Department of Computer Science
Series of Publications A
Report A-2026-13

Learning under Data Shift: From Distributional Invariance to Structural Adaptation

Shayan Gharib

Doctoral dissertation, to be presented for public examination with the permission of the Faculty of Science of the University of Helsinki in the Main Building, Hall F3017, on 11 September 2026 at 13:00.

University of Helsinki
Finland

Supervisor
Arto Klami, University of Helsinki, Finland

Pre-examiners
Jenni Raitoharju, University of Jyväskylä, Finland
Werner Zellinger, Johannes Kepler University Linz, Austria

Opponent
Jenni Raitoharju, University of Jyväskylä, Finland

Custos
Arto Klami, University of Helsinki, Finland

Contact information

Department of Computer Science
P.O. Box 68 (Pietari Kalmin katu 5)
FI-00014 University of Helsinki
Finland

Email address: info@cs.helsinki.fi
URL: http://cs.helsinki.fi/
Telephone: +358 2941 911

Copyright © 2026 Shayan Gharib
ISSN 1238-8645 (print)
ISSN 2814-4031 (online)
ISBN 978-952-84-3246-3 (paperback)
ISBN 978-952-84-3247-0 (PDF)
Helsinki 2026
Unigrafia

Learning under Data Shift: From Distributional Invariance to Structural Adaptation

Shayan Gharib

Department of Computer Science
P.O. Box 68, FI-00014 University of Helsinki, Finland
shayan.gharib@helsinki.fi

PhD Thesis, Series of Publications A, Report A-2026-13
Helsinki, September 2026, 76+68 pages
ISSN 1238-8645 (print)
ISSN 2814-4031 (online)
ISBN 978-952-84-3246-3 (paperback)
ISBN 978-952-84-3247-0 (PDF)

Abstract

Machine learning has established itself as a pivotal technology, empowering significant progress in fields as diverse as industrial automation and autonomous navigation. However, the success of these systems typically hinges on a "static world" assumption: the belief that the conditions encountered during deployment will perfectly mirror those seen during training. In reality, the world is dynamic. Changes in environment, sensor aging, or even the subjective biases of human annotators create a phenomenon known as data shift, which can cause once-reliable models to fail catastrophically. This thesis explores the fundamental challenge of ensuring that machine learning remains adaptable when faced with such inevitable changes.

We frame this challenge as a progression from *distributional invariance* to *structural adaptation*. In computer vision tasks, we address covariate shift, where changes such as variations in lighting or background alter the appearance of objects without changing their identity. Such variations often lead to performance degradation as the model's learned patterns fail to generalize to new observations. To address this, we propose strategies to compensate for the performance drop by learning representations that are invariant to these environmental fluctuations, effectively teaching models to focus on the essence of an image rather than its environmental artifacts. We then extend this perspective to the "noisy" world of weak supervision, where we interpret the challenge of learning from incomplete labels as a

form of concept shift. By leveraging structural information in the form of label cardinality, we demonstrate how to reconstruct reliable models from biased and limited observations.

The final part of this thesis moves beyond deep-learning based methods to address the physical reality of sensing networks. In structural health monitoring, data shifts are often caused by tangible changes, such as the relocation of a transducer or the gradual accumulation of rust and fouling. For these scenarios, we introduce a self-adaptation framework that utilizes a physics-informed forward simulation to align a model with its altered environment.

In summary, this thesis provides a unified perspective on managing diverse data shifts across a range of practical applications. The proposed strategies demonstrate that successful adaptation is achievable across a spectrum ranging from static environmental shifts in image data to dynamic structural changes in physical sensing networks by actively adapting to changes and ensuring reliable operation in ever-evolving conditions.

Computing Reviews (2012) Categories and Subject Descriptors:

Computing methodologies → Machine learning → Machine learning approaches → Neural networks
Computing methodologies → Artificial intelligence → Computer vision
Applied computing → Physical sciences and engineering

General Terms:
Domain adaptation, Machine learning

Additional Key Words and Phrases:
Weak supervision, Structural health monitoring

Acknowledgements

I would like to express my deepest gratitude to my supervisor, Prof. Arto Klami, for fostering a relaxed working environment, providing invaluable guidance throughout my doctoral studies, and helping me grow as a researcher. My sincere appreciation also goes to Assoc. Prof. Jenni Raitoharju and Dr. Werner Zellinger for examining my thesis and offering insightful feedback. Furthermore, a special thank you goes to all my co-authors; your insights and collaboration made this work possible.

I am also grateful to the Finnish Center for Artificial Intelligence (FCAI), a flagship project of the Research Council of Finland (RCoF), for supporting this research. During this time, I was also privileged to contribute to other RCoF projects, in particular Sustainable Industrial Ultrasonic Cleaning (SIUC) and the Ultrasonic AI-powered Industrial Sensing Platform (UA-ISP). Additionally, I am sincerely thankful to the Nokia Foundation for awarding me the Nokia Scholarship, which served as a great source of encouragement during the final stages of my work.

A heartfelt thank you goes to Prof. Tuomas Virtanen, my Master's thesis supervisor, who was the first to give me the opportunity to step into the world of research. I am deeply grateful for his guidance and collaboration even during my doctoral studies.

I would also like to thank the past and present members of the Multisource Probabilistic Inference (MUPI) group whom I have crossed paths with over the past few years. Thank you for the engaging discussions and the constant exchange of ideas, especially Bernardo and Ville, who were the steadfast pillars of our *Paper Discussion Forum* (PDF) reading club.

Last but certainly not least, I would like to thank the two people to whom I owe the most in my life: my parents, Tahereh and Mohammad. I am especially grateful to my mother, who has been my greatest inspiration from my very first days of school to this day. I would definitely not be standing where I am today were it not for their sacrifices.

Helsinki, August 2026
Shayan Gharib

Contents

List of Publications **ix**

1 Introduction **1**
1.1 Motivation 1
1.2 Problem Statement 4
1.3 Thesis Contributions 5
1.4 Thesis Structure 6

2 Background **7**
2.1 Machine Learning 7
2.1.1 Learning Setups 7
2.1.2 Neural Networks 9
2.2 Distribution Shift 11
2.2.1 Real-World Causes of Shift 11
2.2.2 Categories of Shift 12
2.3 Domain Adaptation 14
2.3.1 Problem Setup and Definitions 14
2.3.2 Theoretical Foundations 15

3 Covariate Shift in Vision Applications **19**
3.1 Explicit Alignment via Geometric Approach 20
3.1.1 Joint Moment Representation via Siegel Embeddings 21
3.1.2 Learning Objective and Optimization Strategy . . . 22
3.1.3 Computational Stability 22
3.1.4 Empirical Validation 23
3.2 Implicit Alignment via Adversarial Consistency Training . . 26
3.2.1 Bridging SSL and UDA 26
3.2.2 Formalization of Interpolation Consistency Training 27
3.2.3 Adversarial Adaptation 28
3.2.4 Method summary 28
3.2.5 Empirical Validation 29

4 Label Observation Shift in Weak Supervision **31**
4.1 The Challenge of Partial Supervision 32
4.2 Hierarchy of Cardinality Information 33
4.3 Cardinality-Aware Single-Label Classification 35
4.3.1 SPMLL Loss 35
4.3.2 Cardinality Statistics Loss 35
4.3.3 Instance Cardinality Loss 36
4.3.4 Integrated SPMLL Algorithms 37
4.4 Experiments 37
4.4.1 Experiment 1: Performance Comparison and Benchmarking 37
4.4.2 Experiment 2: Value of Cardinality Information . . . 38
4.4.3 Experiment 3: Robustness to Estimation Error . . . 40
4.4.4 Discussion 41

5 Covariate Drift in Sensing Networks **43**
5.1 Physical Origins of Covariate Drift 43
5.2 Problem Setup 44
5.3 Differentiable Forward Simulation 45
5.3.1 Trajectory Length Calculation 46
5.3.2 Material Parameterization 47
5.3.3 Differentiable Mask 49
5.3.4 Optimization 50
5.4 Empirical Validation 50
5.4.1 Experimental Details 50
5.4.2 Results 52
5.4.3 Discussion 55

6 Conclusion **59**
6.1 Summary of Findings 59
6.2 Limitations 60
6.3 Outlook 62

References **63**

List of Publications

This thesis is based on the following original publications, referred to as Paper I to Paper IV throughout the text. The author of this thesis is the first author of all four papers. The author's specific contributions to each publication are outlined below:

I. **Shayan Gharib**, Marcelo Hartmann, and Arto Klami. Geometric Moment Alignment for Domain Adaptation via Siegel Embeddings. *Conference on Lifelong Learning Agents (CoLLAs)*, 2026.

    Contribution: The initial method idea emerged from joint brainstorming between the present author and Dr. Hartmann, and the concept was subsequently developed collaboratively by both. The present author formulated the method for domain adaptation. The experimental design was formalized by all the authors. The present author was responsible for the implementations and analysis of the results. The present author drafted the manuscript, and the co-authors contributed to revising the paper.

II. **Shayan Gharib** and Arto Klami. Extending Interpolation Consistency Training for Unsupervised Domain Adaptation. *International Joint Conference on Neural Networks (IJCNN)*, 2023.

    Contribution: The method was originally proposed by the present author and subsequently refined to its final form through collaboration between both authors. The present author was responsible for the experimental design and implementation. Both authors contributed to the writing of the paper.

III. **Shayan Gharib**, Pierre-Alexandre Murena, and Arto Klami. Single-positive Multi-Label Learning with Label Cardinality. *Transactions on Machine Learning Research (TMLR)*, 2025.

Contribution: The method was mainly developed by the present author and Prof. Klami, with contributions from Prof. Murena. The experiments were designed by the present author and Prof. Klami. The present author carried out the implementation and analysis of the results. The paper was written by the present author with contributions from the co-authors.

IV. **Shayan Gharib**, Denys Iablonskyi, Joonas Mustonen, Julius Korsimaa, Petteri Salminen, Burla Nur Korkmaz, Martin Weber, Ari Salmi, Arto Klami. Self-adaptation of Ultrasound Sensing Networks. *Mechanical Systems and Signal Processing (MSSP)*, 224:112214, 2025.

Contribution: The present author, together with Dr. Iablonskyi and Prof. Klami, developed the method. The data was collected by the co-authors. The implementation and analysis of the results were carried out by the present author. The paper was written by the present author with contributions from all co-authors.

Chapter 1

Introduction

The proliferation of sensors in daily life has led to a significant increase in data accumulation. These sensors that are embedded in a vast array of consumer devices, including smartphones, tablets, and wearables, fundamentally transform how data is acquired and processed [1, 2, 3]. Consequently, Machine Learning (ML) algorithms have been widely deployed to leverage this abundance of information across domains ranging from computer vision and audio processing to biomedicine [4, 5, 6]. Deep Learning (DL) models, in particular, currently demonstrate state-of-the-art (SOTA) performance on complex tasks such as object detection and semantic segmentation [7, 8]. However, despite these algorithmic advancements, significant challenges remain regarding the generalization capability and robustness of these models [9]. The aim of this thesis is to address specific challenges related to the reliability of ML models when facing imperfect data and shifting environments. This chapter introduces the thesis by motivating the problem, stating the research objectives, and summarizing the major contributions.

1.1 Motivation

The availability of large-scale annotated datasets was a primary factor contributing to the initial success of supervised DL models. For example, one of the first powerful image classifiers, AlexNet, was trained on a subset of the ImageNet dataset containing over 1.2 million labeled images [10]. However, obtaining high-quality expert annotations for massive datasets is often prohibitively expensive or practically impossible in specialized applications [11, 12, 13]. This bottleneck has motivated the exploration of alternative learning frameworks, including semi-supervised, weakly-supervised, and unsupervised learning [14].

A fundamental limitation of standard ML approaches is that training on data from a specific source does not guarantee generalization to data originating from other sources. A major barrier to the wider adoption of ML models is their lack of robustness and transferability, the ability to adapt learned knowledge across domains and tasks without catastrophic forgetting. This issue is particularly pronounced in real-world deployments where the operating environment differs significantly from the training conditions. These discrepancies, broadly referred to as distribution shifts or dataset shifts, can arise from changes in the data collection pipeline, variations in the physical environment, or the degradation of sensor hardware over time [15].

Example 1: Audio

To illustrate the severity of this problem, consider a sound event detection (SED) system [16]. Suppose a model is trained to detect acoustic events (such as footsteps, alarms, or glass breaking) using audio data collected in a controlled studio environment. The training set consists of clean recordings captured with high-fidelity microphones. Under these conditions, the model may achieve near-perfect accuracy. However, once deployed in a real-world setting, such as a home monitoring system, the performance often degrades largely due to environmental mismatches from hardware and acoustic environment variability to background noise. As a result of these mismatches, a previously high-performing model may fail to detect critical events or suffer from high false-alarm rates.

Example 2: Vision

The problem is equally prevalent in computer vision. Consider a model trained on a standardized dataset of household objects, collected with high-resolution cameras under consistent lighting. When deployed on a mobile robot navigating a cluttered home, the model encounters dim lighting, motion blur, occlusions, and unusual viewing angles. The robot's onboard camera may also possess different optical characteristics than the source camera. This demonstrates how even modest changes in visual conditions can substantially degrade the performance of image classification models [17].

Beyond environmental shifts, many practical applications also face a another hurdle. In domains where full annotation is costly, practitioners

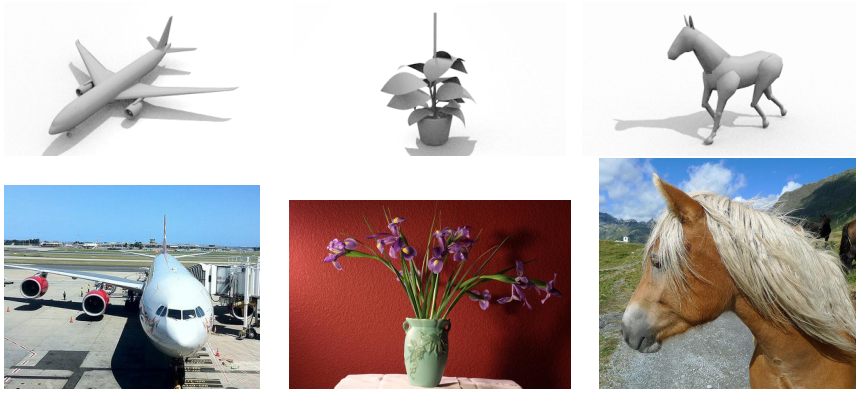


Figure 1.1: Sample of synthetic and real images from the VisDA-2017 dataset.

often rely on weak supervision, such as assigning a single positive label in multi-label tasks – for example labeling an image as containing a "dog" as the primary object without explicitly annotating other objects, such as a "tree" and "car", that may also appear in the background [18, 19]. While this reduces annotation cost, it introduces a shift between the true label distribution and the observed label distribution. If this mismatch is ignored, the model may incorrectly interpret unlabeled objects as absent – for example, treating "tree" and "car" as negative labels even when they are present in the image – which can hinder the model's ability to learn robust visual representations from incomplete or noisy supervision.

Finally, when real-world data is scarce, dangerous, or expensive to collect (e.g., industrial failure cases or medical anomalies), practitioners often turn to simulation. Synthetic data allows for the generation of infinite, perfectly labeled samples. However, this strategy introduces the Sim-to-Real gap [20, 21]: even high-fidelity simulators differ from reality in texture, physics, and noise profiles. Models trained purely on synthetic data often fail in the real world, necessitating domain adaptation methods that can bridge the gap between the synthetic source and the real target. Figure 1.1 illustrates this challenge using the VisDA-2017 dataset [22], highlighting the visual contrast between simulated images and their real-world counterparts, and underscoring the need for the adaptation methods proposed in this thesis.

1.2 Problem Statement

The overarching problem addressed in this thesis is the failure of ML models to generalize when the fundamental assumption of independent and identically distributed (i.i.d.) data is violated. Standard empirical risk minimization assumes that the training data and the test data are drawn from the same underlying distribution [23]. However, as motivated in the previous section, this assumption rarely holds in dynamic real-world environments or under constrained annotation budgets. The generalization failures are broadly referred to as *dataset shift* [15].

The widespread prevalence of these issues necessitates algorithms that can robustly transfer knowledge across different environments. To address these challenges, this thesis focuses on the family of methods collectively referred to as *domain adaptation (DA)* [24].

While a formal mathematical definition of a "domain" is provided in Chapter 2, in the context of this thesis, the adaptation process is fundamentally defined by the relationship between a source and a target. In general, the *source domain* refers to a dataset that is fully annotated and used to train a model for a specific task. The *target domain*, conversely, consists of data (often unlabeled) that serves as a proxy for the environments the model will encounter during deployment. The core objective is to utilize the labeled source data to maximize performance on the target data. When the target data is entirely unlabeled, this paradigm is specifically known as **Unsupervised Domain Adaptation (UDA)**.

However, dataset shift is not a monolithic problem; it manifests differently depending on whether the discrepancy lies in the external environment, the internal sensing mechanism, or the annotation process itself. Consequently, this thesis dissects the broad problem of adaptation into three specific challenge areas:

1. **Covariate Shift in Vision Applications:** This is the classical UDA scenario where the marginal distribution of input features shifts between the source (training) and target (deployment) domains (see Figure 1.1), while the task itself remains constant. The challenge is to align these feature distributions without access to target labels.

2. **Label Observation Shift in Weak Supervision:** Here, the "shift" occurs between the *true* label distribution and the *observed* label distribution. In multi-label settings with partial annotations (e.g., single-positive labels), the observed supervision is a biased proxy of the ground truth. The problem becomes adapting the model to infer the full label set despite this observation gap.

3. **Covariate Drift in Sensing Networks:** Distinct from static covariate shift, this refers to dynamic changes in the input distribution over time within a distributed sensing network. As sensors age or environmental conditions fluctuate, the data distribution "drifts", requiring mechanisms for self-adaptation without human intervention.

1.3 Thesis Contributions

The primary objective of this thesis is to develop robust learning algorithms capable of generalizing across distribution shifts and data imperfections. To this end, the thesis makes the following key contributions to the fields of computer vision, weakly supervised learning, and sensing networks:

- **Geometric Moment Alignment (Paper I):** We introduce a novel UDA framework that aligns the statistical moments of source and target feature distributions within the Symmetric Positive Definite (SPD) geometry of the Siegel embeddings [25]. Unlike standard Euclidean alignment, our method leverages the geometry of SPD matrices to capture higher-order statistical discrepancies, resulting in more robust feature invariance in visual DA tasks.

- **Interpolation Consistency Training for UDA (Paper II):** We propose an extension to Interpolation Consistency Training (ICT) [26] tailored for UDA. By enforcing prediction consistency across interpolated samples from target domains, we effectively smooth the decision boundary in low-density regions. This regularization strategy mitigates the risk of overfitting to the source domain and improves the transferability of learned features to the target domain.

- **Single-label to Multi-Label Adaptation (Paper III):** We address the label observation shift in multi-label classification, where only a single positive label is observed per image [19]. We propose a loss function that incorporates label cardinality constraints (prior knowledge of the expected number of positive labels). This approach effectively corrects the bias introduced by missing annotations, enabling the model to recover performance comparable to fully supervised training without requiring full annotations.

- **Self-Adaptation for Sensing Networks (Paper IV):** To tackle covariate drift in distributed ultrasound sensing networks, we develop a self-adaptation mechanism that can be employed directly on the deployment sites. This method estimates the drift explicitly so that

the network can easily compensate for changes in the data acquisition pipeline (e.g., sensor location changes or environmental fluctuations), ensuring continuous monitoring reliability without manual recalibration.

1.4 Thesis Structure

The remainder of this thesis is organized as follows. Chapter 2 provides the necessary theoretical foundations. It defines the relevant ML paradigms, formalizes the concepts of dataset shift and briefly review the DA methodology. Chapter 3 details the proposed DA techniques in the presence of covariate shift for vision applications, covering both the geometric alignment approach (Paper I) and the interpolation consistency training extension (Paper II). Chapter 4 presents the method for handling label observation shift in the context of single-positive multi-label learning through cardinality constraints (Paper III). Chapter 5 describes the self-adaptation framework developed for robust ultrasound sensing to address covariate drift problem (Paper IV). Finally, Chapter 6 summarizes the findings, discusses the limitations of the proposed methods, and outlines potential directions for future research.

Chapter 2

Background

This chapter presents a background on ML setups followed by an overview on domain shift problems and adaptation methods.

2.1 Machine Learning

ML constitutes a subfield of computer science dedicated to the development of algorithms capable of identifying patterns in data to perform predictive or decision-making tasks. Fundamentally, learning is formulated as the process of inferring an underlying functional relationship from observed data [27]. Formally, ML methods aim to approximate a function $f : \mathcal{X} \mapsto \mathcal{Y}$, parameterized by a vector $\theta \in \Theta$, such that the function maps input variables $\mathbf{x} \in \mathcal{X}$ to target outputs $\mathbf{y} \in \mathcal{Y}$. The specific formulation of this optimization problem depends on the nature of the available data and the task constraints.

2.1.1 Learning Setups

In the following, we briefly review the main learning setups and algorithms in ML relevant to this thesis.

Supervised learning

In the supervised setting, the training dataset $D = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N}$ consists of N input-output pairs. The objective is not merely to learn a function f_θ that minimizes the discrepancy between the predicted output $\hat{\mathbf{y}} = f_\theta(\mathbf{x})$ and the true target $\mathbf{y}$, but also to ensure that the learned function f_θ exhibits high *generalization*: the ability to perform accurately on previously unseen

samples drawn from the same joint distribution $P(\mathbf{x}, \mathbf{y})$. Standard supervised learning is fundamentally grounded in the principle of expected risk minimization $\mathbb{E}_{(\mathbf{x},\mathbf{y})\sim P}[\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y})]$, where $\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y})$ is a loss function to measure discrepancy. However, as the underlying data-generating distribution P is typically unknown, this objective is realized in practice through empirical risk minimization (ERM), which minimizes the average loss over the available training data [28]. Therefore, the optimal parameters $\hat{\theta}$ are obtained by minimizing the empirical risk over the training set:

$$\hat{\theta} := \arg\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(f_\theta(\mathbf{x}_i), \mathbf{y}_i). \tag{2.1}$$

Crucially, this framework relies on the assumption that the training data D and the unseen test data D' are *Independent and Identically Distributed* (i.i.d.), meaning they are drawn from the same joint distribution $P(\mathbf{x}, \mathbf{y})$. As discussed in Section 2.2, this assumption is frequently violated in real-world deployments, leading to the phenomenon of dataset shift. Common supervised tasks include *classification*, where $\mathcal{Y}$ is a discrete set of class labels; *multi-label learning*, where each instance can be assigned multiple labels and $\mathcal{Y} \subseteq \{0, 1\}^K$, with K denoting the total number of possible labels; and *regression*, where $\mathcal{Y} \subseteq \mathbb{R}^n$ is continuous.

Semi-supervised learning

Semi-supervised learning (SSL) addresses scenarios where labeled data is scarce but unlabeled data is abundant. The dataset comprises a small labeled set $\mathcal{D}_l = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{n_l}$ and a much larger unlabeled set $\mathcal{D}_u = \{\mathbf{x}_j\}_{j=1}^{n_u}$ [29]. The goal is to leverage the structural information in $\mathcal{D}_u$, such as data density or cluster assumptions, to improve the generalization of the model trained on $\mathcal{D}_l$ [30]. SSL is particularly valuable when manual annotation is prohibitively expensive, yet data acquisition itself is cheap. Similar to supervised learning, standard SSL methods typically assume that both labeled and unlabeled data originate from the same underlying distribution.

Unsupervised Learning

In unsupervised learning, the dataset consists solely of unlabeled inputs $\mathcal{D} = \{\mathbf{x}_i\}_{i=1}^{N}$. Since no target labels $\mathbf{y}$ are provided, the objective is to discover inherent latent structures or representations within the data distribution $P(\mathbf{x})$. Common tasks include clustering, dimensionality reduction, and density estimation [31]. Rather than minimizing a prediction error, unsupervised objectives often optimize surrogates for data likelihood [32], mu-

tual information [33], or reconstruction fidelity (e.g., in autoencoders) [34], facilitating exploratory data analysis and feature learning.

Weakly Supervised Learning

Weakly Supervised Learning (WSL) covers a broad spectrum of scenarios where the available supervision is strictly less informative than the ground truth required for the task. Unlike SSL where labels are scarce but exact or Unsupervised learning where labels are absent, WSL deals with labels that are noisy, inexact, or incomplete [18]. These forms of weak supervision can be broadly categorized into three types:

- **Inexact Supervision:** The labels are coarser than the target task (e.g., image-level labels for an object detection task) [35].
- **Inaccurate Supervision:** The labels contain noise or errors (e.g., crowdsourced annotations) [36].
- **Incomplete Supervision:** Only a subset of the true labels is provided. This is particularly relevant to *multi-label learning*, where an image may contain multiple objects, but only one is annotated (the single-positive setting) [19, 37].

WSL methods aim to infer the true latent labels or train robust models despite this supervision gap, often by imposing constraints on the label distribution or utilizing robust loss functions.

2.1.2 Neural Networks

A Neural Network (NN) is a computational model composed of interconnected processing units (neurons) organized into layers [38]. These models learn hierarchical representations of data by adjusting the connection strengths (weights) during training. The fundamental building block is the artificial neuron, which computes a weighted sum of its inputs, adds a bias term, and passes the result through a non-linear activation function.

Deep Learning (DL) refers to the use of neural networks with multiple stacked hidden layers, enabling the modeling of complex, high-dimensional functions [39]. A deep neural network serves as a powerful universal function approximator f_θ in the optimization objective defined in Equation 2.1. Modern architectures extend this foundation with specialized inductive biases, such as Convolutional Neural Networks for processing grid-structured data (e.g., images) [40] and Transformers for modeling inter-object dependencies via attention mechanisms [41].

Learning Process

The training of a neural network involves two iterative computational steps: the *forward pass* and the *backward pass*.

Forward Pass In this phase, input data propagates through the network layer-by-layer to generate a prediction. For a layer l with input vector $\mathbf{z}^{(l-1)}$, weight matrix $W^{(l)}$, and bias vector $\mathbf{b}^{(l)}$, the output $\mathbf{z}^{(l)}$ is computed as

$$\mathbf{z}^{(l)} = \sigma\left(\mathbf{W}^{(l)}\mathbf{z}^{(l-1)} + \mathbf{b}^{(l)}\right), \tag{2.2}$$

where $\sigma(\cdot)$ is a non-linear activation function (e.g., ReLU, Sigmoid). The parameters of the network θ consist of the aggregate of all weights and biases across layers. The final output $\hat{\mathbf{y}} = f_\theta(\mathbf{x})$ is compared against the ground truth $\mathbf{y}$ using the loss function $\mathcal{L}$. The loss function measures how different the predictions are with respect to the true labels $\mathbf{y}$.

Backward Pass In the backward pass, or *backpropagation*, the gradient of the loss function with respect to the network parameters, $\nabla_\theta\mathcal{L}$, is computed. In modern deep learning frameworks, this process is formalized as reverse-mode automatic differentiation, which applies the chain rule of calculus recursively to evaluate derivatives from the output layer to the input layer with machine precision [42]. These gradients then guide the optimization of parameters, typically via gradient descent (GD) [43],

$$\theta \leftarrow \theta - \eta\nabla_\theta\mathcal{L}, \tag{2.3}$$

where η is the learning rate.

In practice, computing the gradients over the entire dataset (Batch GD) is computationally prohibitive for large datasets. Therefore, modern training utilizes *mini-batch stochastic gradient descent (SGD)*, where the gradient is estimated using a small random subset (mini-batch) of data at each iteration. This improves computational efficiency and often also enhances generalization through its implicit regularization effects [44]. This stochasticity introduces noise that can help escape shallow local minima but requires careful tuning of the learning rate [39]. To address this, adaptive optimization algorithms, such as *Adam* [45], are commonly employed. These methods dynamically adjust the learning rate for each parameter based on estimates of the first and second moments of the gradients, ensuring faster convergence and improved stability compared to fixed learning rates.

Together, the forward and backward passes enable a neural network to learn complex patterns by repeatedly propagating inputs forward to compute predictions and propagating gradients backward to refine the network parameters.

2.2 Distribution Shift

As established in the previous section, the standard supervised learning framework relies on the minimization of the empirical risk (Equation 2.1) as a proxy for the expected risk. This approach is theoretically grounded in the i.i.d. assumption [28]. Formally, let the train dataset $\mathcal{D}_{tr}$ and the test dataset $\mathcal{D}_{te}$ be drawn from joint probability distributions $P_{tr}(\mathbf{x}, \mathbf{y})$ and $P_{te}(\mathbf{x}, \mathbf{y})$, respectively. The i.i.d. assumption posits that

$$P_{tr}(\mathbf{x}, \mathbf{y}) = P_{te}(\mathbf{x}, \mathbf{y}). \tag{2.4}$$

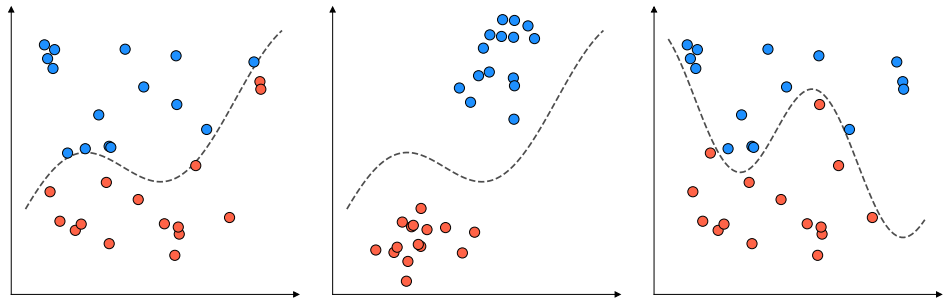
Under this condition, as the number of training samples $N \to \infty$, the empirical risk converges to the expected risk, and the learned model f_θ is expected to generalize well to unseen data. Consequently, standard generalization strategies such as regularization [46], dropout [47], or early stopping [48] are specifically designed to mitigate overfitting when N is finite, ensuring the model generalizes to unseen samples drawn from the *identical* underlying distribution.

However, in most real-world applications, this assumption is violated [49]. When $P_{tr} \neq P_{te}$, the learning process suffers from dataset shift (or domain shift). In such scenarios, relying on these conventional regularization techniques or merely increasing N through additional source data is insufficient to guarantee performance on the target domain; instead, the model may converge to a biased solution that fails to capture the variations present in the test environment.

To categorize different types of shifts, it is necessary to decompose the joint probability distribution $P(\mathbf{x}, \mathbf{y})$. Using the chain rule of probability, the joint distribution can be factorized in two primary ways. For the purpose of this thesis, we primarily utilize the standard view in discriminative learning to formalize the specific shifts addressed in our work, where the decision $\mathbf{y}$ is inferred from the input $\mathbf{x}$, i.e. $P(\mathbf{x}, \mathbf{y}) = P(\mathbf{y}|\mathbf{x})P(\mathbf{x})$ [50].

2.2.1 Real-World Causes of Shift

In practical deployments, distribution shifts arise from a variety of mechanisms. Sample selection Bias occurs when the training data is not a



(a) Original data (b) Covariate shift (c) Concept shift

Figure 2.1: An illustration of primary dataset shifts. (a) Original data operating under standard conditions. (b) Covariate shift, where the input distribution $p(x)$ changes, but the decision boundary $p(y|x)$ remains static. (c) Concept shift, where the underlying decision boundary $p(y|x)$ changes. The dashed gray line represents the decision boundary between the two classes.

representative sample of the operational environment (e.g., medical data collected only from a specific demographic) [51]. Non-stationary environments introduce shifts due to temporal changes, such as sensor degradation or seasonal variations in visual scenes [52]. Furthermore, annotation artifacts can introduce discrepancies between the true label distribution and the labels provided by human annotators, particularly in weakly supervised settings [18, 53].

2.2.2 Categories of Shift

Based on the decomposition of $P(\mathbf{x}, \mathbf{y})$, we define the specific categories of dataset shift addressed in this thesis. These are visually summarized in Figure 2.1.

Covariate Shift

Covariate shift is the most common setup in UDA. It occurs when the distribution of the input features $P(x)$ changes between the training and test domains, while the conditional distribution of the labels remains constant [54]:

$$P_{tr}(\mathbf{x}) \neq P_{te}(\mathbf{x}), \quad \text{but} \quad P_{tr}(\mathbf{y}|\mathbf{x}) = P_{te}(\mathbf{y}|\mathbf{x}). \tag{2.5}$$

This implies that the fundamental relationship between the inputs and outputs is preserved, but the model encounters inputs in regions of the feature

space that were sparsely populated or unseen during training. Papers I and II specifically address this challenge in the context of visual recognition.

Additionally, while mathematically similar to Covariate Shift, *Covariate Drift* (also known as *virtual drift* [55]) explicitly refers to the continuous, time-dependent evolution of the input distribution, often seen in dynamic environments:

$$P_{t_1}(\mathbf{x}) \neq P_{t_2}(\mathbf{x}) \quad \text{for} \quad t_1 < t_2. \tag{2.6}$$

Unlike static covariate shift, which compares two fixed domains, covariate drift implies a non-stationary process where the distribution $P_t(x)$ is a function of time. Paper IV addresses this dynamic challenge through a self-adaptation mechanism.

Concept Shift

Concept shift is formally defined as a change in the conditional distribution of the target variable given the input [50]:

$$P_{tr}(\mathbf{x}) = P_{te}(\mathbf{x}), \quad \text{but} \quad P_{tr}(\mathbf{y}|\mathbf{x}) \neq P_{te}(\mathbf{y}|\mathbf{x}). \tag{2.7}$$

In the literature, this is interpreted as a change in the true underlying semantics of the class (e.g., the definition of "spam" evolving over time or the definition of a "positive medical diagnosis" evolving due to updated clinical guidelines) [56]. However, following the characterization by Webb et al. [57], concept shift also encompasses any change in the posterior class probabilities, referred to as *class drift*.

Paper III specifically addresses a form of annotation-induced concept shift which we refer to it as *label observation shift*. Here, the physical semantics of objects (e.g., a "dog") remain constant, but the *labeling function* mapping inputs to observed labels changes. In our single-positive multi-label setting, the training distribution $P_{tr}(\mathbf{y}_{\text{obs}} \mid \mathbf{x})$ reflects a partial annotation policy in which a single positive label is observed per instance, whereas the test distribution $P_{te}(\mathbf{y}_{\text{true}} \mid \mathbf{x})$ corresponds to the complete multi-label ground truth. We therefore rigorously classify this discrepancy as a shift in the conditional distribution $P(\mathbf{y} \mid \mathbf{x})$, necessitating specialized methods to recover the true labels from partial observations.

A potential secondary effect of such a change in the labeling mechanism is a shift in the marginal distribution of observed labels, $P(\mathbf{y}_{\text{obs}})$. The policy of annotation selection itself is strictly class-agnostic: the observed label $\mathbf{y}_{\text{obs}}$ is assumed to be selected uniformly at random from the true positive set. However, because the probability of sampling a specific label is inversely proportional to the number of true positive labels for that instance,

this selection alters the observed marginal distribution as well. Classes that appear in smaller positive sets are systematically overrepresented in the observations, inducing a marginal shift ($P_{tr}(\mathbf{y}_{\text{obs}}) \neq P_{te}(\mathbf{y}_{\text{true}})$). While the severe mismatch between the observed training conditional $P_{tr}(\mathbf{y}_{\text{obs}} \mid \mathbf{x})$ and the complete test conditional $P_{te}(\mathbf{y}_{\text{true}} \mid \mathbf{x})$ remains the primary challenge, the method proposed in Paper III directly addresses both this dominant conditional shift and the secondary marginal shift.

2.3 Domain Adaptation

Having established the types of distribution shifts, we now turn to the algorithmic framework developed to address them. DA refers to the process of transferring knowledge from a *source* domain to a different but related *target* domain. The landscape of DA is diverse and can be categorized based on the availability of labeled data, the number of distinct domains involved, or access to source data during adaptation.

Common settings include *Semi-supervised DA*, where a small portion of the target data is labeled, allowing the model to leverage both abundant source labels and sparse target supervision [58]. Another variation is *Multi-source DA*, which involves multiple distinct source domains adapting to a single target; here, the challenge lies in determining which sources are most relevant and how to aggregate their knowledge [59]. *UDA* represents a more challenging and practical setting where the target domain is entirely unlabeled, requiring the learner to extract knowledge using only a labeled source dataset and an unlabeled target dataset [60]. More recently, *Source-free DA* has emerged as a privacy-preserving alternative where adaptation is performed using a pre-trained source model without access to the original source data [61].

While these categories are by no means exhaustive, this thesis primarily focuses on the *Single-Source UDA* setting. In this scenario, the objective is to learn a robust classifier solely by aligning the distributions of a single source and a single target, without relying on any direct supervision in the target domain.

2.3.1 Problem Setup and Definitions

We formally define the UDA problem in the context of covariate shift following the learning-theoretic framework established by Ben-David et al. [62] and Zhao et al. [63]. Let $\mathcal{X}$ denote the input space and $\mathcal{Y}$ denote the output space. A domain is defined as a pair $\langle \mathcal{D}, f \rangle$, where $\mathcal{D}$ represents a marginal probability distribution over the inputs $\mathcal{X}$, and $f : \mathcal{X} \to \mathcal{Y}$ is

the labeling function. In the specific context of UDA, we consider a source domain $\langle \mathcal{D}_S, f_S \rangle$ and a target domain $\langle \mathcal{D}_T, f_T \rangle$ defined on the same input space. The fundamental assumption of covariate shift posits that while the marginal distributions differ ($\mathcal{D}_S \neq \mathcal{D}_T$), the conditional relationship between inputs and outputs remains consistent, implying $f_S(\mathbf{x}) = f_T(\mathbf{x})$ for all $\mathbf{x}$.

The learning algorithm is provided with a labeled source dataset $S = \{(\mathbf{x}_i^s, y_i^s)\}_{i=1}^{n_s}$ where samples are drawn i.i.d. from $\mathcal{D}_S$ and labeled according to $y_i^s = f_S(\mathbf{x}_i^s)$. Simultaneously, the learner has access to an unlabeled target dataset $T = \{\mathbf{x}_j^t\}_{j=1}^{n_t}$ drawn i.i.d. from $\mathcal{D}_T$. Given a hypothesis class $\mathcal{H}$ of functions $h : \mathcal{X} \to \mathcal{Y}$ (e.g., $\{0, 1\}$ for binary classification), we define the risk of a hypothesis h with respect to the true labeling function f under a distribution $\mathcal{D}$ as the expected disagreement

$$\varepsilon(h, f) = \mathbb{E}_{\mathbf{x} \sim \mathcal{D}}\big[|h(\mathbf{x}) - f(\mathbf{x})|\big]. \tag{2.8}$$

For clarity, we adopt the shorthand notation $\varepsilon_S(h) := \varepsilon(h, f_S)$ and $\varepsilon_T(h) := \varepsilon(h, f_T)$ to denote the risks measured on the source distribution $\mathcal{D}_S$ and the target distribution $\mathcal{D}_T$, respectively. The ultimate goal of UDA is to select a hypothesis $h \in \mathcal{H}$ that minimizes the target risk $\varepsilon_T(h)$ by leveraging the labeled source data and the unlabeled target data.

2.3.2 Theoretical Foundations

A fundamental question in DA is: *under what conditions can a predictor trained on a source domain generalize to a target domain?* To answer this question, it is necessary to quantify the discrepancy between the two domains in a way that is relevant to the learning task.

For binary classification, let $\mathcal{H}$ be a class of functions $h, h' : \mathcal{X} \to \{0, 1\}$. The symmetric difference hypothesis space is defined as

$$\mathcal{H}\Delta\mathcal{H} = \{\mathbf{x} \mapsto h(\mathbf{x}) \oplus h'(\mathbf{x}) \ : \ h, h' \in \mathcal{H}\}, \tag{2.9}$$

where $\oplus$ denotes the XOR operation. The associated divergence between two distributions $\mathcal{D}_S$ and $\mathcal{D}_T$ is

$$d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_S, \mathcal{D}_T) = 2 \sup_{h, h' \in \mathcal{H}} \left| \Pr_{\mathbf{x} \sim \mathcal{D}_S}[h(\mathbf{x}) \neq h'(\mathbf{x})] - \Pr_{\mathbf{x} \sim \mathcal{D}_T}[h(\mathbf{x}) \neq h'(\mathbf{x})] \right|. \tag{2.10}$$

This divergence can be interpreted as follows. The difference between the source and target distributions is characterized by comparing the disagreement behavior of pairs of hypotheses h and h' across the two domains. The

XOR operation identifies the input regions where the two hypotheses disagree, thereby defining a disagreement region. The $\mathcal{H}\Delta\mathcal{H}$-divergence then searches for the pair of hypotheses whose probability of disagreement differs the most between the source and target domains. In other words, it identifies hypotheses that frequently disagree on the source domain but rarely on the target domain, or vice versa. Building on this notion, Ben-David et al. [62] derived the following finite-sample generalization bound.

Theorem 2.1 (Ben-David et al. [62]) *Let $\mathcal{H}$ be a hypothesis class with VC-dimension d. Given empirical distributions $\widehat{\mathcal{D}}_S$ and $\widehat{\mathcal{D}}_T$ constructed from m samples each, and letting $\widehat{\varepsilon}_S(h)$ denote the empirical source risk of a hypothesis h, then for any $\delta \in (0,1)$, with probability at least $1-\delta$, the following holds for all $h \in \mathcal{H}$:*

$$\varepsilon_T(h) \leq \widehat{\varepsilon}_S(h) + \frac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(\widehat{\mathcal{D}}_S, \widehat{\mathcal{D}}_T) + 4\sqrt{\frac{2d\log(2m) + \log(2/\delta)}{m}} + \lambda, \quad (2.11)$$

where

$$\lambda = \min_{h \in \mathcal{H}} \left(\varepsilon_S(h) + \varepsilon_T(h)\right). \quad (2.12)$$

This bound highlights two key factors governing target performance: the source risk and the divergence between marginal input distributions. This naturally motivates approaches that seek to minimize source error while aligning feature distributions across domains, i.e., addressing *covariate shift*. However, the bound also depends on the joint error term λ, which captures the existence of a hypothesis that performs well on both domains simultaneously. When the source and target labeling mechanisms differ, this term may remain large even if the marginal distributions are well aligned.

To make this dependence explicit, Zhao et al. [63] refined the analysis by separating marginal and conditional discrepancies. Let $\mathcal{H} \subseteq [0,1]^{\mathcal{X}}$ be a class of real-valued hypotheses. The induced hypothesis class is then defined as

$$\widetilde{\mathcal{H}} = \left\{ \mathbf{x} \mapsto \mathbb{I}\big(|h(\mathbf{x}) - h'(\mathbf{x})| > t\big) \ : \ h, h' \in \mathcal{H},\ t \in [0,1] \right\}, \quad (2.13)$$

where $\mathbb{I}(\cdot)$ is the indicator function.

Theorem 2.2 *For any hypothesis $h \in \mathcal{H}$,*

$$\begin{aligned} \epsilon_T(h) \leq \epsilon_S(h) + d_{\widetilde{\mathcal{H}}}(\mathcal{D}_S, \mathcal{D}_T) \quad (2.14) \\ + \min\Big\{ \mathbb{E}_{\mathbf{x}\sim\mathcal{D}_S}\big[|f_S(\mathbf{x}) - f_T(\mathbf{x})|\big],\ \mathbb{E}_{\mathbf{x}\sim\mathcal{D}_T}\big[|f_S(\mathbf{x}) - f_T(\mathbf{x})|\big] \Big\}. \end{aligned}$$

Theorem 2.2 explicitly separates the effects of marginal and conditional shift. The second term on the right hand side of the equation reflects discrepancies in the input distributions, while the third term quantifies the difference between the source and target labeling functions.

If the conditional distributions are identical, i.e., $f_S = f_T$, the final term vanishes, and minimizing marginal discrepancy is sufficient for adaptation. However, when the labeling functions differ, aligning feature distributions alone is not sufficient to guarantee low target error. In such cases, the residual risk is fundamentally controlled by the discrepancy between the labeling mechanisms across domains.

This decomposition provides a more precise interpretation of the term λ in Theorem 2.1, revealing it as a consequence of conditional shift. This perspective is particularly relevant for the setting studied in this thesis, where the source and target domains may exhibit covariate shift and conditional shift. The results above therefore motivate methods that address not only feature alignment, but also the preservation, retrieval, or correction of label semantics across domains.

Chapter 3

Covariate Shift in Vision Applications

The transition from curated training datasets to unconstrained real-world deployments is frequently hindered by the discrepancy between training and testing data distributions. In computer vision, this phenomenon is pervasive; even when models are trained on diverse images, they often encounter significant covariate shifts arising from variations in lighting, hardware-induced artifacts, or synthetic-to-real domain gaps. As established in the theoretical framework of Chapter 2, the primary objective in addressing covariate shift is to minimize the target risk by mitigating the distribution mismatch without the aid of target-domain labels.

This section presents two distinct yet complementary methodologies for achieving domain-invariant representations in the presence of covariate shift. We approach the problem through the lens of Ben-David's generalization bound (Equation 2.11), which identifies the domain divergence and source-domain error as the key components to minimize.

The first methodology, which we refer to as **explicit alignment**, focuses on directly minimizing the statistical discrepancy between domains. In practice, this requires learning a representation in which the discrepancy between domains, measured by an appropriate distance metric, is minimized. Conventional methods often rely on standard Euclidean distances between summaries of the distributions, such as low-order moments. In contrast, we propose a principled geometric framework. By leveraging Siegel embeddings [25], we map the first- and second-order moments of the distributions into a Symmetric Positive Definite (SPD) manifold, where we can employ geometry-aware distances to achieve a more faithful and mathematically grounded alignment.

The second methodology addresses the problem through **implicit alignment** and decision boundary refinement. We explore the synergy between SSL and UDA, building on the principle of consistency regularization. By extending interpolation consistency training [26], we encourage the model to produce stable predictions in low-density regions of the target domain. However, as consistency training alone can be detrimental under significant distribution shifts, we integrate an adversarial learning mechanism using a Gradient Reversal Layer (GRL) [60] to enforce domain-invariant features while simultaneously smoothing the classifier's predictions.

Together, these approaches demonstrate that robust vision adaptation can be achieved either by explicitly measuring and reducing geometric distance or by implicitly regularizing the model's behavior across domains using adversarial and consistency-based constraints.

3.1 Explicit Alignment via Geometric Approach

The first methodological pillar of this thesis addresses covariate shift by explicitly minimizing the statistical discrepancy between the source and target domains in a latent representation space $\mathcal{Z}$. To reach this space, we define a neural network-based feature encoder $e_\theta : \mathcal{X} \to \mathcal{Z}$, which transforms each input sample into a latent representation $\mathbf{z}$. We make no assumptions about the distributions of the source and target domains in the latent space, except that their first- and second-order moments exist. For each domain, the empirical mean μ and covariance Σ of the distributions in the latent space are estimated from a mini-batch of size b as follows:

$$\mu = \frac{1}{b}\sum_{i=1}^{b}\mathbf{z}_i, \quad \Sigma = \frac{1}{b-1}\sum_{i=1}^{b}(\mathbf{z}_i - \mu)(\mathbf{z}_i - \mu)^\top. \tag{3.1}$$

These statistical moments, particularly the second-order covariances, do not reside in a flat Euclidean space but are naturally constrained to the manifold of SPD matrices, denoted as $\mathcal{P}(n)$. In this non-Euclidean space, the standard "straight-line" Euclidean distance is often misleading, as it ignores the spectral structure and curvature of the manifold [64]. Consequently, aligning distributions effectively requires distances that respect the geometry of the SPD space. Our approach leverages this geometric perspective by treating the first two statistical moments not as separate entities, but as a single point on a higher-dimensional SPD manifold.

3.1.1 Joint Moment Representation via Siegel Embeddings

To enable simultaneous adaptation of mean and covariance, we employ Siegel embeddings to map these statistics into a unified SPD matrix. Given the empirical first moment $\mu \in \mathbb{R}^n$ and second moment $\Sigma \in \mathcal{P}(n)$ estimated from a mini-batch of data, we define the diffeomorphic embedding $\psi : \mathbb{R}^n \times \mathcal{P}(n) \to \mathcal{P}(n+1)$ as:

$$\psi(\mu, \Sigma) = \begin{bmatrix} \Sigma + \mu\mu^\top & \mu \\ \mu^\top & 1 \end{bmatrix}. \tag{3.2}$$

This transformation is central to our framework because it maps the manifold of Gaussians into the submanifold of $\mathcal{P}(n+1)$. This ensures that the information-geometric structure of the distributions is isometrically preserved within this representation [65].

Geometry-Aware Distances

Once the source and target moments are embedded as points $P_S = \psi(\mu_S, \Sigma_S)$ and $P_T = \psi(\mu_T, \Sigma_T)$, we quantify their discrepancy using distances native to the SPD manifold. We consider two primary distance measures, which are described below.

Affine-Invariant Riemannian Distance (AIRD): Derived from the intrinsic geometric properties of the SPD manifold, the AIRD accounts for the non-Euclidean curvature of $\mathcal{P}(n+1)$ and is defined as [65, 66]:

$$d_A(P_S, P_T) = \sqrt{\frac{1}{2}\sum_{i=1}^{n+1} \log^2 \lambda_i(P_S^{-1} P_T)}, \tag{3.3}$$

where $\lambda_i(P_S^{-1}P_T)$ denotes the i-th eigenvalue of the matrix $P_S^{-1}P_T$. The distance d_A between embeddings provides a principled lower bound to the computationally intensive Fisher-Rao distance on the Gaussian manifold between latent features of source and target distributions. Although the embedded submanifold is not totally geodesic, implying the shortest path on the SPD manifold may be shorter than the path constrained to the Gaussian submanifold, d_A remains a faithful and rigorous metric for alignment.

While AIRD is theoretically robust, its computation requires the extraction of all eigenvalues. This motivates the use of a more computationally efficient alternative that maintains geometric awareness while scaling more effectively.

Hilbert Projective Distance (HPD): For high-dimensional embeddings, we adopt the HPD as a computationally efficient approximation of the AIRD [65]. It simplifies the discrepancy measure by depending only on the extreme eigenvalues of the matrix product $P_S^{-1}P_T$:

$$d_H(P_S, P_T) = \log\left(\frac{\lambda_{max}(P_S^{-1}P_T)}{\lambda_{min}(P_S^{-1}P_T)}\right), \tag{3.4}$$

where λ_{max} and λ_{min} represent the maximum and minimum eigenvalues of $P_S^{-1}P_T$, respectively. By focusing only on these eigenvalues, HPD can be efficiently approximated using fast iterative solvers [67].

3.1.2 Learning Objective and Optimization Strategy

To integrate the geometric alignment into the learning process, we define the relationship between the feature encoder F_θ and the downstream classifier C. The encoder transforms the input data into a latent representation $\mathbf{z} = F_\theta(\mathbf{x})$, which subsequently serves as the input to the classifier to produce the final prediction. The parameters of both components are optimized simultaneously through a combined objective function that balances task-specific performance with domain invariance.

The total loss function used to train the network is formulated as follows:

$$\mathcal{L} = \mathcal{L}_{task} + \beta\mathcal{L}_{dist}, \tag{3.5}$$

where $\mathcal{L}_{task}$ represents the task-specific loss (in our case cross-entropy for classification) measured on the labeled source domain, and β is a hyperparameter that weights the contribution of the adaptation component [68, 69].

The adaptation loss, $\mathcal{L}_{dist}$, is where the geometry-aware distances are utilized to minimize the shift between the domains. Depending on the chosen metric, this term is substituted with either d_A or d_H between the Siegel-embeddings of the source and target distributions.

By iteratively minimizing this combined objective, the model learns a latent space $\mathcal{Z}$ that is both discriminative for the intended task and invariant to the distribution shift between the source and target domains, consistent with the theoretical framework introduced in Chapter 2.

3.1.3 Computational Stability

A critical aspect of the proposed method is ensuring the numerical stability of the AIRD and HPD distances, which rely on matrix inversions and eigenvalue decompositions. To guarantee that the embedded SPD matrix

P_S remains invertible, we leverage the properties of the Schur complement. For a block matrix $M = \begin{bmatrix} A & B \\ G & E \end{bmatrix}$, the matrix is invertible if and only if E and the Schur complement $A - BE^{-1}G$ are non-singular. In our Siegel embedding $\psi(\mu, \Sigma)$, the Schur complement simplifies directly to the covariance matrix Σ. Thus, the invertibility of the entire joint embedding is equivalent to the invertibility of the feature covariance.

In practice, we monitor this condition through two primary mechanisms:

1. **Dimensionality Constraints:** We ensure the mini-batch size is significantly larger than the latent feature dimensionality ($b \gg n$) to prevent the covariance matrix from becoming rank-deficient.

2. **Two-Phase Optimization:** We utilize a gradual learning strategy where the model first optimizes only the task objective on the source domain while monitoring the determinant of P_S; the geometric adaptation loss is only activated ($\beta > 0$) once the determinant of the matrix P_S reaches a stability threshold η.

3.1.4 Empirical Validation

Data & Setup Our first evaluation is conducted on the *Office-31* dataset, a standard benchmark for UDA comprising 4,110 images across 31 classes. The images are collected from three distinct domains: *Amazon* (A), containing a single object usually on a white background and centered; *Webcam* (W), consisting of low-resolution images from web cameras; and *DSLR* (D), featuring high-resolution images taken with a digital single-lens reflex camera [70]. We construct five pairs of source-target adaptation scenarios from these domains to evaluate the model's ability to generalize to target domains with varying levels of shift. We also evaluate our methods on the large-scale VisDA-2017 benchmark dataset [22], which is specifically designed for challenging DA and exhibits a pronounced "Sim-to-Real" covariate shift (see Figure 1.1). The dataset comprises three domains: a synthetic training domain with rendered 3D objects, a validation domain containing cropped images from Microsoft COCO [71], and a test domain consisting of cropped images from YouTube-BoundingBox [72].

For the backbone architecture, we utilize a ResNet-50 model [73] pretrained on ImageNet [74]. To facilitate adaptation, a bottleneck adaptation layer is inserted before the final classifier to extract latent features. The model is trained using the Adam optimizer. We benchmark our methods, referred to as *GeoAdapt*, against several moment-matching baselines, including Deep Domain Confusion (DDC) [75], Deep Correlation Alignment

Table 3.1: Classification accuracy (↑) on the target domain for the Office-31 benchmark dataset.

Method	Moment	A→W	D→W	A→D	D→A	W→A	Avg
Source-only	-	0.698 ± 0.001	0.950 ± 0.001	0.714 ± 0.018	0.597 ± 0.01	0.601 ± 0.011	0.712
DDC	1	0.786 ± 0.016	$\mathbf{0.962 \pm 0.002}$	$\mathbf{0.846 \pm 0.030}$	0.599 ± 0.016	0.596 ± 0.018	0.758
DCORAL	2	0.797 ± 0.006	0.867 ± 0.01	0.776 ± 0.002	0.604 ± 0.014	0.637 ± 0.037	0.736
MECA	2	0.800 ± 0.010	$\mathbf{0.962 \pm 0.003}$	0.776 ± 0.007	0.632 ± 0.006	0.647 ± 0.008	0.763
CMD	1, 2	0.774 ± 0.018	0.946 ± 0.003	0.792 ± 0.006	0.557 ± 0.036	0.555 ± 0.005	0.725
HoMM	1, 2	0.797 ± 0.012	0.931 ± 0.004	0.776 ± 0.007	0.580 ± 0.021	0.601 ± 0.026	0.737
CMD	1, 2, 3	0.789 ± 0.002	0.953 ± 0.001	0.809 ± 0.017	0.602 ± 0.018	0.610 ± 0.009	0.753
HoMM	1, 2, 3	0.835 ± 0.019	0.950 ± 0.004	0.814 ± 0.006	0.619 ± 0.012	0.624 ± 0.022	0.768
GeoAdapt-HPD (ours)	1, 2	0.830 ± 0.004	$\mathbf{0.962 \pm 0.002}$	0.817 ± 0.006	0.606 ± 0.011	0.624 ± 0.013	0.768
GeoAdapt-AIRD (ours)	1, 2	$\mathbf{0.846 \pm 0.009}$	0.961 ± 0.003	0.828 ± 0.005	$\mathbf{0.647 \pm 0.009}$	$\mathbf{0.661 \pm 0.010}$	**0.789**

(DCORAL) [68], Minimal-Entropy Correlation Alignment (MECA) [76], Central Moment Discrepancy (CMD) [69], and Higher-Order Moment Matching (HoMM) [77]. As a reference point to measure how much DA methods actually help, we separately train the ResNet-50 model only on labeled source domain data without any adaptation to the target domain data and report its performance on the target domain. We refer to this model as *Source-only* method.

Results & Discussion As shown in Table 3.1, our geometry-aware methods outperform baseline approaches that also utilize moment-matching mechanisms for adaptation on the Office-31 dataset. Notably, GeoAdapt-AIRD achieves the highest average accuracy across all Office-31 tasks. This suggests that respecting the manifold structure of the moments enables a more robust estimation of inter-domain distance, which in turn leads to improved alignment.

The hyperparameter tuning protocols differ between the datasets to reflect their respective evaluation standards. For Office-31, hyperparameters are selected based on the $A \rightarrow W$ setup and applied to all other setups. In contrast, for VisDA-2017, hyperparameters are tuned on the validation domain, while performance is reported on the test domain, which serves as the primary adaptation target. This protocol reflects an out-of-the-box deployment scenario, as the target domain is not observed during hyperparameter tuning. Table 3.2 presents the results on VisDA-2017, averaged over ten runs. GeoAdapt-AIRD achieves the best overall performance again, followed by another geometry-aware MECA method among the moment-matching approaches in the table.

To isolate the individual contributions of our framework, we conduct an ablation study comparing our methods against a variant, *GeoAdapt-Euc*, which employs Siegel embeddings but replaces the proposed distances with a standard Euclidean distance. As detailed in Table 3.3, the geometry-aware

Table 3.2: Classification accuracy (↑) on the target domain for the VisDA-2017 benchmark.

Method	Moment	Accuracy
Source-only	-	0.345 ± 0.021
DDC	1	0.526 ± 0.016
DCORAL	2	0.700 ± 0.012
MECA	2	0.736 ± 0.014
CMD	1, 2	0.634 ± 0.038
HoMM	1, 2	0.717 ± 0.007
CMD	1, 2, 3	0.733 ± 0.046
HoMM	1, 2, 3	0.705 ± 0.028
GeoAdapt-HPD (ours)	1, 2	0.715 ± 0.022
GeoAdapt-AIRD (ours)	1, 2	$\mathbf{0.748 \pm 0.021}$

Table 3.3: Classification accuracy (↑) on the target domain for the VisDA-2017 benchmark, evaluating different distance metrics in our ablation study.

Method	Moment	Accuracy
GeoAdapt-HPD	1, 2	0.715 ± 0.022
GeoAdapt-AIRD	1, 2	0.748 ± 0.021
GeoAdapt-Euc	1, 2	0.705 ± 0.010

methods outperforms *GeoAdapt-Euc* on VisDA-2017, indicating that performance gains are primarily driven by the distance measure rather than the embedding alone. Notably, *GeoAdapt-Euc* yields results comparable to the Euclidean-based DCORAL (0.700 ± 0.012), whereas MECA, which similarly utilizes a geometry-aware metric, achieves significantly higher accuracy (0.736 ± 0.014). These findings underscore that regardless of the embedding strategy, respecting the underlying manifold geometry is critical for achieving robust domain alignment.

To evaluate the dimensionality constraints, we provide an analysis of feature dimensionality, as illustrated in Figure 3.1, to shed light on selecting the optimal dimensionality. We use compact embedding spaces with dimensionality in the order of tens, in contrast to the high-dimensional feature vectors (e.g., $n = 2048$) typically extracted from standard ResNet backbones. Figure 3.1 shows the results on the A→W setup of the Office-31 dataset across dimensions n, indicating that a compact adaptation layer benefits *all* baseline methods as the peak accuracy occurs for $n \in [32, 128]$. This suggests that embeddings with reduced dimensions should be more widely considered in DA, with the possibility of offering gains in both accuracy and efficiency. Our distance measures perform best in low-to-mid

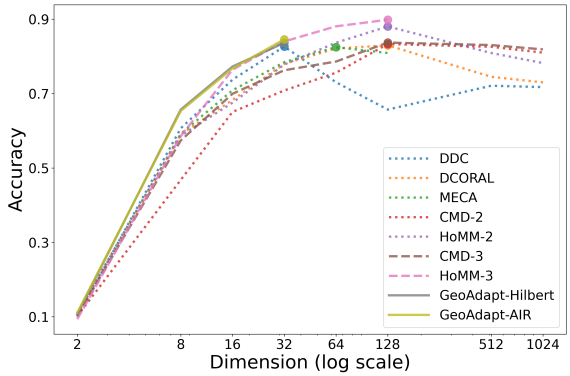


Figure 3.1: Accuracy on the A→W setup of the Office-31 dataset as a function of the embedding dimensionality (x-axis). Our distances are the best for the dimensionalities up to our conservative choice of maximum dimensionality where P_S can be robustly inverted.

dimensions and could likely scale to higher dimensions using larger mini-batches or covariance shrinkage methods [78]. As discussed in Section 3.1.3, we adopt a conservative approach that avoids computational issues, without exploring such approximations.

3.2 Implicit Alignment via Adversarial Consistency Training

The second methodological pillar of this thesis approaches covariate shift through the lens of implicit alignment and decision boundary refinement. While the geometric approach focuses on aligning global statistical moments, this methodology leverages the synergy between SSL and UDA to learn robust representations. By integrating consistency regularization [79, 80] with adversarial learning [60], we ensure that the model remains stable in the presence of distribution shifts while simultaneously smoothing the decision boundary in low-density regions of the unlabeled target domain.

3.2.1 Bridging SSL and UDA

SSL and UDA are fundamentally related; both seek to leverage unlabeled data to improve generalization. In SSL, it is typically assumed that labeled and unlabeled data follow the same distribution, whereas UDA explicitly accounts for a domain shift between labeled source data and unlabeled target data. An advanced technique in SSL is ICT [26], which is grounded in



(a) Image 1 (b) Image 2 (c) MixUp Result

Figure 3.2: MixUp augmentation applied to two images from the COCO dataset [71].

the *cluster assumption*. This assumption posits that the decision boundary between classes should reside in low-density regions of the feature space. Therefore, enforcing consistency there is particularly beneficial, while high-density regions are typically learned well regardless. ICT encourages this by ensuring that the model's predictions for an interpolated sample are consistent with the interpolation of the predictions for the original samples.

3.2.2 Formalization of Interpolation Consistency Training

We utilize the MixUp augmentation strategy [81] to explore these low-density regions. For any two target-domain samples $\mathbf{x}'_T$ and $\mathbf{x}''_T$, we define their linear interpolation as

$$\text{Mix}_\lambda(\mathbf{x}'_T, \mathbf{x}''_T) = \lambda \mathbf{x}'_T + (1-\lambda)\mathbf{x}''_T, \tag{3.6}$$

where the mixing rate λ is drawn from a symmetric Beta distribution $Beta(\alpha, \alpha)$. MixUp effectively explores low-density regions because with sufficiently high number of classes the mixed inputs are likely from different classes, producing samples that lie between class clusters. When $\alpha < 1$, λ tends to be near 0 or 1, so samples remain close to the originals, while larger α yields more evenly mixed samples that extend further beyond the original data distribution. Figure 3.2 illustrates this phenomenon: the resulting image produced by Mixup on the right does not correspond to a representative sample from the dataset and therefore resides in a low-density region.

To maintain stable adaptation targets, ICT also employs a *mean-teacher* framework [80]. It maintains a student network (F, C) and a teacher network (F', C'), where the weights of the teacher are updated as the exponential moving average (EMA) of the student's parameters. The consistency regularization encourages the student's prediction on the mixture to match the teacher's mixed predictions:

$$C(F(\text{Mix}_\lambda(\mathbf{x}'_T, \mathbf{x}''_T))) \approx \text{Mix}_\lambda(y', y''), \tag{3.7}$$

where $y' = C'(F'(\mathbf{x}'_T))$ and $y'' = C'(F'(\mathbf{x}''_T))$ are the predictions for the original samples. This behavior is enforced by minimizing the consistency loss $\mathcal{L}_{cons}$, measuring the distance between these two terms using a standard metric such as mean squared error or cross entropy as the function d:

$$\mathcal{L}_{cons} = \mathbb{E}_{(\mathbf{x}'_T, \mathbf{x}''_T) \sim X_T} \left[d\Big(C(F(\mathrm{Mix}_\lambda(\mathbf{x}'_T, \mathbf{x}''_T))), \mathrm{Mix}_\lambda(y', y'') \Big) \right]. \tag{3.8}$$

3.2.3 Adversarial Adaptation

While ICT is highly effective in standard SSL [26], our empirical findings indicate that consistency training alone is insufficient for handling significant distribution shifts and can even deteriorate performance. To alleviate this, we integrate an adversarial component to explicitly account for the covariate shift. We introduce a domain classifier D that attempts to distinguish between source and target features by minimizing the adversarial loss:

$$\mathcal{L}_{adv} = -\mathbb{E}_{\mathbf{x}_S \sim X_S}[\log D(F(\mathbf{x}_S))] - \mathbb{E}_{\mathbf{x}_T \sim X_T}[\log(1 - D(F(\mathbf{x}_T)))]. \tag{3.9}$$

To discover domain-invariant representations, we utilize a GRL [60] between the feature extractor F and the domain classifier D. During backpropagation, the GRL multiplies the gradients by a negative constant ($-\gamma \frac{\partial \mathcal{L}_{adv}}{\partial \theta_F}$ for $\gamma > 0$), effectively forcing F to produce features that are indistinguishable to the domain classifier, while D simultaneously aims at maximizing the domain classification accuracy. The total objective thus becomes a min-max problem.

3.2.4 Method summary

The full training objective consists of the aforementioned losses together with an image classification loss $\mathcal{L}_{cls}$ (e.g., cross-entropy) applied to the labeled source domain. In summary, our method optimizes three objectives:

$$\min_{F,C} \mathcal{L}_{cls}, \quad \min_{F,C,F',C'} \mathcal{L}_{cons}, \quad \text{and} \quad \min_{D} \max_{F} \mathcal{L}_{adv}.$$

These objectives respectively minimize classification error, enforce an interpolation consistency regularizer adapted for domain adaptation, and apply adversarial min-max training to learn a domain-invariant representation. Figure 3.3 illustrates an overall architecture of the proposed method.

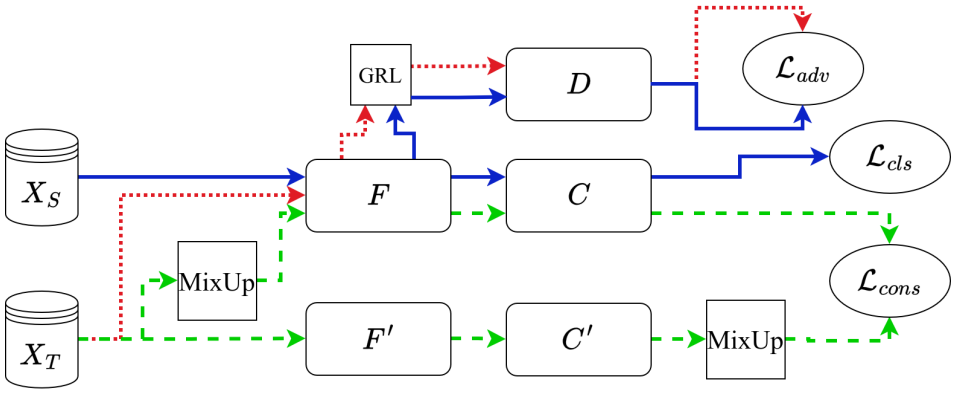


Figure 3.3: Schematic overview of the proposed method. The solid blue path illustrates how labeled source data is utilized. The dotted red path shows how unlabeled target data is incorporated to address covariate shift. The dashed green path highlights the components related to ICT.

Training follows standard DNNs optimization with gradient descent and automatic differentiation. Most model parameters are updated using ordinary gradients, but the feature extractor F must be updated as

$$\theta_F \longleftarrow \theta_F - \eta \left(\frac{\partial \mathcal{L}_{cons}}{\partial \theta_F} + \frac{\partial \mathcal{L}_{cls}}{\partial \theta_F} - \gamma \frac{\partial \mathcal{L}_{adv}}{\partial \theta_F} \right), \tag{3.10}$$

where η denotes the learning rate, to account for gradient reversal. The teacher networks F' and C' are not optimized directly; instead, their parameters are maintained as EMA of the student networks F and C.

3.2.5 Empirical Validation

Data & Setup To validate the efficacy of our proposed framework, we evaluate performance on three standard digits benchmarks: *MNIST* [40], *USPS* [82], and *MNIST-M* [60]. While MNIST and USPS represent conventional digit recognition tasks, MNIST-M introduces a synthetic distribution shift by blending digits with patches from the BSDS500 dataset [83]; see [60] for details. We compare our proposed method against three primary baselines: a *Source-only* model trained without target domain data; *ICT*, which applies interpolation consistency in a semi-supervised manner to target samples; and *DANN*, a standard adversarial domain adaptation method. The comparison with DANN is particularly instructive, as it isolates the relative importance of interpolation consistency versus standard adversarial training. To ensure a fair comparison, all networks (F, C, and

Table 3.4: The accuracy of different models. The results are averaged over 5 runs for each setup.

Method	**MNIST → MNIST-M**	**MNIST → USPS**	**USPS → MNIST**
Source-only	0.42 ± 0.04	0.81 ± 0.01	0.54 ± 0.02
ICT	0.35 ± 0.03	0.75 ± 0.05	0.43 ± 0.02
DANN	0.73 ± 0.02	0.85 ± 0.02	0.79 ± 0.03
Ours	$\mathbf{0.85 \pm 0.02}$	$\mathbf{0.91 \pm 0.01}$	$\mathbf{0.83 \pm 0.03}$

D) utilize the standardized "MNIST architecture" originally introduced by Ganin et al. [60].

Results & Discussion In this analysis, we focus on highlighting the core effects of our method through ablation-style experiments, specifically examining the performance comparison between our approach and the standalone ICT and DANN baselines. Table 3.4 reports the mean classification accuracies and standard deviations across five independent runs on the test set of the target domains. The results indicate that while standard ICT fails to generalize under significant domain shifts, underperforming even the Source-only baseline across all three configurations, DANN successfully manages these covariate shifts. Ultimately, our method consistently outperforms all baselines across every setup. These findings demonstrate that ICT can be effectively extended to enhance the robustness of DA methods.

Notably, these results were achieved using established benchmark architecture used by Ganin et al. [60] without any specific hyperparameter tuning. While we acknowledge that the absolute accuracy of the current SOTA for these datasets are much higher, reporting such values is not the primary aim of this validation. Performance gains could be readily achieved by adopting more sophisticated backbones, such as ResNets; however, a direct comparison with those results would be misleading given our reliance on a much simpler architecture. Our framework remains essentially backbone-agnostic, and we contend that the choice of different backbones should be evaluated separately in terms of the specific trade-offs between computational cost and accuracy [84].

Chapter 4

Label Observation Shift in Weak Supervision

This chapter addresses a specific form of concept shift: *annotation-induced concept shift*. In this scenario, the underlying semantics of objects remain constant, but the annotation process is such that the label information is only partially observed. Figure 4.1 provides a visual representation of this shift, contrasting standard ground truth annotations with the more restricted labeling regimes. Within the taxonomy of weakly supervised learning discussed in Section 2.1.1, this setting is classified as *incomplete supervision*. Specifically, we consider *single-positive multi-label learning* (SPMLL), where the annotation process during training reveals only a single relevant label per instance, despite the testing environment requiring the prediction of all relevant labels [19, 85]. As a result, relevant semantic concepts may remain unlabeled, hindering the learning of complete label associations.



	Full	Partial	SPMLL
bench	✓	✓	✓
sheep	✓		
cow	✓	✓	
car	✗		
apple	✗	✗	

Figure 4.1: Example image from the COCO dataset [71] and the annotations available during training under the *Full*, *Partial*, and *SPMLL* settings.

4.1 The Challenge of Partial Supervision

The SPMLL problem presents a severe case of label observation shift. Formally, each q-dimensional data point $\mathbf{x} \in \mathcal{X} = \mathbb{R}^q$ is associated with a binary label vector $\mathbf{y} \in \mathcal{Y} = \{0,1\}^C$, where C is the number of classes. In a fully labeled setting, we would have access to the complete vector $\mathbf{y}_i$ of instance i for training. However, in the SPMLL setup, we only observe a single-positive label vector $\mathbf{l}_i \in \{0,1\}^C$, where $l_i^j = 1$ indicates that the j-th class is the only observed positive label for $\mathbf{x}_i$, and $l_i^j = 0$ for all other classes. This implies that for the set of unobserved classes $\mathbb{U}_i$, the true label values are unknown (not necessarily negative), i.e. $\sum_{j=1}^{C} l_i^j = 1$.

This setup relates to the UDA framework discussed in the previous section, but with a subtle and critical difference. In covariate shift, we assume a shift in the marginal distribution of features $P_S(\mathbf{x}) \neq P_T(\mathbf{x})$ while the labeling function remains unchanged. In SPMLL, the "shift" occurs in the observation of the labeling function itself: the training phase operates under a restricted observation of labels, whereas the test phase requires recovery of the full multi-label function.

A common baseline in the literature is the Assumed Negative (AN) loss [19], which treats all unobserved labels as negatives. As illustrated in Figure 4.2 with L_{AN}, this tends to assign near-zero probabilities to all unknown labels, which misleads the model and causes severe false negatives. For example, in Figure 4.1, the unannotated "sheep" and "cow" labels can be treated as negative targets during training, despite their presence in the image. A model trained in this way may therefore miss them in other examples as well. Conversely, Zhou et al. [86] attempt to mitigate this by maximizing the entropy of predictions (L_{EM} in Figure 4.2) for unlabeled classes. However, this pushes unobserved probabilities toward 0.5, a value that is typically far too high for large-scale multi-label datasets where the true number of positive labels is much smaller than the total class count C. Other recent approaches have focused on preventing false negatives through a combination of pseudo-labeling strategies with consistency regularization [87], used a mutual information bottleneck to retain label-specific features during the pseudo-labeling process [88], or introduced a noise-robust loss function to address class imbalance in SPMLL [89]. A more similar approach to ours is the method by Cole et al. [19], which regularizes predictions using the expected label cardinality and incorporates label smoothing. Their regularizer can be viewed as a special case of our more general framework, which considers various forms of additional information about label cardinality.

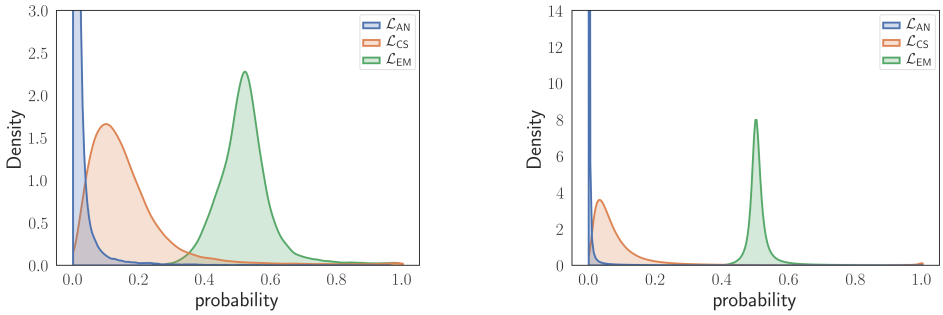


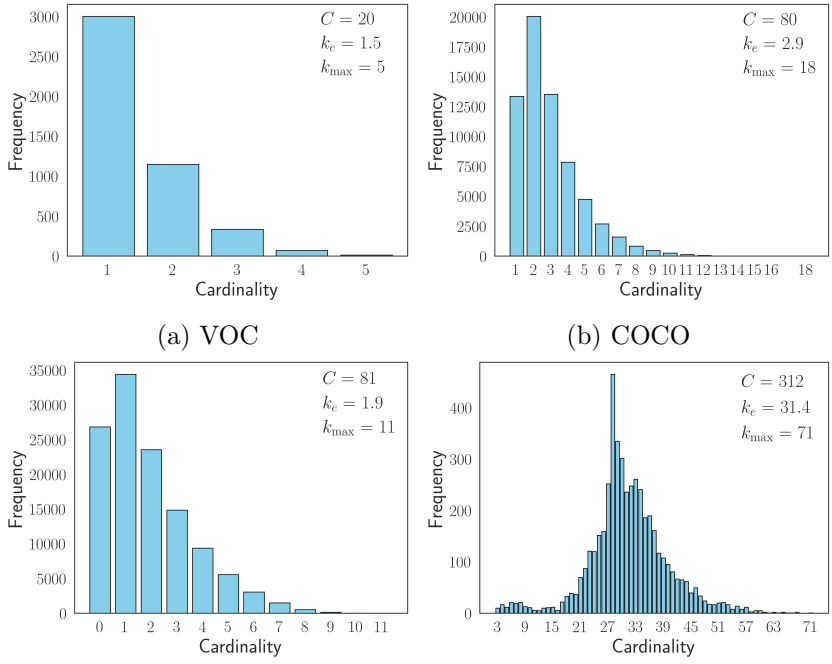
Figure 4.2: Distribution of model outputs for unobserved labels on VOC (left) and NUS (right). $\mathcal{L}_{\text{AN}}$ induces false negatives by pushing predictions toward 0, while $\mathcal{L}_{\text{EM}}$ induces false positives by shifting them toward 0.5. Our $\mathcal{L}_{\text{CS}}$ promotes predictions consistent with cardinality statistics. The y-axis is truncated at 20% of the maximum density of $\mathcal{L}_{\text{AN}}$ for clarity.

Building upon these foundations, we propose to leverage auxiliary information regarding the *label cardinality*, the total number of positive labels per instance, to recover the true multi-label distribution. Our core idea is to utilize this cardinality as a structural constraint that guides the model in identifying hidden positives among the unobserved labels of the training samples. While our primary focus is the SPMLL setting where only one label is known, our framework is designed to be generalizable to various partial labeling setups.

4.2 Hierarchy of Cardinality Information

We categorize cardinality information into a hierarchy based on its granularity:

- **Instance Cardinality (IC):** The most informative scenario, where the exact number of true positive labels $k_i = \sum_j y_i^j$ is known for each instance i.
- **Cardinality Distribution (CD):** A more relaxed assumption where we know or can estimate the discrete probability distribution $P(k)$ of the label cardinalities across the dataset.
- **Cardinality Statistics (CS):** The crudest tier, utilizing simple summary statistics such as the maximum ($k_{\max}$) and mean (k_e) label cardinality.



(a) VOC (b) COCO

(c) NUS (d) CUB

Figure 4.3: Cardinality distributions and statistics for multi-label image benchmarks show that low-cardinality instances dominate when labels represent objects, whereas the attribute labels in CUB exhibit a distinctly different distribution. Here, C represents the number of classes, while k_e and $k_{\max}$ denote the mean and maximum label cardinalities in the datasets.

The motivation for defining this multi-tiered hierarchy is to provide a flexible framework that seamlessly adapts to varying levels of available information, ensuring the method remains effective across different practical constraints. Additionally, such cardinality constraints can often be obtained without incurring the high cost of full multi-label annotation. For example, IC can be collected by asking annotators simply to count the objects in an image – a task significantly faster and cheaper than identifying and categorizing every instance [90]. At coarser levels, CD and CS can be reliably estimated from domain priors, metadata from similar historical datasets [91], or class-agnostic object proposal algorithms [92].

As visualized in Figure 4.3, these distributions vary significantly across benchmark datasets: object-oriented datasets like VOC and COCO are dominated by low cardinalities, whereas attribute-rich datasets like CUB exhibit much higher and wider distributions.

4.3 Cardinality-Aware Single-Label Classification

4.3.1 SPMLL Loss

To integrate cardinality into the training process, we first define a factorized SPMLL objective:

$$\mathcal{L}_{\text{SPMLL}} = \frac{1}{NC} \sum_{i=1}^{N} \sum_{j=1}^{C} \left(\mathbb{I}_{[l_i^j=1]} \mathcal{L}^+(\hat{y}_i^j) + \mathbb{I}_{[l_i^j=0]} \mathcal{L}^u(\hat{y}_i^j) \right), \tag{4.1}$$

where $\hat{y}_i^j = h(x_i)$ is the predicted probability for class j, $\mathcal{L}^+$ is the loss for observed positive labels (e.g., cross-entropy), and $\mathcal{L}^u$ is the loss for unobserved classes. This provides a unifying framework where the task is to design an informative $\mathcal{L}^u$ that prevents the collapse observed in prior methods.

4.3.2 Cardinality Statistics Loss

Designed for cases with only basic statistics, this loss defines a non-penalized range for unobserved output probabilities $\hat{y}_i^j$. We use an upper threshold $T_{up} = (k_{\max} - k_i^o)/C_i^u$ and a lower threshold $T_{lo} = (k_e - k_i^o)/C_i^u$ to ensure the model does not prematurely commit to a label being positive or negative in the absence of evidence. Here, k_i^o is the number of observed positive labels for instance i ($k_i^o = 1$ in standard SPMLL) and $C_i^u = C - k_i^o$ is the number of unobserved classes. The loss $\mathcal{L}_{CS}^u$ is defined as:

$$\mathcal{L}_{\text{CS}}^u(f_i^j) = \begin{cases} \lambda \left(\hat{y}_i^j - T_{up} \right)^2 & \hat{y}_i^j > T_{up} \\ \lambda \left(\hat{y}_i^j - T_{lo} \right)^2 & \hat{y}_i^j < T_{lo} \\ 0 & \text{Otherwise} \end{cases} \tag{4.2}$$

where λ is a scaling hyperparameter. This loss applies a quadratic penalty only when predictions fall outside the designated range. The reasoning for this design is three-fold: the model should not be a priori certain about unobserved labels, the loss should not favor any specific output within the statistically likely range, and the range should align with prior knowledge about label cardinality.

4.3.3 Instance Cardinality Loss

When k_i is known, we encourage predictions to be consistent with the instance cardinality. The loss is formulated as:

$$\mathcal{L}_{IC}^{u}(\hat{y}_i) = \left(\left(\sum_{j \in \Omega_i} \hat{y}_i^j \right) - k_i^u \right)^2 + \beta \left(\sum_{j \notin \Omega_i} \hat{y}_i^j \right)^2, \tag{4.3}$$

where $k_i^u = k_i - k_i^o$ is the target cardinality for unobserved classes and Ω_i is the set of M_i classes with the highest predicted probabilities. The first term aligns the probability mass of the most likely classes with the target, while the second term drives the remaining labels toward zero. We observe two critical practical remarks: (1) we set $M_i > k_i$ (e.g., $M_i = 2k_i$ or $k_{\max} - k_i^o$) to avoid enforcing false negatives if current predictions are inaccurate, and (2) we use a scaled sigmoid transformation that saturates to one faster, as logits $\boldsymbol{z}_i$ often result in $\hat{y}_i^j < 1$ even for correct classes. Therefore, we use the transformed outputs $\tilde{y}_i^j = \frac{1}{1+\exp\{-a(\boldsymbol{z}_i)\}}$ in place of $\hat{y}_i^j$ where $a > 1$.

Bipartite Matching Estimator

The effectiveness of the instance loss $\mathcal{L}_{IC}$ depends on the availability of the instance cardinality k_i. When only the dataset-level distribution $P(k)$ is available, we need to estimate $\hat{k}_i$ for each instance to use in Equation 4.3. We cast this as a bipartite matching problem between the model's class-sum predictions and the known cardinality distribution.

We define a score vector $\mathbf{s} \in \mathbb{R}^N$, where each element $s_i = \sum_{j \in \Omega_i} \tilde{y}_i^j$ aggregates the predicted probabilities of the classes within Ω_i according to the current model's outputs. We also construct a target vector $\mathbf{v} \in \{0, \ldots, k_{max}\}^N$ as a discrete representation of $P(k)$ by assigning values according to its frequencies in increasing order (e.g., if 15% of samples have three labels, then $0.15N$ entries of $\mathbf{v}$ are set to 3).

The goal is to find a permutation Π of the indices in $\mathbf{v}$ that minimizes $\sum_{i=1}^{N}(s_i - \Pi_i v)^2$, yielding the estimated cardinality $\hat{k}_i = \Pi_i v$ for each instance i. Although general bipartite matching is expensive [93], this reduces to one-dimensional Euclidean matching [94, 95] and can be solved in $\mathcal{O}(N \log N)$ by sorting $\mathbf{s}$ and matching it sequentially to $\mathbf{v}$. This ensures that the global cardinality constraints are strictly satisfied while providing the necessary instance-level targets for the $\mathcal{L}_{IC}$ loss to refine the decision boundaries and recover unobserved labels.

4.3.4 Integrated SPMLL Algorithms

We introduce three practical algorithms for training under different tiers of cardinality information:

- `CS`: The baseline method that uses only $\mathcal{L}_{CS}$. It is simple to implement and serves as an informative initialization for more complex variants.
- `CS` $\Rightarrow$ `CD`: A multi-stage approach for the CD setup. The model is first initialized with $\mathcal{L}_{CS}$ so that predictions $\hat{y}_i^j$ start becoming informative. Then, instance cardinalities $\hat{k}_i$ are estimated using the known distribution and the bipartite matching algorithm. Finally, the model is optimized using $\mathcal{L}_{IC}$ with these estimated targets, re-estimating $\hat{k}_i$ after each epoch.
- `CS` $\Rightarrow$ `IC`: Similar to the `CS` $\Rightarrow$ `CD` approach but skips the estimation step by using the known true k_i directly in the $\mathcal{L}_{IC}$ loss. This serves as the upper bound for what cardinality information can provide.

4.4 Experiments

This section provides a comprehensive evaluation of the proposed methods to quantify the value of different levels of cardinality information and assess the robustness of the estimation process. We analyze the performance across diverse benchmarks and provide insights into the relationship between cardinality estimation quality and classification accuracy.

4.4.1 Experiment 1: Performance Comparison and Benchmarking

The primary objective of the first experiment is to contrast our proposed methods against the current SOTA, including ROLE [19], EM+APL [86], LL-R + BoostLU [96], and the Generalized Robust (GR) Loss [89]. We report the Mean Average Precision (MAP) for three variants of our framework: the baseline `CS` method, the distribution-aware `CS` $\Rightarrow$ `CD` approach, and the instance-aware `CS` $\Rightarrow$ `IC` upper bound. As a standard metric for multi-label classification, MAP computes the average of per-class Average Precision scores, thereby measuring how well the model ranks relevant labels above irrelevant ones. The results are summarized in Table 4.1 for all benchmark datasets in Figure 4.3: Pascal VOC 2012 (VOC) [97], MS COCO 2014 (COCO) [71], NUS-Wide (NUS) [98], and CUB-200-2011 (CUB) [99].

Table 4.1: Test MAP across all methods. The CS method is directly comparable to prior approaches.

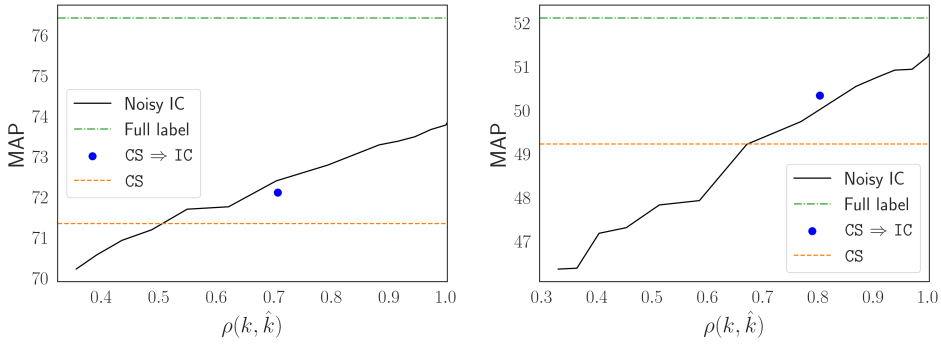
Method	VOC	COCO	NUS	CUB
Full label	89.56 ± 0.35	76.43 ± 0.09	52.12 ± 0.09	32.73 ± 0.20
AN	85.51 ± 0.41	64.33 ± 0.10	42.68 ± 0.09	18.65 ± 0.31
ROLE	88.10 ± 0.16	66.89 ± 0.16	41.74 ± 0.21	14.52 ± 0.47
EM+APL	89.19 ± 0.29	70.92 ± 0.23	47.56 ± 0.20	21.13 ± 0.60
LL-R+BoostLU	88.81 ± 0.24	73.11 ± 0.21	49.76 ± 0.51	19.75 ± 0.23
GR Loss	89.38 ± 0.12	71.33 ± 0.21	47.08 ± 0.43	20.69 ± 0.28
CS (ours)	88.97 ± 0.03	71.35 ± 0.10	49.23 ± 0.45	21.56 ± 0.08
CS $\Rightarrow$ CD (ours)	89.10 ± 0.17	72.07 ± 0.08	50.26 ± 0.08	21.58 ± 0.21
CS $\Rightarrow$ IC (ours)	89.69 ± 0.16	73.45 ± 0.12	50.92 ± 0.48	21.73 ± 0.35

On the low-cardinality VOC dataset, our CS variant and most SPMLL baselines achieve approximately 89% MAP, matching the full-label upper bound and indicating that crude statistics are sufficient when semantic separation is clear. However, on the more challenging COCO and NUS benchmarks, our hierarchical approach demonstrates clear advantages. Specifically, the CS method (71.35% MAP on COCO; 49.23% on NUS) outperforms prior baselines like ROLE.

The other two variants are not directly comparable to the rest, as they require additional information; instead, the focus should be on the differences among the three proposed methods. For COCO and NUS, we can see that having access to CD yields a clear improvement, and knowing IC provides further gains. The CS $\Rightarrow$ IC variant (73.45% on COCO; 50.92% on NUS) effectively bridges the gap toward full-label performance (76.43% and 52.12%, respectively). In contrast, the attribute-rich CUB dataset remains a challenge for all methods (MAP $< 22\%$), highlighting that the efficacy of cardinality information is contingent on the specific nature of the data and labels.

4.4.2 Experiment 2: Value of Cardinality Information

While our experiment in Section 4.4.1 establishes that exact label cardinalities are highly informative, they are often difficult to obtain in practice. This motivates a deeper investigation into the information content of the cardinality signal, which we address by posing two fundamental research questions: (Q1) How do improvements in the quality of cardinality estimates directly translate to improvements in classification performance (MAP)? and (Q2) How accurate does a cardinality estimator need to be to still provide a benefit over simple dataset-level statistics?



(a) COCO (b) NUS

Figure 4.4: MAP improves roughly linearly with better cardinality estimates (black), and the real estimator (blue) aligns closely with the simulation. Full-label (green) and CS (orange) MAP scores are shown for scale reference.

To address these questions, we conduct a simulation experiment using a noisy oracle. For the `CS ⇒ CD` method, instead of using the actual model outputs to estimate cardinalities via bipartite matching as it is done in Section 4.3.3, we generate synthetic scores $s_i = k_i + \epsilon$, where k_i is the true instance cardinality and $\epsilon \sim \mathcal{N}(0, \sigma^2)$ is Gaussian noise. By varying the variance σ^2, we can produce cardinality estimates $\hat{k}_i$ of varying quality, ranging from perfect estimates ($\sigma^2 = 0$) to a random permutation of the cardinalities ($\sigma^2 \to \infty$). We quantify the quality of these estimates using the Pearson correlation $\rho(\hat{k}, k)$ between the estimated and true cardinalities over the entire training set.

As illustrated in Figure 4.4, the relationship between the estimation quality and the final MAP is essentially near linear for both COCO and NUS. This provides a clear answer to Q1: improvements in the cardinality estimator are reflected directly and gradually in the MAP estimates of classification accuracy. Regarding Q2, the point where this linear trend intersects the accuracy of the `CS` method identifies the minimum required correlation. For the COCO dataset, any estimator with $\rho(\hat{k}, k) > 0.5$ provides a tangible benefit, whereas for NUS, a correlation of at least 0.65 is required. We also observe that the real results from our `CS ⇒ CD` method[1] align closely with this simulation trend, confirming that the simulation process closely replicates the real estimation error, making it informative for real use cases.

[1]Figure 4.4 reproduces the corresponding figure from the published paper. In both figures, the legend incorrectly labels the method as `CS ⇒ IC`; the correct label is `CS ⇒ CD`.

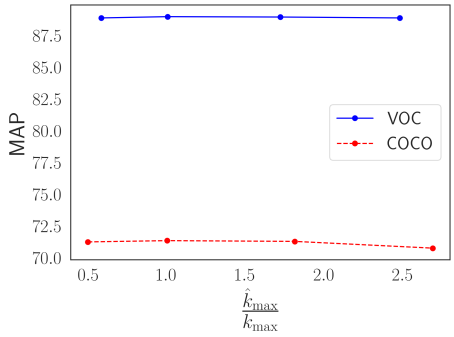
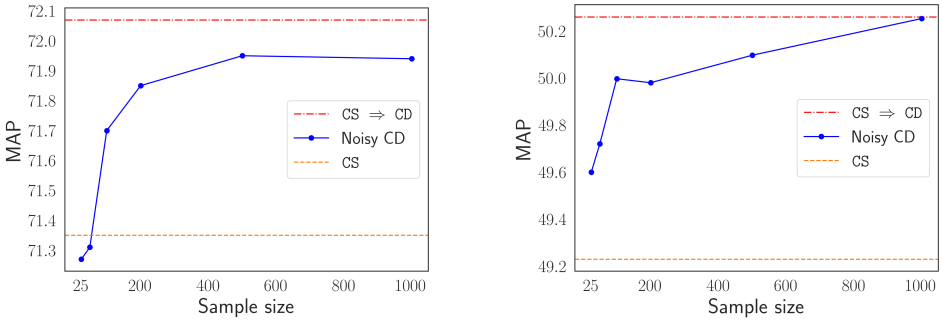


Figure 4.5: Sensitivity analysis of $k_{\max}$, equivalently reflecting the upper threshold T_{up} in the `CS` method.



(a) COCO (b) NUS

Figure 4.6: MAP steadily improves as more precise estimates of cardinality density are used, illustrated here as a function of sample size B with known instance cardinalities.

4.4.3 Experiment 3: Robustness to Estimation Error

In real-world applications, the exact CS or CD information are rarely known a priori and must be estimated from data. This experiment evaluates the sensitivity of our methods to errors in these estimates.

For the `CS` method, the primary parameter is the maximum cardinality $\hat{k}_{\max}$ used to set the upper threshold T_{up}. Our analysis shows that the method is remarkably robust to this choice; we observed virtually no significant change in MAP for COCO or VOC when $\hat{k}_{\max}$ was varied from 0.5 to 2.5 (Figure 4.5) times its true value.

Figure 4.6 illustrates how MAP evolves when CD is estimated from a sample of B instances with known k_i (without requiring fully annotated

Table 4.2: Test MAP for all methods using validation data restricted to a single positive label per instance (denoted as "Single" in the table). For comparison, results obtained on fully annotated validation data ("Full") from Table 4.1 are also reported.

Method	Full		Single	
	COCO	NUS	COCO	NUS
AN	64.33 ± 0.10	42.68 ± 0.09	64.13 ± 0.21	42.33 ± 0.54
ROLE	66.89 ± 0.16	41.74 ± 0.21	66.95 ± 0.09	41.16 ± 0.69
EM+APL	70.92 ± 0.23	47.56 ± 0.20	70.76 ± 0.18	47.56 ± 0.20
LL-R+BoostLU	73.11 ± 0.21	49.76 ± 0.51	70.71 ± 0.18	48.31 ± 0.91
GR Loss	71.33 ± 0.21	47.08 ± 0.43	71.06 ± 0.13	47.38 ± 0.02
`CS` (ours)	71.35 ± 0.10	49.23 ± 0.45	71.19 ± 0.19	49.02 ± 0.19

samples). We use a simple baseline estimator: set $\hat{k}_{\max}$ to the maximum of the largest observed k in the sample and a user-defined upper bound (e.g., $1.2\,k_{\max}$), and estimate $P(k)$ as the histogram over the B samples with additive Laplace smoothing. For NUS, even $B = 25$ already yields a clear improvement over `CS`, and for both datasets MAP increases consistently with larger B, as expected.

4.4.4 Discussion

Robustness to Validation Data Bias As in prior comparison methods, we used a fully labeled validation set to tune hyperparameters, in order to keep the results comparable with the existing literature and comparison methods. This may slightly inflate all reported MAP scores, since such data would not be available in a real deployment, but the effect is small and affects all methods similarly. In addition, `CS` has fewer hyperparameters than most competing methods, so it is less susceptible to gains from tuning. To verify this empirically, Table 4.2 reproduces Table 4.1 for the two most relevant datasets using a validation set with only one positive label per instance. The results are very similar overall, except for LL-R+BoostLU, whose many hyperparameters lead to a noticeable drop in MAP. Under this setting, our `CS` method now outperforms all others on both datasets.

Failure Mode and Generality Analysis We consistently achieved strong performance across four datasets, but the benefit of cardinality information varied. We hypothesize that this failure relates to the ratio of observed to unobserved information. In attribute-rich datasets like CUB, unobserved labels far outnumber observed ones ($k_i^u \gg k_i^o$), making the true target cardinality k_i^u almost conditionally independent of the single

observed label. When the observed label does not carry sufficient signal to distinguish between instances of high and low cardinality, the model outputs will not become informative. Thus, recovering the labeling function from cardinality is not universal but may depend on data complexity and the strength of dependencies between observed and unobserved labels.

Value of Cardinality Information The empirical investigation reveals several key insights into the utility of cardinality as a supervisory signal in SPMLL:

1. **Robustness of `CS`:** The `CS` method serves as a strong baseline that functions effectively with minimal prior information. It relies on CS to determine its thresholds and remains robust even when the maximum cardinality estimate has an error of 50%.

2. **Benefits of CD Information:** Moving from simple statistics to CD, whether provided a priori or estimated from a small labeled subset, consistently enhances the MAP scores compared to the `CS` baseline.

3. **Impact of IC Estimation Quality:** More accurate estimates of instance cardinalities improve overall performance, with MAP increasing roughly linearly with estimation quality as measured by correlation.

4. **Comparison with Full Supervision:** While the absolute gains in MAP may seem modest in some cases, they should be interpreted relative to the achievable upper bound. On VOC and NUS, our approach effectively closes the gap to fully supervised performance, enabling the recovery of the latent multi-label function from a single positive label.

Chapter 5

Covariate Drift in Sensing Networks

The final methodological pillar of this thesis moves from the latent spaces of neural networks and label observation shift into the tangible realm of physical sensing systems. Here, we tackle a complex form of covariate drift: changes in the signal distribution of ultrasound sensing networks caused by physical alterations in the sensing environment. In structural health monitoring (SHM) [100] and non-destructive testing (NDT) [101], a system's ability to detect damage or fouling relies on a stable deployment state. However, as sensors are relocated during maintenance or as material properties drift due to corrosion and temperature changes, the recorded signals undergo a significant distribution shift.

5.1 Physical Origins of Covariate Drift

In traditional UDA, methods typically aim to align the source and target distributions without explicitly modeling the underlying mechanism that caused the shift. In the context of ultrasound sensing, however, we have the unique advantage of understanding the physical laws governing the drift. While SHM has traditionally been implemented using largely static sensing systems, such as rigid transducer collars, the emergence of distributed sensing solutions [102] suggests a need to account for potential changes in structure, operational conditions, and resource constraints, which motivates the proposed self-adaptation framework for dynamic reconfiguration and adaptive behavior [103, 104]. We can categorize the origins of these shifts into two distinct types:

- **Discrete Operational Shifts:** These may occur during maintenance or system re-deployment. For example, a technician might remove a transducer to clean a pipe and re-attach it a few centimeters away from

its original location. Even such minor displacements can invalidate an ML model analysis of the signal recordings, as the acoustic waves now traverse different trajectories.

- **Continuous Environmental Shifts:** These represent gradual changes in the sensed medium itself. The accumulation of rust, fouling on the internal surface of a pipe, or variations in the batches of liquid flowing through a refinery pipe all alter the acoustic velocity and dispersion characteristics of the waveguide. All these continuous changes lead to covariate drift (discussed in Section 2.2.2) in the recorded signals.

From an ML perspective, the recorded ultrasound signals are the covariates that can serve as inputs to a model, for example, to predicts location of fouling [105, 106]. Any physical change in the transducer locations or the material properties result in a covariate drift, as the high-energy regions of the signal's time-frequency representation shift in the spectrogram space. To address this, our approach performs simulator-based parameter adaptation. Instead of passively collecting data, we leverage a physical forward simulator to re-align our model with the current physical state of the domain. This is inherently more efficient than traditional adaptation because we do not require a new labeled dataset; we simply find the set of physical parameters that maximizes the energy match between our simulation and the observed, drifted signals.

5.2 Problem Setup

The primary goal of this work is to enable the *self-adaptation* of the ultrasound sensing network. Specifically, we aim to automatically adapt and recover the estimated physical configuration parameters of the system (subsequently defined as θ) directly from the observed ultrasonic signals. The main benefit of this data-driven adaptation is that it eliminates the need for precise manual measurements and calibration by engineers during installation or after maintenance.

To accurately model the physical system and facilitate the adaptation, we must establish a rigorous mathematical notation for the sensing network. We consider a network of N transducers positioned on a structure. The fundamental unit of our data collection is the transducer pair. When sensor i transmits an ultrasound signal, the remaining $N-1$ sensors, denoted by j, record the resulting waveforms $s^{i\to j}$ in a synchronized manner. The transmitted energy propagates as multimodal guided ultrasonic waves within the

structure, with trajectories governed by the geometry, boundary conditions, and material properties of the waveguide. A detailed discussion of guided waves is beyond the scope of this thesis; for a comprehensive review of the underlying physics, the reader is referred to the works of Viktorov [107] and Rose [108].

The overall configuration of the sensing system is determined by a joint parameter set $\theta = \{\theta_L, \theta_M, \theta_G\}$, which represents the latent state of the physical domain:

1. **Transducer Locations (θ_L):** We parameterize the sensor locations on a flattened 2D representation of the structure's surface (see Figure 5.1). For a cylindrical pipe, we define the coordinates as $\theta_L = \{(x_i, y_i)\}_{i=1}^{N}$. To remove translational ambiguity, we fix the first transducer as the origin: $(x_1, y_1) = (0, 0)$.

2. **Material Characteristics (θ_M):** These are characterized by the dispersion curve [108] $f^1 : \mathbb{R} \to \mathbb{R}$, which describes the material-specific relationship between the wave packet group velocity and its frequency.

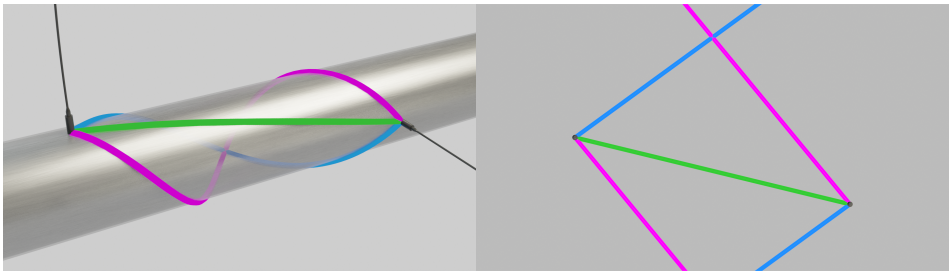
3. **Geometrical Parameters (θ_G):** This accounts for the known geometry of the structure, such as the pipe diameter d.

A critical complexity in hollow cylindrical structures is the existence of multiple trajectories. A signal does not simply travel the straight-line path; it also wraps around the pipe in clockwise and anti-clockwise directions [109]. We collect these trajectory lengths into a vector. This multi-path nature ensures that each transducer pair provides a rich, multi-modal signal that we can exploit for adaptation.

5.3 Differentiable Forward Simulation

Our solution utilizes a dual-branch architecture that aligns empirical observations with a theoretical model. The first branch processes the measurement signals using the Continuous Wavelet Transform (CWT) features. Unlike standard time-domain signals, CWT features $\boldsymbol{S}$ provide a variable time-frequency resolution, allowing us to capture the non-stationary and localized characteristics of ultrasound wave packets [110]. This representation is ideal for our task because it makes the arrival of different frequencies along different trajectories visually and computationally distinct.

[1] In this chapter, f denotes the dispersion curve function. This notation should not be confused with its use in the previous chapters, where it referred to a labeling function.



(a) (b)

Figure 5.1: Illustration of the first three trajectories, namely the ballistic path and the first-order helical paths in both clockwise and counterclockwise directions within a straight pipe between two transducers (a), along with their corresponding 2D projections (b). The sensor positions are parameterized on the 2D surface.

The second branch is the forward simulation, which constructs a differentiable mask M that characterizes the expected times of arrival (TOAs) for a given parameter set θ. Figure 5.2 illustrates a diagram overview of our method. This simulation process is executed in three primary stages: trajectory length calculation, material parameterization, and mask construction.

5.3.1 Trajectory Length Calculation

The first stage in constructing the simulated mask involves determining the spatial distance traversed by the acoustic waves for every observed signal pair $(i \rightarrow j)$. This step is fundamentally dependent on the underlying geometry of the structure. While a simple Euclidean distance would suffice for flat surfaces, complex or enclosed structures require the consideration of multiple propagation paths.

For the straight pipe geometry considered in our experiments, the transmitted signal travels within the pipe wall via multiple helical trajectories. To account for this, the length of the η-th helical path between a pair of sensors is calculated as follows [109]:

$$\begin{aligned} \ell_\eta &= \sqrt{(\Delta\theta_{L_x})^2 + (\Delta\theta_{L_y} + \eta\pi d)^2}, \\ \eta &= -\mathcal{W}, \cdots, -1, 0, 1, \cdots, \mathcal{W} \end{aligned} \tag{5.1}$$

where $\Delta\theta_{L_x}$ and $\Delta\theta_{L_y}$ represent the difference between the x and y coordinates of the transducers in the flattened 2D space, and d is the diameter of

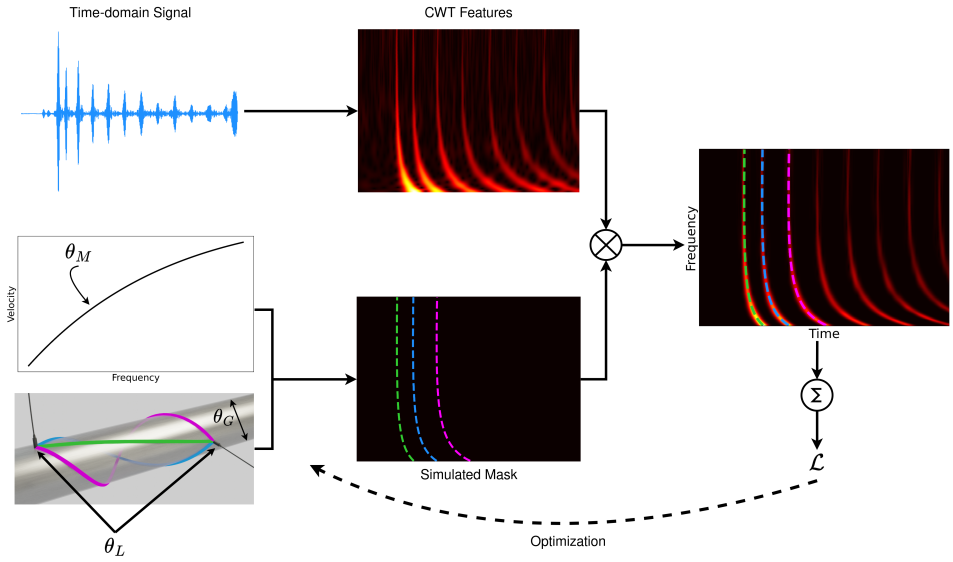


Figure 5.2: Overview of the proposed method. The top branch shows the transformation of a measured time-domain signal into its CWT features, while the bottom branch illustrates the forward simulation used to construct a mask of expected signal TOAs for a given parameter set $\theta = \{\theta_L, \theta_M, \theta_G\}$. The match between observations and simulations is quantified by the objective function $\mathcal{L}$, defined as the sum of the element-wise product between the CWT features and the mask. The black dashed arrow indicates optimization with respect to θ.

the pipe. The index η ranges from $-\mathcal{W}$ to $\mathcal{W}$, accounting for both clockwise and anti-clockwise helical paths up to a specified maximum order.

For each signal, these lengths are collected into a trajectory vector $\boldsymbol{\ell} \in \mathbb{R}^{2\mathcal{W}+1}$. While analytic expressions are utilized for the straight pipe geometry, this stage generalizes to more complex manifolds by replacing the analytic formula with numerical computations of geodesics [111, 112].

5.3.2 Material Parameterization

Once trajectory lengths are established, the wave velocity must be determined to estimate TOAs. Because the group velocity of an acoustic wave packet depends on its frequency, the material properties are parameterized as a free-form function f representing the dispersion curve [113]. This function characterizes the unique velocity-frequency relationship of the waveguide material. Two methods are proposed for this parameterization.

Frequency-Velocity Scaling

This approach builds upon a pre-computed nominal dispersion curve, denoted as f_0, which is typically obtained for the assumed material using theoretical models or numerical software libraries [114, 115]. To compensate for the discrepancy between the idealized theoretical model and the physical reality (often caused by material impurities, surface corrosion, or deviations in wall thickness), the curve is adjusted via two free scaling parameters. The estimated dispersion curve $f(\varphi)$ is defined by

$$f(\varphi) = \beta f_0(\alpha\varphi), \tag{5.2}$$

where φ represents the frequency. In this formulation, α serves as the frequency scaling parameter, which is primarily influenced by the wall thickness of the structure, while β acts as the velocity scaling parameter, depending on the material composition. Empirical results demonstrate that this simple scaling is often sufficient for adapting the model to diverse environments, even when the optimization is initialized with a fundamentally different material assumption (e.g. copper instead of stainless steel), as the underlying shape of dispersion curves for various metals remains structurally similar. Additionally, if the precise dispersion curve is available, we can set $\alpha = \beta = 1$ to effectively disable material optimization.

Gaussian Process

In scenarios where a feasible initial guess for the dispersion curve is unavailable, such as in cases of severe corrosion or unknown waveguide materials, we employ a non-parametric approach. Here, the dispersion curve is modeled as a smooth function using Gaussian process (GP) machinery [116]. We assume that the function follows a GP prior:

$$f(\varphi) \sim \mathcal{GP}\big(m(\varphi), \kappa(\varphi, \varphi')\big), \tag{5.3}$$

where $m(\cdot)$ is the mean function and $\kappa(\cdot, \cdot)$ is the covariance function (kernel). We utilize a squared exponential kernel to quantify the covariance between any two frequencies φ_1 and φ_2:

$$\kappa(\varphi_1, \varphi_2) = \exp\left(-\frac{(\varphi_1 - \varphi_2)^2}{2l^2}\right), \tag{5.4}$$

where the length-scale l controls the assumed smoothness of the function.

Given a set of z noisy observations of frequency-velocity pairs $(\boldsymbol{\varphi}_z, \mathbf{v}_z)$, specifying θ_M, we condition the GP on these data. Assuming a Normal likelihood, the posterior velocity distribution at arbitrary new frequencies $\boldsymbol{\varphi}_*$ is obtained from the standard GP predictive mean ($\boldsymbol{\mu}$) and covariance ($\mathbf{K}$):

$$\mathbf{v}_* \mid \boldsymbol{\varphi}_*, \boldsymbol{\varphi}_z, \mathbf{v}_z \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{K}), \tag{5.5}$$

$$\boldsymbol{\mu} = \mathbf{K}_{*z} \left[\mathbf{K}_{zz} + \sigma^2 \mathbf{I}\right]^{-1} \mathbf{v}_z, \tag{5.6}$$

$$\mathbf{K} = \mathbf{K}_{**} - \mathbf{K}_{*z} \left[\mathbf{K}_{zz} + \sigma^2 \mathbf{I}\right]^{-1} \mathbf{K}_{z*}, \tag{5.7}$$

where $\mathbf{K}_{zz}$ represents the covariance matrix of the z observed points, $\mathbf{K}_{*z}$ ($\mathbf{K}_{z*}$) is the covariance matrix between the prediction and observed points, and $\mathbf{K}_{**}$ is the covariance matrix of the prediction points. In addition, σ^2 represents the observation noise variance, which is assumed to be negligible in the implementation.

5.3.3 Differentiable Mask

The final stage links the physical parameters to the output space via a differentiable mask M. This mask matches the dimensions of the CWT features (Frequency × Time). Each entry in the mask represents the simulation's weight assigned to the expected presence of a signal feature. A value of zero indicates no expected signal feature at that time-frequency point, while positive values reflect increasing simulation weight for a feature's presence there. To enable gradient-based optimization, each mask entry at the time-frequency coordinate (p, q) for a given sensor pair is formally defined as:

$$\boldsymbol{M}_{p,q}^{(i \to j)} = \sum_{\eta=-\mathcal{W}}^{\mathcal{W}} \exp\left(-\gamma \left(\boldsymbol{T}_{p,q} - \frac{\boldsymbol{\ell}_\eta}{\boldsymbol{V}_{p,q}}\right)^2\right). \tag{5.8}$$

Here, $\boldsymbol{T}$ and $\boldsymbol{V}$ are matrices of time steps and velocities, formed by repeating their respective vectors to match the dimensions of the mask $\boldsymbol{M}$. As shown in Figure 5.2, each $\boldsymbol{M}$ also aggregates information across all helical paths.

The parameter γ controls the smoothness of the mask: larger values produce broader masks that capture information over wider regions of the CWT features. In our experiments, we use $\gamma = 10^{10}$; performance is generally insensitive to this choice, provided γ is not too small (as $\gamma \to 0$ yields a binary mask and hinders optimization).

5.3.4 Optimization

Alignment is achieved by maximizing the element-wise product of the mask and the observed CWT features $\boldsymbol{S}$. The optimization minimizes the following loss function:

$$\mathcal{L}(\theta_L, \theta_M, \theta_G) = -\frac{1}{N'} \sum_{\iota=1}^{N'} \boldsymbol{M}_{\iota}^{(i \to j)} \cdot \boldsymbol{S}_{\iota}^{(i \to j)}, \tag{5.9}$$

where N' is the number of observed signal from sensor pairs.

This objective is physically inspired, aiming to find a configuration where the simulated mask aligns with the regions of maximum signal energy in the recorded data (see Figure 5.2). This formulation is also robust against isolated physical phenomena such as destructive interference, where wave packets from different paths might cancel each other out, because the loss function aggregates information across all available trajectories and transducer pairs.

The optimization is performed iteratively using the Adam algorithm. In each iteration, gradients of the loss function are computed with respect to the latent parameters θ by backpropagating through the differentiable forward simulation, followed by a standard parameter update.

5.4 Empirical Validation

This section presents the empirical validation of the proposed self-adaptation framework using two independent ultrasonic sensing configurations. The experiments evaluate the framework's ability to recover the sensing geometry and material properties from measured data and to adapt to changes in sensor deployment.

5.4.1 Experimental Details

To evaluate the proposed self-adaptation framework, we utilize two distinct ultrasonic sensing configurations: a Wireless Sensor Network (WSN) [102] and a Laser-ultrasonic setup [117]. These setups represent independent deployment environments with significant differences in sensing technology. The WSN datasets (WSN1, WSN2) utilize transducers that function as both transmitters and receivers within a relatively narrow frequency band, whereas the Laser dataset employs a non-contact excitation point with a broad frequency range. Detailed descriptions of the electronic components and specific hardware configurations for these setups are available in Paper IV at the end of this thesis.

The measured coordinates for each layout are summarized in Table 5.1. It is important to note that even in controlled laboratory settings, obtaining perfectly accurate ground truth for sensor locations is difficult due to the curvature of the pipe and the finite size of the transducer footprint. Consequently, the measured values should be treated as indicative estimates with an expected margin of error of approximately 1 cm.

To ensure the identifiability of the system, since trajectory lengths are a function of both sensor locations and pipe diameter, we fix the diameter to its manufactured value of 15.4 cm for WSN and 12.7 cm for Laser setups to remove scaling ambiguity. Additionally, we fix the location of the first transducer at $(0, 0)$ as our coordinate origin, as the analysis relies solely on relative positioning.

The CWT features are extracted using frequency ranges tailored to each setup: 280 to 480 kHz for WSN and 100 to 1000 kHz for the Laser setup. In both cases, the frequency range is divided into 150 bins, and a Morlet wavelet with 9 cycles is employed. The sampling frequency is set to 4 MHz for the WSN data and 5 MHz for the Laser experiments. To test the robustness of the method and ensure the discovery of the global minimum, we introduce uniformly distributed noise ranging from 5 to 50 cm to the initial measured coordinates. This randomization of initial coordinates is an integral part of the routine to navigate the non-convex loss surface and guarantee convergence to a global minimum. For the GP material parameterization, we utilize $z = 5$ inducing points at evenly spaced frequencies and a length scale of 5 MHz.

Table 5.1: Experimental datasets used for initial network deployment. Measured and optimized (x, y) coordinates (cm) are given for the flattened pipe geometry, relative to Sensor 1 at $(0, 0)$. In the Laser dataset, Sensor 1 is the Nd:YAG excitation laser; Sensor 2 is the LDV; Sensors 3–4 are broadband transducers.

Datasets	State	Sensor 2	Sensor 3	Sensor 4	Sensor 5
WSN1	Measured	20.0,-30.2	30.0,-24.2	60.0,-36.3	10.0,-24.2
	Optimized	19.7,-30.4	29.7,-24.2	59.2,-36.5	10.1,-24.2
WSN2	Measured	5.0, -22.9	55.0,-13.4	60.0,-33.6	25.0,0.0
	Optimized	5.3, -23.8	54.2,-14.1	59.1,-33.9	24.5,0.0
Laser	Measured	26.0,-10.0	28.0,-10.0	28.0,-29.9	-
	Optimized	26.6,-10.1	27.7,-10.4	27.8,-30.2	-

Table 5.2: The experimental dataset used for the *adaptation* experiment, in which Sensor 5 was manually shifted by 3 cm in the y-direction before *Measured 2*. The measured, adapted, and optimized (x, y) coordinates (cm) are shown for the flattened pipe geometry, relative to the fixed Sensor 1 at $(0, 0)$.

Datasets	State	Sensor 2		Sensor 3		Sensor 4		Sensor 5	
WSN3	Measured 1	0.0,	-24.2	50.0,	-12.1	50.0,	-36.3	25.0,	**0.0**
	Optimized	0.2,	-24.2	49.1,	-12.0	49.2,	-36.3	24.6,	0.0
	Measured 2	0.0,	-24.2	50.0,	-12.1	50.0,	-36.3	25.0,	**-3.0**
	Adapted	0.3,	-24.3	49.3,	-12.5	49.6,	-36.7	24.6,	-3.0

5.4.2 Results

Initial Deployment and Adaptation For the initial deployment phase, the method was evaluated on the WSN1, WSN2, and Laser datasets to estimate parameters for the first time. As summarized in Table 5.1, the optimized coordinates utilizing the frequency-velocity scaling approach for modeling the dispersion curve agree with the measured sensor locations within the expected measurement uncertainty of 1 cm across all setups.

To evaluate the system's capacity for continuous adaptation to operational changes, we created the WSN3 dataset. This dataset was collected several months after the initial deployment, during which time the pipe structure had sustained minor structural damage from unrelated experiments to simulate a realistically degraded environment. In this scenario, Sensor 5 was removed and intentionally re-attached approximately 3 cm away from its original y-coordinate, while all other sensors remained at their measured locations. Upon re-evaluating the objective function using the initial deployment phase parameters, the loss increased from -35.65 to -30.42, effectively flagging that a physical change had occurred. By re-optimizing the configuration starting from the previously known solution, the framework correctly identified the new location of Sensor 5 and maintained the coordinate accuracy for the stationary sensors, as reported in Table 5.2. Minor variations in the coordinates of the stationary sensors are attributed to finite measurement precision, as the hardware-specific resolution of the TOA estimation limits the maximum achievable accuracy to the millimeter scale.

Illustration The overall effectiveness of the self-adaptation framework is summarized in Figure 5.3, which illustrates a single optimization run for the WSN3 dataset. Figure 5.3a depicts the update on the transducer coordi-

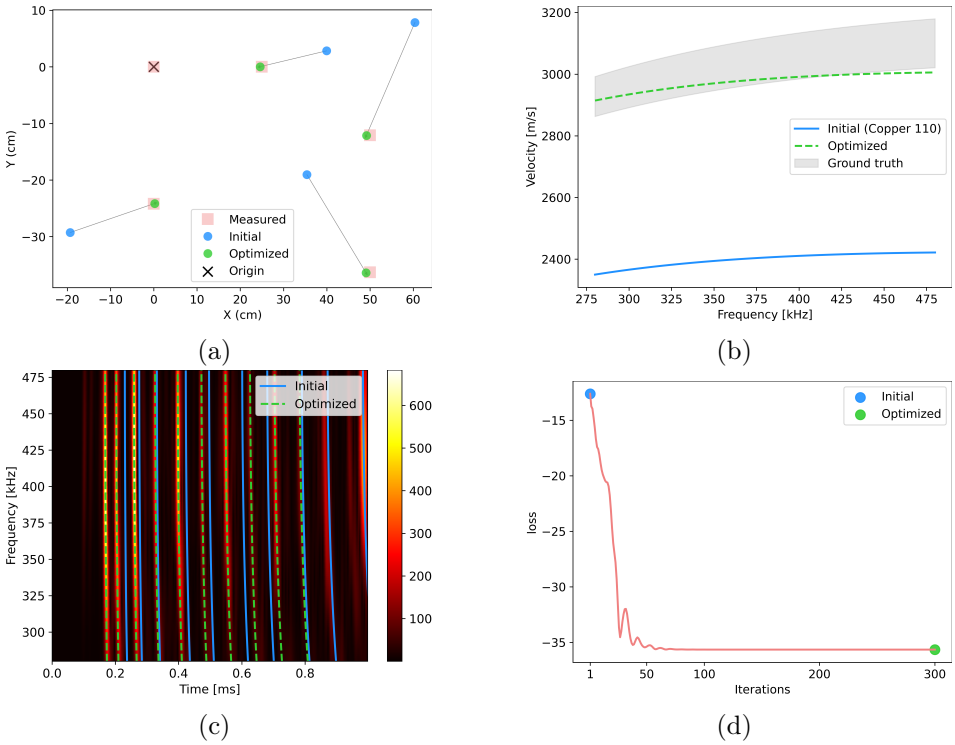


Figure 5.3: Initial (blue) and optimized (green) states for WSN3 dataset: (a) transducer locations relative to a fixed origin (0,0); (b) dispersion curves compared to the theoretical ground truth (gray); (c) simulated mask alignment with observed CWT features; and (d) objective function convergence.

nates from their initial randomized states to the optimized positions, which is aligned with the measured physical locations. The material adaptation is shown in Figure 5.3b, where the estimated dispersion curve, obtained by frequency-velocity scaling approach, converges within the ground truth range calculated for stainless steel. The validity of the final parameter set θ is confirmed in Figure 5.3c, where the simulated mask generated with ten trajectories accurately overlays the high-energy wave packets in the observed CWT features. Finally, Figure 5.3d displays the reduction of the loss function, demonstrating stable convergence over 300 iterations.

Verification of Global Optimum Due to the non-convex nature of the loss surface, where local minima can occur if helical paths are misaligned by one or more orders, we conducted 1200 experiments to verify the probability of reaching the global optimum. These experiments were grouped

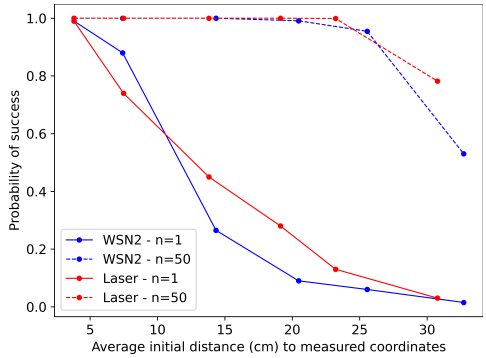


Figure 5.4: Probability of success for single ($n = 1$, solid) and combined ($n = 50$, dashed) optimization runs versus initial coordinate error for the WSN2 and Laser datasets. Results are derived from 1200 total experiments binned into groups of 200.

into bins of 200 and analyzed based on the average initial distance error introduced by random noise. As shown in Figure 5.4, while a single optimization run ($n = 1$) has a diminishing success rate as the initial error increases, performing $n = 50$ runs with different random initializations ensures a success probability close to 1.0 even for initial coordinate errors as large as 20 cm. This corresponds to a relative error of up to 71% in the Laser dataset, highlighting the robustness of the method in recovering the true sensing configuration even with highly vague initial prior information.

Informed & Uninformed Material Priors The flexibility of the material modeling was tested on the broadband Laser dataset using both the frequency-velocity scaling and GP approaches. We initialized the optimization with two different priors: "Steel 1020" as an informed guess and "Copper 110" as an uninformed guess to simulate a complete lack of material knowledge. As shown in Figure 5.5, all combinations of methods and priors successfully converged to the ground truth dispersion curve of the steel pipe. Notably, the marginal differences observed between the dispersion curves estimated by the two methods are primarily a consequence of the limited number of inducing points utilized for the GPs. Additionally, the scaling method initialized with copper produced results identical to those initialized with steel because the dispersion curve of steel resembles a scaled version of copper. While scaling is efficient for most metals, the GP approach provides an alternative for materials with fundamentally different or unknown dispersion shapes.

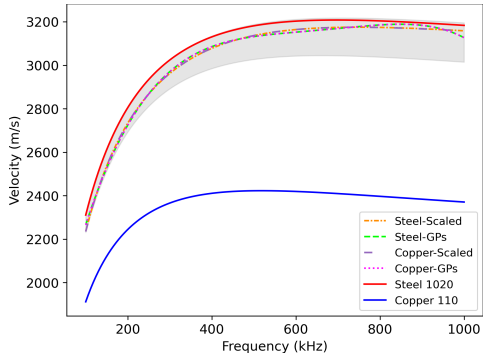


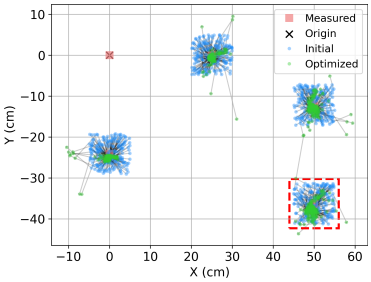
Figure 5.5: Comparison of initial dispersion curves (Steel 1020 and Copper 110) with optimized results from GP and Scaling approaches. The gray region represents the ground truth accounting for manufacturing tolerances.

Value of Helical Paths The impact of utilizing multiple helical trajectories on coordinate precision is demonstrated in Figure 5.6 using the WSN3 dataset. We compared the optimization outcomes using 1, 2, 4, and 10 trajectories to construct the simulated mask. The results in Figures 5.6c confirm that a single trajectory (representing the TOA of the ballistic path) is inadequate for uniquely identifying transducer locations. As the number of helical paths increases, the diversity of the geometric information allows for better estimated coordinates to progressively approach the measured ground truth, as shown in Figure 5.6b. The highest precision was achieved using 10 trajectories, which is the maximum number detectable in our setup before signal overlap with reflections from the pipe edges becomes significant.

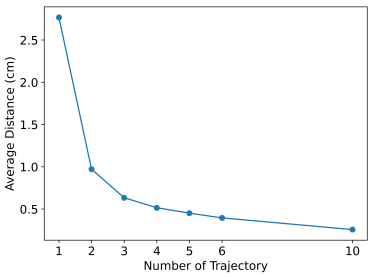
5.4.3 Discussion

The empirical evaluations demonstrate that the self-adaptation framework successfully estimates sensing network configurations during both initial deployment and maintenance-driven re-calibration. The method exhibits robustness across diverse ultrasonic technologies, varying initial estimation errors, and different trajectory counts. Because the algorithm focuses on detecting wave packets, a task fundamentally less complex than quantifying structural damage or fouling, it remains reliable for a wide range of practical sensing configurations.

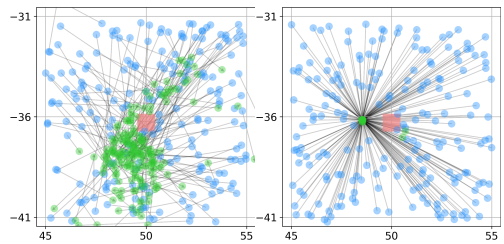
Signal Reflections The forward model currently accounts only for initial propagation paths, intentionally excluding reflections from material bound-



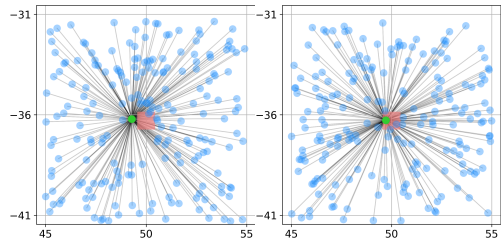
(a) all transducers – 1 trajectory



(b)



(c) 1 trajectory (d) 2 trajectories



(e) 4 trajectories (f) 10 trajectories

Figure 5.6: Influence of trajectory count on WSN3 coordinate precision: (a) global sensor layout; (c–f) magnified views of a single transducer, shown by a frame of red dashed lines in (a) using 1, 2, 4, and 10 trajectories under ±5 cm initial randomization from the measured coordinates; and (b) mean distance between optimized and measured coordinates as a function of the number of trajectories.

aries by truncating the sensing time. This truncation is a strategic choice to eliminate crosstalk from reflections, though it imposes an upper limit on the number of observable helical paths. Note that, direct paths are always observable and provide sufficient information for system identification given an adequate sensor count. Moreover, placing sensors away from reflective boundaries increases the number of helical paths that can be reliably detected. Future iterations of the forward simulation could be expanded to explicitly incorporate reflections as additional information sources.

Dynamic Environments The experiments presented were conducted in static environments using empty pipes. In active industrial settings involving flowing liquids or structural vibrations, the framework will require further refinement. Although the underlying optimization principles are

expected to generalize, adaptation to dynamic environments would likely necessitate signal aggregation over extended periods and specialized extensions to the forward simulation to account for the influence of the pipe contents on recorded signals.

Chapter 6

Conclusion

This chapter concludes the thesis by synthesizing the research findings, critically evaluating the limitations of the proposed methods, and outlining an outlook for the future of adaptive ML systems.

6.1 Summary of Findings

This thesis has provided a unifying perspective on managing diverse data shifts across a spectrum of practical applications. We have framed this challenge as a progression from achieving distributional invariance in latent feature spaces to performing structural adaptation in physical sensing systems.

The first stage of the thesis, detailed in Papers I and II, addressed covariate shift in vision applications, where environmental variations alter the appearance of objects without changing their underlying identity. In Paper I, we introduced a principled geometric framework for explicit alignment. By utilizing Siegel embeddings, we mapped the statistical moments of source and target domains into a SPD manifold. This allowed geometry-aware distances like AIRD to more faithfully align the domains compared to standard Euclidean measures. In Paper II, we evaluated the robustness of the ICT method, a SOTA semi-supervised method, under significant distribution shifts. Our findings revealed that standalone consistency regularization can deteriorate under shift, underperforming even a naive source-only baseline. By integrating adversarial training via a GRL, we demonstrated that consistency and adversarial mechanisms can be synergistically combined to learn domain-invariant representations that are both stable and robust.

In the second stage, presented in Paper III, we moved toward addressing label observation shift, a form of annotation-induced concept shift com-

mon in weakly supervised settings. We addressed the SPMLL problem, where only a single relevant label is observed per instance during training. We showed that leveraging label cardinality, the number of positive labels, serves as a vital structural constraint to recover the true multi-label distribution. Our results indicated that even the crudest tier of cardinality information, referred to as CS in the thesis, significantly improves performance and exhibits high robustness to estimation errors. We also showed that estimating the CD from even a small number of samples can consistently outperform the `CS` baseline. On the NUS and VOC datasets, incorporating CD information allows us to nearly close the performance gap with models trained using full label information. Furthermore, we demonstrated that as the estimation of IC becomes more accurate, the classification performance increases near-linearly.

The final part of the thesis, Paper IV, shifted the focus from model-centric adaptation to system-centric adaptation within physical sensing networks. We addressed covariate drift in ultrasound sensing, where signal distributions can change due to tangible alterations in the sensing structure or environment continuously. By leveraging physical laws governing wave propagation, we developed a self-adaptation framework that is able to detect a shift in the system, and utilize a differentiable forward simulator to re-align the model with its current environment without requiring labeled target data or model retraining. Crucially, this development helps the industry move away from rigid collars and expensive mechanical jigs currently required to ensure static sensor placement [118]. By enabling self-adaptation for distributed sensing networks, our method relaxes installation restrictions and reduces the overall cost of deployment in industrial SHM.

6.2 Limitations

While the methodologies proposed in this thesis offer robust solutions to specific shifts, some limitations remain that suggest the need for more integrated approaches.

A primary limitation of the work in Papers I and II is the assumption of isolated covariate shift. In real-world deployments, it is common for multiple shifts to occur simultaneously, such as a vision system encountering lighting variations (covariate shift) alongside evolving class priors (label shift) and noisy labels (annotation-induced concept shift). Although the standard benchmarks utilized in this study, may inherently contain secondary shifts like label imbalance, our analysis did not explicitly isolate or investigate the impact of these phenomena in detail. Consequently, evaluating the

performance of our framework on datasets with confirmed and concurrent multiple types of shift remains an essential future direction to ascertain its robustness under more intricate, real-world conditions.

Furthermore, our experiments were evaluated on standard benchmark architectures, such as ResNet-50 pre-trained on ImageNet. While our framework is backbone-agnostic, the practical trade-offs between alignment accuracy and the high computational costs of modern backbones, such as Vision Transformers, were not empirically benchmarked. As noted by Goldblum et al. [84], the choice of feature extractor can significantly impact adaptation performance, and the performance-versus-cost trade-off of our adaptation methods on more advanced architectures remains an open empirical question.

Regarding the weak supervision framework in Paper III, we found that while cardinality constraints are highly effective for object-centric datasets, they exhibit limitations on attribute-rich datasets like CUB. We hypothesize that this is due to a low signal-to-noise ratio in the single-positive setting: when unobserved labels far outnumber the only single observed label, the observed signal may not provide sufficient information to distinguish between instances of high and low cardinality. Additionally, while we showed that CD can be estimated from small labeled subsets of data, the practical utility of this estimation remains contingent on the stability of the underlying distribution, which is often violated in unconstrained, non-stationary environments. If the operational setting undergoes a significant dynamic change, the initial cardinality priors may no longer accurately reflect the target domain. Furthermore, in specialized industrial or biomedical applications, obtaining high-quality expert annotations for even a small subset of samples can be prohibitively expensive or practically impossible, leaving the framework without the necessary structural guidance to resolve label ambiguity.

Finally, the self-adaptation framework in Paper IV is currently limited by the complexity of its forward simulation. While the model is currently developed for specific geometric configurations such as straight pipes, it can generalize to any structure where propagation distances can be computed through a differentiable algorithm. However, this involves the numerical computation of geodesics for complex or irregular structures, which can be computationally more intensive. Distinct from these geometric constraints, there is a need for the extension of the adaptive system to account for internal contents, such as flowing liquids, which necessitates more intricate parameterization of the waveguide medium. Furthermore, the forward simulator currently truncates the time window to ignore signal reflections from

the pipe edges. While this successfully avoids complex interference, it fundamentally throws away information, and limits the inability of the forward simulator to use these reflections as additional, valuable trajectories for more robust system identification.

6.3 Outlook

Building on the success of Paper IV, the synergy between physical simulators and ML offers a powerful path forward. An exciting frontier is the use of domain-conditional generative modeling to bridge the "Sim-to-Real" gap. Instead of waiting for real target data, practitioners could use physics-informed or advanced generative models to synthesize realistic target domain samples for a wide range of anticipated environmental conditions. This generation of synthetic targets could provide the cheap, high-fidelity labels needed for continuous alignment in dynamic real-world deployment.

The future of reliable ML lies in adaptation mechanisms that are both computationally efficient and inexpensive to deploy. Traditional adaptation methods that require iterative backpropagation for large ML backbones for every new domain shift are often too expensive for industrial deployment. While Test-Time Adaptation (TTA) [119] is a step in the right direction, many current TTA methods still require backpropagation at test time, which can be prohibitive for real-time applications. Future work could place more emphasize on methods that can update their behavior during a single forward pass, perhaps by adjusting lightweight, modular adapters to reconcile shifts in real-time.

Finally, the ultimate goal of this research trajectory is the move from DA toward domain generalization. The necessity of tuning adaptation hyperparameters on the target domain often limits the "out-of-the-box" utility of these methods. The future lies in developing models that are intrinsically robust to shift through the development of models that do not require adaptation because they have already internalized the structural and physical invariants of the world, ensuring reliable operation across any environment they may encounter.

References

[1] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *International Conference on Artificial Intelligence and Statistics*, 2017.

[2] Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, and Jorge Reyes. A public domain dataset for human activity recognition using smartphones. In *International European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, 2013.

[3] Zahraa S. Abdallah, Mohamed Medhat Gaber, Bala Srinivasan, and Shonali Krishnaswamy. Activity recognition with evolving data streams: A review. *ACM Computing Surveys*, 51(4):71, 2018.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016.

[5] Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Qiang Cheng, Guoliang Chen, Jie Chen, Jingdong Chen, Zhijie Chen, Mike Chrzanowski, Adam Coates, Greg Diamos, Ke Ding, Niandong Du, Erich Elsen, Jesse Engel, Weiwei Fang, Linxi Fan, Christopher Fougner, Liang Gao, Caixia Gong, Awni Hannun, Tony Han, Lappi Johannes, Bing Jiang, Cai Ju, Billy Jun, Patrick LeGresley, Libby Lin, Junjie Liu, Yang Liu, Weigao Li, Xiangang Li, Dongpeng Ma, Sharan Narang, Andrew Ng, Sherjil Ozair, Yiping Peng, Ryan Prenger, Sheng Qian, Zongfeng Quan, Jonathan Raiman, Vinay Rao, Sanjeev Satheesh, David Seetapun, Shubho Sengupta, Kavya Srinet, Anuroop Sriram, Haiyuan Tang, Liliang Tang, Chong Wang, Jidong Wang, Kaifu Wang, Yi Wang, Zhijian Wang, Zhiqian

Wang, Shuang Wu, Likai Wei, Bo Xiao, Wen Xie, Yan Xie, Dani Yogatama, Bin Yuan, Jun Zhan, and Zhenyao Zhu. Deep speech 2: End-to-end speech recognition in English and Mandarin. In *International Conference on Machine Learning*, 2016.

[6] Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M. Summers. ChestX-Ray8: Hospital-Scale Chest X-Ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017.

[7] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel Ni, and Heung-Yeung Shum. DINO: DETR with improved denoising anchor boxes for end-to-end object detection. In *International Conference on Learning Representations*, 2023.

[8] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. Segment anything. In *IEEE/CVF International Conference on Computer Vision*, 2023.

[9] Olaf Dünkel, Artur Jesslen, Jiaohao Xie, Christian Theobalt, Christian Rupprecht, and Adam Kortylewski. CNS-Bench: Benchmarking image classifier robustness under continuous nuisance shifts. In *IEEE/CVF International Conference on Computer Vision*, 2025.

[10] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, 2012.

[11] Hang Qiu, Krishna Chintalapudi, and Ramesh Govindan. MCAL: Minimum cost human-machine active labeling. In *International Conference on Learning Representations*, 2023.

[12] Artur Yakimovich, Anaël Beaugnon, Yi Huang, and Elif Ozkirimli. Labels in a haystack: Approaches beyond supervised learning in biomedical applications. *Patterns*, 2(12):100383, 2021.

[13] Bernhard Kratzwald, Stefan Feuerriegel, and Huan Sun. Learning a Cost-Effective Annotation Policy for Question Answering. In *Conference on Empirical Methods in Natural Language Processing*, 2020.

[14] Jesper E. van Engelen and Holger H. Hoos. A survey on semi-supervised learning. *Machine Learning*, 109(2):373–440, 2020.

[15] Joaquín Quiñonero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D. Lawrence, editors. *Dataset Shift in Machine Learning*. The MIT Press, 2008.

[16] Annamaria Mesaros, Toni Heittola, Tuomas Virtanen, and Mark D. Plumbley. Sound event detection: A tutorial. *IEEE Signal Processing Magazine*, 38(5):67–83, 2021.

[17] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019.

[18] Zhi-Hua Zhou. A brief introduction to weakly supervised learning. *National Science Review*, 5(1):44–53, 2018.

[19] Elijah Cole, Oisin Mac Aodha, Titouan Lorieul, Pietro Perona, Dan Morris, and Nebojsa Jojic. Multi-label learning from single positive labels. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[20] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Joshua Susskind, Wenda Wang, and Russell Webb. Learning from simulated and unsupervised images through adversarial training. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017.

[21] Konstantinos Bousmalis, Alex Irpan, Paul Wohlhart, Yunfei Bai, Matthew Kelcey, Mrinal Kalakrishnan, Laura Downs, Julian Ibarz, Peter Pastor, Kurt Konolige, Sergey Levine, and Vincent Vanhoucke. Using simulation and domain adaptation to improve efficiency of deep robotic grasping. In *IEEE International Conference on Robotics and Automation*, 2018.

[22] Xingchao Peng, Ben Usman, Neela Kaushik, Dequan Wang, Judy Hoffman, and Kate Saenko. VisDA: A synthetic-to-real benchmark for visual domain adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2018.

[23] Vladimir N. Vapnik. An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, 10(5):988–999, 1999.

[24] Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. Analysis of representations for domain adaptation. In *Advances in Neural Information Processing Systems*, 2006.

[25] Miquel Calvo and Josep M. Oller. A distance between multivariate normal distributions based in an embedding into the Siegel group. *Journal of Multivariate Analysis*, 35(2):223–242, 1990.

[26] Vikas Verma, Kenji Kawaguchi, Alex Lamb, Juho Kannala, Arno Solin, Yoshua Bengio, and David Lopez-Paz. Interpolation consistency training for semi-supervised learning. *Neural Networks*, 145:90–106, 2022.

[27] Christopher M. Bishop. *Pattern recognition and machine learning.* Springer, 2006.

[28] V. Vapnik. Principles of risk minimization for learning theory. In *Advances in Neural Information Processing Systems*, 1991.

[29] Olivier Chapelle, Vikas Sindhwani, and Sathiya S. Keerthi. Optimization techniques for semi-supervised support vector machines. *Journal of Machine Learning Research*, 9(7):203–233, 2008.

[30] Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien, editors. *Semi-Supervised Learning.* The MIT Press, 2006.

[31] Zoubin Ghahramani. Unsupervised learning. In *Advanced Lectures on Machine Learning*, pages 72–112. Springer, 2004.

[32] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2014.

[33] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeshwar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and Devon Hjelm. Mutual information neural estimation. In *International Conference on Machine Learning*, 2018.

[34] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *International Conference on Machine Learning*, 2008.

[35] Hakan Bilen and Andrea Vedaldi. Weakly supervised deep detection networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016.

[36] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *Advances in Neural Information Processing Systems*, 2018.

[37] Thibaut Durand, Nazanin Mehrasa, and Greg Mori. Learning a deep convnet for multi-label classification with partial labels. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[38] R. Lippmann. An introduction to computing with neural nets. *IEEE ASSP Magazine*, 4(2):4–22, 1987.

[39] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. The MIT Press, 2016.

[40] Yann Lecun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[41] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

[42] Atılım Günes Baydin, Barak A. Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. Automatic differentiation in machine learning: a survey. *Journal of Machine Learning Research*, 18(1):5595–5637, 2018.

[43] Léon Bottou. On-line learning and stochastic approximations. In *On-Line Learning in Neural Networks*, pages 9–42. Cambridge University Press, 1999.

[44] Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. In *International Conference on Learning Representations*, 2017.

[45] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

[46] Stephen Hanson and Lorien Pratt. Comparing biases for minimal network construction with back-propagation. In *Advances in Neural Information Processing Systems*, 1988.

[47] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014.

[48] Lutz Prechelt. Automatic early stopping using cross validation: quantifying the criteria. *Neural Networks*, 11(4):761–767, 1998.

[49] Marvin Mengxin Zhang, Henrik Marklund, Nikita Dhawan, Abhishek Gupta, Sergey Levine, and Chelsea Finn. Adaptive risk minimization: Learning to adapt to domain shift. In *Advances in Neural Information Processing Systems*, 2021.

[50] Jose G. Moreno-Torres, Troy Raeder, Rocío Alaiz-Rodríguez, Nitesh V. Chawla, and Francisco Herrera. A unifying view on dataset shift in classification. *Pattern Recognition*, 45(1):521–530, 2012.

[51] Bianca Zadrozny. Learning and evaluating classifiers under sample selection bias. In *International Conference on Machine Learning*, 2004.

[52] Judy Hoffman, Trevor Darrell, and Kate Saenko. Continuous manifold based adaptation for evolving visual domains. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2014.

[53] Xiyu Yu, Tongliang Liu, Mingming Gong, Kun Zhang, Kayhan Batmanghelich, and Dacheng Tao. Label-noise robust domain adaptation. In *International Conference on Machine Learning*, 2020.

[54] Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244, 2000.

[55] João Gama, Indrė Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. A survey on concept drift adaptation. *ACM Computing Surveys*, 46(4):44, 2014.

[56] David J. Hand. Classifier Technology and the Illusion of Progress. *Statistical Science*, 21(1):1–14, 2006.

[57] Geoffrey I. Webb, Roy Hyde, Hong Cao, Hai Long Nguyen, and Francois Petitjean. Characterizing concept drift. *Data Mining and Knowledge Discovery*, 30(4):964–994, 2016.

[58] Kuniaki Saito, Donghyun Kim, Stan Sclaroff, Trevor Darrell, and Kate Saenko. Semi-supervised domain adaptation via minimax entropy. In *IEEE/CVF International Conference on Computer Vision*, 2019.

[59] Han Zhao, Shanghang Zhang, Guanhang Wu, José M. F. Moura, Joao P. Costeira, and Geoffrey J. Gordon. Adversarial multiple source domain adaptation. In *Advances in Neural Information Processing Systems*, 2018.

[60] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(1):2096–2030, 2016.

[61] Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? Source hypothesis transfer for unsupervised domain adaptation. In *International Conference on Machine Learning*, 2020.

[62] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine Learning*, 79(1):151–175, 2010.

[63] Han Zhao, Remi Tachet Des Combes, Kun Zhang, and Geoffrey Gordon. On learning invariant representations for domain adaptation. In *International Conference on Machine Learning*, 2019.

[64] Vincent Arsigny, Pierre Fillard, Xavier Pennec, and Nicholas Ayache. Geometric means in a novel vector space structure on symmetric positive-definite matrices. *SIAM Journal on Matrix Analysis and Applications*, 29(1):328–347, 2007.

[65] Frank Nielsen. Fisher-Rao and pullback Hilbert cone distances on the multivariate Gaussian manifold with applications to simplification and quantization of mixtures. In *Workshop on Topology, Algebra, and Geometry in Machine Learning*, 2023.

[66] Rajendra Bhatia. Geometry of positive matrices. In *Positive Definite Matrices*, pages 201–236. Princeton University Press, 2007.

[67] Andrew V. Knyazev. Toward the optimal preconditioned eigensolver: Locally optimal block preconditioned conjugate gradient method. *SIAM Journal on Scientific Computing*, 23(2):517–541, 2001.

[68] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European Conference on Computer Vision – Workshops*, 2016.

[69] Werner Zellinger, Thomas Grubinger, Edwin Lughofer, Thomas Natschläger, and Susanne Saminger-Platz. Central moment discrepancy (CMD) for domain-invariant representation learning. In *International Conference on Learning Representations*, 2017.

[70] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *European Conference on Computer Vision*, 2010.

[71] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision*, 2014.

[72] Esteban Real, Jonathon Shlens, Stefano Mazzocchi, Xin Pan, and Vincent Vanhoucke. Youtube-boundingboxes: A large high-precision human-annotated data set for object detection in video. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017.

[73] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016.

[74] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[75] Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474*, 2014.

[76] Pietro Morerio, Jacopo Cavazza, and Vittorio Murino. Minimal-entropy correlation alignment for unsupervised deep domain adaptation. In *International Conference on Learning Representations*, 2018.

[77] Chao Chen, Zhihang Fu, Zhihong Chen, Sheng Jin, Zhaowei Cheng, Xinyu Jin, and Xian-Sheng Hua. HoMM: Higher-order moment

matching for unsupervised domain adaptation. In *AAAI Conference on Artificial Intelligence*, 2020.

[78] Olivier Ledoit and Michael Wolf. Honey, I shrunk the sample covariance matrix. *Journal of Portfolio Management*, 30(4):110–119, 2004.

[79] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *International Conference on Learning Representations*, 2017.

[80] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in Neural Information Processing Systems*, 2017.

[81] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018.

[82] Jonathan J. Hull. A database for handwritten text recognition research. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(5):550–554, 1994.

[83] Pablo Arbeláez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(5):898–916, 2011.

[84] Micah Goldblum, Hossein Souri, Renkun Ni, Manli Shu, Viraj Uday Prabhu, Gowthami Somepalli, Prithvijit Chattopadhyay, Mark Ibrahim, Adrien Bardes, Judy Hoffman, Rama Chellappa, Andrew Gordon Wilson, and Tom Goldstein. Battle of the backbones: A large-scale comparison of pretrained models across computer vision tasks. In *Advances in Neural Information Processing Systems*, 2023.

[85] Donghao Zhou, Pengfei Chen, Qiong Wang, Guangyong Chen, and Pheng-Ann Heng. Acknowledging the unknown for multi-label learning with single positive labels. In *European Conference on Computer Vision*, 2022.

[86] Donghao Zhou, Pengfei Chen, Qiong Wang, Guangyong Chen, and Pheng-Ann Heng. Acknowledging the unknown for multi-label learning with single positive labels. In *European Conference on Computer Vision*, 2022.

[87] Ming-Kun Xie, Jiahao Xiao, and Sheng-Jun Huang. Label-aware global consistency for multi-label learning with single positive labels. In *Advances in Neural Information Processing Systems*, 2022.

[88] Biao Liu, Ning Xu, Jiaqi Lv, and Xin Geng. Revisiting pseudo-label for single-positive multi-label learning. In *International Conference on Machine Learning*, 2023.

[89] Yanxi Chen, Chunxiao Li, Xinyang Dai, Jinhuan Li, Weiyu Sun, Yiming Wang, Renyuan Zhang, Tinghe Zhang, and Bo Wang. Boosting single positive multi-label classification with generalized robust loss. In *International Joint Conference on Artificial Intelligence*, 2024.

[90] Hisham Cholakkal, Guolei Sun, Fahad Shahbaz Khan, and Ling Shao. Object counting and instance segmentation with image-level supervision. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[91] Novi Quadrianto, Alex J. Smola, Tibério S. Caetano, and Quoc V. Le. Estimating labels from label proportions. *Journal of Machine Learning Research*, 10(82):2349–2374, 2009.

[92] Jasper R. R. Uijlings, Koen E. A. van de Sande, Theo Gevers, and Arnold W. M. Smeulders. Selective search for object recognition. *International Journal of Computer Vision*, 104(2):154–171, 2013.

[93] Harold W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955.

[94] Sanjeev Arora. Nearly linear time approximation schemes for Euclidean TSP and other geometric problems. In *Annual Symposium on Foundations of Computer Science*, 1997.

[95] Pankaj Agarwal and Kasturi Varadarajan. A near-linear constant-factor approximation for Euclidean bipartite matching? In *Annual Symposium on Computational Geometry*, 2004.

[96] Youngwook Kim, Jae Myung Kim, Jieun Jeong, Cordelia Schmid, Zeynep Akata, and Jungwoo Lee. Bridging the Gap Between Model Explanations in Partially Annotated Multi-Label Classification. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[97] Mark Everingham, S. M. Ali Eslami, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The Pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136, 2015.

[98] Tat-Seng Chua, Jinhui Tang, Richang Hong, Haojie Li, Zhiping Luo, and Yantao Zheng. NUS-WIDE: a real-world web image database from National University of Singapore. In *ACM International Conference on Image and Video Retrieval*, 2009.

[99] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The Caltech-UCSD Birds-200-2011 dataset. Technical report, California Institute of Technology, USA, 2011.

[100] Tomasz Wandowski, Paweł H. Malinowski, and Wiesław M. Ostachowicz. Circular sensing networks for guided waves based structural health monitoring. *Mechanical Systems and Signal Processing*, 66-67:248–267, 2016.

[101] Ayoub Keshmiry, Sahar Hassani, Mohsen Mousavi, and Ulrike Dackermann. Effects of environmental and operational conditions on structural health monitoring and non-destructive testing: A systematic review. *Buildings*, 13(4):918, 2023.

[102] Julius Korsimaa, Martin Weber, Petteri Salminen, Joonas Mustonen, Denys Iablonskyi, Edward Hæggström, Arto Klami, and Ari Salmi. Wireless and battery-operatable iot platform for cost-effective detection of fouling in industrial equipment. *Scientific Reports*, 14(1):14084, 2024.

[103] Haiqing Yao, Heng Cao, and Jin Li. Design and implementation of a portable wireless system for structural health monitoring. *Measurement and Control*, 49(1):23–32, 2016.

[104] Arka P. Reksowardojo, Gennaro Senatore, and Ian F. C. Smith. Design of structures that adapt to loads through large shape changes. *Journal of Structural Engineering*, 146(5):04020068, 2020.

[105] Burla Nur Korkmaz, Shayan Gharib, Haoyu Wei, Julius Korsimaa, Petteri Salminen, Martin Weber, Denys Iablonskyi, Ari Salmi, and Arto Klami. Physical simulator-based neural networks for real-time fouling tomography. *ACM Transactions on Probabilistic Machine Learning*, 2026. Accepted.

[106] Eva Wallhäußer, Mohammed A. Hussein, and Thomas Becker. Detection methods of fouling in heat exchangers in the food industry. *Food Control*, 27(1):1–10, 2012.

[107] Igor A. Viktorov. *Rayleigh and Lamb Waves: Physical Theory and Applications*. Springer US, 2013.

[108] Joseph L. Rose. *Ultrasonic Guided Waves in Solid Media*. Cambridge University Press, 2014.

[109] Ehsan Dehghan-Niri and Salvatore Salamone. A multi-helical ultrasonic imaging approach for the structural health monitoring of cylindrical structures. *Structural Health Monitoring*, 14(1):73–85, 2015.

[110] Praveen Kumar and Efi Foufoula-Georgiou. Wavelet analysis for geophysical applications. *Reviews of Geophysics*, 35(4):385–412, 1997.

[111] Carlos-Omar Rasgado-Moreno and Madis Ratassepp. Geodesic equations for guided wave helical path separation for a pipe bend. *Mechanical Systems and Signal Processing*, 204:110820, 2023.

[112] Denys Iablonskyi, Carlos-Omar Rasgado-Moreno, Madis Ratassepp, Arto Klami, Edward Hæggström, and Ari Salmi. Unsupervised fouling reconstruction in the pipe bend. In *International Ultrasonics Symposium*, 2023.

[113] Ka Lok Jimmy Fong. *A study of curvature effects on guided elastic waves*. PhD thesis, University of London, UK, 2005.

[114] Armin M. A. Huber and Markus G. R. Sause. Classification of solutions for guided waves in anisotropic composites with large numbers of layers. *The Journal of the Acoustical Society of America*, 144(6):3236–3251, 2018.

[115] Brian Pavlakovic, Mike Lowe, David Alleyne, and Peter Cawley. Disperse: A general purpose program for creating dispersion curves. In *Review of Progress in Quantitative Nondestructive Evaluation*, pages 185–192. Springer US, 1997.

[116] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2005.

[117] Joonas Mustonen, Denys Iablonskyi, Shayan Gharib, Joonas Suorsa, Martin Weber, Arto Klami, Edward Hæggström, and Ari Salmi. Noncontacting laser-ultrasonic fouling detection on steel pipes. *Ultrasonics*, 151:107617, 2025.

[118] Habiba Lais, Premesh S. Lowe, Tat-Hean Gan, Luiz C. Wrobel, and Jamil Kanfoud. Characterization of the use of low frequency ultrasonic guided waves to detect fouling deposition in pipelines. *Sensors*, 18(7):2122, 2018.

[119] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2021.