Domain Adaptation for Medical Image Classification

Inter-Disciplinary Dual Degree Project: Phase 1

Submitted in partial fulfillment of the requirements for the degree of

Bachelor of Science in **Mathematics**

and

Master of Technology in **Centre for Digital Health**

Chirag P

(18B090003)

Guide: **Prof. Amit Sethi**

Department of Electrical Engineering

October 11, 2022

Contents

1 Introduction 3

2 Divergences in Probability measures 3

2.1 f-Divergences 4

2.2 Integral Product Measures 4

3 Loss functions in generalized UDA 5

4 Data-set description 5

5 Gradually Vanishing Bridge 8

6 Minimum Class Confusion 9

7 Maximum Density Divergence 10

8 Domain Conditioned Adaptation Network 11

9 Experiments - UDA 12

9.1 Dataset and Settings 12

9.2 Baseline on FHIST 13

9.3 DCAN on FHIST 13

9.4 GVB on X-ray data-sets 14

9.5 GVB and its variations on FHIST 14

9.5.1 Ideating on Loss functions 14

9.5.2 Experiments 15

10 Observations based on UDA experiments 16

11 Experiments - SSDA 17

12 Observations based on SSDA experiments 18

13 Grad-Cam visualizations 19

1 Introduction

Domain adaptation is a transfer learning method which aims to make classification and other tasks more consistent with data from multiple sources. In this setting, we consider different domains to be different centres or scanners which provided the image. Domain adaptation methods are of three types: Adversarial, Distance-based and Optimal Transport based. However, other innovative methods have also been successful in ameliorating the domain discrepancy between two or more domains. Domain adaptation techniques on medical image data-sets have been scarce. Only shallow domain adaptation methods have been employed, with CyCADA (Cycle-Consistent Adversarial Domain Adaptation), a very primitive algorithm, being a bench-marked technique in radiology. However, there is a need to apply domain adaptation techniques on histopathology data-sets, which have a lot of inherent randomness in terms of where the attention is being given, during classification. In this document, we experiment with state of the art unsupervised domain adaptation techniques like DANN [9], CDAN [10] and GVB [1] on three types of medical image data-sets- Retinal image Photography images, X-Ray images and Colon Cancer histopathology images, and employ a loss function called "Island loss" used in facial recognition. Other methods are also experimented. They include MDD [3] and DCAN [4], and a first in class approach called Minimum Class Confusion [2] by back-propagating their losses.

First, we conduct a study of loss functions used in the domain adaptation setting. We also benchmark adversarial and distance based methods and propose that adversarial methods perform better for the FHIST data-set, a few shot domain adaptation data-set with two domains (CRC and NCT). On the retinal image data-set, unsupervised adversarial methods also beat state of the art semi-supervised adversarial methods in the case of severe class imbalance. Potential reasons are also proposed for it. As an interesting segue, temperature conditioning in the Minimum Class Confusion has also been experimented with, and better results were yielded. Throughout, SSDA stands for Semi-Supervised Domain Adaptation and UDA stands for Unsupervised Domain Adaptation.

2 Divergences in Probability measures

Let (Ω, F, P_1) and (Ω, F, P_2) be two probability measures on the same underlying space. Then, the divergence between the probability measures P_1 and P_2 gives a measure of how much the two probability distributions assign different masses or densities to different points in Ω. A popular measure of the divergence is the $KL-$ Divergence.

The three broad types of probability divergences are given below. We will discuss the first two, since they are the most relevant.

- f-divergence (KL Divergence, Jensen Shannon Divergence)
- Integral Product Measures (Wasserstein Distance, Maximum Mean Discrepancy)
- H-divergence

2.1 f-Divergences

Given any convex function $f : \mathbb{R} \to \mathbb{R}$ which is continuous such that $f(1) = 0$. Then, an f-divergence can be formulated as $D_f(p||q) = \mathbb{E}_q(f(p(X), q(X)))$ where p, q are PDFs/PMFs on the underlying space Ω. Then sigma field is defined to be F by default.

Some examples are given below:

- $t \to t \cdot log(t)$ (KL Divergence)
- $t \to (t+1) \cdot log(\frac{2}{t+1}) + t \cdot log(t)$ (Jensen Shannon Divergence)
- $t \to t^2 \cdot log(t)$
- $t \to t^3 \cdot log(t)$

2.2 Integral Product Measures

Let p, q be PDFs/PMFs on the underlying space Ω. Let $\Im$ be a space of functions defined from $\mathbb{R} \to \mathbb{R}$.
Then, $IPM_{\Im}(p||q) = \sup_{f\in\Im}[\mathbb{E}_p(f(X) - \mathbb{E}_q(f(X)]$. Trivially, if $f == Id$ and for $Id \in \Im$, the IPM becomes the maximum mean discrepancy between the two distributions.
Similarly, if $\Im$ is chosen to be the set of 1-Lipschitz functions, the IPM reduces to the Wasserstein Distance.

We see extensive usage of the KL divergence to measure the domain shift, in the case of distance-based unsupervised domain adaptation methods, as in case of DCAN

(Domain-Conditioned Adaptation Networks) [4] and MDD (Maximum Density Divergence) [3]. IPMs are also very common, although they do not yield results as good. Deep CORAL (Deep-Correlation Alignment) [12], MMD and its variants [7], HoMM (Higher-order Moment Matching) [13] are all in essence, using an IPM to measure domain discrepancy. Even DCAN uses a variant of MMD. However, we observe that the best performing unsupervised methods on our data-sets are adversarial in nature. Hence, although domain discrepancy measures exist so that we may appreciate the domain shifts, we opine that minimizing them directly will not yield state of the art results.

3 Loss functions in generalized UDA

There are two types of losses that aim to reduce domain shift and make the classifier more resistant to data from different sources: Adversarial losses like domain confusion loss and Domain discrepancy losses. The classification loss is also present in the backdrop of the domain alignment problem, since we seek to utilize the source classifier on the target. The general problem of domain adaptation can be formulated as:

$$\min_{G_*}[L_{cls} + L^{adv}_{trans} + L_{ext}]$$

$$\max_{D_*}[L^{adv}_{trans}]$$

L_{cls} is the classifier loss on the source domain and L^{adv}_{trans} is the adversarial transfer loss.

This is a two-step optimization problem, commonly referred to as a **mini-max** problem in mathematical literature. In the first step, we freeze the parameters of D_*, the discriminator and update the parameters of G_* through back-propogation. Subsequently, we freeze the parameters of G_* and update only those of D_*. Previously state of the art methods like DANN [9], DCAN [4] , MDD [3], MCC [2], MCD [8], CDAN [10] and GVB [1] use their own characteristic external loss functions.

4 Data-set description

The problem of domain adaptation is entirely data-set dependent. We use three medical image data-sets from different domains of radiography and imaging, apart from natural image data-sets. In general, radio-images taken from two different scanners are bereft of severe domain shifts. This is maybe because of the sensitivity of the X-Ray/ CT sensor to the X-rays and because of the uniformity in body compositions of individuals. We indeed validate this statement. However, it was not so in the case of the other two data-sets. We are also currently working on patch-extraction on a more complex dataset called the CAMELYON17.

- **X-ray data-set**: The first X-ray data-set comprises chest X-ray images selected from retrospective cohorts of pediatric patients of one to five years old from Guangzhou Women and Children's Medical Center, Guangzhou. The second X-ray data-set was compiled by a team of researchers from Qatar University, Doha, Qatar, and the University of Dhaka, Bangladesh along with their collaborators from Pakistan and Malaysia in collaboration with doctors. We train our models with a ResNet backbone on the latter, and adapt it onto the former. Both are open-source and are available on Kaggle.
- **RFMiD data-set**: RFMiD is a new publicly available retinal images data-set with presence of 45 different types of pathologies. Three different cameras have been used: The TOPCON 3D OCT-2000, Kowa VX-10a TOPCON and TRC-NW 300. These cameras have different resolutions, and make up 2427, 467 and 306 of all the images. Retinas examined through different cameras may conjure a domain shift in the process. Our initial goal was to group all the pathological retinas as one, and normal retinas as another, and apply semi-supervised and unsupervised domain adaptation between images from different cameras.
- **FHIST data-set**: It was originally curated for few-shot classification of near-domain target samples, where the source domain is CRC-TP data-set and (near-domain) target is NCT-CRC-HE-100K (NCT). It consists of colorectal cancer histology images from two different domains, with 6 classes: Benign, Muscle, Stroma, Inflammatory, Debris and Tumor. For each class, we have close to 20,000 labeled patches in the source domain, and around 10,000 unlabeled patches in the target domain.

  - **Benign** class consists of normal epithelial cells which line the colon.
  - **Muscle** class consists of normal muscle tissue found in the colon, like sphincter muscle tissue.

  - **Stroma** class consists of precursor cells which develop into muscle tissue.
  - **Inflammatory** class consists of cells which are marked by inflammation markers like presence of eosinophils, lymphocytes and neutrophils. Common causes of inflammation in the colon are Lactose intolerance, Crohn's disease, and IBS.
  - **Debris** class consists of organic waste materials preceded by the apoptosis of cells. they also consist of cells locked in interphase and the G-metaphase.
  - **Tumor** class consists of cells marked by either dysplasia, metaplasia or metastasis in the epithelial cells. The most common origin of colon cancer is adenocarcinoma, a cancer starting in the glandular epithelial cells which line the colon.

- **CAMELYON17**: This data-set contains lymph node biopsy images from 5 different centres. At the WSI (Whole-Slide Image) level, three types of metastases are distinguished, based on size: macro-metastases, micro-metastases and ITC (isolated tumor cells).

  - **Macro-metastasis** : Larger than 2 mm
  - **Micro-metastasis** : Larger than 0.2 mm or containing more than 200 cells, but not larger than 2 mm
  - **Isolated tumor cells** : Single tumor cell or a cluster of 200 or less cells, and tumor region smaller than 0.2 mm

  If tumor cells are found, the pathologist measures their extent in order to determine the pathological N stage (pN-stage) of the tumor. PN-Stage Slide Labels

  - **PN0** : No micro-metastases or macro-metastases or ITC found.
  - **PN0(i+)** : No micro-metastases or macro-metastases or ITC found.
  - **PN1mi** : Micro-metastases found, but no macro-metastases found.
  - **PN1** : Metastases found in 1-3 lymph nodes, of which at least 1 is a macro-metastasis
  - **PN2** : Metastases found in 4-5 lymph nodes, of which at least 1 is a macro-metastasis.

  On the CAMELYON17 data-set, as a first, we are proceeding with patch extraction of metastasized patches taken from WSI images. We successfully extracted patches marked as "metastasized".

5 Gradually Vanishing Bridge

Gradually Vanishing Bridge for Adversarial Domain Adaptation (GVB) is a technique for unsupervised domain adaptation that encourages domain-invariant feature extraction. The main reason for discrepancy between domains is the existence of domain-specific features in the latent space. GVB aims to reduce this discrepancy by using bridges to subtract domain-specific features from the representation of an image. The bridges are used on the generator (GVB-G) as well as the discriminator (GVB-D). The end-to-end framework of the GVB is shown below:

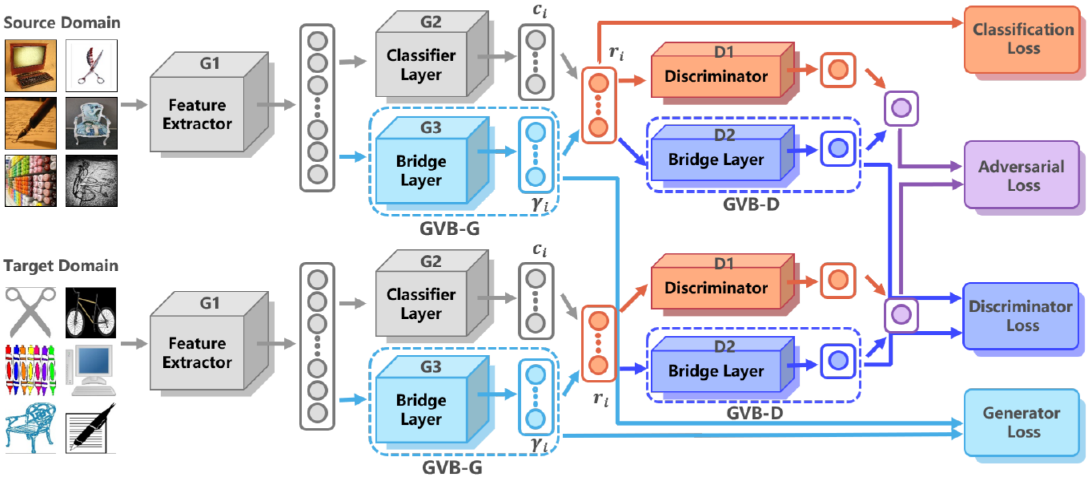


Figure 1: GVB Framework

The entire network is shared by source and target domains. The feature extractor G1 is a ResNet50, whose output is fed into the classifier (G2) and bridge (G3) layers. The output of G3 is γ_i - the domain specific features, which are subtracted from the classifier output c_i to yield domain-invariant features r_i. The r_i vectors from source and target images are then passed on to the discriminator D1 and its corresponding bridge D2, that provides additional discriminating ability to the model. Three kinds of loss functions are used here:

1. **Classification loss**: This is the cross-entropy loss, calculated over the source domain images only. The mathematical formulation is given below, where L_{ce} is cross entropy, $G_\star$ is the operator that results in class responses, and (x_i^s, y_i^s) is a source domain sample and its label.

$$L_{cls} = \frac{1}{N_s}\sum_{i=1}^{N_s} L_{ce}(G_\star(x_i^s), y_i^s); \qquad G_\star(x_i^s) = G_2(G_1(x_i^s)) - G_3(G_1(x_i^s))$$

2. **Adversarial loss**: A mini-max loss function that is minimized over the generator and maximised over discriminator. This is calculated using both source and target domain samples, where $D_\star$ is D_1 and its output is a single value, the probability of the sample belonging to the source domain. Ideally, the output of the discriminator should be 0.5. The following is minimized, in order to maximise the adversarial loss:

$$-L_{trans}^{adv} = \frac{1}{N_s}\sum_{i=1}^{N_s} log(D_\star(G_\star(x_i^s))) \; + \; \frac{1}{N_t}\sum_{j=1}^{N_t} log(1 - D_\star(G_\star(x_j^t)))$$

3. **Extra loss**: This can be any additive loss function, such as a clustering loss over class clusters, or class confusion loss, or domain divergence based loss. Here, we explore multiple extra loss functions and study their impact on the domain adaptation accuracy.

The code for GVB is heavily borrowed from **CDAN + E**, which performs the mini-max optimization by back-propagating on the tensor product of **f** (parameters of the feature map) and **g** (parameters of the classifier) rather than the concatenation of **f** and **g**. It is a more robust framework than vanilla **DANN**, which does not consider the interactions between the parameters of the classifier and the feature map. It was indeed shown to be superior on natural image datasets like OfficeHome and DomainNet.

6 Minimum Class Confusion

Probability re-scaling according to a temperature coefficient produces nuances in the classifier, by redressing overconfident predictions made by it. The minimum class confusion loss function seeks to minimize **confusion terms** between classes j and j', such that $j \neq j'$ where the indices are exhaustive over the set of classes.
The class confusion term between two classes j and j' is set to be equal to $C_{jj'} = \hat{\mathbf{y}}_{\cdot j}^{\intercal}\mathbf{W}\hat{\mathbf{y}}_{\cdot j'}^{\intercal}$ which is a better estimate of the class confusion than $\hat{\mathbf{y}}_{\cdot j}^{\intercal}\hat{\mathbf{y}}_{\cdot j'}^{\intercal}$. The matrix $\mathbf{W}$ is a diagonal matrix with diagonal terms W_{ii} given as the softmax outputs of the entropies in classifying a sample i. $\hat{\mathbf{y}}_{ij}$ is given by

$$\hat{\mathbf{y}}_{ij} = \frac{\exp(Z_{ij}/T)}{\sum_{j'=1}^{|C|}\exp(Z_{ij'}/T)}$$

$|C|$ is the number of classes, T is the temperature coefficient and Z_{ij} is the logistic output of the classifier layer for the class j and the sample i. After normalizing the class confusion terms, the final MCC Loss function is given as:

$$L_{MCC} = \frac{1}{|C|}\sum_{j=1}^{|C|}\sum_{j'\neq j}^{|C|}|C_{jj'}|$$

Further, we observe that back-propagating the MCC loss increases the accuracy in the first few iterations of the algorithm , probably because the primitive classifier needs more assistance in classifying hard samples. So, we gradually reduce the temperature coefficient as the number of iterations grows. A lower value of temperature penalizes the logistic output Z_{ij} more in the case of faulty classifications, thereby curbing the classifier more strictly in the final iterations.

$$T = 0.5 * \frac{1}{\exp(\frac{i}{10000})}$$

7 Maximum Density Divergence

This MDD loss function can be used to minimize the distance between the source and target distributions, while maximizing intra-domain class density.
A confused domain discriminator doesn't necessarily imply that the domains have been replied. This is termed as the **Equilibrium Challenge**. To address this challenge, we employ an additional loss function: MDD Loss. For distributions P and Q, MDD can be theoretically formulated as:

$$MDD(P,Q) = E_{X_s\sim P,X_t\sim Q}[\| X_s - X_t \|_2^2] + E_{X_s,X'_s\sim P}[\| X_s - X'_s \|_2^2] + E_{X_t,X'_t\sim Q}[\| X_t - X'_t \|_2^2]$$

Minimizing the first term will essentially minimize gap between distributions P and Q, and minimizing the next 2 terms will maximize the density within the domains. Instead of the Euclidean distance, other norm functions can also be used.
In this MDD formulation, to compute expected value we need to sum over all the samples from the distributions. The distribution in this case is random sampling of data points from source and target domains. It is practically impossible to sum over all samples since the number of samples is very high and therefore, we employ batch-processing so we don't have access to all samples in each iteration. To address this issue, a computationally

effective implementation of MDD is as follows:

$$L_{mdd} = \frac{1}{n_b}\sum_{i}^{n_b} \parallel f_{s,i} - f_{t,i} \parallel_2^2 + \frac{1}{m_s}\sum_{y_{s,i}={y'}_{s,i}} \parallel f_{s,i} - {f'}_{s,j} \parallel_2^2 + \frac{1}{m_t}\sum_{y_{t,i}={y'}_{t,i}} \parallel f_{t,i} - {f'}_{t,j} \parallel_2^2$$

where n_b is half of the batch size, m_s and m_t are the number of source and target samples in n_b, f and f' are the feature vectors, and y, y' are labels (in case of source domain) and pseudo-labels (in case of target domain). This loss is used along with the adversarial loss in our experiments.

8 Domain Conditioned Adaptation Network

Domain Conditioned Adaptation Network (DCAN) aims to capture domain-specific information in low-level features extracted by the initial convolution layers of the backbone ResNet. It uses domain conditioned channel attention mechanism that gives attention weights to a global pooling of each channel of a convolution layer. Separate FC layers are used to compute the attention weights for source sample and target sample after the global pooling layer. The reason behind this is to model the domain-wise channel dependencies for each domain. This module is followed by a domain conditioned feature correction module, that modifies the task-specific layer outputs of the target domain ($H(x_t)$) to get it closer to the source domain ($H(x_s)$). Their distance is further reduced by using the classic MMD criterion on $\hat{H}(x_t)$ and $H(x_s)$.As a regularization step, some source samples are also passed through the feature correction module.

The loss function is formulated as:

$$\min_{G} \quad L = L_s + \alpha \sum_{l=1}^{L} (L_M^l + L_{reg}^l) + \beta L_e$$

L_s is the classification loss for source samples (cross-entropy loss)
$L_M^l = MMD(H_l(x_t), \hat{H}_l(x_t))$ which is the MMD loss for feature correction module,
$L_{reg}^l = MMD(H_l(x_s), H_l(x_t))$ which is the corresponding regularization loss,
L_e is the target entropy loss computed using the class predictions of target samples.

The entire model architecture is shown below:

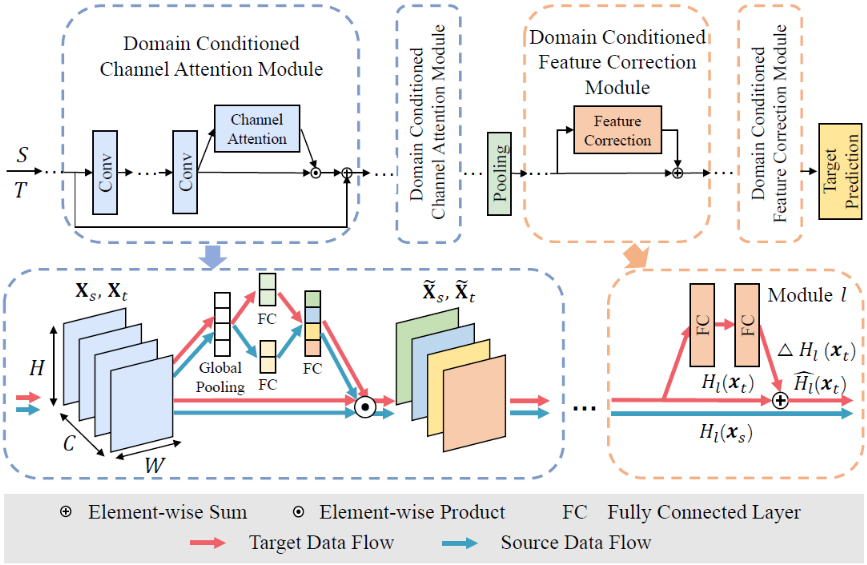

Domain Conditioned Channel Attention Module
S
T
Conv
Conv
Channel Attention
Domain Conditioned Channel Attention Module
Pooling
Domain Conditioned Feature Correction Module
Feature Correction
Domain Conditioned Feature Correction Module
Target Prediction
$\mathbf{X}_s, \mathbf{X}_t$
H
C
W
Global Pooling
FC
FC
FC
$\tilde{\mathbf{X}}_s, \tilde{\mathbf{X}}_t$
Module l
FC
FC
$\triangle H_l(\boldsymbol{x}_t)$
$H_l(\boldsymbol{x}_t)$
$\widehat{H_l}(\boldsymbol{x}_t)$
$H_l(\boldsymbol{x}_s)$
⊕ Element-wise Sum
⊙ Element-wise Product
FC Fully Connected Layer
Target Data Flow
Source Data Flow

Figure 2: DCAN Framework

9 Experiments - UDA

We begin by training an un-adapted ResNet, and then apply state of the art adaptation architectures. The best UDA architecture is GVB, with significantly improved benchmark results on natural image data-sets. When applied to the FHIST, we expect the accuracy of GVB to subsist when we add more loss functions which aid domain alignment synergistically. But, we observe that this interaction is in fact, antagonistic. We present our results systematically.

9.1 Dataset and Settings

We employ the FHIST data-set to compare the performance of different loss functions. It was originally curated for the task of few-shot classification of near-domain target samples, where the source domain is CRC-TP data-set and (near-domain) target is NCT-CRC-HE-100K (NCT). It consists of colorectal cancer histology images from two different domains, with 6 classes: Benign, Muscle, Stroma, Inflammatory, Debris and Tumor, with close to 20,000 labeled patches in the source domain, and around 10,000 unlabeled patches in the target domain.

All experiments are run on NVIDIA GeForce GTX Titan (6GB GDDR6). The batch size is kept constant at 8 and the number of iterations is 10,000.

9.2 Baseline on FHIST

We investigate the need for domain adaptation on the FHIST data-set. For this, we train three classification models using each of the domains separately. These models are trained for 10,000 iterations using ResNet50 backbone and a cross-entropy loss.

Model description	Best Accuracy
Trained and tested on CRC-TP	0.87262
Trained and tested on NCT	0.98701
Trained on CRC-TP and tested on NCT	0.61382

From the accuracies, we can conclude that the dataset is suitable for domain adaptation. For CRC-TP to NCT adaptation, the lower bound of accuracy is 61.38% and the upper bound is 98.70%, which means the accuracy of the adapted model must lie between these two values. Further, we will experiment with a distance-based UDA technique.

9.3 DCAN on FHIST

DCAN is the current best distance-based method for UDA on natural image datasets. Implementing this approach on FHIST gave a best accuracy of = 76.6337%

We mention in the next section that the GVB benchmark outclasses the DCAN on the FHIST data-set. We proceed and modify the extra loss functions in the GVB architecture.

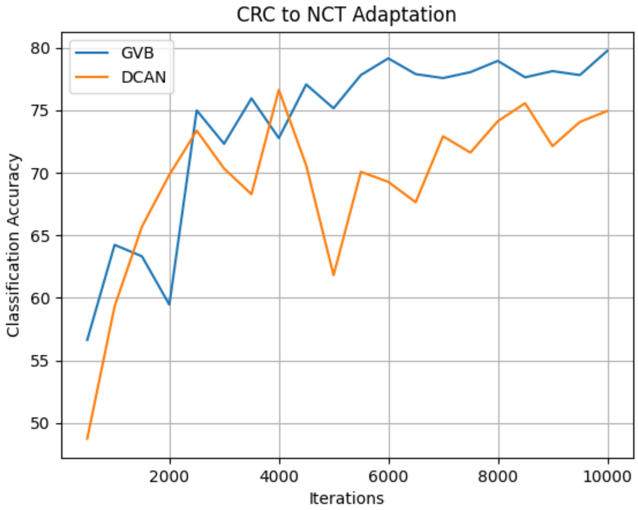


Figure 3: Accuracy of DCAN and GVB

9.4 GVB on X-ray data-sets

We also investigate the importance of domain adaptation on radiography data-sets. We obtain the following:

Model description	**Best Accuracy**
Trained on source and tested on target	0.996
Trained on source and adapted onto target	0.997

We see only a negligible increase in accuracy on applying a SOTA domain adaptation framework. The hypothesis is that X-ray images taken using different scanners do not produce a significant domain shift.

9.5 GVB and its variations on FHIST

9.5.1 Ideating on Loss functions

On top of the Binary Cross entropy classifier loss, we add more loss functions which effectively capture the separation of the feature representations of the source and target samples. These loss functions must incorporate a distance measure like a Minkowski distance or a cosine similarity metric so that the classification clusters can be made more compact, and therefore, aid the classifier. The centre of a cluster is taken to be the weight prototype of the feature map in the final ResNet layer. If there are c classes, there will be c such prototypes. The prototype is the same for both the source and target domains. This is because in domain adaptation frameworks, we use a single feature extractor.

- **$\mathbf{L^1}$-type loss** : The $\mathbf{L^1}$ distances between the clusters are minimized.

$$\sum_{c \in C} \sum_{c' \in C, c' \neq c} \frac{||c - c'||}{1 + ||c - c'||}$$

- **$\mathbf{L^2}$-type loss** : The $\mathbf{L^2}$ distances between the clusters are minimized.

$$\sum_{c \in C} \sum_{c' \in C, c' \neq c} \frac{||c - c'||^2}{1 + ||c - c'||^2}$$

- **Island loss** : This loss draws its roots from facial recognition. It minimizes the cosine similarity between prototypes of different classes, and in doing so, the angles between the clusters $c \in C$ are maximized. The the distances from the centres are minimized using a centre loss component. Here, B is the batch taken, and in our experiments it is fixed to be of size 8. α is the trade-off parameter. However, using the island loss may not work so well in the case of a large number of classes. This is because of crowding in the latent feature space, making it harder for the classifier to confidently make predictions. But in the medical setting, due to the comparatively fewer number of classes (<20), it serves well.

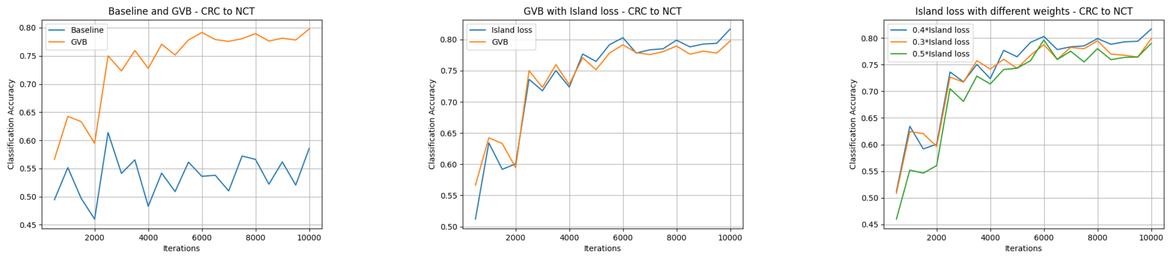
$$\sum_{c \in C} \sum_{i \in B} ||x_i - c||^2 + \alpha \sum_{c \in C} \sum_{c' \neq c} \frac{c \cdot c'}{||c|| \cdot ||c'||}$$

9.5.2 Experiments

In this subsection, we first implement the GVB on FHIST, specifically from CRC to NCT, and consequently add different types of extra loss functions mentioned above in place of L_{ext} mentioned in section 3. Here, GVB heavily borrows code from CDAN, which in turn heavily borrows code from MCD (Maximum Classifier Discrepancy for UDA) [8]. MCD employs two distinct classifiers to classify the samples in the source, and maximizes the disagreement between these two classifiers as a part of the mini-max optimization, so that either classifier learns different but equally important representations in the feature space. The distinct classifiers are created by initializing them differently.

Model description	**Best Accuracy**
GVB	79.758%
GVB + 1*MDD	77.766%
GVB + 0.05*MDD	77.033%
GVB + 0.05*MDD + 0.5*MCC	77.412%
GVB + 0.02*Island Loss	78.552%
GVB + 0.1*Island Loss	78.656%
GVB + 0.3*Island Loss	79.841%
GVB + 0.4*Island Loss	**81.671%**
GVB + 0.5*Island Loss	79.621%
GVB + $\mathbf{L^1}$ type Loss	78.755%
GVB + $\mathbf{L^2}$ type Loss	77.683%

Here, we refrain from tuning the weights of the loss functions as a hyper-parameter.



(a) Baseline GVB

(b) GVB + Island Loss

(c) GVB+Isl. Loss(weighted)

Doing so would only make the results valid on one ordered pair of data-sets. We even do not consider the reverse problem of switching the Source and Target domains. It is our intent to be as general as possible.
Loss functions fall into two categories by their nature:

1. Loss functions that **minimize the domain discrepancy**: MDD jointly minimizes the inter-domain divergence and maximizes the intra-domain density. MCC minimizes the class confusion using class predictions.

2. Loss functions that **aid clustering in the source and target domain**: Island loss, Center loss etc. which makes the clusters more dense and maximises angle between them in the latent space.

As an interesting segue, we run some experiments that leverage properties of MCC as well as temperature conditioning.

Model description	**Best Accuracy**
MCC (for the first 2500 iterations) + GVB + 0.4*Island loss	81.084%
GVB + 0.4*$\mathbf{L^1}$ type Loss + 0.2*MCC (Temperature) + 0.4*Island Loss	**81.603%**

10 Observations based on UDA experiments

The GVB, which is an adversarial method, gives better results than DCAN which is a distance-based method with attention. Adding the MDD loss to GVB decreases the accuracy. Since DCAN, the best distance based DA method was outperformed by GVB, and even adding the MDD loss reduced the performance of GVB by a few points. We hypothesize that a distance based metric degrades the performance of DA in the medical setting. This inclines us to believe that any kind of mean operator or moment operator

adds random noise to the medical image datasets, which may confuse the diagnosis. We shall proceed to test this hypothesis as a part of DDP Phase 2, using multiple other medical image datasets, including the CAMELYON 17 and the ADNI Brain imaging database.
Further, addition of MCC loss also decreased the accuracy. Adding a clustering loss increased the accuracy of GVB, while adding domain alignment /domain discrepancy losses compromised on the accuracy. Also, an interesting thing to note is that island loss makes the accuracy worse off in the initial few iterations while it seems to converge to the target accuracy faster. We also observe that MCC improves accuracy in the initial few iterations. This is probably because the classifier is confused when it is beginning to be trained, and later when it starts to make confident predictions on it's own accord, will not benefit much from class confusion minimization.

11 Experiments - SSDA

We compare the results of SOTA unsupervised domain adaptation methods with those of SOTA semi-supervised domain adaptation methods. Here, we employ Cross-Domain Adaptive clustering (CDAC) [5] in the 3-shot setting. We employ this method on the retinal imaging data-set and the FHIST data-set. CDAC uses a contrastive loss to align positive and negative samples. It initially groups features of unlabeled target data into clusters and then performs cluster-wise feature alignment across the source and target domains rather than sample-wise or distribution-wise feature alignment. It is the SOTA SSDA algorithm. We run the experiments on FHIST and RFMiD.

FHIST: CRC → NCT

Model description	**Best Accuracy**
Baseline ResNet	61.382%
3-shot CDAC	**73.470%**
GVB+CDAN	**79.758%**

CDAC uses strong augmentation on the images as a pre-processing measure. This might result in important subtleties of medical images being lost. Histology images have innate randomness in terms of the region of the image being given attention to. We further show

this in the next section, where we employ a heat-map of attention regions in a few sample images of both the source and the target.
On the RFMiD (Retinal imaging) data-set, we group all the 46 disease classes into one, to reduce computational complexity while clustering.

RFMiD: TOPCON 3D OCT-2000 → Kowa VX-10 α TOPCON

Model description	Best Accuracy
Baseline ResNet	**80.418%**
3-shot CDAC	71.458%
DANN	**81.087%**
MDD	**81.288%**

12 Observations based on SSDA experiments

We suspect that the degraded accuracy of the semi-supervised method may be attributed to the following reasons:

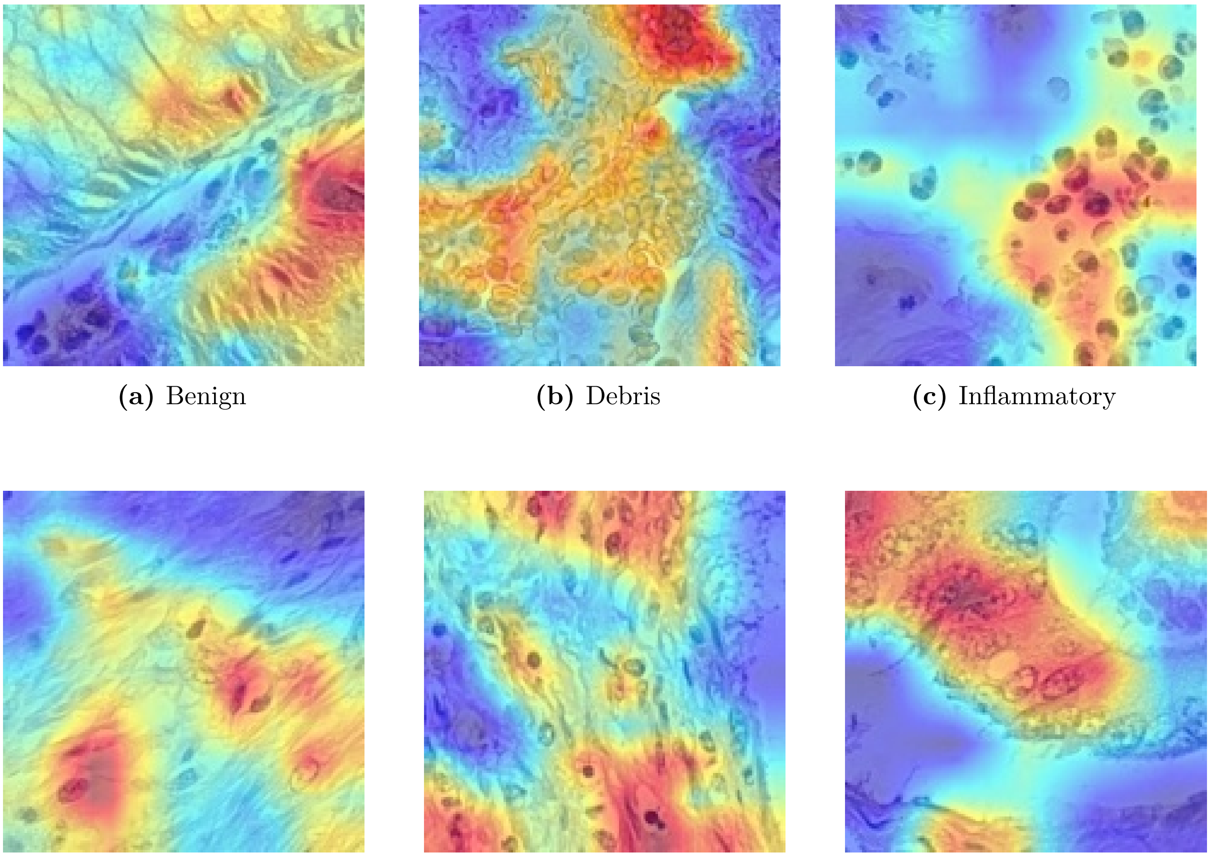
1. Due to severe class imbalance in cameras 2 and 3, all modes of variation were not effectively captured in a domain-adaptation framework. Therefore, the baseline ResNet performs as good as the domain-adapted ResNet. In fact, we think that on top of the accuracy gained during the adaptation process, the accuracy took an equally scathing hit because of the symmetric treatment of the two domains by the UDA method.

2. In the 3-shot SSDA setting, 3 samples are randomly selected from the pool of the target samples. In doing so, we may not be able to capture other modes of variation in the SSDA method. In this case, there are multiple modes of variation from the normal- presence of cotton wool spots, boat haemorrhages, macular exudates, discoloration of the macula etc. are a few examples of aberrations. So, picking just 3 samples will not capture all aberrations.

3. Classification itself might be a caveat- one retinal image might be clinically positive for two or more conditions. In this case, the logistic classifier assigns them under the class with the highest soft-max output. To address this issue, a $\mathbf{L^1}$ loss or a $\mathbf{L^2}$ loss may be used, on a subset of the data-set with a low class imbalance.

13 Grad-Cam visualizations

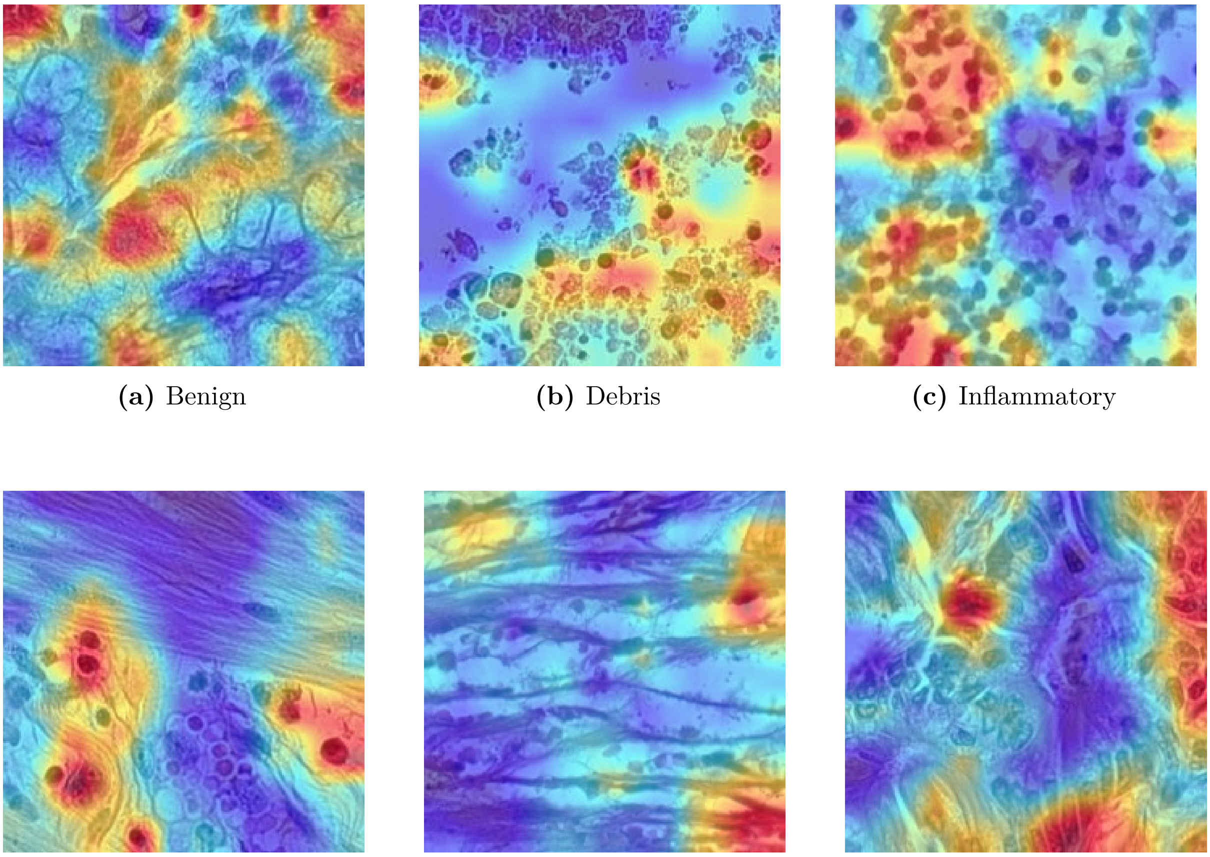
Grad-CAM is a visual explanation algorithm that helps us visualise contribution of each area in deciding the class of the image. It works by averaging the gradient-flow heat maps from each layer of the network. For a single layer, we cut-off the model at the layer for which we wish to build a Grad-CAM heat-map. The model is then applied to the input image, and the layer output and loss are collected to compute the gradient of the output w.r.t loss. In order to overlay the heat-map with the original image, we take portions of the gradient that contribute to the prediction and resize them.



(a) Benign **(b)** Debris **(c)** Inflammatory

(a) Muscle **(b)** Stroma **(c)** Tumor

Figure 6: CRC-TP



(a) Benign **(b)** Debris **(c)** Inflammatory

(a) Muscle **(b)** Stroma **(c)** Tumor

Figure 8: NCT

Observe that in all classes, the highest weight is given to some portion of the cellular region. The most attended regions are not very intuitive and the shape of region-of-attention is highly random, unlike natural images. Thus, there is a propensity for the classifier to attend to fallacious regions in the image.

References

[1] S. Cui, S. Wang, J. Zhou, C. Su, Q. Huang, and Q. Tian. Gradually vanishing bridge for adversarial domain adaptation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020

[2] Y. Jin, X. Wang, M. Long, and J. Wang, Minimum class confusion for versatile domain adaptation, in Proc. ECCV, 2020, pp. 464–480

[3] Jingjing Li, Erpeng Chen, Ding Zhengming, Lei Zhu, Ke Lu, and Heng Tao Shen. Maximum density divergence for domain adaptation. IEEE Trans. Pattern Anal. Mach. Intell., pages 1–1, 2020.

[4] Li, Shuang and Liu, Chi Harold and Lin, Qiuxia and Xie, Binhui and Ding, Zhengming and Huang, Gao and Tang, Jian, "Domain Conditioned Adaptation Network", 2020, *arxiv.2005.06717*

[5] Li, Jichang and Li, Guanbin and Shi, Yemin and Yu, Yizhou, Cross-Domain Adaptive Clustering for Semi-Supervised Domain Adaptation, 2021, *arxiv.2104.09415*

[6] Singh, Ankit, Contrastive Learning for Domain Adaptation, 2021, *arxiv.2107.00085*

[7] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, “A kernel two-sample test,” JMLR, vol. 13, no. Mar, pp. 723–773, 2012.

[8] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In Proceedings of the IEEE Con ference on Computer Vision and Pattern Recognition, pages, 3723–3732, 2018, *arxiv.1712.02560*

[9] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, Francois Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. The Journal of Machine Learning Research, 17(1): 2096–2030, 2016 *arxiv.1505.07818*

[10] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adapta tion. In Advances in Neural Information Processing Systems, pages 1647–1657, 2018. *arxiv.1705.10667*

[11] GRADCAM

[12] Baochen Sun and Kate Saenko, Deep CORAL: Correlation Alignment for Deep Domain Adaptation, 2016, *arxiv.1607.01719*

[13] Chao Chen and Zhihang Fu and Zhihong Chen and Sheng Jin and Zhaowei Cheng and Xinyu Jin and Xian-Sheng Hua, HoMM: Higher-order Moment Matching for Unsupervised Domain Adaptation, 2019, *arxiv.1912.11976*