), n02090721 (Irish wolfhound), n02396427 (wild boar), n02108000 (EntleBucher), n02391049 (zebra), n02412080 (ram), n02108915 (French bulldog), n02480495 (orangutan), n02110806 (basenji), n02128385 (leopard), n02107683 (Bernese mountain dog), n02085936 (Maltese dog), n02094114 (Norfolk terrier), n02087046 (toy terrier), n02100583 (vizsla), n02096177 (cairn), n02494079 (squirrel monkey), n02105056 (groenendael), n02101556 (clumber), n02123597 (Siamese cat), n02481823 (chimpanzee), n02105505 (komondor), n02088094 (Afghan hound), n02085782 (Japanese spaniel), n02489166 (proboscis monkey).

The selected labels for ImageNet-200 are the previous labels of ImageNet-100 we provided with the following classes: n02364673 (guinea pig), n02114548 (white wolf), n02134084 (ice bear), n02480855 (gorilla), n02090622 (borzoi), n02113624 (toy poodle), n02093859 (Kerry blue terrier), n02403003 (ox), n02097298 (Scotch terrier), n02108551 (Tibetan mastiff), n02493793 (spider monkey), n02107142 (Doberman), n02096585 (Boston bull), n02107574 (Greater Swiss Mountain dog), n02107908 (Appenzeller), n02086240 (Shih-Tzu), n02102973 (Irish water spaniel), n02112018 (Pomeranian), n02093647 (Bedlington terrier), n02397096 (warthog), n02437312 (Arabian camel), n02483708 (siamang), n02097047 (miniature schnauzer), n02106030 (collie), n02099601 (golden retriever), n02093991 (Irish terrier), n02110627 (affenpinscher), n02106166 (Border collie), n02326432 (hare), n02108089 (boxer), n02097658 (silky terrier), n02088364 (beagle), n02111129 (Leonberg), n02100236 (German short-haired pointer), n02486261 (patas), n02115913 (dhole), n02486410 (baboon), n02487347 (macaque), n02099849 (Chesapeake Bay retriever), n02108422 (bull mastiff), n02104029 (kuvasz), n02492035 (capuchin), n02110958 (pug), n02099429 (curly-coated retriever), n02094258 (Norwich terrier), n02099267 (flat-coated retriever), n02395406 (hog), n02112350 (keeshond), n02109961 (Eskimo dog), n02101388 (Brittany spaniel), n02113799 (standard poodle), n02095570 (Lakeland terrier), n02128757 (snow leopard), n02101006 (Gordon setter), n02115641 (dingo), n02097209 (standard schnauzer), n02342885 (hamster), n02097474 (Tibetan terrier), n02120079

(Arctic fox), n02095314 (wire-haired fox terrier), n02088238 (basset), n02408429 (water buffalo), n02133161 (American black bear), n02328150 (Angora), n02410509 (bison), n02492660 (howler monkey), n02398521 (hippopotamus), n02112137 (chow), n02510455 (giant panda), n02093428 (American Staffordshire terrier), n02105855 (Shetland sheepdog), n02111500 (Great Pyrenees), n02085620 (Chihuahua), n02123045 (tabby), n02490219 (marmoset), n02099712 (Labrador retriever), n02109525 (Saint Bernard), n02454379 (armadillo), n02111889 (Samoyed), n02088632 (bluetick), n02090379 (red-bone), n02443114 (polecat), n02361337 (marmot), n02105412 (kelpie), n02483362 (gibbon), n02437616 (llama), n02107312 (miniature pinscher), n02325366 (wood rabbit), n02091032 (Italian greyhound), n02129165 (lion), n02102318 (cocker spaniel), n02100877 (Irish setter), n02074367 (dugong), n02504013 (Indian elephant), n02363005 (beaver), n02102480 (Sussex spaniel), n02113023 (Pembroke), n02086646 (Blenheim spaniel), n02497673 (Madagascar cat), n02087394 (Rhodesian ridgeback).