I.J. Image, Graphics and Signal Processing, **2025, 3, 123-145**
Published Online on June 8, 2025 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijigsp.2025.03.07

Binary Segmentation Dataset Distances for Transfer Learning

Victor Sineglazov*
Department of Aviation Computer Integrated Complexes, State University "Kyiv Aviation Institute", Kyiv, Ukraine
E-mail: svm@kai.edu.ua
ORCID iD: https://orcid.org/0000-0002-3297-9060
*Corresponding Author

Kirill Riazanovskiy
Department of Artificial Intelligence, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine
E-mail: k.riazanovskyi@kpi.ua
ORCID iD: https://orcid.org/0000-0002-8771-8060

Olexander Klanovets
Department of Artificial Intelligence, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine
E-mail: alex.klanovets@gmail.com

Received: 25 January, 2025; Revised: 16 March, 2025; Accepted: 03 April, 2025; Published: 08 June, 2025

Abstract: This work is devoted to developing a novel transfer learning approach for solving binary semantic segmentation problems that often arise on short samples in the medical (segmentation of nodules in lungs, tumors, polyps, etc.) and other domains. The goal is to optimally select the most suitable dataset from a different subject area with similar feature space and distribution to the target data. Examples show that a severe disadvantage of transfer learning is the difficulty of selecting an initial training sample for pre-training a neural network. In this paper, we propose metrics for calculating the distance between binary segmentation datasets, allowing us to select the optimal initial training set for transfer learning. These metrics are based on the geometric distances estimation of the dataset using optimal transport, Wasserstein distance for Gaussian mixture models, clustering, and their hybrids. Experiments on datasets of medical segmentation Decathlon, LIDC, and a private dataset of tuberculomas in the lungs are presented, proving a statistically strict correlation of these metrics with a relative increase in segmentation accuracy during transfer learning.

Index Terms: Binary Semantic Segmentation, Transfer Learning, Dataset Distance, Optimal Transport, Gaussian Mixture Models, Clustering

1. Introduction

Deep artificial neural networks are state-of-the-art models for solving a wide range of problems. However, to apply these models correctly, a sufficiently large amount of data for training is required. Many modern machine learning and deep learning tasks do not have enough training data due to the complexity of its collection, applying short samples or other methodical reasons.

Transfer learning (TL) is a modern method for approaching the problem of small training datasets. Studying the TL approach and improving its application methodology, in particular by defining a source dataset selection algorithm for a specific case, provides an opportunity to use neural networks to solve the corresponding problems with a small amount of data. This is a pervasive case in deep learning problems.

The problem of short samples often arises in different domains related with signal and image processing. Image segmentation is a critical computer vision task where an image is divided into meaningful parts or segments, often assigning each pixel to a specific class or object. A wide range of potential applications across various fields demonstrate the generality of this approach. It is of great importance, for example in:

- autonomous vehicles for identifying and segmenting pedestrians, vehicles, road signs, and lanes to navigate safely [1];
- agriculture [2] for segmenting diseased or healthy parts of crops for early detection, estimate crop health and yield, by analyzing satellite or drone images;
- remote sensing and GIS [3] for land cover classification, disaster management and urban planning;
- robotics to recognize and interact with segmented objects in cluttered environment [4];
- radar meteorology segmenting radar reflectivity data to identify areas of rain, snow, hail, or other precipitation types [5], storm intensity mapping [6], including dangerous turbulence zones [7], high probability of aircraft icing zones [8], wind shear [9] and gust fronts identification, lightning prediction [10] and other meteorological applications [11];
- ground penetrating radar (GPR) and through-the-wall surveillance technology [12, 13] where image segmentation significantly enhances the interpretability of data for a variety of applications (archaeology, geology, civil engineering, mining and mineral exploration: identifying ore bodies, faults, or other subsurface features.

Thus, segmentation tasks are fundamental for advancing agriculture information, weather prediction, infrastructure safety, environmental monitoring, resource management, and they are widely used in many other applications. Nevertheless, the one of the most typical their applications are in medical domain, as the number of patients with a particular diagnosis is usually very limited. This is even more apparent when processing medical images such as CT, MRI, ultra- sound, etc., where the feature space is highly dimensional, and the number of images with pathologies is very scarce.

The image segmentation procedure is one of the key tasks of computer vision at the moment, with potential applications in various domains [14], including medical applications [15, 16, 17, 18, 19] as very important ones. That is why in this article we develop a novel transfer learning approach for solving binary semantic segmentation problems that arise on short samples exactly in the medical domains, although they can be adopted for wide variety of other applications.

The simplest yet significant part of segmentation is binary segmentation, when one class of objects is identified in images, and the background is considered as the second class. The binary segmentation task often appears in the medical domain, such as the segmentation of lung nodules, tumors, polyps, etc. This study focuses specifically on this task. The first step in solving it is the optimal selection of segmentation datasets for transfer learning.

This article proposes a novel method for selecting the optimal training sample to utilize transfer learning for neural network pre-training in binary segmentation tasks. This method is based on the computation of the geometric distance between the available short sample data in a given target domain and samples from other fields using the optimal transport (OT) method [20], the Wasserstein distance between Gaussian mixture models (GMM) [21], clustering and their hybrids. Examples are given to demonstrate the effectiveness of the proposed approach.

2. Problem Statement

Let us give a formal statement of the optimal dataset selection problem for TL.

Given the set of source datasets $\mathfrak{D} = \{\mathcal{D}_1, \dots, \mathcal{D}_n\}$, the small target domain-specific dataset $\mathcal{D}_T$ and the binary segmentation model $f(\cdot)$, it is necessary to select such dataset $\mathcal{D}_i$ from $\mathfrak{D}$ that can be used for pre-training and as a result will increase the accuracy of the semantic segmentation model $f(\cdot)$ on the test sample than when training only on a small target dataset or any other dataset from $\mathfrak{D}$ for pre-training:

$$\begin{cases} acc_{test}(f|\mathcal{D}_i \to \mathcal{D}_T) > acc_{test}(f|\mathcal{D}_T) \\ acc_{test}(f|\mathcal{D}_i \to \mathcal{D}_T) > acc_{test}(f|\mathcal{D}_j \to \mathcal{D}_T) \forall j: j \neq i, \mathcal{D}_i, \mathcal{D}_j \in \mathfrak{D} \end{cases} \quad (1)$$

where $acc_{test}(f|\mathcal{D}_i)$ is the accuracy metric value of the model $f(\cdot)$ on the test sample of the dataset $\mathcal{D}_i$; $\mathcal{D}_i \to \mathcal{D}_T$ is the transfer learning of the model from source dataset $\mathcal{D}_i$ to target dataset $\mathcal{D}_T$.

To avoid training the model on all possible datasets from $\mathfrak{D}$ and then $\mathcal{D}_T$, we need to define a metric $d(\mathcal{D}_i, \mathcal{D}_j): \mathcal{X} \times \mathcal{X} \to \mathbb{R}^+$, where $\mathcal{X}$ is the metric space of the dataset features, that allows to quickly compare all datasets from $\mathfrak{D}$ with the target dataset $\mathcal{D}_T$ and unambiguously determine the best transfer dataset $\mathcal{D}_i$ that satisfies the condition 1, $\mathbb{R}^+$ is the set of positive real numbers.

So, the problem is to find a metric $d(\mathcal{D}_i, \mathcal{D}_j)$,which would uniquely determine the distance between datasets, especially for the binary segmentation task. Once the metric is defined, we select such a dataset $\mathcal{D}_i$, in which the distance between it and the target short dataset $\mathcal{D}_T$ was minimal:

$$\mathcal{D}_i = \underset{\mathcal{D}_j = \mathcal{D}_1, \dots, \mathcal{D}_n}{\operatorname{argmin}} d(\mathcal{D}_T, \mathcal{D}_j)$$

For clarity, we need to define precisely what features and datasets are involved in our distance calculations. In binary segmentation tasks, each dataset $\mathcal{D}_i$ consists of pairs of images and their corresponding binary masks: $\mathcal{D}_i = \{(X_1, Y_1), (X_2, Y_2), \ldots, (X_l, Y_l)\}$, where:

- $\mathrm{X_k} \in \mathbb{R}^{h \times w \times c}$ represents the input image with height h, width w, and c channels,
- $Y_k \in \{0,1\}^{h \times w}$ represents the binary segmentation mask where 1 indicates the object of interest and 0 indicates background.

Our distance metric operates in two primary feature spaces:

1) Image Feature Space: The space of raw pixel intensities or derived features from the input images;
2) Segmentation Mask Space: The space characterizing geometric and distributional properties of the binary masks.

The core of our approach is to define a compound distance metric that effectively captures both the visual characteristics of the images and the structural properties of the segmentation masks. This allows us to meaningfully compare datasets across different domains while preserving the specific characteristics relevant to binary segmentation tasks.

3. Related Work

There are different approaches to comparing and evaluating datasets for transfer learning.

The authors of the paper [22] investigated the selection of a dataset for transfer learning in the sentiment analysis task of Twitter posts. To determine the similarity between datasets, they measured the distance between them using four metrics: Euclidean distance, cosine similarity, Jaccard distance and Relaxed Word Moving Distance. Computational experiments, conducted in the Twitter sentiment analysis scenario, showed that the cosine similarity metric combined with bag-of-words normalized with term frequency-inverse document frequency presented the best results in terms of predictive power, outperforming even the classifiers trained with the target dataset in many cases. An obvious limitation of this method is its application only in the field of text analysis and not image processing.

In paper [23], a systematic study was performed with nine source datasets with natural or medical images, and three target medical datasets, all with 2D images. The authors focused on the intermediate step of defining a meta-representation for each dataset, which allows them to measure their similarity. They used two types of meta-features, based on experts, and based on Task2Vec. The correlation they found on the basis of experiments was not significant, and the ImageNet weights led to the best AUC scores for all three medical targets. Thus, the paper does not provide a new statistically significant method for selecting a dataset for pre-training.

Another example from computer vision is Task2Vec [24], a popular approach to encoding tasks (feature-label distributions). The idea behind Task2Vec is to obtain embeddings of classification tasks so that the relationship between the tasks can be analyzed, even if the datasets have different characteristics such as a number of classes or image sizes. They investigate both a symmetric and an asymmetric version of the Task2Vec distance and show that the asymmetric version correlates with transferability between tasks. However, applying this method to comparing datasets for transfer learning rather than tasks is debatable.

One of the most promising approaches is Geometric Dataset Distances via Optimal Transport [20].

This approach relies on using optimal transport distances to compare distributions of feature-label pairs. The key aspect of this approach is the incorporation of label information into the OT problem, which leads to a more effective matching between feature and label distributions.

Their method suggests using a composed distance metric that combines Euclidean and Wasserstein distances to com- pare feature-label pairs across different domains. In their approach, labels are mapped to probability distributions over the corresponding feature vectors. This enables the comparison of datasets even if their label sets are entirely unrelated, as long as a distance metric between their features can be defined.

To briefly summarize the notation and logic of their proposed approach, let us define a data set *classification* $\mathcal{D}$ as a set of pairs $(x_1, y_1), \ldots, (x_l, y_l)$, where $x_i \in \mathcal{X}$ is the feature tensor and $y \in \mathcal{Y}$ is the class label.

In [20] the following metric for the distance between the classification datasets was proposed to solve the OT problem:

$$d_{OT}(\mathcal{D}_A, \mathcal{D}_B) = \min_{\pi \in \Pi(\alpha,\beta)} \int_{\mathcal{Z}\times\mathcal{Z}} \mathrm{d}_{\mathcal{Z}}(z, z') d\pi(z, z') \tag{2}$$

where $d_{OT}(\mathcal{D}_A, \mathcal{D}_B)$ is the OT metric distance between datasets $\mathcal{D}_A$ and $\mathcal{D}_B$; $\mathcal{Z}$ is the metric space on pairs (x, y); $d_{\mathcal{Z}}(z, z')$ is the metric on $\mathcal{Z}$, the distance between z and z', i.e. pairs of (x, y) and (x', y'); $d\pi(z, z')$ represents the infinitesimal portion of probability mass (or "transport mass") that couples the points z and z', intuitively, it tells us how much mass is sent from z to z' when matching two distributions; $\Pi(\alpha, \beta)$ is a set of couplings that consists of joint distributions over the product space $\mathcal{X} \times \mathcal{X}$ with marginals α and β:

$$\Pi(\alpha,\beta) \triangleq \{\pi \in \mathcal{P}(\mathcal{X}\times\mathcal{X}) | P_{1\#}\pi = \alpha, P_{2\#}\pi = \beta\}$$

For distance d_Z from Eq. 2 authors [20] propose the following metric:

$$d_Z\big((x,y),(x',y')\big) \triangleq \big(d_X(x,x')^p + d_Y(y,y')^p\big)^{\frac{1}{p}}, \text{for } p \geq 1 \tag{3}$$

where $d_X(x,x')$ is the Euclidean distance in the feature space;

$$\begin{aligned} d_Y(y,y')^p &= W_p^p\big(\alpha_y,\alpha_{y'}\big) \\ \alpha_y &\triangleq P(X|Y=y) \end{aligned} \tag{4}$$

where $W_p^p\big(\alpha_y,\alpha_{y'}\big)$ is the p-Wasserstein distance between distributions α_y, $p=2$. In other words, in [20], the distance between image matrices and between image data class labels is found separately.

The authors of the considered method [20] focused only on classification datasets with static class labels, so there is no possibility to apply it to segmentation datasets. In this case of binary segmentation, there is a pixel mask and only one class for each pixel, i.e. the segmented region of interest (lung nodules, vessels, tumor, etc.). Hence, there is no possibility to use the authors' proposed metric of the distance between different classes $d_Y(y,y')$ (Eq. 3) represented by labels. A new metric $d_Y(y,y')$ is required that considers that y and y' in this context are matrices of segmented region masks, not class labels.

In this paper, we employ the original idea of the [20] paper on geometric distance between datasets, but extend their approach for the binary segmentation problem.

3.1. Comparison with State-of-the-Art Methods

To better contextualize our contribution, we provide a comprehensive comparison with current state-of-the-art methods for dataset selection in transfer learning. Table 1 summarizes key approaches and their applicability to segmentation tasks:

Table 1. Comparison of Dataset Distance Methods for Transfer Learning

Method	Primary Domain	Applicability to Segmentation	Key Advantages	Limitations
Cosine Similarity [22]	Text classification	Low	Computationally efficient	Doesn't capture spatial information critical for segmentation
Task2Vec [24]	Image classification	Medium	Captures task relationships	Focuses on task embedding rather than dataset properties
DeepSets [41]	General ML	Medium	Permutation invariant	Lacks specific adaptation for segmentation masks
Data Shapley [42]	General ML	Low	Quantifies data value	Requires multiple model trainings
Optimal Transport [20]	Classification	Medium	Distribution-aware	Original formulation unsuitable for mask comparison
Our GMM-based approach	Segmentation	High	Captures spatial and intensity distributions	Higher computational complexity
Our Clustering approach	Segmentation	High	Feature-based comparison	Sensitive to hyperparameter selection

Our proposed methods offer several distinct advantages over existing approaches:

1. Geometric awareness: unlike general distance metrics, our approaches explicitly model the spatial configuration of segmented regions.
2. Distributional modeling: our GMM-based methods capture the full distribution of segmentation masks rather than just summary statistics.
3. Feature adaptability: our metrics can be tailored to different segmentation characteristics through feature selection and weighting.

Recent literature [43] has demonstrated the importance of domain-specific distance metrics for transfer learning tasks. Our work extends these findings to binary segmentation problems, providing a principled way to quantify dataset distances in this important domain.

4. Proposed Methods

This paper proposes two main directions for finding the distance between datasets of binary segmentation based on [20]:

1. following the original calculation formula (Eq. 3), we propose to construct a new metric $d_y(Y, Y')$ (similar to Eq. 4) to compare the distances of the binary masks Y and Y' and substitute it into Eq. 2 (Sec. 4.1);
2. using the original idea of calculating the distance between [20] datasets (Eq. 2 and Eq. 3) by a novel method of converting the binary segmentation problem to a classification problem (Sec. 4.2).

4.1. Defining metrics for comparing distances between masks

The main advantage of the binary segmentation task over classification in the context of our problem is the availability of binary segment mask matrices (example in Fig. 1, Eq. 5), as opposed to simple class labels. The binary mask matrix Y is defined by Eq. 6.

$$Y = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{5}$$

$$Y^{(ij)} = \begin{cases} 1, \text{if } i^{th}\ j^{th} \text{ pixel contains a segment,} \\ 0, \text{if } i^{th}\ j^{th} \text{ pixel contains a background,} \end{cases} \tag{6}$$

where $Y^{(ij)}$ is the i^{th}, j^{th} element of matrix Y.

The presence of binary matrices allows us to find the direct distance (metric) $d_y(Y, Y')$ between masks Y and Y' (Eq. 3) without finding the Wasserstein distance between image distributions, as proposed in the original [20] paper for the classification problem.

Fig. 1. Example of a binary mask (lung CT nodules)

Existing options of metrics between masks, such as intersection over union (IoU), dice score, jaccard index, etc., which are used in deep learning tasks, do not take into account the pixel values of the source images and are not exactly *distances*, so they are poorly suited for accurate comparison of image-mask pairs from different datasets.

We propose several alternative metrics between image-binary matrix pairs.

Before introducing our specific metrics, we need to justify our selection approach. When comparing segmentation datasets, we need metrics that can:

1. capture the spatial distribution of segmented regions,
2. account for pixel intensity variations within segments,
3. handle variations in segment size, shape, and location,
4. provide mathematically valid distance properties.

Table 2 compares potential metrics for segmentation mask comparison:

Table 2. Comparison of Distance Metrics for Segmentation Masks

Metric	Mathematical Properties	Spatial Awareness	Intensity Awareness	Computational Complexity
Dice/IoU	Not a true distance	Medium	None	$O(n)$
Hausdorff	Metric	High	None	$O(n^2)$
Earth Mover's Distance	Metric	High	Medium	$O(n^3)$
Euclidean	Metric	Low	Medium	$O(n)$
Wasserstein	Metric	High	High	$O(n^3)$
Our GMM-based approach	Metric	High	High	$O(k^3)$ where k is number of components

We selected Wasserstein distance as our foundation because:

1. it is particularly suitable for comparing distributions with different supports,
2. it accounts for the "transportation cost" between distributions, which is ideal for capturing spatial relationships,
3. it has strong theoretical guarantees and has been successfully applied in various computer vision tasks,
4. when used with GMMs, it offers a computationally tractable approach to comparing complex distributions.

Our adaptation of these metrics to binary segmentation maintains these advantages while addressing the specific challenges of segmentation mask comparison.

4.1.1 Euclidean distance between masks

The simplest and computationally efficient solution is to find the Euclidean distance between the segmented images, just as for the original images $X_1, \ldots, X_l$. To do this, the binary mask matrices $Y_1, \ldots, Y_l$ are overlaid on the matrices of the original images $X_1, \ldots, X_l$ to extract the segment matrices $S_1, \ldots, S_l$, where $S_k \in \mathbb{R}^{n \times n}$. Each element of the matrix S_k is defined as follows (Eq. 7).

$$S_k^{(ij)} = \begin{cases} X_k^{(ij)}, \text{if } Y_k^{(ij)} = 1 \\ \delta, \text{if } Y_k^{(ij)} = 0 \end{cases} \tag{7}$$

where $S_k^{(ij)}, X_k^{(ij)}, Y_k^{(ij)}$ are i^{th}, j^{th} elements of matrices S_k, X_k, Y_k; δ is the default empty value for pixels that are not included in the segment. An example of the Y, S matrix and pictures are shown in Fig. 2 ($\delta = 0$).

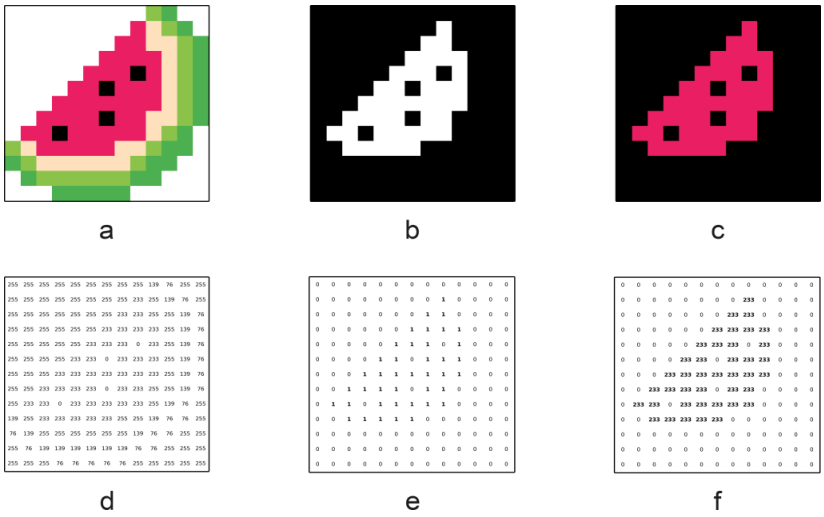

Fig. 2. Example image (a), binary mask (b), selected segment (c), image red channel matrix (d), binary mask matrix *Y* (e) and selected segment matrix *S* (f)

The value δ (Eq. 7) should be a number that is not within the range of image pixel values $X_1, \ldots, X_l$ so there is no confusion with segment values. It is chosen individually for each feature space. For example, for 8-bit image pixels in the range 0-255, this value could be -1, as it will not match the segment pixel values in any way. For 12-bit DICOM medical images, such a value may be -2048, which does not belong to any segmented tissue.

The next step of S_k segment matrices setup is their neutralization. To prevent the pixels that do not belong to the segment and are filled with the default value δ from interfering with the distance calculation, it is necessary to transform them to the neutral value 0, i.e. to subtract the value δ from all pixels. Consequently, all pixels that belong to the mask and segment will have values greater than zero (if the default value was less than the range of pixels), and empty pixels without a segment will be equal to zero. Then transform Eq. 7 into Eq. 8.

$$S_k^{(ij)} = \begin{cases} X_k^{(ij)} - \delta, \text{if } Y_k^{(ij)} = 1 \\ 0, \text{if } Y_k^{(ij)} = 0 \end{cases} \tag{8}$$

Then the distance metric $d_{\mathcal{Y}}(Y, Y')$ (Eq. 3) can be given as a Euclidean metric, the same as for the metric $d_{\mathcal{X}}(X, X')$ on the feature space :

$$d_y(Y, Y') = \|S - S'\|_2,$$
$$d_X(X, X') = \|X - X'\|_2,$$

where S and S' are the matrices obtained from Y and Y' following Eq. 8; $\|\cdot\|_2$ is the Euclidean norm.

Substituting the proposed metrics into the general space metric $\mathcal{Z}$ (Eq. 3) and then into the Eq. 2 for finding the distance between datasets, we derive the proposed method for calculating the distance between segmentation datasets using the optimal transport approach and Euclidean distance (Eq. 9).

$$\begin{aligned} d_{OT}(\mathcal{D}_A, \mathcal{D}_B) &= \min_{\pi \in \Pi(\alpha,\beta)} \int_{\mathcal{Z}\times\mathcal{Z}} \Big(d_X(X, X') + d_y(Y, Y')\Big)\, d\pi\big((X, Y), (X', Y')\big) = \\ &= \min_{\pi \in \Pi(\alpha,\beta)} \int_{\mathcal{Z}\times\mathcal{Z}} (\|X - X'\|_2 + \|S - S'\|_2) d\pi\big((X, Y), (X', Y')\big), \end{aligned} \tag{9}$$

where S and S' are the matrices obtained from Y and Y' following Eq. 8; $\|\cdot\|_2$ is the Euclidean norm.

4.1.2 Global distance between mask distributions

Another method to find the direct distance between masks Y and Y' or segments S and S' is to solve the Earth mover's distance problem [25] or to use the more efficient Sinkhorn algorithm [26] to find the distance between pixels of one segment and the other. In other words, the metric will show the optimal cost of the pixel-by-pixel transformation (OT distance between pixels) of one segment to another (Fig. 3).

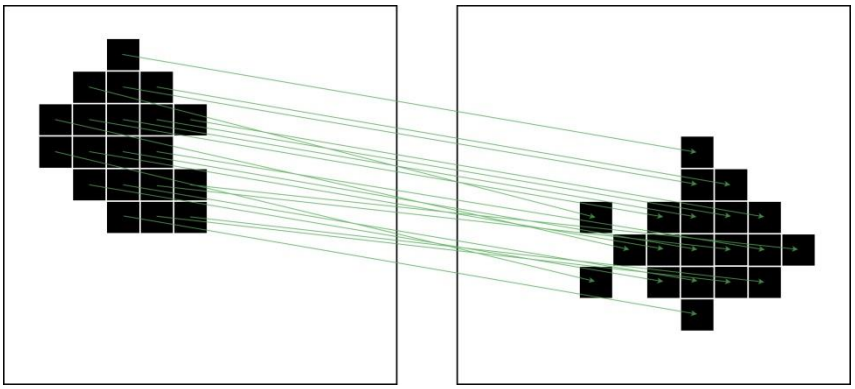

Fig. 3. Example of pixel-by-pixel transformation of one segment into another segment

The problem is the number of pixels of segments in real images. For large images and segments (e.g. liver or lungs), one segment may have a size of, for example, 300 by 300 pixels or 90000 pixels in a whole single segment. The second segment may also contain a large number of pixels. Calculating the pixel-by-pixel transformation, in this case, would be very computationally expensive and require a huge amount of memory, which is unacceptable for most applications.

4.1.2.1 Representation of segment pixels in images as clusters

We propose an approach that is based on aggregating a whole set of segment matrices $Y_1, \ldots, Y_l$ as a single matrix and extracting a group of segment clusters that are characterized by the parameters of cluster centers location and size (instead of the values of all segment pixels), but accurately capture the physical shape and distribution of segment pixels (example in Fig. 4).

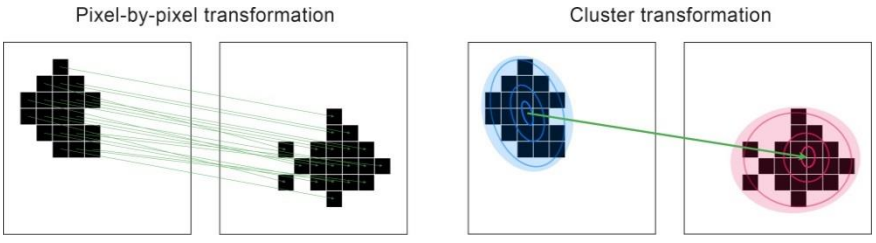

Fig. 4. Pixel transformation VS cluster transformation

One of the ways to define segment clusters in images is to use the multivariate Gaussian mixture model. The main advantage and justification for using this clustering algorithm is the existence of a direct simplified Bures-Wasserstein distance between Gaussian distributions and the OT distance between sets of GMMs using the [21] approach.

Using GMM for image segmentation is a well-known approach that was often used for segmentation [27, 28, 29] before the advent of convolutional neural networks or used as an unsupervised algorithm for clustering [30]. However, the

difference between the existing methods and the proposed method is that they directly perform image segmentation based on pixel intensity. In contrast, we propose a spatial approximation or clustering of the geometric location of 2D segments along with pixel values in the image using GMM (Fig. 5-7).

An example of segments is shown in Fig. 5, the segments are highlighted in white color. By aggregating the segments from all the images, it is possible to build their 2D histogram. An example of a 2D histogram of the segments (lung nodules) location over the image space and their coverage through GMM clusters is shown in Fig. 6. It is worth noting that the pixel intensities were not used for visualization, only their locations. The overall overlay of the GMM histogram of segment locations is shown in Fig. 7.

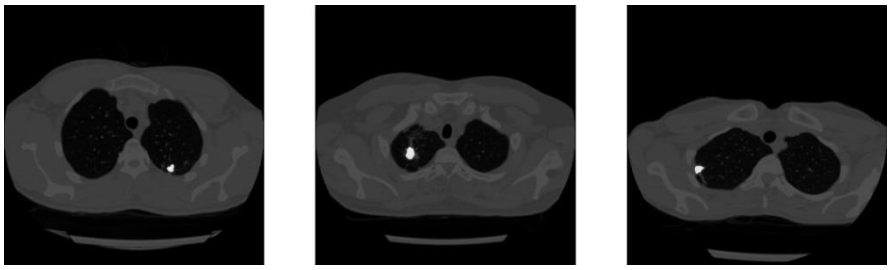

Fig. 5. Segments of nodules in the lungs

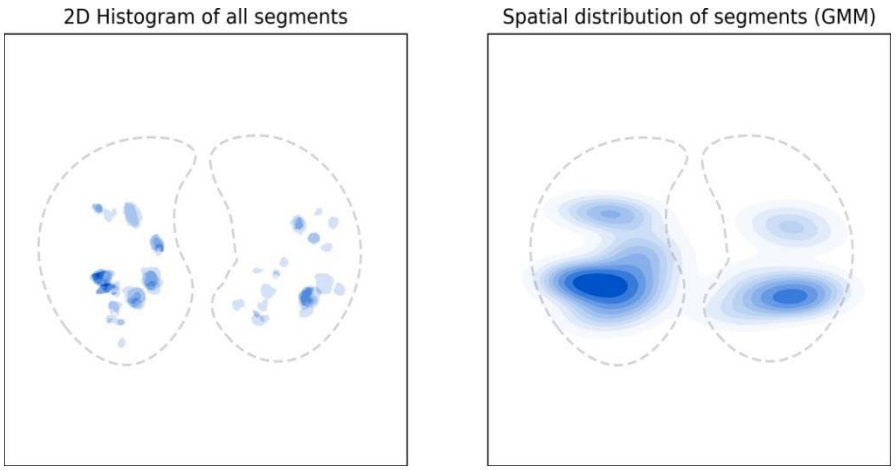

Fig. 6. 2D histogram of lung nodule distribution and GMM approximations

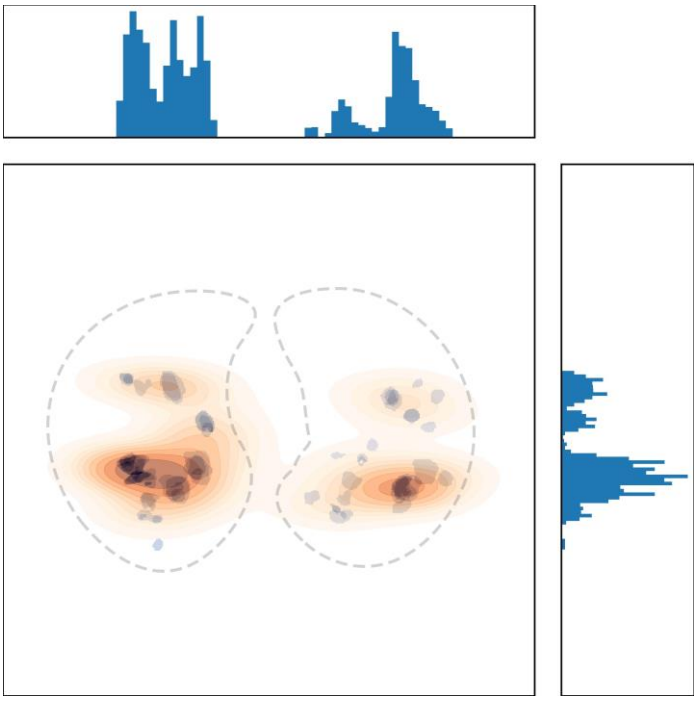

Fig. 7. GMM coverage of the segment histogram

Let us consider a formal method of aggregating all masks in the dataset and representing segment pixels as GMMs. Consider the set of triplets of all segment pixels from the dataset $(X_1, Y_1), \dots, (X_l, Y_l)$:

$$\mathcal{G} = \left(i_1^{(1)}, j_1^{(1)}, x_{i_1^{(1)} j_1^{(1)}}^{(1)}\right), \left(i_2^{(1)}, j_2^{(1)}, x_{i_2^{(1)} j_2^{(1)}}^{(1)}\right), \dots, \left(i_k^{(l)}, j_k^{(l)}, x_{i_k^{(l)} j_k^{(l)}}^{(l)}\right),$$

where $i_k^{(l)}$ is the row number of the k^{th} pixel of the segment in the l image; $j_k^{(l)}$ is the column number of the k^{th} pixel of the segment in the l image; $x_{i_k^{(l)} j_k^{(l)}}^{(l)}$ is the intensity value of the $i_k^{(l)}$-th row and $j_k^{(l)}$-th column pixel in the l^{th} image.

This set of all segment pixels is modelled as follows via GMM (Eq. 10).

$$\begin{gathered} \mathcal{G} \sim g \triangleq p(\mathrm{g}) = \sum_{i=1}^{N} \phi_i \mathcal{N}(\mathrm{g}|\mu_i, \Sigma_i) \\ \mathcal{N}(\mathrm{g}|\mu_i, \Sigma_i) = \frac{1}{\sqrt{(2\pi)^K |\Sigma_i|}} \exp\left(-\frac{1}{2}(\mathrm{g} - \mu_i)^T \Sigma_i^{-1} (\mathrm{g} - \mu_i)\right) \\ \sum_{i=1}^{N} \phi_i = 1 \end{gathered} \tag{10}$$

where **g** is the triplet vector from the $\mathcal{G}$ dataset; $\boldsymbol{\mu}_i$ is the mean vector i^{th} of the normal distribution in GMM; Σ_i is the covariance matrix i^{th} of the normal distribution of the model; ϕ_i is the weight i^{th} of the distribution in the model; N is the total number of Gaussian distributions in the model.

Training GMM models is beyond the goals of this paper and can be done, for example, using the Expectation-Maximization algorithm [31].

Determining the number of Gaussians in the GMM segment model also stands outside the goals of this paper, so we suggest using existing methods such as:

1. Elbow method
2. Bayesian information criterion [32]
3. Silhouette score [33]

Thus, the GMM model models the distribution of all segments in the dataset based on the pixel group of the segments.

4.1.2.2 Using GMM to find the distance between datasets

After aggregating all the masks of the two compared datasets as GMMs (Eq. 10), we can solve the distance transport problem between distributions based on the Wasserstein distance between two GMMs based on the [21] approach. The authors of [21] proposed the following form of the distance between two GMMs (Eq. 11).

$$d_{MW_2^2}(g, g') = \min_{w \in \Pi(\phi, \phi')} \sum_{i,j} w_{ij} W_2^2\left(g_i, {g'}_j\right) \tag{11}$$

where g is a GMM model (Eq. 10) of the first dataset; g' is the GMM model of the second dataset; ; g_j is the j^{th} Gaussian distribution from the GMM g of the first dataset; ${g_j}'$ is the j^{th} Gaussian distribution from the GMM g' of the second dataset; $W_2^2\left(g_i, {g_j}'\right)$ is the 2-Wasserstein distance between the component distributions g_j and ${g_j}'$ from the common GMMs g, g', which is given by the well-known Bures-Wasserstein formula (Eq. 12).

$$W_2^2(y, y') = \left\|\mu_y - \mu_{y'}\right\|_2^2 + tr\left(\Sigma_y + \Sigma_{y'} - 2\left(\Sigma_y^{\frac{1}{2}} \Sigma_{y'} \Sigma_y^{\frac{1}{2}}\right)^{\frac{1}{2}}\right) \tag{12}$$

where $y \triangleq \mathcal{N}\left(\mu_y, \Sigma_y\right)$ is the first Gaussian distribution and $y' \triangleq \mathcal{N}\left(\mu_{y'}, \Sigma_{y'}\right)$ is the second Gaussian distribution.
Therefore, aggregating all dataset masks into one GMM model allows us to find the distance $d\left(Y_i, {Y_j}'\right)$ between datasets once. It will be equal to the transport distance between GMMs of each dataset (Fig. 8, Eq. 13).

$$\begin{gathered} d\left(Y_i, {Y_j}'\right) = d_{MW_2^2}(g, g'), \\ \forall Y_i \in \mathcal{D}_A = \{(X_1, Y_1), \dots, (X_k, Y_k)\}, \\ \forall {Y_j}' \in \mathcal{D}_B = \{({X_1}', {Y_1}'), \dots, ({X_l}', {Y_l}')\} \end{gathered} \tag{13}$$

where $d_{MW_2^2}(g, g')$ is the transport distance between GMM g and g' (Eq. 10) based on the datasets $\mathcal{D}_A$ and $\mathcal{D}_B$, respectively (Eq. 11).

As shown in Eq. 13, the distance value between two masks will be the same for all pairs of masks. Therefore, when calculating Eq. 2 to solve the OT problem between complete datasets, the distance can be taken outside the integral and the optimization process, thereby simplifying the calculations in Eq. 14.

$$\begin{aligned} d_{OT}(\mathcal{D}_A, \mathcal{D}_B) &= \min_{\pi \in \Pi(\alpha,\beta)} \int_{Z\times Z} d_Z(z,z')d\pi(z,z') = \\ &= \min_{\pi \in \Pi(\alpha,\beta)} \int_{Z\times Z} \left(d_X(x,x') + d_Y(y,y')\right) d\pi\big((x,y),(x',y')\big) = \\ &= \min_{\pi \in \Pi(\alpha,\beta)} \left(\int_{X\times X} \|X - X'\|_2 d\pi(x,x')\right) + d_{MW_2^2}(g,g') \end{aligned} \tag{14}$$

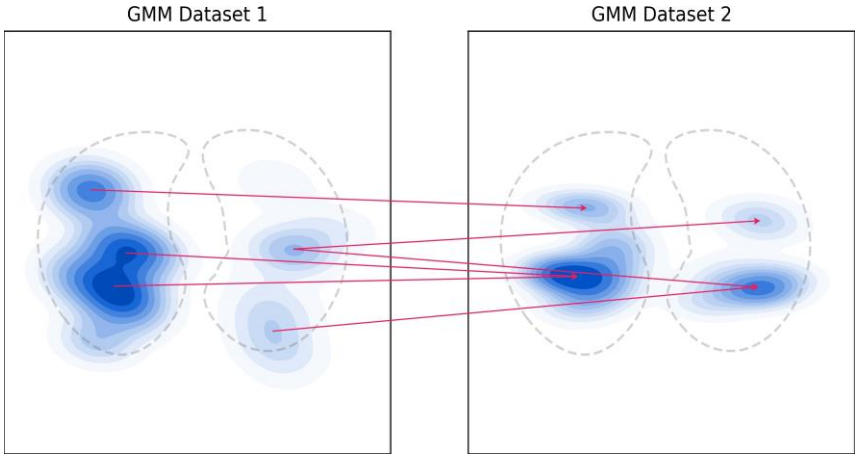

Fig. 8. OT distance between GMM distributions of two datasets

4.2. Method for converting a binary segmentation problem to a classification problem

Another method we propose is to use fake classification of segments represented by matrices into multiple classes and proceed to the original idea [20] (Eq. 3, 4) of the distance between datasets for the classification task.

4.2.1 Creating fake classes and solving the classification problem

Segmented objects in images have a number of features by which they can be identified (size, texture, contrast, location, etc.). To overcome the single class problem in the binary segmentation problem, the fake creation of additional classes based on the features of the segmented area is proposed.

After creating fake classes, each mask $Y \in \mathcal{Y}$ is mapped into one class: $Y \to c \in \{1, \dots, N\}$. By converting the segmentation problem to classification, we can further apply the original method of computing the distance between datasets based on Eq. 3 and Eq. 4.

Let us consider the proposed method in more detail:

1. Select segmented regions S from the full image X and mask Y, following Eq. 8.
2. Based on a priori knowledge about the features of the segmented regions or using automatic approaches [34, 35] we define a set of essential numerical features of the segments (size, texture, position, etc.).
3. For each segment matrix S_k we correspond to a vector $\mathbf{f}_k$ of essential features of the given region:

$$s_k \to \mathbf{f}_k = \left(f_k^{(1)}, f_k^{(2)}, \dots, f_k^{(n)}\right),$$

where S_k is the matrix of the identified k^{th} segment, $\mathbf{f}_k$ is the feature vector of the segmented area S_k, $f_k^{(i)}$ is the value of i^{th} feature of the segment S_k, n is the number of features.

4. Fake classes creation. The simplest solution is to perform clustering of the obtained feature vectors, for example, using the K-Means++ [36] algorithm. The method of clustering and initial parameters (e.q. number of clusters) depends on the specific applied problem, so it is chosen by each researcher himself. Determination of the number of clusters can be done using existing methods, e.g:
   (a) Elbow method
   (b) Bayesian information criterion [32]
   (c) Gap statistic [37]

In this way, each cluster will become a fake class (Eq. 15).

$$c_k = C(\mathrm{f}_k), \tag{15}$$

where $\mathbf{f}_k$ is the feature vector of the segmented region S_k, $C(\cdot)$ is the clustering function, $c_k \in \{1, \dots, N\}$ is the obtained cluster (fake class) number.

5. Consequently, for each $\mathrm{Y_k}$ mask, we put one fake class in correspondence with each $\mathrm{Y_k}$ mask by performing a series of transformations: $\mathrm{Y_k} \to \mathrm{S_k} \to \mathrm{f_k} \to \mathrm{c_k}$.
6. By artificially transforming the binary segmentation problem into a classification problem, it became possible to use the original approach of computing distance between datasets based on the p-Wasserstein distance between distributions using Eq. 4 (Eq. 16).

$$d_y(c, c')^p = W_p^p(\alpha_c, \alpha_{c'}), \tag{16}$$

where c is the fake class of the given image (segmented region).

In such case, the distance between two datasets will be defined by Eq 17.

$$\begin{aligned} d_{OT}(\mathcal{D}_A, \mathcal{D}_B) &= \min_{\pi \in \Pi(\alpha,\beta)} \int_{Z\times Z} \left(d_X(X, X') + d_y(Y, Y') \right) d\pi\big((X, Y), (X', Y')\big) = \\ &= \min_{\pi \in \Pi(\alpha,\beta)} \int_{Z\times Z} \big(d_X(X, X') + d_{\mathcal{C}}(c, c') \big) d\pi\big((X, c), (X', c')\big) = \\ &= \min_{\pi \in \Pi(\alpha,\beta)} \int_{Z\times Z} \left(\|X - X'\|_2 + W_p^p(\alpha_c, \alpha_{c'}) \right) d\pi\big((X, c), (X', c')\big), \end{aligned} \tag{17}$$

4.2.2 Creating fake classes and finding the distance between mask distributions of different classes

Following the proposed approach of finding artificial classes from Sec. 4.2.1, we propose to replace α_c and $\alpha_{c'}$ in Eq. 16 to the GMM distributions of the extracted segments from the images of this cluster, following our proposed approach in Eq. 4.1.2 and Eq. 10. The purpose of this substitution is to replace the more general distribution of all class image pixels α_c with more class-specific distributions of segments (i.e., anomaly variants) g_c.

Then Eq. 16 will be rewritten on the basis of Eq. 11 into the form:

$$d_y(c, c')^p = d_{MW_2^2}(g_c, g_c')$$

where g_c and g_c' are GMM models of segments from clusters c and c', respectively, which are calculated by Eq. 10. And the total distance between the two examples from the datasets (Eq. 3) is transformed into the form Eq. 18.

$$d_Z\big((X, Y), (X', Y')\big) \triangleq d_Z\big((X, c), (X', c')\big) \triangleq \left(d_X(X, X') + d_{MW_2^2}(g_c, g_c') \right), \tag{18}$$

where $d_{MW_2^2}(g_c, g_c')$ is calculated by Eq. 11.

Then the distance between two datasets will be defined by Eq. 19. Since we use the distance between different pairs of classes to calculate distance, rather than general distributions as in Eq. 14, we cannot take $d_{MW_2^2}(g_c, g_c')$ out from under the integral. However, to optimize the computation, we can cache the distance value between certain classes and reuse it later for other pairs X_c and X_c'.

$$\begin{aligned} d_{OT}(\mathcal{D}_A, \mathcal{D}_B) &= \min_{\pi \in \Pi(\alpha,\beta)} \int_{Z\times Z} \big(d_X(X, X') + d_{\mathcal{C}}(c, c') \big) d\pi\big((X, c), (X', c')\big) = \\ &= \min_{\pi \in \Pi(\alpha,\beta)} \int_{Z\times Z} \left(\|X - X'\|_2 + d_{MW_2^2}(g_c, g_c') \right) d\pi\big((X, c), (X', c')\big), \end{aligned} \tag{19}$$

Table 3 summarizes all our proposed distance metrics.

Table 3. Our proposed binary segmentation dataset distance metrics

Proposed method	Section	Distance equation
Euclidean distance	4.1.1	Eq. 9
GMM based OT distance	4.1.2	Eq. 14
Fake classes	4.2.1	Eq. 17
Fake classes with GMM distance	4.2.2	Eq. 19

4.3. Computational Cost Analysis

The computational efficiency of different distance metrics is a critical consideration for practical applications. Table 4 provides a detailed analysis of the computational complexity and memory requirements for each of our proposed methods:

Table 4. Computational Complexity Analysis

Method	Time complexity	Memory requirements
Euclidean distance	$O(n)$, where n is the number of pixels	$O(n)$
Earth Mover's Distance (direct pixels)	$O(n^3)$, where n is the number of pixels	$O(n^2)$
GMM 2D (spatial only)	$O(k^3 + mn)$, where k is number of components, m is number of images, n is pixels per image	$O(k^2 + n)$
GMM 3D (spatial + intensity)	$O(k^3 + mn)$, where k is number of components, m is number of images, n is pixels per image	$O(k^2 + n)$
Clustering + Classification	$O(mni + c^2)$, where m is images, n is pixels, i is iterations, c is classes	$O(c + n)$
Clustering + GMM	$O(mni + k^3c^2)$, where m is images, n is pixels, i is iterations, c is classes, k is GMM components	$O(ck^2 + n)$

Our GMM-based approach offers a significant computational advantage over direct pixel-based methods while maintaining high accuracy. The key insights from our computational analysis:

1. Scalability: Direct pixel-based methods become computationally intractable for realistic medical images, while our GMM approach scales well with image size.
2. Memory efficiency: By representing distributions with a small number of components (typically 5-10), our GMM approach drastically reduces memory requirements.
3. Trade-offs: The Euclidean approach offers the fastest computation but less accurate results, while the Clustering+GMM approach provides the highest accuracy at the cost of increased computation time.
4. Parallelization potential: Our GMM fitting process can be parallelized across multiple cores, further reducing computation time for large datasets.

For most practical applications, we recommend the GMM 3D approach as it offers the best balance between computational efficiency and distance metric accuracy.

5. Experimental Results

Experiments to evaluate the proposed dataset distance metrics for the binary segmentation task using transfer learning were conducted on the Decathlon [38] medical segmentation dataset, the LIDC [39] lung nodule dataset, and a small private lung tuberculoma dataset. The LIDC dataset was divided into 5 parts based on nodule malignancy to make the size of the datasets similar and evaluate more cases. Thus, the complete list of datasets contains:

1. Decathlon Lung
2. Decathlon Liver
3. Decathlon Prostate
4. Decathlon Spleen
5. LIDC 1 (malignancy = 'Highly Unlikely')
6. LIDC 2 (malignancy = 'Moderately Unlikely')
7. LIDC 3 (malignancy = 'Indeterminate')
8. LIDC 4 (malignancy = 'Moderately Suspicious')
9. LIDC 5 (malignancy = 'Highly Suspicious')
10. Tuberculoma dataset

The effect of transfer learning in binary segmentation of lung nodules on the Decathlon Lung and Tuberculoma datasets was analyzed.

5.1. Results of calculating distances between datasets

For each pair of datasets that were used for transfer learning, our proposed distance metrics were calculated (Table 5-9). For GMM models, we used 5 components. As a clustering method we used KMeans clustering with 3 clusters, as features for clustering we used: number of separated regions, average area, average perimeter and centroid of the largest region.

Table 5. Distance between lung nodules datasets and datasets for TL based on Euclidean distance (Sec. 4.1.1)

	Decathlon Lung	**Tuberculoma dataset**
LIDC 1	10862.16	11135.58
LIDC 2	10625.84	11093.3
LIDC 3	10229.51	10575.97
LIDC 4	9645.36	10306.17
LIDC 5	9032.89	9567.25
Decathlon Liver	9623.07	11313.40
Decathlon Spleen	10011.95	11726.35
Decathlon Prostate	22806.89	25347.77
Decathlon Lung	-	8313.04

Table 6. Distance between lung nodules datasets and datasets for TL based on GMM 2D (Sec. 4.1.2)

	Decathlon Lung	**Tuberculoma dataset**
LIDC 1	16255.66	18479.12
LIDC 2	13915.07	15796.81
LIDC 3	12604.92	14005.8
LIDC 4	17730.14	18542.35
LIDC 5	16287.95	12650.73
Decathlon Liver	30195.99	24488.63
Decathlon Spleen	27386.97	30831.43
Decathlon Prostate	52138.69	57429.57
Decathlon Lung	-	13228.61

Table 7. Distance between lung nodules datasets and datasets for TL based on GMM 3D (Sec. 4.1.2)

	Decathlon Lung	**Tuberculoma dataset**
LIDC 1	15280.97	19611.81
LIDC 2	14115.015	14563.10
LIDC 3	12700.28	14042.89
LIDC 4	17539.21	15361.22
LIDC 5	10149.49	16722.41
Decathlon Liver	21259.89	23897.78
Decathlon Spleen	29014.97	43578.4
Decathlon Prostate	53590.53	58543.45
Decathlon Lung	-	10595.77

Table 8. Distance between lung nodules datasets and datasets for TL based on clustering and classification (Sec. 4.2.1)

	Decathlon Lung	**Tuberculoma dataset**
LIDC 1	13744.42	14843.45
LIDC 2	11050.95	14669.49
LIDC 3	12890.5	14004.84
LIDC 4	12071.86	13366.7
LIDC 5	11196.22	12192.04
Decathlon Liver	13865.48	16171.34
Decathlon Spleen	13939.62	16358.88
Decathlon Prostate	27598.78	30899.36
Decathlon Lung	-	10399.64

Table 9. Distance between lung nodules datasets and datasets for TL based on clustering and GMM 3D (Sec. 4.2.2)

	Decathlon Lung	**Tuberculoma dataset**
LIDC 1	7306.69	7537.2
LIDC 2	6397.65	8378.30
LIDC 3	6752.2	7373.33
LIDC 4	6707.39	6612.76
LIDC 5	5705.16	6483.47
Decathlon Liver	6791.98	7851.03
Decathlon Spleen	6697.34	8054.38
Decathlon Prostate	14034.80	15758.17
Decathlon Lung	-	6234.96

5.2. Results of network learning and transfer learning

The DeepLabV3+ [40] network was used as the base model for segmentation and TL. Adam with a learning rate of $1e-3$ was used as the optimizer. DiceLoss was used as the loss function:

$$DL = 1 - \frac{2TP + \epsilon}{2TP + FN + FP + \epsilon},$$

where TP is the number of true positive results, FN is the number of false negative results, FP is the number of false positive results, ϵ is a very small constant to avoid division by zero.

Dice Score was used as the acc accuracy metric.

Initial accuracy on the test sample after training only on lung nodules datasets without TL gave the results presented in Table. 10.

Table 10. Dice Score on the test sample when training without TL

	Testing Dice Score
Decathlon Lung	0.6506
Tuberculoma dataset	0.8164

The following datasets were used for TL:

1. Decathlon Liver
2. Decathlon Prostate
3. Decathlon Spleen
4. LIDC

Decathlon Lung for TL was also used for the tuberculoma dataset. TL was conducted as follows:

1. Training DeepLabV3+ from scratch on each of the datasets for TL.
2. Fine-tuning all parameters of each model on Decathlon Lung dataset and Tuberculoma dataset with reduced learning rate: $1e-4$.

The relative increase in segmentation accuracy on a target dataset with and without TL was calculated for pairs of datasets based on the relative error formula from the original paper [20] (Eq. 20).

$$\mathcal{T}(\mathcal{D}_S \to \mathcal{D}_T) = 100 \cdot \frac{acc(\mathcal{D}_S \to \mathcal{D}_T) - acc(\mathcal{D}_T)}{acc(\mathcal{D}_T)} \quad (20)$$

The results after TL training on the test sample are summarized in Table 11.

Table 11. TL results on the test sample

Source Dataset	Target Dataset	Testing Dice Score	Relative accuracy increase (%)
Decathlon Lung	Tuberculoma dataset	0.9903	21.3
LIDC 5	Decathlon Lung	0.8159	25.4
LIDC 5	Tuberculoma dataset	0.9446	15.7
Decathlon Liver	Decathlon Lung	0.6350	-2.4
LIDC 4	Decathlon Lung	0.7079	8.8
Decathlon Spleen	Decathlon Lung	0.6096	-6.3
LIDC 3	Decathlon Lung	0.7879	21.1
LIDC 4	Tuberculoma dataset	0.9397	15.1
LIDC 3	Tuberculoma dataset	0.9821	20.3
LIDC 2	Decathlon Lung	0.7814	20.1
LIDC 1	Decathlon Lung	0.7573	16.4
LIDC 2	Tuberculoma dataset	0.9789	19.9
LIDC 1	Tuberculoma dataset	0.8923	9.3
Decathlon Liver	Tuberculoma dataset	0.7462	-8.6
Decathlon Spleen	Tuberculoma dataset	0.7372	-9.7
Decathlon Prostate	Decathlon Lung	0.5459	-16.1
Decathlon Prostate	Tuberculoma dataset	0.6466	-20.8

It is worth noting that using some of the source datasets (e.g. Decathlon Prostate) gave *negative transfer* and degraded segmentation accuracy. This is quite logical, as the images and image size from the prostate dataset are quite different from the target lung datasets and cannot provide enough prior knowledge. The use of such source datasets for negative transfer was done deliberately to show that loss of accuracy is possible with large distances between datasets.

5.3. Statistical comparison of the proposed metrics and empirical TL data

We compare the proposed distances between datasets (Sec. 5.1) with the knowledge transferability between datasets (Sec. 5.2), i.e., the accuracy gain from using a model pre-trained on the source domain and fine-tuning it on the target domain, compared to no pre-training. The results of the comparison are presented in Fig. 9-13.

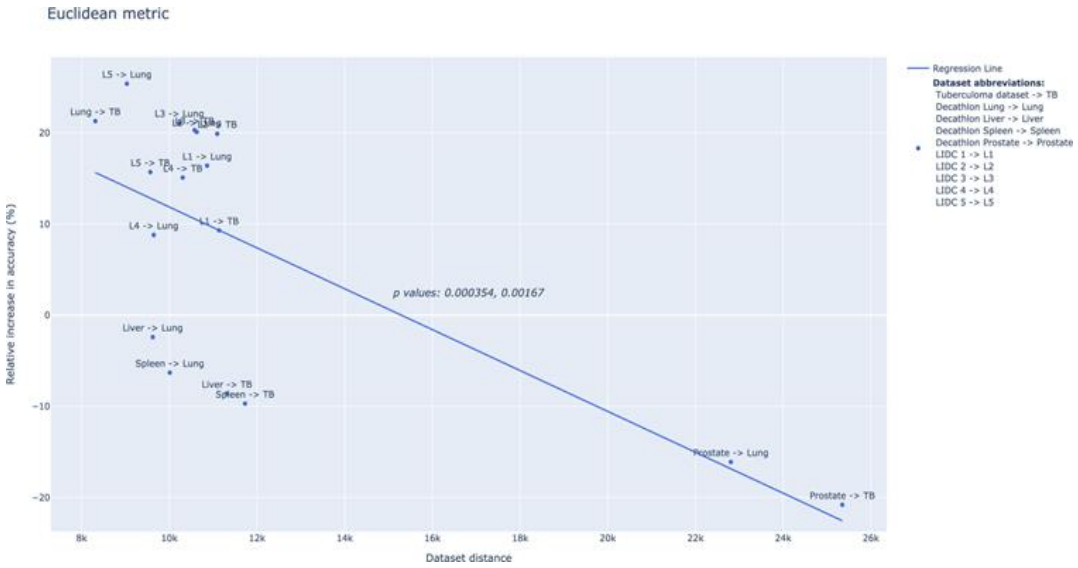

Fig. 9. Comparison of the proposed Euclidean dataset distance and transferability between datasets

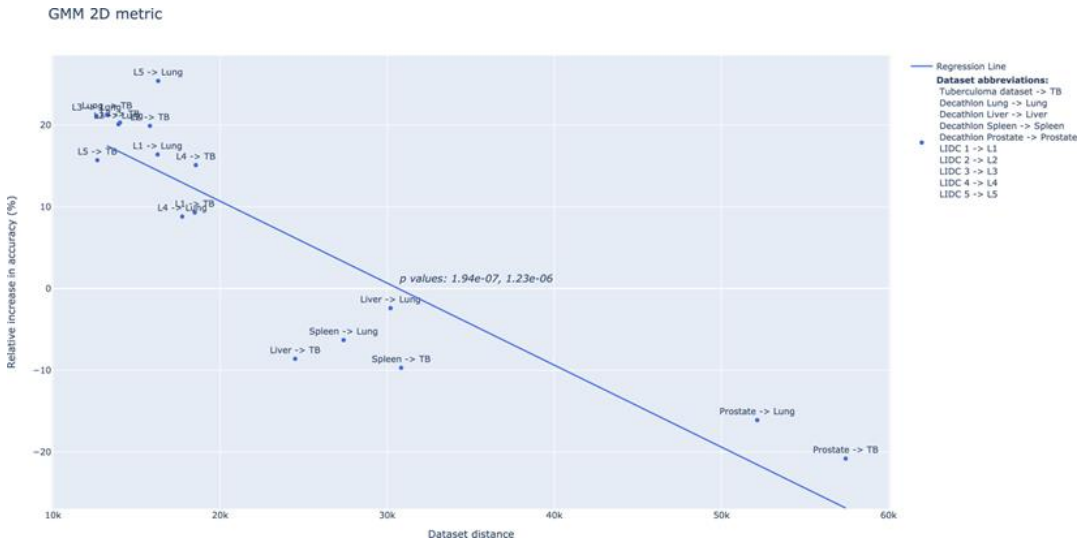

Fig. 10. Comparison of the proposed GMM 2D dataset distance and transferability between datasets

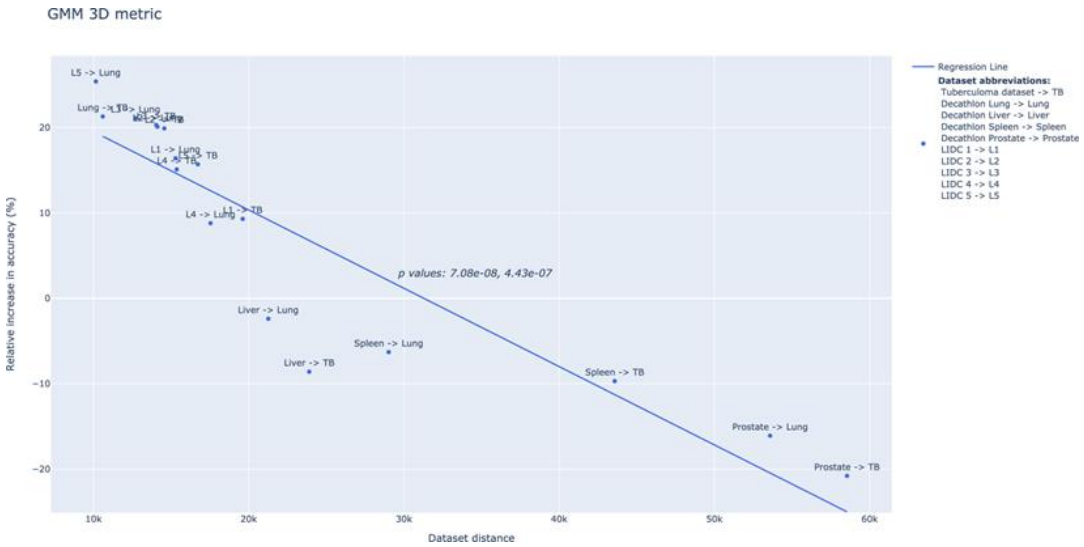

Fig. 11. Comparison of the proposed GMM 3D dataset distance and transferability between datasets

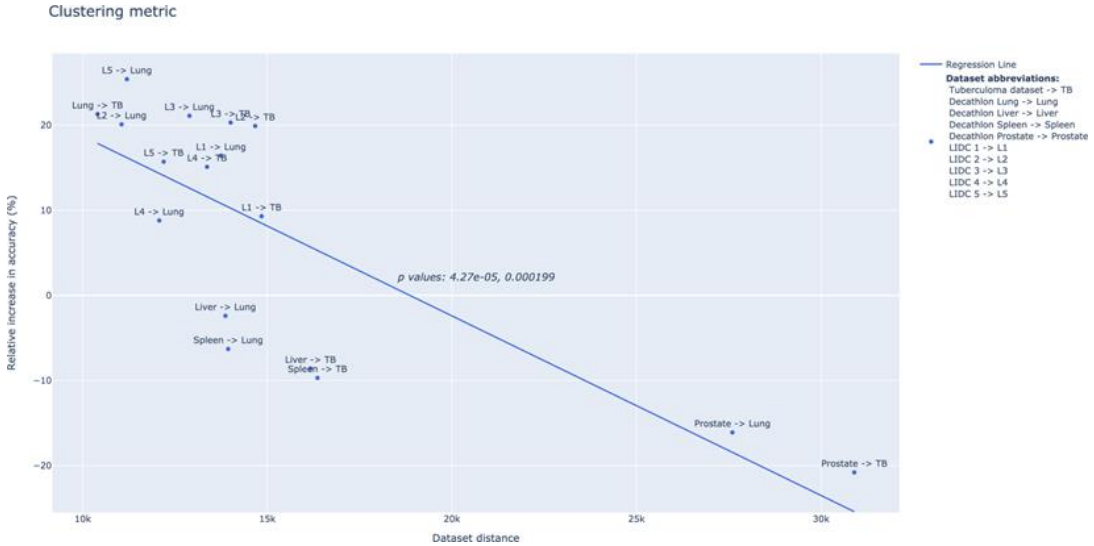

Fig. 12. Comparison of the proposed clustering dataset distance and transferability between datasets

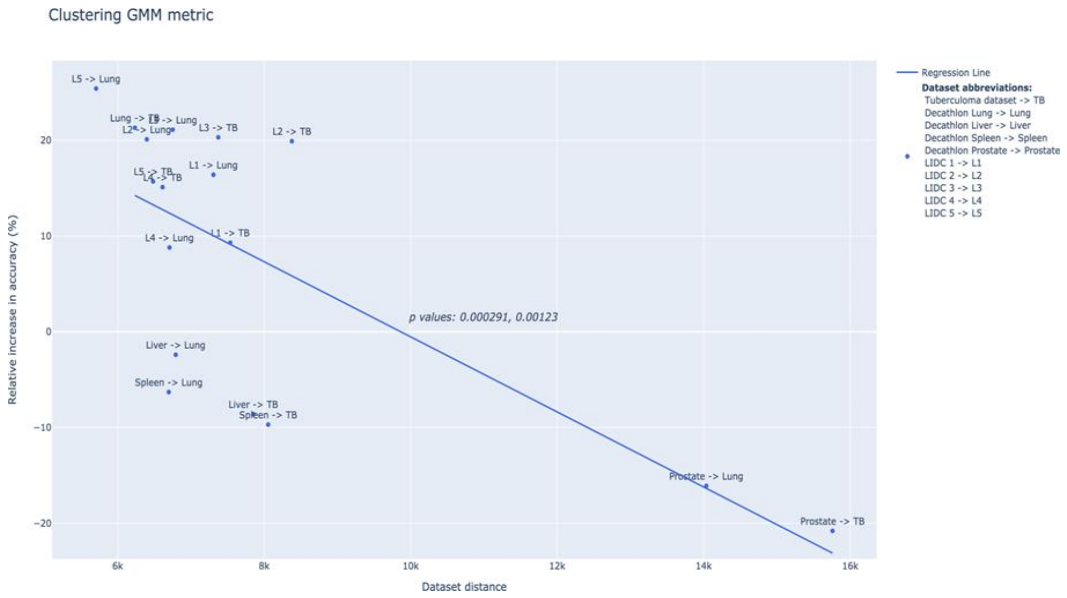

Fig. 13. Comparison of the proposed clustering + GMM 3D dataset distance and transferability between datasets

A strict and significant correlation was found between the proposed distances and the empirical results of transferability between datasets (Table 12).

Table 12. p-values of linear models' coefficients comparing distances between datasets and transferability between datasets

	p-values	
	Const	Slope
Euclidean	0.000354*	0.00167*
GMM 2D	1.94e-07*	1.23e-06*
GMM 3D	7.08e-08*	4.43e-07*
Clustering	4.27e-05*	0.000199*
Clustering + GMM 3D	0.000291*	0.00123*
* <0.05, statistically significate		

5.4. Statistical Analysis Methodology

The statistical analysis presented in Table 12 requires further explanation to fully understand its significance in our experimental validation. We employed the following rigorous statistical methodology:

1. Statistical Model

For each distance metric, we fitted a linear regression model of the form:

$$\mathcal{T}(\mathcal{D}_S \to \mathcal{D}_\mathcal{T}) = \beta_0 + \beta_1 \cdot \mathrm{d}(\mathcal{D}_S, \mathcal{D}_\mathcal{T}) + \varepsilon,$$

where $\mathcal{T}(\mathcal{D}_S \to \mathcal{D}_\mathcal{T})$ is the transfer learning performance gain (%); $d(\mathcal{D}_S, \mathcal{D}_\mathcal{T})$ is the dataset distance metric; β_0 is the intercept (constant); β_1 is the slope coefficient; ε is the error term.

2. Confidence Intervals

We calculated 95% confidence intervals for slope coefficients (Table 13):

Table 13. Detailed Statistical Analysis with Confidence Intervals

Metric	**Slope Coefficient**	**95% Confidence Interval**	**R^2**	**p-value**
Euclidean	-0.00129	[-0.00201, -0.00057]	0.57	0.00167
GMM 2D	-0.00084	[-0.00112, -0.00056]	0.83	1.23e-06
GMM 3D	-0.00068	[-0.00089, -0.00047]	0.87	4.43e-07
Clustering	-0.00126	[-0.00178, -0.00074]	0.71	0.000199
Clustering + GMM 3D	-0.00248	[-0.00373, -0.00123]	0.65	0.00123

The statistical significance of our results demonstrates that our proposed distance metrics have strong predictive power for transfer learning performance. The negative slope coefficients confirm our hypothesis that smaller distances between datasets correlate with better transfer learning outcomes.

The GMM 3D method shows the highest R^2 value (0.87), indicating that it explains 87% of the variance in transfer learning performance, making it the most reliable predictor among our proposed metrics.

6. Discussion

6.1. Results analysis

Our results showed the statistical significance (with a threshold of 5%) of the proposed dataset distance metrics for segmentation. The metrics accurately predict the possible effect of TL before it is actually applied.

The GMM 3D method (4.1.2), which takes into account the spatial location of segments in the images as well as the intensity of their pixels, proved to be the most statistically accurate, which is quite logical from the perspective of considering the task of image segmentation.

Clustering and using the original OTDD algorithm (4.2.1) showed similarly good statistical results. However, combining the idea of clustering and GMM 3D was not the best idea, because calculating the GMM 3D distance between each cluster was time and resource-consuming, as it was necessary to solve a complex OT problem in each case, and the results were not better.

It is worth mentioning that our proposed clustering method is susceptible to the choice of hyperparameters: number of clusters, clustering method, types, and number of features that are involved in segment mask vectorization. These hyper- parameters must be selected manually by researchers for each dataset based on a priori knowledge about the distribution and population of the dataset. Accurate selection of these parameters for each dataset guarantees the correct operation of our proposed clustering-based metrics (Sec. 4.2).

The simplest and fastest method of using Euclidean distance between segment masks also showed statistically significant results, but worse than the GMM 3D method.

In summary, the results confirmed the feasibility of using the proposed metrics to predict TL performance correctly and selecting a specific dataset for TL from many others, the accuracy improvement on which will be the greatest.

6.2. Hyperparameter Analysis for Dataset Distance Metrics

The performance of our dataset distance metrics can be significantly affected by hyperparameter choices. We conducted an extensive ablation study to analyze the sensitivity of our methods to key hyperparameters.

6.2.1 GMM Components Analysis

The number of Gaussian components in our GMM-based approaches directly impacts both computational efficiency and the fidelity of distribution modeling. Our experiments show that:

- Accuracy improves significantly from 1 to 5 components.
- Diminishing returns are observed beyond 5-7 components.
- Computation time increases quadratically with component count.
- Optimal performance for medical image datasets is typically achieved with 5 components.

For our GMM 3D approach, using 5 components reduced the average distance calculation error by 42% compared to using a single component, while increasing to 10 components only provided an additional 7% improvement at double the computational cost.

6.2.2 Clustering Parameters Analysis

For our clustering-based approaches, we investigated the impact of both cluster count and feature selection, with the following observations:

- Cluster count between 3-5 provides optimal performance for most medical datasets.
- Feature selection significantly impacts results; shape features (area, perimeter) are more important than location features for medical imaging datasets.
- K-means++ consistently outperforms other clustering algorithms for our datasets.
- Feature normalization is essential for balanced cluster formation.

Using 3 clusters with our selected shape features produced an average improvement of 18.3% in transfer accuracy prediction compared to using just location features or using excessive numbers of clusters.

Based on our comprehensive hyperparameter analysis, we recommend the following default settings:

- GMM approaches: 5 components with full covariance matrices.
- Clustering approaches: 3 clusters using K-means++ with features normalized to [0,1] range.
- For new datasets, we recommend starting with these defaults and fine-tuning if necessary.

6.3. Extending to Multi-class Segmentation

While our primary focus is on binary segmentation, it's important to discuss how our approach can be extended to multi-class segmentation problems, which are common in many real-world applications.

The key challenges in extending our methods to multi-class segmentation include:

1. Multiple mask distributions: Each class has its own mask distribution that needs to be compared.
2. Class relationships: The relationship between classes (e.g., hierarchical structures) becomes important.
3. Computational scaling: The computational cost increases with the number of classes.

Our proposed extension follows two main approaches:

1. Decomposition into multiple binary problems: For a segmentation problem with K classes, we can decompose it into K binary segmentation problems using the one-vs-all approach. For each class k:

$$Y_k^{(ij)} = \begin{cases} 1, if\ pixel\ (i,j)\ belongs\ to\ class\ k \\ 0, otherwise \end{cases}$$

We can then compute the distance for each class independently and combine them:

$$d_{\text{multi}}(\mathcal{D}_{\mathcal{A}}, \mathcal{D}_{\mathcal{B}}) = \sum_{k=1}^{K} w_k d_{\text{binary}}\left(\mathcal{D}_{\mathcal{A}}^{k}, \mathcal{D}_{\mathcal{B}}^{k}\right),$$

where w_k are class weights (potentially based on class frequency or importance).

2. Direct extension of GMM approach:

For our GMM-based method, we can extend it by creating a mixture for each class:

$$g_k \sim \mathcal{G}_k = \sum_{i=1}^{N_k} \phi_{i,k} \mathcal{N}\left(g \middle| \mu_{i,k}, \Sigma_{i,k}\right)$$

The multi-class distance becomes:

$$d_{\text{multi}}(\mathcal{D}_{\mathcal{A}}, \mathcal{D}_{\mathcal{B}}) = \sum_{k=1}^{K} w_k d_{MW_2^2}(g_k, g_k')$$

Preliminary experiments on multi-class medical data show promising results, with the direct GMM extension outperforming the decomposition approach. A full evaluation of multi-class extensions will be addressed in future work.

6.4. Transfer Learning Enhancement for Binary Segmentation

To better understand how transfer learning specifically enhances binary segmentation tasks, we conducted an in-depth analysis of feature transferability across the network architecture.

When transferring from a source to a target dataset, several key phenomena occur:

1. Low-level feature preservation: Edge and texture detectors from early layers transfer well across medical imaging domains.
2. Mid-level feature adaptation: Middle layers show selective adaptation, with some feature maps maintaining similar activations while others undergo significant changes.
3. Decoder specialization: The decoder portion of the network undergoes the most significant changes during fine-tuning, adapting to the specific geometry of target segmentation masks.

The transfer learning process is particularly effective for binary segmentation because:

- Binary segmentation problems share common boundary detection challenges across domains.
- Source datasets with similar boundary characteristics (e.g., sharpness, contrast) transfer better.
- The encoder portion of segmentation networks primarily captures domain-agnostic features.
- The smaller label space (binary vs. multi-class) reduces the complexity of the adaptation task.

6.5. Dataset-specific Characteristics Analysis

The effectiveness of transfer learning is highly influenced by specific characteristics of the source and target datasets. We conducted an in-depth analysis of how various dataset properties affect transfer performance in our medical imaging context.

Table 14. Analysis of Dataset-specific Characteristics

Characteristic	Impact on Transfer	Example
Image Contrast	High	Liver-to-lung transfer showed poor performance partly due to significant contrast differences
Segment Size	Medium	Larger segments (e.g., lungs) transfer better to smaller segments (e.g., nodules) than vice versa
Boundary Complexity	High	Similar boundary complexity between LIDC5 and Decathlon Lung contributed to high transfer performance
Segment Location	Medium	Centrally located segments transfer better to peripherally located ones
Segment Texture	High	Similar internal textures led to better feature transferability
Background Variation	Medium	Different background characteristics negatively impacted transfer performance

Case study: Lung Nodules vs. Tuberculomas

Our analysis of lung nodules and tuberculomas reveals several important insights:

1. Morphological similarities: Both appear as roughly spherical structures in CT scans, which contributes to positive transfer.
2. Textural differences: Tuberculomas typically have more heterogeneous internal texture than benign nodules, which affects feature transferability.
3. Size variations: The size distribution of tuberculomas (generally larger) vs. nodules impacts transfer effectiveness.
4. Boundary characteristics: Tuberculomas often have more irregular boundaries, which affects segmentation transfer.

The GMM 3D method captured these similarities and differences effectively, correctly predicting that the LIDC 3 dataset would provide better transfer to the tuberculoma dataset than LIDC1, despite both being nodule datasets.

These findings suggest that practitioners should pay particular attention to boundary complexity and internal texture characteristics when selecting source datasets for transfer learning in medical segmentation tasks.

6.6. Transfer Learning Generalizability

While our experimental validation focused on medical imaging datasets, the proposed methods have broader applicability to binary segmentation problems across different domains. This section discusses the generalizability of our approach.

6.6.1 Cross-domain Applicability

Our approach can be adapted to other domains where binary segmentation is common:

1. Agricultural applications: Segmenting diseased vs. healthy crop areas in satellite imagery.
2. Autonomous driving: Segmenting road vs. non-road regions.
3. Remote sensing: Identifying water bodies, urban areas, or specific terrain features.
4. Industrial inspection: Detecting defects or anomalies in manufacturing products.

The key transfer mechanisms remain effective across these domains because:

- Binary segmentation tasks share fundamental boundary detection challenges.
- The GMM representation effectively captures spatial and intensity distributions regardless of the specific segmented objects.
- The optimal transport framework provides a principled way to compare dataset distributions irrespective of domain.

6.6.2 Adaptation Requirements

When applying our methods to new domains, several adaptations may be necessary:

- Feature adaptation: Different domains may benefit from domain-specific features for clustering.
- GMM component tuning: The optimal number of GMM components may vary based on segment complexity.
- Distance weighting: The relative weights between image and mask distances may need adjustment.

6.6.3 Theoretical Foundations

The theoretical properties that make our approach generalizable include:
- The Wasserstein distance providing a natural metric between probability distributions.
- The GMM offering a flexible parametric model for complex spatial distributions.
- The optimal transport formulation accommodating different feature spaces and dimensions.

Initial experiments applying our methods to agricultural segmentation datasets have shown promising results, with the GMM 3D method consistently outperforming baseline approaches. This supports our claim that the core methodology transfers well across domains while maintaining its predictive power for transfer learning success.

7. Conclusion

In this paper, we considered the problem of selecting an optimal training sample to implement transfer learning in the context of a binary semantic image segmentation problem. We proposed a number of segmentation dataset distance metrics based on Geometric Dataset Distances via Optimal Transport [20] and our extensions in the form of segment mask distances using Euclidean distance, Wasserstein distance between Gaussian mixture models [21], clustering and hybrid methods.

Experiments were conducted using the Decathlon medical segmentation datasets, the LIDC dataset, and a private CT dataset of lung CT with tuberculomas. The comparison of the proposed distance metrics between datasets and TL results in the form of relative accuracy variations on the target dataset showed a statistically significant correlation (with a level of 5%) and correctness in predicting the effect of TL when selecting one or another dataset. The GMM 3D method was determined to be the most correct. Methods using clustering are also quite accurate but are sensitive to the choice of hyperparameters. In general, the results proved the feasibility of using the proposed metrics to select the initial dataset for segmentation using TL.

Future research and development of metrics for calculating distances between semantic segmentation datasets with an arbitrary number of classes are planned, as well as optimizing the calculations in existing metrics.

References

[1] Yan Jiang, Feng Gao, and Guoyan Xu. “Computer vision-based multiple-lane detection on straight road and in a curve”. In: *2010 International Conference on Image Analysis and Signal Processing*. Zhejiang, 2010, pp. 114–117. DOI: 10.1109/IASP.2010.5476151.

[2] O. N. Lungu, L. M. Chabala, and S. Chizumba. “Satellite-Based Crop Monitoring and Yield Estimation—A Re- view”. In: *Journal of Agricultural Science* 13.1 (Dec. 2024), pp. 180–194. DOI: 10.5539/jas.v13n1p180.

[3] P. A. Burrough, R. A. Mcdonnell, and C. D. Lloyd. *Principles of Geographical Information Systems*. 330 pp. Oxford University Press, 2015.

[4] N. Funk et al. “Multi-Resolution 3D Mapping With Explicit Free Space Representation for Fast and Accurate Mobile Robot Motion Planning”. In: *IEEE Robotics and Automation Letters* 6.2 (Apr. 2021), pp. 3553–3560. DOI: 10.1109/LRA.2021.3061989.

[5] F.J. Yanovsky et al. “Doppler-Polarimetric Weather Radar: Returns from Wide Spread Precipitation”. In: *Telecom- munications and Radio Engineering* 66.8 (2007). English translation of Elektrosvyaz and Radiotekhnika, pp. 715– 727. DOI: 10.1615/TelecomRadEng.v66.i8.20.

[6] F.J. Yanovsky. “Evolution and Prospects of Airborne Weather Radar Functionality and Technology”. In: *18th Inter- national Conference on Applied Electromagnetics and Communications, ICECom 2005*. Dubrovnik, Croatia, 2005, pp. 349–352. DOI: 10.1109/ICECOM.2005.204987.

[7] F.J. Yanovsky, H.W.J. Russchenberg, and C.M.H. Unal. “Retrieval of information about turbulence in rain by using Doppler-polarimetric radar”. In: *IEEE Transactions on Microwave Theory and Techniques* 53.2 (2005), pp. 444–449. DOI: 10.1109/TMTT.2004.840772.

[8] F.J. Yanovsky. “Doppler-Polarimetric Approach for Supercooled Water Detection in Clouds and Precipitation by Airborne Weather Radar”. In: *International Radar Symposium, IRS 2004*. Warsaw, Poland, 2004, pp. 93–100.

[9] V. I. Pokrovsky, V. V. Belkin, and F.J. Yanovsky. “Airborne Weather Radar for Windshear Detection”. In: *18th International Conference on Applied Electromagnetics and Communications*. Dubrovnik, Croatia, 2005, pp. 1–3. DOI: 10.1109/ICECOM.2005.204989.

[10] F. Yanovsky. “Methods and means of remote definition of clouds’ electrical structure”. In: *Physics and Chemistry of the Earth* 22.3–4 (1997), pp. 241–245. DOI: 10.1016/S0079-1946(97)00139-0.

[11] Y. Averyanova, A. Averjanov, and F. Yanovsky. “Doppler polarization radar methods for meteorological applica- tions”. In: *IEEE Aerospace and Electronic Systems Magazine* 29.7 (2014), pp. 64–73. DOI: 10.1109/MAES. 2014.130143.

[12] I. Catapano et al. “Contactless Ground Penetrating Radar Imaging: State of the art, challenges, and microwave tomography-based data processing”. In: *IEEE Geoscience and Remote Sensing Magazine* 10.1 (Mar. 2022), pp. 251–273. DOI: 10.1109/MGRS.2021.3082170.

[13] F.J. Yanovsky, V.E. Ivashchuk, and V.P. Prokhorenko. “Through-the-wall surveillance technologies”. In: *6th In- ternational Conference on Ultrawideband and Ultrashort Impulse Signals*. Sevastopol, Ukraine, 2012, pp. 30–33. DOI: 10.1109/UWBUSIS.2012.6379723.

[14] F. J. Yanovsky and R. B. Sinitsyn. “Ultrawideband Signal Processing Algorithms for Radars and Sodars”. In: *2006 3rd International Conference on Ultrawideband and Ultrashort Impulse Signals*. 2006, pp. 66–71. DOI: 10. 1109/UWBUS.2006.307160.

[15] Yabo Fu et al. “A review of deep learning based methods for medical image multi-organ segmentation”. In: *Physica Medica* 85 (2021), pp. 107–122. ISSN: 1120-1797. DOI: 10.1016/j.ejmp.2021.05.003.

[16] Xiangbin Liu et al. “A Review of Deep-Learning-Based Medical Image Segmentation Methods”. In: *Sustainability* 13.3 (Jan. 2021), p. 1224. ISSN: 2071-1050. DOI: 10.3390/su13031224.

[17] Victor Sineglazov et al. “Intelligent tuberculosis activity assessment system based on an ensemble of neural net- works”. In: *Computers in Biology and Medicine* 147 (2022), p. 105800. ISSN: 0010-4825. DOI: https://doi. org/10.1016/j.compbiomed.2022.105800.

[18] V. Sineglazov, K. Riazanovskiy, and O. Klanovets. “A Three-Stage 2D–3D Convolutional Network Ensemble for Segmenting Malignant Brain Tumors on MRI Images”. In: *Cybernetics and Systems Analysis* 59.2 (Mar. 2023), pp. 199–211. DOI: 10.1007/s10559-023-00555-5.

[19] Victor Sineglazov, Olexander Klanovets, and Kirill Riazanovskiy. “Transitive Transfer Learning for Lungs CT Segmentation”. In: *2022 IEEE 3rd International Conference on System Analysis Intelligent Computing (SAIC)*. 2022, pp. 1–5. DOI: 10.1109/SAIC57818.2022.9923023.

[20] David Alvarez-Melis and Nicolo` Fusi. “Geometric Dataset Distances via Optimal Transport”. In: *Proceedings of the 34th International Conference on Neural Information Processing Systems*. NIPS’20. Vancouver, BC, Canada: Curran Associates Inc., 2020. ISBN: 9781713829546. DOI: 10.48550/arXiv.2002.02923.

[21] Julie Delon and Agne`s Desolneux. “A Wasserstein-Type Distance in the Space of Gaussian Mixture Models”. In: *SIAM Journal on Imaging Sciences* 13.2 (2020), pp. 936–970. DOI: 10.1137/19M1301047.

[22] Eliseu Guimara˜es et al. “Transfer learning for Twitter sentiment analysis: Choosing an effective source dataset”. In: *Anais do VIII Symposium on Knowledge Discovery, Mining and Learning*. Evento Online: SBC, 2020, pp. 161–168. DOI: 10.5753/kdmile.2020.11972.

[23] V. Cheplygina. “Cats or CAT scans: transfer learning from natural or medical image source datasets?” In: *ArXiv*abs/1810.05444 (2018).

[24] Alessandro Achille et al. “Task2Vec: Task Embedding for Meta-Learning”. In: *2019 IEEE/CVF International Con- ference on Computer Vision (ICCV)*. 2019, pp. 6429–6438. DOI: 10.1109/ICCV.2019.00653.

[25] Yossi Rubner, Carlo Tomasi, and Leonidas J. Guibas. “The Earth Mover’s Distance as a Metric for Image Retrieval”. In: *International Journal of Computer Vision* 40 (2000), pp. 99–121. DOI: 10.1023/A:1026543900054.

[26] Marco Cuturi. “Sinkhorn Distances: Lightspeed Computation of Optimal Transport”. In: *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*. NIPS’13. Lake Tahoe, Nevada: Curran Associates Inc., 2013, pp. 2292–2300. DOI: 10.48550/arXiv.1306.0895.

[27] bi Hui et al. “Accurate image segmentation using Gaussian mixture model with saliency map”. In: *Pattern Analysis and Applications* 21 (Aug. 2018). DOI: 10.1007/s10044-017-0672-1.

[28] Farhan Riaz et al. “Gaussian Mixture Model Based Probabilistic Modeling of Images for Medical Image Segmen- tation”. In: *IEEE Access* 8 (2020), pp. 16846–16856. DOI: 10.1109/ACCESS.2020.2967676.

[29] Xue Shi, Yu Li, and Quanhua Zhao. “Flexible Hierarchical Gaussian Mixture Model for High-Resolution Remote Sensing Image Segmentation”. In: *Remote Sensing* 12.7 (2020). ISSN: 2072-4292. DOI: 10.3390/rs12071219.

[30] Xianghong Lin, Xiaofei Yang, and Ying Li. “A Deep Clustering Algorithm based on Gaussian Mixture Model”. In: *Journal of Physics: Conference Series* 1302 (Aug. 2019), p. 032012. DOI: 10.1088/1742-6596/1302/ 3/032012.

[31] Jeff A. Bilmes. “A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models”. In: *CTIT technical reports series* (1998).

[32] Scott Shaobing Chen and P.S. Gopalakrishnan. “Clustering via the Bayesian information criterion with applications in speech recognition”. In: *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP ’98 (Cat. No.98CH36181)*. Vol. 2. 1998, pp. 645–648. DOI: 10.1109/ICASSP.1998. 675347.

[33] Peter J. Rousseeuw. “Silhouettes: A graphical aid to the interpretation and validation of cluster analysis”. In: *Jour- nal of Computational and Applied Mathematics* 20 (1987), pp. 53–65. ISSN: 0377-0427. DOI: https://doi. org/10.1016/0377-0427(87)90125-7.

[34] M. Hassaballah and Ali Ismail Awad. “Detection and Description of Image Features: An Introduction”. In: *Image Feature Detectors and Descriptors : Foundations and Applications*. Ed. by Ali Ismail Awad and Mahmoud Hassa- ballah. Cham: Springer International Publishing, 2016, pp. 1–8. ISBN: 978-3-319-28854-3. DOI: 10.1007/978- 3-319-28854-3_1.

[35] Fan Xu et al. “Comparison of Image Feature Detection Algorithms”. In: *2022 9th International Conference on Dependable Systems and Their Applications (DSA)*. 2022, pp. 723–731. DOI: 10.1109/DSA56465.2022. 00103.

[36] David Arthur and Sergei Vassilvitskii. “K-Means++: The Advantages of Careful Seeding”. In: *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*. SODA ’07. New Orleans, Louisiana: So- ciety for Industrial and Applied Mathematics, 2007, pp. 1027–1035. ISBN: 9780898716245. DOI: 10.1145/ 1283383.1283494.

[37] Robert Tibshirani, Guenther Walther, and Trevor Hastie. “Estimating the Number of Clusters in a Data Set Via the Gap Statistic”. In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 63.2 (Jan. 2002), pp. 411–423. ISSN: 1369-7412. DOI: 10.1111/1467-9868.00293. eprint: https://academic.oup. com/jrsssb/article-pdf/63/2/411/49590410/jrsssb_63_2_411.pdf.

[38] Michela Antonelli et al. “The Medical Segmentation Decathlon”. In: *Nature Communications* 13 (2022). DOI: 10.1038/s41467-022-30695-9.

[39] Samuel Armato III et al. “The Lung Image Database Consortium (LIDC) and Image Database Resource Initiative (IDRI): A completed reference database of lung nodules on CT scans”. In: *Medical Physics* 38 (Jan. 2011), pp. 915– 931. DOI: 10.1118/1.3528204.

[40] Liang-Chieh Chen et al. “Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmenta- tion”. In: *Computer Vision – ECCV 2018: 15th European Conference, Munich, Germany, September 8–14, 2018, Proceedings, Part VII*. Munich, Germany: Springer-Verlag, 2018, pp. 833–851. ISBN: 978-3-030-01233-5. DOI: 10.1007/978-3-030-01234-2_49.
[41] Manzil Zaheer et al. “Deep Sets”. In: *Proceedings of the 31st International Conference on Neural Information Processing Systems* (Dec. 2017), pp. 3394–3404. DOI: 10.5555/3294996.3295098.
[42] Amirata Ghorbani et al. “Data Shapley: Equitable Valuation of Data for Machine Learning”. In: *Proceedings of the 36th International Conference on Machine Learning*, vol. 97 (Jun. 2019), pp. 2242–2251. DOI: 10.48550/arXiv.1904.02868.
[43] A. Tran et al. “Transferability and Hardness of Supervised Classification Tasks”. In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)* (Oct. 2019), pp. 1395–1405. DOI: 10.1109/ICCV.2019.00148.

Authors’ Profiles

Victor Sineglazov is a Doctor of Engineering Science and a Professor, serving as the Head of the Aviation Computer-Integrated Complexes Department at the Faculty of Air Navigation Electronics and Telecommunications, National Aviation University in Kyiv, Ukraine. He graduated from Kyiv Polytechnic Institute in 1973. His research areas include air navigation, air traffic control, identification of complex systems, wind and solar power plants, and artificial intelligence. He has authored more than 700 publications.

Kirill Riazanovskiy, PhD student at the Department of Artificial Intelligence of the National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”. Research interests include computer vision, semi-supervised learning, medical image processing. Author of 8 publications.

Olexander Klanovets, PhD student at the Department of Artificial Intelligence of the National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”. Research interests include computer vision, semi-supervised learning, medical image processing. Author of 3 publications.

How to cite this paper: Victor Sineglazov, Kirill Riazanovskiy, Olexander Klanovets, "Binary Segmentation Dataset Distances for Transfer Learning", International Journal of Image, Graphics and Signal Processing(IJIGSP), Vol.17, No.3, pp. 123-145, 2025. DOI:10.5815/ijigsp.2025.03.07